

Visual CE[®]

Version 12

SYWARE[®]
Inc.

Table of Contents

Introduction	9
Overview of Visual CE	9
Visual CE Editions	10
Visual CE Controls	11
About this Documentation.....	13
Installation and Support.....	14
System Requirements.....	14
Installing Visual CE	15
About the Sample Applications	15
Technical Support	16
Visual CE Add-on Tools	17
PART I: QUICK TOURS.....	19
Quick Tour I: Using a Visual CE Application on a Touchscreen Device.....	20
Overview of Quick Tour I	20
Quick Lesson 1: Running a Visual CE Application on the CE Device	21
Quick Lesson 2: Creating Records	22
Quick Lesson 3: Displaying a Record List	25
Quick Lesson 4: Filtering the Record List.....	26
Quick Lesson 5: Searching for Records	27
Quick Lesson 6: Using Command Buttons	28
Quick Lesson 7: Exiting a Visual CE Application.....	29
Quick Tour Ia: Using a Visual CE Application on a Smartphone.....	30
Overview of Quick Tour Ia	30
Quick Lesson 1a: Running a Visual CE Application on a Smartphone	31
Quick Lesson 2a: Creating Records	32
Quick Lesson 3a: Displaying a Record List	34
Quick Lesson 4a: Searching for Records	35
Quick Lesson 5a: Using Command Buttons to Move to Different Regions of the Form.....	36
Quick Lesson 6a: Exiting a Visual CE Application.....	37
Quick Tour II: Creating a Visual CE Application for Touchscreen Devices.....	38
Overview of Quick Tour II	38
Quick Lesson 1: Starting Visual CE's Development Environment.....	39
Quick Lesson 2: Creating a Table.....	40
About Tables and Forms.....	42
Quick Lesson 4: Creating an Edit Box and a Label	44
Quick Lesson 5: Creating the Remaining Edit Boxes and Labels	48
Quick Lesson 6: Arranging Controls on the Form.....	49
Quick Lesson 7: Creating a Calculated Field.....	51
Quick Lesson 8: Creating a Checkbox	52
Quick Lesson 9: Creating Radio Buttons.....	53
Quick Lesson 10: Creating a Note Box.....	55
Quick Lesson 11: Creating a Drop-Down List.....	56
Quick Lesson 12: Creating Command Buttons.....	58
Quick Lesson 13: Creating a Timestamp Button	60
Quick Lesson 14: Creating a Scribble Box	61
Quick Lesson 15: Creating a Title for the Application.....	62
Quick Lesson 16: Specifying an Initial Sort/Search Order.....	63
Quick Lesson 17: Specifying the Tab and Column Order	64
Quick Lesson 18: Downloading Your Application.....	66

Quick Lesson 19: Testing Your Application.....	67
Quick Tour IIa: Creating a Visual CE Application for Smartphones	68
Overview of Quick Tour IIa	68
Quick Lesson 1a: Starting Visual CE's Development Environment	69
Quick Lesson 2a: Creating a Table	70
Quick Lesson 3a: Selecting a Table	73
About Tables and Forms	73
Quick Lesson 4a: Creating an Edit Box.....	75
Quick Lesson 5a: Creating a Command Button to Sort Records.....	78
Quick Lesson 6a: Creating the Remaining Edit Boxes and Labels.....	79
Quick Lesson 7a: Arranging Controls on the Form	80
Quick Lesson 8a: Creating a Calculated Field and Button.....	82
Quick Lesson 9a: Creating a Checkbox	84
Quick Lesson 10a: Creating a Drop-Down List	85
Quick Lesson 11a: Creating Radio Buttons	87
Quick Lesson 12a: Creating a Note Box	89
Quick Lesson 13a: Creating Command Buttons to Move to Different Regions of the Form.	90
Quick Lesson 14a: Creating a Timestamp Button.....	92
Quick Lesson 15a: Creating a Title for the Application	93
Quick Lesson 16a: Specifying an Initial Sort/Search Order	94
Quick Lesson 17a: Specifying the Tab and Column Order	95
Quick Lesson 18a: Downloading Your Application	97
Quick Lesson 19a: Testing Your Application.....	98
Quick Tour III: Sharing Data with Desktop Database Applications.....	99
Overview of Quick Tour III	99
Quick Lesson 1: Downloading a Table to Create a New Application.....	100
Quick Lesson 2: Looking at the Synchronization Options.....	103
Quick Tour IV: Getting Acquainted with Macros and Events	106
Overview of Quick Tour IV	106
Quick Lesson 1: Exploring the Sample Application.....	107
Quick Lesson 2: Creating the Form.....	108
Quick Lesson 3: Planning the Macro.....	109
Quick Lesson 4: Opening the Macro Builder	110
Quick Lesson 5: Writing a Show/Hide Control Command.....	111
Quick Lesson 6: Adding a Skip Command with an "If" Expression.....	113
Quick Lesson 7: Attaching the Macro to a Control Event.....	115
Quick Tour V: Using the Macro Debugger.....	116
Overview of Quick Tour V	116
Quick Lesson 1: Testing the Sample Application	117
Quick Lesson 2: Opening the Macro in the Debugger	118
Quick Lesson 3: Inserting Breakpoints	120
Quick Lesson 4: Running the Macro with the Debugger.....	121
PART II: CREATING YOUR OWN VISUAL CE APPLICATIONS	123
Overview of Part II	124
Visual CE Development Environment	126
Starting Visual CE's Development Environment	126
Changing the Device Screen Size.....	128
File Menu Commands	129
Edit Menu Commands	130
Form Menu Commands.....	131
Control Menu Commands.....	132
Development Window Toolbar Buttons	133
Development Window Shortcut Keys	134

Exiting Visual CE's Development Environment	135
Getting Started	136
Steps in Creating an Application.....	136
Tips for Planning Your Application.....	137
Creating and Managing Tables.....	138
About Creating Tables	138
Creating a Table Using Visual CE	139
Cloning a Table	142
Creating a Table with Pocket Access	143
Creating a Table with SQL Server CE	144
Copying an Access Database to the CE Device.....	145
Downloading a Table from an ODBC-Enabled Application on the Desktop PC.....	146
Adding, Deleting, or Modifying Columns in an Existing Table	149
Deleting Tables	151
Changing a Form's Table.....	152
Using Global Variables.....	153
About Using Global Variables to Store Temporary Values.....	153
What Controls and Form Features Use Global Variables.....	153
Beginning a New Form	155
Specifying the Table to be Used by a Form.....	155
Creating a Form Without a Table.....	157
Specifying the Form Size	158
Hiding Scroll Bars in the Development Window (Handheld Version of Development Environment Only)	159
Specifying Form Background Color	160
Specifying Color for List Mode Text.....	161
Adding Custom Colors to the Color Palette	162
Customizing or Removing the Title Bar and Start Flag.....	163
Adding Tabs	164
Adding Controls to the Form.....	166
About Controls	166
Creating Edit Boxes	167
Edit Box Input Masks	170
Edit Box Default Values	171
Text Edit Controls.....	171
Numeric Edit Controls (Integer or Floating Point Number)	171
Date/Time Edit Controls	172
Edit Box Money Values	173
Edit Box Date/Time Formats	174
Using Min/Max to Make an Edit Box Required.....	175
Edit Box Word Lists	176
Creating Note Boxes	177
Creating Labels	179
Creating Checkboxes.....	180
Creating Radio Buttons.....	182
Creating Drop-Down Lists.....	184
Using Values from Another Table as a Drop-Down List.....	187
Creating Dependent Drop-Down Lists.....	189
Multi-Tiered Dependent Drop-Down Lists	192
Creating Navigational Drop-Down Lists	199
Creating Calculated Fields.....	201
About Constants, Column Names, Functions, and Operators.....	201
About IF-THEN-ELSE	202
Using Date/Time Values in a Calculated Field Expression	205

Using If-Then-Else in a Calculated Field.....	206
Using Basic Functions in a Calculated Field Expression.....	207
Using Advanced Functions in a Calculated Field	208
Text Functions	208
Numeric Functions.....	209
Date/Time Functions	211
Global Variable Function	211
GPS Functions	212
System Functions	213
Creating Command Buttons	214
About Button Macros	214
Command Buttons, Organized by Category.....	216
Creating Command Buttons as Shortcuts for Menu Picks.....	219
Creating Command Buttons to Search Records.....	220
Creating Command Buttons to Import/Export Text or XML Files.....	221
Creating Command Buttons to Move to Another Region of the Form	223
Creating Command Buttons to Run Macros	225
Creating Command Buttons to Start External Programs	226
Creating Command Buttons to Open Report CE	228
Creating Command Buttons to Sort Records	229
Creating Command Buttons to Send SMS Messages.....	230
Creating Command Buttons to Dial a Phone Number.....	231
Creating Command Buttons to Set Alarms	232
Creating Command Buttons to Control mEnable Connections.....	234
Creating Command Buttons to Put or Get Files via mEnable.....	235
Creating Scribble Boxes.....	237
Creating Timestamps	239
Creating Image Controls.....	241
Creating Image Controls to Take Pictures with Digital Cameras.....	244
Creating Image Controls to Display Web Pages	246
Creating a Jump to Another Table	248
Creating an AutoNumber Control	252
Creating a Lookup	254
Creating a Grid Control.....	257
How the Grid Works When Displaying Related Records from a Different Table	257
How the Grid Works When Displaying Records from the Current Table	257
Creating an mEnable Stoplight Control	262
Modifying Control Properties.....	264
Changing Control Properties	264
Specifying Font Properties for a Control	265
Fixing a Control's Position on the Screen	266
Changing Properties for Multiple Controls.....	267
Arranging Controls on the Form.....	268
Turning Off Snap-to-Grid	268
Aligning and Sizing Controls on the Form	269
Applying Cut, Copy, and Paste to Controls	270
Deleting Controls	271
Customizing the Form.....	272
Customizing or Removing the Menu Bar.....	272
Specifying Control Order	276
Preventing Duplicate Values in a Column	278
Enabling the Auto Recalc Feature.....	279
Enabling Bar Code Scanners	280
Creating Text for the "About Box".....	281

Defining Record Sorts and Searches	282
Specifying an Initial Sort/Search Ordering.....	282
Indexing Columns	283
Creating Pre-defined Filters for Record Display	284
Security	286
Password-Protecting an Application	286
Supplying Passwords for an SQL Server CE Database Table.....	287
Allowing Users to Insert and/or Delete Records	288
Locking the Form Design	289
Downloading, Testing, and Managing Applications.....	290
Saving an Application	290
Saving a Textual Description of the Form.....	291
Downloading an Application (Desktop PC Version Only)	292
Testing the Application.....	293
Opening an Existing Application	294
Modifying an Existing Application	295
Setting Up Visual CE Applications to Run on a Desktop PC.....	296
Distributing Applications	298
Creating Distribution Files (Pro and Mobile 360 Editions Only).....	298
PART III: MACROS AND EVENTS.....	301
Overview of Part III	302
Macros and Events: An Overview	303
Introducing Macros and Events	303
A Sample Macro Script	306
Steps in Creating a Macro	307
Using the Macro Builder.....	309
Starting the Macro Builder	309
Creating a New Macro	312
Applying Cut, Copy, and Paste to Macro Commands	314
Editing an Existing Macro	315
Deleting an Existing Macro	316
Converting a Legacy Button Macro to a Buttonless Macro	317
Creating a Way for Users to Run the Macro.....	319
Triggering Macros with Events.....	320
About Events	320
Using Control Events to Trigger a Macro.....	321
Using Form Events to Trigger a Macro	322
Command Reference	323
Commands by Category	323
Commands Organized Alphabetically.....	328
Using the Macro Debugger	370
Overview	370
Starting the Macro Debugger.....	371
Setting and Removing Breakpoints	373
Running a Macro in the Debugger.....	375
Closing the Debugger	376
PART IV: USING VISUAL CE APPLICATIONS	377
Overview of Part IV	378
Visual CE Application Window	379
Starting Visual CE Applications	379

Record Menu Commands.....	381
Edit Menu Commands	383
Option Menu Commands.....	384
Application Window Toolbar Buttons.....	385
Application Window Shortcut Keys.....	386
Exiting Visual CE Applications	388
Viewing Records.....	389
Scrolling Through Records	389
Displaying a Record List.....	390
Switching from List Display to Single Record Display	391
Sorting and Finding Records	392
Searching for Specific Records in the Table	392
Changing Sort/Search Order	394
Filtering the Record Display	395
Adding, Editing, and Deleting Records.....	397
Creating New Records	397
Creating a New Record by Duplicating an Existing Record	398
Editing Records	399
Saving Records	400
Deleting Records	401
Entering Data into Records	402
Using Scribble Boxes	402
Using Image Controls	403
Using Image Controls to Take Pictures with a Digital Camera	404
Entering Null Values in a Date/Time or Numeric Field.....	405
Entering Negative Numbers and Decimal Points (Non-Touchscreen Devices)	406
Using Word Lists in an Edit or Note Box	407
Copying or Moving Data Between Records	408
Using a Bar Code Scanner with Visual CE	409
Using Alarms	410
Setting and Responding to Alarms.....	410
Printing Records.....	412
Printing a Record	412
Using Report CE to Print a Visual CE Record.....	413
Exporting and Importing.....	414
Export Table	414
Import an ASCII File	415
Clear Table	416
Beaming Records from One CE Device to Another	417
Visual CE and mEnable.....	418
Using Visual CE Applications with mEnable	418
Running Visual CE from Another Application.....	421
Running a Visual CE Application from a Visual Basic or C/C++ Application	421
PART V: DESKTOP CONNECTIVITY	423
Overview of Part V.....	424
Downloading a Desktop PC Table to a Windows CE Table.....	425
Downloading Data from an Excel Spreadsheet to a Windows CE Table.....	426
Importing a Windows CE Table into Microsoft Excel	427
Importing a Windows CE Table Into Microsoft Access	428
Linking a Windows CE Table with Microsoft Access.....	429
Exporting a Windows CE Table to an ASCII Text or XML File.....	430

Importing an ASCII Text or XML File into a Windows CE Table	431
PART VI: SYNCHRONIZING TABLES.....	433
Overview of Part VI	434
Selecting Table Synchronization Options	435
Custom Synchronization Option	438
Using Timestamp-Based Synchronization.....	439
Synchronizing Multiple Handhelds (Pro and Mobile 360 Editions Only).....	440
Synchronizing Tables On Command Using Visual CE	441
Synchronizing Tables Using Vicesync.....	442
Synchronizing Tables Using mEnable	443
Synchronizing Tables on the CE Device with Excel Spreadsheets on the Desktop PC	444
Index.....	445

Copyright © 2010 SYWARE, Inc. All rights reserved.

SYWARE, Visual CE, Report CE, mEnable, and FoneDB are registered trademarks and sqlceEnable and Mobile 360 are trademarks of SYWARE, Inc. Other brands and their products are trademarks or registered trademarks of their respective holders.

Introduction

Overview of Visual CE

Welcome to Visual CE, the fastest and easiest way to create forms-based applications for Windows CE devices. Visual CE form applications allow you to use your CE device to collect, display, and modify data. You can also transfer data between Visual CE applications and your favorite ODBC-enabled applications on the desktop PC.

Typically, you will create forms on your desktop PC, and then download them to one or more CE devices. (You can also create forms directly on CE devices with touchscreens.) Each Visual CE form is actually a compact application associated with a Windows CE, Pocket Access, or SQL Server CE table on the CE device. The table can be empty initially, or you can initialize it with data from another table or ODBC-enabled database.

You have the option of using a form to collect and modify data on a desktop PC as well as a CE device. When you use the form on a desktop PC, it reads and writes data to a desktop database that is synchronized with the form's table on the CE device.

Visual CE provides 16 types of controls for use on your forms: labels, edit boxes, note boxes, checkboxes, drop-down lists, radio buttons, calculated fields, command buttons, scribble boxes, timestamps, image controls, jump buttons, autoNumbers, lookups, grid controls, and mEnable stoplights.

Visual CE's development environment is simple to use—you specify each control you need and define its properties, including associated table column or global variable, read-only or read/write, font, etc. You can drag and size the controls to create the form's layout, which appears in the development window exactly as it will display on the CE device.

Visual CE supports all devices running Windows Mobile (Classic, Standard, Professional, and Embedded) or Windows CE 3.0 or later, including PocketPCs (P/PCs), H/PC Professional devices (H/PCs), and Smartphones.

The figure consists of two side-by-side screenshots of the "Expense Tracker" application. Both screenshots show the same data entry form. The left screenshot is for a Windows CE device with a touchscreen, showing larger, rounded input fields and a larger notes area. The right screenshot is for a smartphone, showing smaller, more compact input fields and a smaller notes area. Both forms include fields for Date (8/20/00 and 2/5/2008 respectively), Description, Amount (0 and 0.00 respectively), and a radio button group for category (Travel, Food, Entertainment, Other, with Travel selected). They also include fields for Tip (15% Tip and 0.00 respectively), Cost Center (Sales), and a Notes section with a scrollable text area. At the bottom, both screens show standard mobile navigation buttons: Record, Edit, Option, and a set of icons for back, forward, search, and calendar.

Visual CE enables you to create form-based applications for CE devices with touchscreens (example above left) and for Smartphones without a touchscreen (example above right)

Visual CE Editions

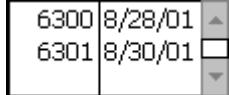
Visual CE is available in three editions:

- **Personal Edition** enables you to create your own Visual CE applications and put them to work using CE devices or desktop PCs. It also gives you the ability to synchronize data between Visual CE applications on your CE device and ODBC-enabled databases on your desktop PC; and it works with **mEnable**, a separate SYWARE product that makes it possible for Visual CE applications to read and write data on remote servers—interactively and in real time—via wired or wireless connections including the Internet.
- **Pro Edition** is designed for developers and others who wish to distribute working versions of Visual CE applications to users who do not have Visual CE. Otherwise, the Personal and Pro Editions are the same.
- **Mobile 360** combines the Pro Editions of Visual CE, **Report CE**, and mEnable. SYWARE's Report CE enables users to design and print customized reports of data collected with Visual CE.

Visual CE Controls

The 16 control types offered by Visual CE are:

Control Type	Description	Example
Label	Text that the user cannot modify. A label can be used to identify other controls, such as edit boxes and note boxes, that do not have their own captions.	Date
Edit box	A single-line text box. An edit box can display the value of a table column or global variable, and/or accept the user's entry of a value for a column or variable.	0.00
Note box	A multi-line text box. A note box can display the value of a table column or global variable, and/or accept the user's entry of a value for a column or variable.	
Checkbox	A choice that the user can select or deselect like a switch. An "X" appears in the checkbox to show a choice is selected.	<input type="checkbox"/> Have Receipt?
Drop-down list	A scrollable list of choices from which the user can choose one or more options or enter new options.	Sales ▾
Radio buttons	A set of choices from which the user can select a single option.	<input checked="" type="radio"/> Travel <input type="radio"/> Food
Calculated field	The result of an expression and one or more field values. Operands may be numeric, text, or date/time.	1.27
Command button	A button that the user clicks to initiate an action, such as move to a different region of the form or start another application.	* Expense
Scribble box	A box in which the user can draw an image freehand using the stylus or mouse (e.g., a simple sketch or signature).	OK!
Timestamp button	A button that the user presses to "stamp" the record with a date and time.	6/18/98 7:36:36 PM
Image control	A box that displays images. The images can be pictures (.bmp, .jpg, or .gif) or even a live web page.	
	An image control can display a picture stored in a table column and/or accept the user's input of a picture to a table column. If the CE device is connected to a digital camera, the image control can	

	be used as a viewfinder and the photos taken automatically stored in the associated table.	
	You can also use image controls to embed a static image, such as a company logo, in the form design.	
Jump button	A button that opens a Visual CE application containing additional information about the current record.	
AutoNumber	A unique, sequential number that Visual CE automatically assigns to each new record (the newest record has the highest number).	1
Lookup	A value that Visual CE reads from a column in a related table and displays in the current form.	Pen
Grid control	Combines the functions of a lookup control and a jump button. It displays rows from a related table and, when a row is double-clicked, opens a second Visual CE application displaying that record. Can also be used to display all records from the current table.	
mEnable stoplight	A button resembling a stoplight that displays and controls the status of an mEnable connection between Visual CE and a remote database server.	

About this Documentation

This documentation covers all editions of Visual CE: Personal, Pro, and Mobile 360.

How To Use This Documentation

Once Visual CE is installed, start with Quick Tour I to see how a sample Visual CE application works on a CE device with a touchscreen. If you have a CE device without a touchscreen (e.g., a Smartphone), take Quick Tour Ia instead.

Next, the Quick Tours show you how to create the sample application using Visual CE on your desktop PC. Take Quick Tour II to create the application for touchscreen devices; take Quick Tour IIa to create the application for non-touchscreen devices.

Take Quick Tour III to learn how to synchronize a Windows CE table with a Microsoft Access database on the desktop PC.

Take Quick Tour IV to get an introduction to macros and events, and Quick Tour V to learn how to debug a macro.

Finally, create and use your own applications using the complete reference information provided for designing and using Visual CE applications.

Installation and Support

System Requirements

Visual CE requires:

- A desktop PC:
 - ◊ IBM-compatible, 80486 processor or higher
 - ◊ Windows 95 or Windows NT 4.0 or higher
 - ◊ At least 8MB RAM (16MB recommended)
 - ◊ 20MB free disk space
 - ◊ Hard disk and mouse
- A Windows CE device (including Smartphones), with Windows Mobile (Classic, Standard, Professional, or Embedded) or Windows CE 3.0 or later

Note

Use of SQL Server CE tables with Visual CE requires SYWARE's `sqlceEnable`, a separate software product.

Installing Visual CE

You install Visual CE on both your desktop PC and CE device at the same time.

Note

Before you begin, be sure to read the README.TXT file.

To install Visual CE:

- 1 Connect your CE device to your desktop PC.
- 2 Insert the Visual CE setup CD in your desktop PC.
- 3 Visual CE prompts you through the installation process.
Follow the instructions on the screen.

About the Sample Applications

The installation program automatically places several sample Visual CE applications on your desktop PC. These samples illustrate many interesting and useful features you may want to incorporate in your own designs. To explore these applications, you must download them to your CE device.

The samples must be run on the CE device, not the desktop PC.

To download the sample applications to the CE device:

- 1 Connect your CE device to your desktop PC.
- 2 On the desktop PC, start Visual CE by clicking **Start-Programs-Visual CE-Visual CE**.
- 3 Select **File-Open**.
- 4 Navigate to the folder in which you installed Visual CE (by default, it's \Program Files\Windows CE Services\SYWARE Visual CE).
- 5 Open the .VCE file you wish to download and explore.
- 6 Select **File-Download Form**.
Visual CE displays the Save dialog box.
- 7 Navigate to the root (top level) directory on the CE device and click **OK**.

Normally, you will save forms in the default location, which is the Microsoft ActiveSync synchronize folder. The root folder is recommended in this one unique case to ensure that all of the samples stay together. In that way, the jumps and other inter-relationships they are intended to illustrate will work properly.

Once you have downloaded a sample application, run it by clicking **Start-Programs-[programName]** on the CE device.

Technical Support

For technical support, contact SYWARE, Inc. at <http://www.syware.com/support>.

In addition, you may wish to visit our web site at www.syware.com. It offers a wealth of information including free software utilities you can download, the latest product development news, and more. Two links are of particular interest for Visual CE users and developers:

- **www.syware.com/faq**

The "Visual CE FAQ" page provides answers to the most frequently asked questions regarding installation, downloading and synchronization, general usage, and other important topics.

- **www.syware.com/totm**

The "Tip of the Month Archive" is a library of practical and creative design ideas from the field, many of them submitted by our users. You may discover the solution to a particular design issue, or find inspiration for a new feature to incorporate in your own Visual CE applications.

Visual CE Add-on Tools

SYWARE provides, free of charge, a collection of software utilities for protecting your Visual CE applications and data, as well as for cleaning your desktop PC and CE device of unwanted files. They are:

- **Rebuild:** Rebuilds a table definition given a form
- **Backup:** Backs up a table into an archival file or restores a table from an archival file
- **Expunge:** Expunges a form from the device, including its shortcut, the registry entries, and (if you choose) its table
- **Clean:** Removes extraneous table definitions
- **Index CDB:** Adds/removes indexes in a .cdb file
- **SSCE Utility:** Allows you to enter and execute SQL commands on an SQL Server CE database. You can use it to add/delete indexes in an SQL Server CE table.

To obtain any of these tools, go to **Support | Downloads** at <http://www.syware.com> and download the **Visual CE Add-on Tools**.

Part I:

Quick Tours

Quick Tour I: Using a Visual CE Application on a Touchscreen Device

Overview of Quick Tour I

In this first quick tour, you will learn how to use a Visual CE application on a CE device with a touchscreen using the sample application provided with Visual CE. In just ten minutes use of the sample application, you will become familiar with many of Visual CE's application features.

Quick Tour I includes the following lessons:

- Quick Lesson 1: Running a Visual CE application on the CE device
- Quick Lesson 2: Creating records
- Quick Lesson 3: Displaying a record list
- Quick Lesson 4: Filtering a record list
- Quick Lesson 5: Searching for records
- Quick Lesson 6: Using command buttons
- Quick Lesson 7: Exiting a Visual CE application

Note

If you have a Smartphone without a touchscreen, take Quick Tour Ia instead.

Quick Lesson 1: Running a Visual CE Application on the CE Device

In this lesson, you will start a Visual CE application on the CE device and get familiar with Visual CE's application window.

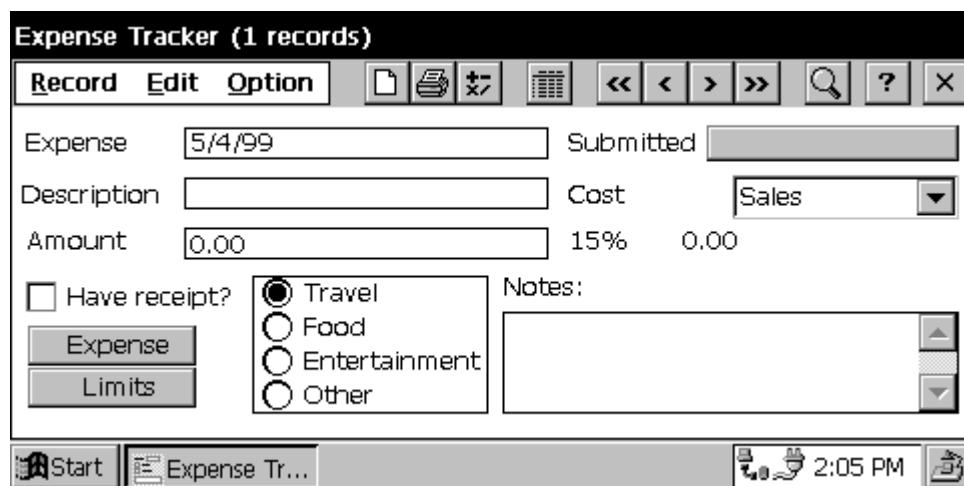
Note

You can run Visual CE applications on either desktop PCs or CE devices. For the purposes of this tutorial, you'll use the CE device.

- To run the sample Visual CE application, click the Windows **Start** button, then select **Programs**, then **Expenses**.

Visual CE displays the EXPENSES application form in its application window.

The illustrations show the horizontal screen format used on H/PC devices, but the procedures for using Visual CE applications are the same on all CE devices with a touchscreen.



The default Visual CE application window provides three menus – Record, Edit, and Option. You can select menu options by clicking them with the stylus or mouse pointer, or by pressing the shortcut keys on your keyboard.

Visual CE toolbar buttons offer a faster way to activate the most frequently-used commands listed on the menus.

Next: Creating Records

Quick Lesson 2: Creating Records

As you create a few records, you will quickly learn the fundamentals of using Visual CE on the CE device.

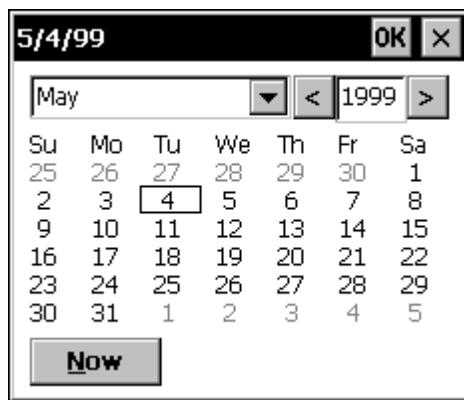
Note

When a Visual CE application is running, it is always “on” a record. When you start an application for the very first time, no user-created records exist, so Visual CE creates a new one to display using the default values supplied by the form designer.

The next time you open the application, Visual CE will initially display the record that you were on at the time you closed the application. To avoid overwriting your work in that case, you must choose **Record–Insert** to start a new one.

- 1 In the **Expense** date field, single-click the date shown.

Visual CE displays a calendar dialog box.



- 2 Enter another date:

- Click the month (from the drop-down list)
- Enter the year, or click the right- and left-arrows to increase/decrease the year
- Click the date in the calendar area
- To change back to today's date, click **Now**
- To confirm your entries, click **OK**

- 3 In the **Description** field, enter **Dinner**.

- 4 In the **Amount** field, single-click the **0.00** value.

Visual CE displays a number pad dialog box



- 5 Enter an amount:

- Click the number buttons to enter an amount, such as 49.55 – the number pad includes a decimal point for money entries. The number you enter displays in the title bar of the dialog box.
- (If you make a mistake, click the **Clear** button, then enter the number again.)
- To confirm your entries, click **OK**

Tip about entering numbers: If you prefer, you can type an amount directly into a number field rather than clicking the number pad. To do so, press the **Tab** key until the number field is highlighted, then type the number.

- 6 This sample application includes a control that can calculate a 15% tip on the amount. To make the calculation, click **Option-Recalculate**.
- 7 Click the **Have Receipt?** checkbox to indicate that you do have a receipt.
- 8 Click the **Entertainment** radio button.
- 9 In the **Cost Center** drop-down list, leave the default (**Sales**) selected.
- 10 In the **Notes** field, enter a brief explanation such as **Client of the Month: Sandy Taylor, SMC Corp.**
- 11 **P/PC users only:** Click the **Limits** button to bring the second region of the form into view.
- 12 Click the **Submitted** button. The current date and time appears. This "timestamp" will be saved with the record.
- 13 **P/PC users only:** Click the **Expense** button to bring the first region of the form back into view.

Now that you have updated the first record, you are ready to save the updates. As you will see, Visual CE automatically saves a record when you insert a new record, move to another record, choose Save, or exit the application.

- 14 Create a new record by selecting **Record-Insert** or clicking the **Insert** button .
- Visual CE displays a new record with default values and, as just explained, saves the record you just updated.
- 15 Make entries or accept the default values in each of the controls.

Control Name	Control Type	Entry	Default
Expense Date	Edit box	Accept today's date or single-click to enter another date	Today's date
Description	Edit box	Enter a short description of the expense	None
Amount	Edit box	Single-click to enter an amount using the number pad or keyboard	0.00
15% Tip	Calculated field	No user entry – Visual CE calculates the value automatically	15% of the value in the Amount field
Have Receipt?	Checkbox	Accept the default (no receipt) or click to indicate that you have a receipt	Not checked
Travel/Food/ Entertainment/Other	Radio button	Click the button that corresponds to the expense category	Travel
Cost Center	Drop-down list	Click the down-arrow to display the drop-down list, then click an option to choose it	Sales
Notes	Note box	Optional – Enter a multi-line description of the expense	None
Submitted	Timestamp	Click button	None

16 Create another record, but this time start by copying data from a previous one:

- Go to the record you want to use as the basis for the new one. To scroll between records, use the **Record-Previous** or **Record-Next** menu commands, or **Ctl+Up Arrow** or **Ctl+Down Arrow** on the keypad, or the VCR buttons (Previous Record , Next Record , etc.) if available on your device. On the P/PC, you can alternatively use the Up/Down switch (down for next record, up for previous).
- While pressing the Shift key, select **Record-Insert** or click the **Insert** button.
Tip for P/PC users: For an explanation of how to use the Shift key, refer to the topic, "Creating a New Record by Duplicating an Existing Record" on page 398.
- Modify the data as desired for your new record.
- Select **Record-Save**.

Tip

You can also copy or move data from one record to another by using the Edit menu's Cut, Copy, and Paste commands.

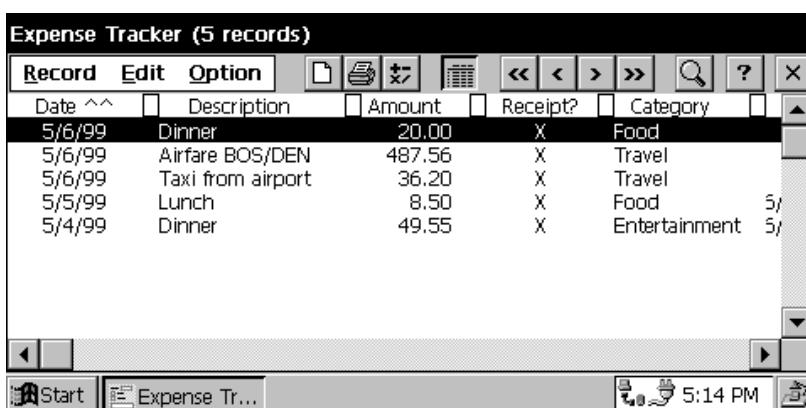
Next: Displaying a Record List

Quick Lesson 3: Displaying a Record List

Normally, Visual CE displays one record at a time in the application window. If you want to display several records at once, use the record list feature as described below:

- 1 Select **Option–List** or click the **List** button  to "turn on" the record list option.

Visual CE displays a list of all records in the Expenses table.



- 2 Adjust the column widths by dragging the squares between the column headings.
- 3 To enlarge the text in the display for easier reading, select **Option-Zoom**. To return to the original size, select **Option-Zoom** again.
- 4 The records are currently sorted by Date (indicated by the $\wedge\wedge$ symbol in the Date column heading). You can sort the records using a different column by clicking on its heading. For example, click **Description**. (If Visual CE asks whether you want to index the column, click **Yes**. Only indexed columns can be used for sorting.) Note that the $\wedge\wedge$ symbol moves next to the Description column heading.
- 5 When the list is sorted on a text column such as Description, you can quickly locate records by initial letters. For example, to find the first record in the list whose description begins with **ai**, simply press the letters **ai** on your keyboard. Assuming that the list is still sorted on Description and that your table includes such a record, Visual CE scrolls the list so that the first record whose description begins with **ai** ("Airfare BOS/DEN" in this example) is at the top of the display.
- 6 Return the list to its original sort order by clicking on the **Date** column heading. Notice that the sort was quicker since the Date column was already indexed. Indexed columns are indicated by the \wedge symbol next to their heading.
- 7 You can edit data in the display. Tap and hold on the value you wish to change.

Notes

- If the table is a Windows CE table (as is the sample), Visual CE automatically indexes a column when you select it for sorting/searching. Up to four columns can be indexed at any one time. When you select a fifth, Visual CE asks which one of the current selections you want to replace.
- As you save new records, Visual CE automatically lists them in the currently-selected sort order. (Saving occurs when you insert a new record, scroll to another record, exit the application, or choose Save.) For example, if the Date column is the current sort/search order, records are automatically ordered from those with the most recent date to those with the oldest date.

Next: Filtering a Record List

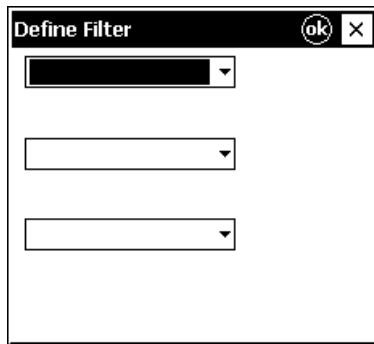
Quick Lesson 4: Filtering the Record List

After you have used your application for a time, your list of records may become very long. To focus on records of interest, you can temporarily "filter" a list to display only records that match specific criteria.

In this lesson, you will ask Visual CE to find records that exceed \$10.00 and contain the word "dinner" in any text column.

1 Select **Option–Filter–Custom**.

Visual CE displays the Define Filter dialog box.

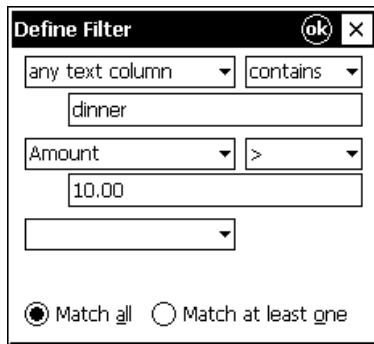


2 Specify the first condition:

- Click the drop-down arrow for the first field in the dialog box, and select the item, **any text column**.
- Leave the default **contains** in place for the comparison operator.
- In the value field (below "any text column"), type **dinner**.

3 Specify the second condition:

- Click the drop-down arrow for the second column name field, and select **Amount**.
- Click the drop-down arrow for the comparison operator field, and select **>**.
- Click in the value field, and use the number pad to enter **10.00**.



4 Since you want to see records that match both conditions, leave the default value **Match all** in place.

5 Click **OK** to close the Define Filter dialog box.

Visual CE displays the filtered list.

6 Select **Option–Filter–Off** to turn off the filter and redisplay all records in the table.

7 To return to the single-record display, select **Option–List** or click the **List** button again.

Next: Searching for Records in the Table

Quick Lesson 5: Searching for Records

You can quickly locate records by values in a table column. In this lesson, you'll search for records with a particular date.

Before you can apply the Search function, you must sort the records according to the column of interest, as described below.

1 Select the "Date" column as the basis for the search:

- Click the **List** button 

- If the "Date" column is not already selected as the "sort/search order" (indicated by the ^ symbol), click on its heading

Visual CE sorts the records by date, from latest to earliest.

2 (Optional) Return to the single record display by clicking the **List** button.

3 Select **Record–Search** or click the **Search** button .

Visual CE displays the calendar dialog box.

4 Click the month, day, and year of the record you want, then click **OK**.

Visual CE displays the record whose date matches your entry.

Next: Using Command Buttons

Quick Lesson 6: Using Command Buttons

A form may be larger than the screen size of the CE device. Command buttons allow you to display different regions of your form without having to scroll. There are two regions in this sample application: one to enter expense information and one to show expense limits.

If you press the **Expense** button, Visual CE displays the region of the form that allows you to enter expense information.

The screenshot shows the "Expense Tracker (5 records)" application. The title bar has "Record", "Edit", and "Option" menus. Below the menu is a toolbar with icons for file operations like Open, Save, Print, and Find. The main area contains fields for "Expense" (5/4/99), "Submitted" (5/6/99 4:59:07 PM), "Description" (Dinner), "Cost" (Sales dropdown), "Amount" (49.55), and "15%". A checked checkbox "Have receipt?" is followed by radio buttons for "Travel", "Food", "Entertainment" (selected), and "Other". To the right is a "Notes" section containing "Client of the Month: Sandy Taylor, SMC Corp". At the bottom left are buttons for "Expense" and "*Limits". The bottom status bar shows "Start", "Visual CE", "Expense Tr...", the date/time "5:35 PM", and a battery icon.

If you press the **Limits** button, Visual CE displays the region of the form that displays expense limits.

The screenshot shows the same "Expense Tracker (5 records)" application. The "Expense" button is highlighted. The main area displays a list of expense limits: Breakfast (8.00), Lunch (10.00), Dinner (35.00), Hotel (150.00), Car rental (daily) (30.00), and Entertainment (including client dinner) (50.00). The bottom status bar shows "Start", "Visual CE", "Expense Tr...", the date/time "5:35 PM", and a battery icon.

Next: Exiting the Visual CE Application

Quick Lesson 7: Exiting a Visual CE Application

To leave the Expense Tracker application, choose **Record–Close** or click the **Exit** button



Quick Tour Ia: Using a Visual CE Application on a Smartphone

Overview of Quick Tour Ia

In this quick tour, you will learn how to use a Visual CE application on a Smartphone using the sample application provided with Visual CE. In just ten minutes use of the sample application, you will become familiar with many of Visual CE's application features on a non-touchscreen device.

Quick Tour Ia includes the following lessons:

- Quick Lesson 1a: Running a Visual CE application on the Smartphone
- Quick Lesson 2a: Creating records
- Quick Lesson 3a: Displaying a record list
- Quick Lesson 4a: Searching for records
- Quick Lesson 5a: Using command buttons to move to different regions of the form
- Quick Lesson 6a: Exiting a Visual CE application

Note

If you have a CE device with a touchscreen, take Quick Tour I instead.

Quick Lesson 1a: Running a Visual CE Application on a Smartphone

In this lesson, you will start a Visual CE application on the Smartphone.

- On the Home screen, click **Start–Expense**.

Visual CE displays the EXPENSE application form in its application window.

The screenshot shows the "Expense Tracker" application window. At the top, it says "Expense Tracker (1 record)" with a delete icon. The main area contains the following fields:

Date	2 / 5 /2008
Description	[empty]
Amount	0.00
Tip	0.00
Cost Center	Sales
<input checked="" type="radio"/> Travel	Notes:
<input type="radio"/> Food	[empty]
<input type="radio"/> Other	
Expense	Limits
Dial	Record

Next: Creating Records

Quick Lesson 2a: Creating Records

As you create a few records, you will quickly learn the fundamentals of using Visual CE on the Smartphone.

Note

When a Visual CE application is running, it is always “on” a record. When you start an application for the very first time, no user-created records exist, so Visual CE creates a new one to display using the default values supplied by the form designer.

The next time you open the application, Visual CE will initially display the record that you were on at the time you closed the application. To avoid overwriting your work in that case, you must choose **Record–Insert** to start a new one.

The illustration on the left shows a record with the default values for the sample application; the illustration on the right shows a completed record.

Expense Tracker (1 record)	
Date	5 / 5 /2008
Description	
Amount	0.00
Tip	0.00
Receipt?	<input type="checkbox"/>
Cost Center	Sales
Travel	<input checked="" type="radio"/>
Food	<input type="radio"/>
Other	<input type="radio"/>
Notes:	
Expense	Limits
Dial	Record

Expense Tracker (1 record)	
Date	5 / 5 /2008
Description	dinner
Amount	34.00
Tip	5.10
Receipt?	<input checked="" type="checkbox"/>
Cost Center	Production
Travel	<input type="radio"/>
Food	<input type="radio"/>
Other	<input checked="" type="radio"/>
Notes:	abc co rep
Expense	Limits
Dial	Record

Tip

Use the up/down rocker to move the cursor from field to field. Use the right/left rocker to move the cursor inside a field. Press the Action key (a.k.a. center joystick) to enter a selection.

1 When the form first opens, input focus is set on the Date button. Use the down rocker to set focus on the date value.

2 Leave the default date in place, or enter a new date.

3 Press the down rocker twice to set the focus on the Description value, and enter **dinner**.

4 Set focus in the **Amount** field, and enter **34.00**.

Tip about entering numbers in a number field: Use the * key to enter a minus sign and the # key for a decimal point.

5 This sample application includes a control that can calculate a 15% tip on the amount. Set focus on the **Tip** button and press the Action key to calculate and display the tip.

6 Set focus on **Receipt?** Use either the right/left rocker or the Action key to toggle the checkbox on to indicate that you do have a receipt.

7 Set focus on the **Cost Center** drop-down. Press the Action key to display the list of options. Use the right/left rocker to select **Production**, and press the Action key.

Alternatively, with the focus on the Cost Center drop-down, and without the list of options displayed, use the right/left rocker to cycle through the list of options until **Production** is visible.

- 8 Use the up/down rocker to set focus on the group of radio buttons for Travel/Food/Other. Use the right/left rocker to select **Food**.
- 9 In the **Notes** field, enter a brief explanation of the expense.
- 10 Set focus on the **Limits** button and press the Action key to bring the second region of the form into view.
- 11 Set focus on the **Submitted** button. Press the Action key. The current date and time appears. This "timestamp" will be saved with the record.
- 12 Set focus on the **Expense** button and press the Action key to bring the first region of the form back into view.

Now that you have updated the first record, you are ready to save the updates. As you will see, Visual CE automatically saves the current record when you insert a new record, move to another record, or exit the application.

- 13 Create a second record by clicking **Record-Insert**. Visual CE displays a new record with default values and, as just explained, saves the record you just updated.
- 14 Make entries or accept the default values in each of the controls for the new record.

Control Name	Control Type	Entry	Default
Expense Date	Edit box	Accept today's date or enter another date	Today's date
Description	Edit box	Enter a short description of the expense	None
Amount	Edit box	Enter an amount using the phone keypad. Use # key to enter decimal point.	0.00
15% Tip	Calculated field	Set focus on the Tip button and press the Action key.	15% of the value in the Amount field
Have Receipt?	Checkbox	Accept the default (no receipt) or check to indicate that you have a receipt	Not checked
Cost Center	Drop-down list	Press Action key to display the drop-down list; use right/left rocker to scroll options; press Action key to select. Or use right/left rocker to cycle through list of options, stopping on the one you want	Sales
Travel/Food//Other	Radio button	Use right/left rocker to scroll options; press Action key to select option	Travel
Notes	Note box	Optional – Enter a multi-line description of the expense	None
Submitted	Timestamp	Set focus on button and press Action key	None

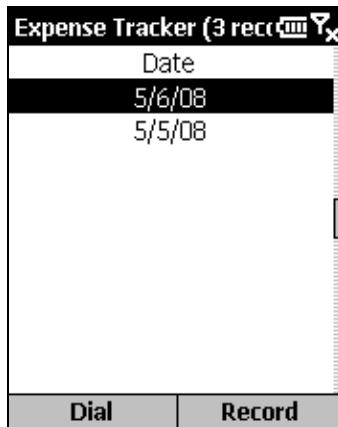
Next: Displaying a Record List

Quick Lesson 3a: Displaying a Record List

You can display a list of all records in the Expenses table.

1 Click **Record–List**.

On a Smartphone, just one table column is displayed in the list, in this case Date.



- 2 This sample application also lets you sort and list records by Description. To use this feature, return to the record view by clicking **Record–List**. Then, set focus on the **Description** button and click the Action key.



- 3 Click **Record–List** to return to list view and see the effect of your choice.
- 4 *If your device has a keyboard:* When the list is sorted on a text column such as Description, you can quickly locate records by initial letters. For example, to find the first record in the list whose description begins with **ai**, simply press the letters **ai** on your keypad. Assuming that the list is still sorted on Description and that your table includes such a record, Visual CE scrolls the list so that the first record whose description begins with **ai** ("air fare" in this example) is at the top of the display.
- 5 Click **Record–List** to return to record view.

Note

- As you save new records, Visual CE automatically lists them in the currently-selected sort order. (Saving occurs when you insert a new record, scroll to another record, exit the application, or choose Save.) For example, if the Date column is the current sort/search order, records are ordered from those with the most recent date to those with the oldest.

Next: Searching for Records in the Table

Quick Lesson 4a: Searching for Records

You can quickly locate records by values in a table column. In this lesson, you'll search for records with a particular date.

Before you can apply the Search function, you must sort the records according to the column of interest, as described below.

- 1 In record view, set the focus on the **Date** button and press the Action key.
Visual CE sorts the records by date, from latest to earliest.
- 2 Click **Record–Search**.
Visual CE displays the Move to dialog box.
- 3 Enter the month, day, and year of the record you want, then click **OK**.
Visual CE displays the first record whose date matches your entry.

Next: Using Command Buttons to Move to a Different Region of the Form

Quick Lesson 5a: Using Command Buttons to Move to Different Regions of the Form

A form may be larger than the screen size of the CE device. In this sample application, command buttons allow you to display different regions of your form without having to scroll. There are two regions in this sample application: one to enter expense information and one to show expense limits.

If you set focus on the **Expense** button and press the Action key, Visual CE displays the region of the form that allows you to enter expense information.

The screenshot shows the "Expense Tracker (3 records)" application. The current view is the expense entry region. It contains fields for Date (5/5/2008), Description (dinner), Amount (34.00), Tip (5.10), Receipt? (checked), Cost Center (Production), and Notes (abc co rep). Below these are three radio buttons: Travel, Food, and Other, with Other selected. At the bottom are two buttons: * Expense (highlighted) and Limits.

If you set focus on the **Limits** button and press the Action key, Visual CE displays the region of the form that displays expense limits.

The screenshot shows the "Expense Tracker (3 records)" application. The current view is the expense limits region. It displays a list of expense limits: Breakfast (8.00), Lunch (10.00), Dinner (35.00), Hotel (150.00), Car rental (daily) (30.00), and Entertainment (50.00). Below this is the text "Submitted" followed by the date and time (5/6/08 10:53:19 AM). At the bottom are two buttons: Expense (highlighted) and * Limits.

Next: Exiting the Visual CE Application

Quick Lesson 6a: Exiting a Visual CE Application

To leave the Expense Tracker application, click **Record–Close**.

Quick Tour II: Creating a Visual CE Application for Touchscreen Devices

Overview of Quick Tour II

You can learn the basics of developing Visual CE applications in under ten minutes. In this quick tour, you will create an application that is nearly identical to the sample Expense Tracker program that you explored in Quick Tour I.

Note

If you are creating applications for non-touchscreen Smartphones, take Quick Tour IIa instead.

Creating the application involves the following steps:

- Quick Lesson 1: Starting Visual CE's Development Environment
- Quick Lesson 2: Creating a table
- Quick Lesson 3: Selecting a table
- Quick Lesson 4: Creating an edit box and its label
- Quick Lesson 5: Creating a variety of edit boxes and labels
- Quick Lesson 6: Arranging controls on the form
- Quick Lesson 7: Creating a calculated field
- Quick Lesson 8: Creating a checkbox
- Quick Lesson 9: Creating radio buttons
- Quick Lesson 10: Creating a note box
- Quick Lesson 11: Creating a drop-down list
- Quick Lesson 12: Creating command buttons
- Quick Lesson 13: Creating a timestamp button
- Quick Lesson 14: Creating a scribble box
- Quick Lesson 15: Creating a title for the application
- Quick Lesson 16: Specifying an initial sort/search order
- Quick Lesson 17: Specifying the column order
- Quick Lesson 18: Downloading the application
- Quick Lesson 19: Testing the application

Note

To best understand the effect of your development choices, you should use the Expense Tracker application on the CE device before developing an identical program. See Quick Tour I.

Quick Lesson 1: Starting Visual CE's Development Environment

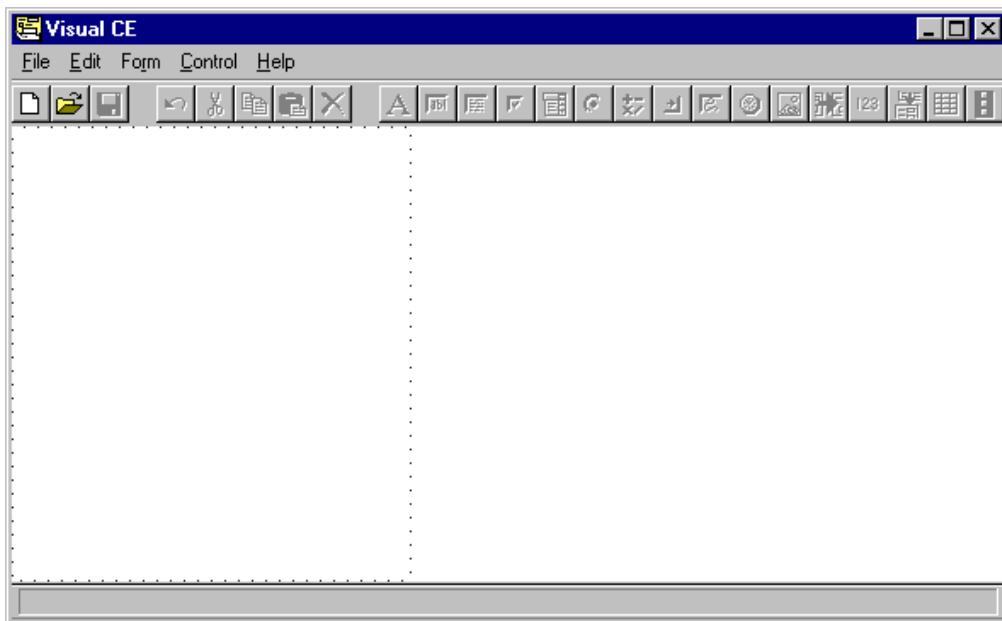
In this lesson, you will start Visual CE's development environment on the desktop PC and get acquainted with its features.

Note

You can build applications on either the desktop PC or the CE device. For the purposes of this Quick Tour, you will use Visual CE's development environment on the desktop PC.

To start Visual CE's development environment on the desktop PC:

- 1 Connect your CE device to your desktop PC.
- 2 On your desktop PC, click the Windows **Start** button, then **Programs–Visual CE–Visual CE**. Visual CE displays its development environment window.



Visual CE's development environment window has five menus: File, Edit, Form, Control, and Help. You can select menu options by clicking them with the mouse pointer or by pressing the shortcut keys.

Its toolbar buttons provide a faster way to activate the most frequently used commands.

The dotted outline in the development window indicates the screen size for the CE device. (Visual CE automatically detects which type of CE device is connected, and displays the screen size for that type.) If you have a device with a larger or smaller display, you can adjust the default using the instructions in the topic "Device Screen Size" on page 128.

Visual CE's default form size matches the screen size. You can change this to any desired size, even a size larger than the CE device screen size. (See "Form Size" on page 158 for additional information.)

The status bar in the lower-left corner (empty in the illustration above) provides helpful feedback as you build the form.

Next: Creating a Table

Quick Lesson 2: Creating a Table

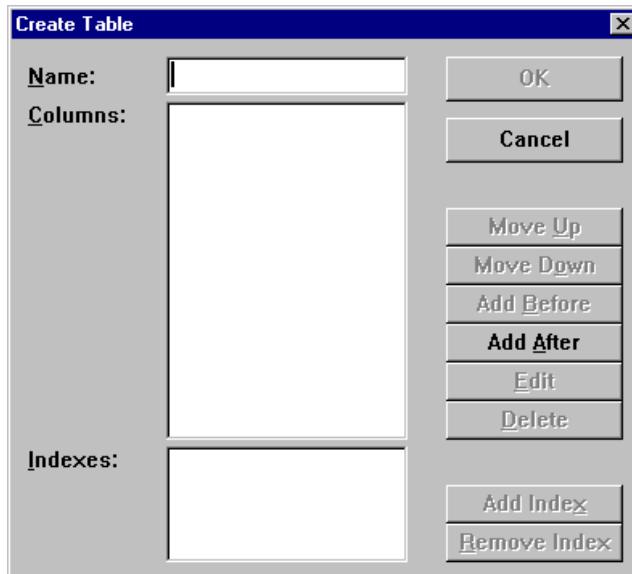
Your first step in creating a new application is to select an existing table or create a new one. For the purposes of this Quick Tour, you will create a new table that is very similar to the one belonging to the Expense Tracker.

You must create one column for each category of data that you want to collect and store in your application.

To create the sample table using Visual CE on the desktop PC:

- 1 Select **File—Create Table**.

Visual CE displays the Create Table dialog box.

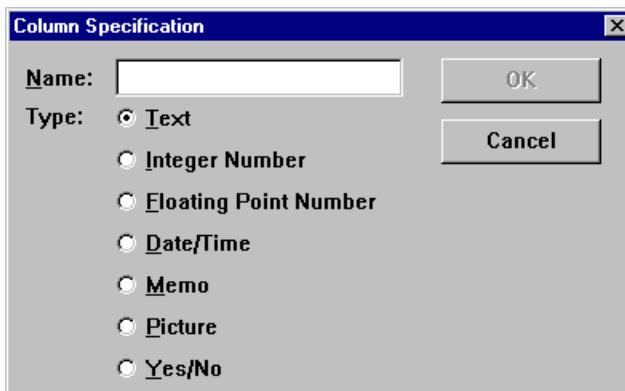


- 2 For Name, type **MyExpenses**.

- 3 Define the first column:

- Click the **Add After** button

Visual CE displays the Columns Specifications dialog box.



- For name, type **Date**

- For Type, click **Date/Time**

- Click **OK**

4 Define the next column:

- Click the **Add After** button
- For name, type **Description**
- For Type, click **Text**
- Click **OK**

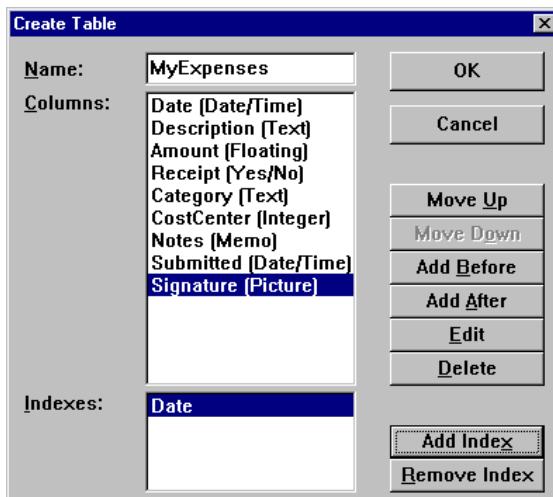
5 Define the remaining columns using the same method as described in Step 4:

Name	Type
Amount	Floating Point Number
Receipt	Yes/No
Category	Text
CostCenter	Integer Number
Notes	Memo
Submitted	Date/Time
Signature	Picture

6 "Indexing" a column enables Visual CE to sort the records in the table by values stored in that column:

- Click the **Add Index** button
- In the dialog that appears, select **Date**
- Click **OK**

Your table definition so far should look like this:



7 Click **OK** to save your table definition.

8 Visual CE asks if you would like the Form Wizard to create a form for you. (This convenient feature automatically creates edit boxes and labels for each column in the table. You can then modify and rearrange the labels and controls as desired.) However, to get the most practice now, click **No**.

Visual CE's status bar (at the bottom of the development environment window) now displays the name of your new table – indicating that the form you'll create in the following lessons will be built on that table.

You might want to take a break before starting your form design. If so, close Visual CE by selecting **File-Close**.

Next: Selecting a Table

Quick Lesson 3: Selecting a Table

About Tables and Forms

To create a Visual CE application, you typically build two components: a table and a form. A table is a structure that stores and organizes your data inside the computer. It is invisible to the user. You cannot view or modify data stored in a table—except via a form. A form is a program (.vce file) that makes it possible for you to view, add, modify, and delete data stored in the table. A form is the user's interface with the data—it's what you see on the screen.

In the previous lessons, you created a table. In this and the following lessons, you will build a form on that table.

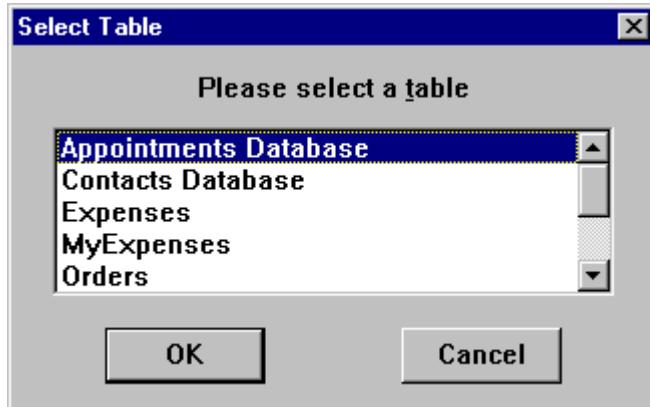
Note

If Visual CE is still open from the previous lesson, the table you just created is already selected. You can tell whether or not a table is selected by looking at the development environment's status bar in the lower left corner – it should display the name of your table.

If so, skip ahead to Quick Lesson 4. Otherwise – if you closed Visual CE after the previous lesson – reopen Visual CE's development environment on the desktop PC by selecting **Start–Programs–Visual CE–Visual CE**, and continue below.

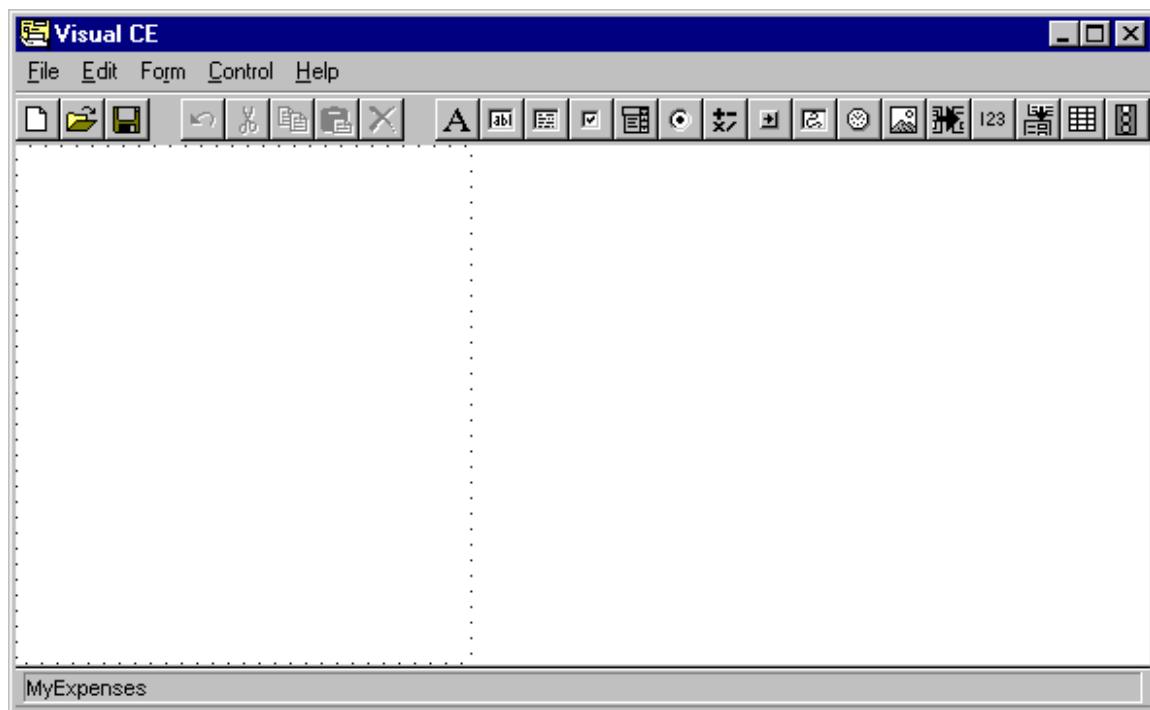
- 1 In Visual CE's development environment, select **File–New Form** or click the **New Form** button .
- 2 If Pocket Access or SQL Server CE databases are supported, Visual CE asks you to specify the location of the table. In that case, click **Windows CE Store**.

The Select Table dialog box appears.



- 3 In the Select Table dialog box, click **MyExpenses** and **OK**.
- 4 Visual CE asks if you want the Form Wizard to create the form. For the purposes of this Quick Tour, click **No**.

Visual CE displays the name of the selected table in its status bar.



You are now ready to build your first form!

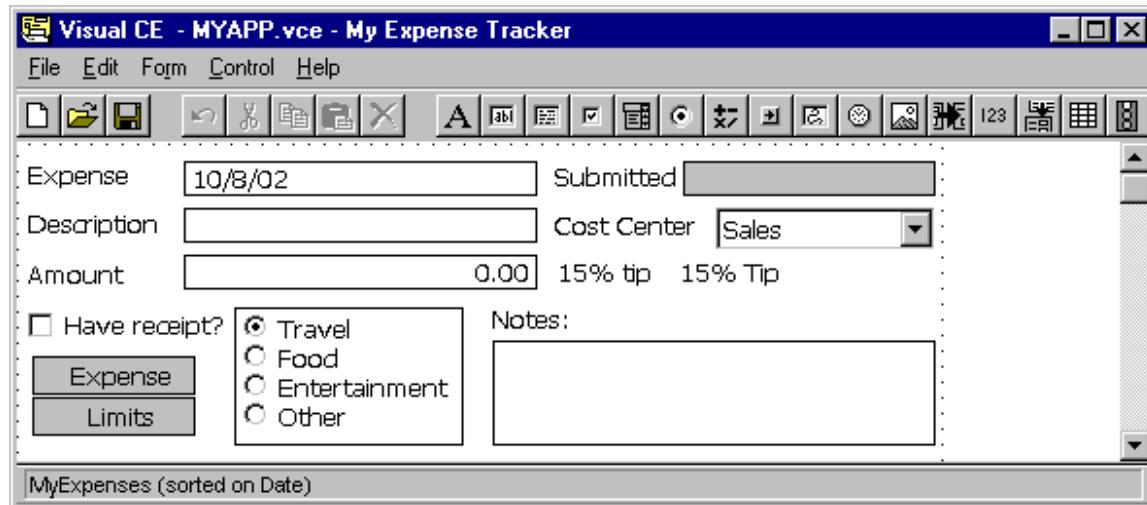
Next: Creating an Edit Box and a Label

Quick Lesson 4: Creating an Edit Box and a Label

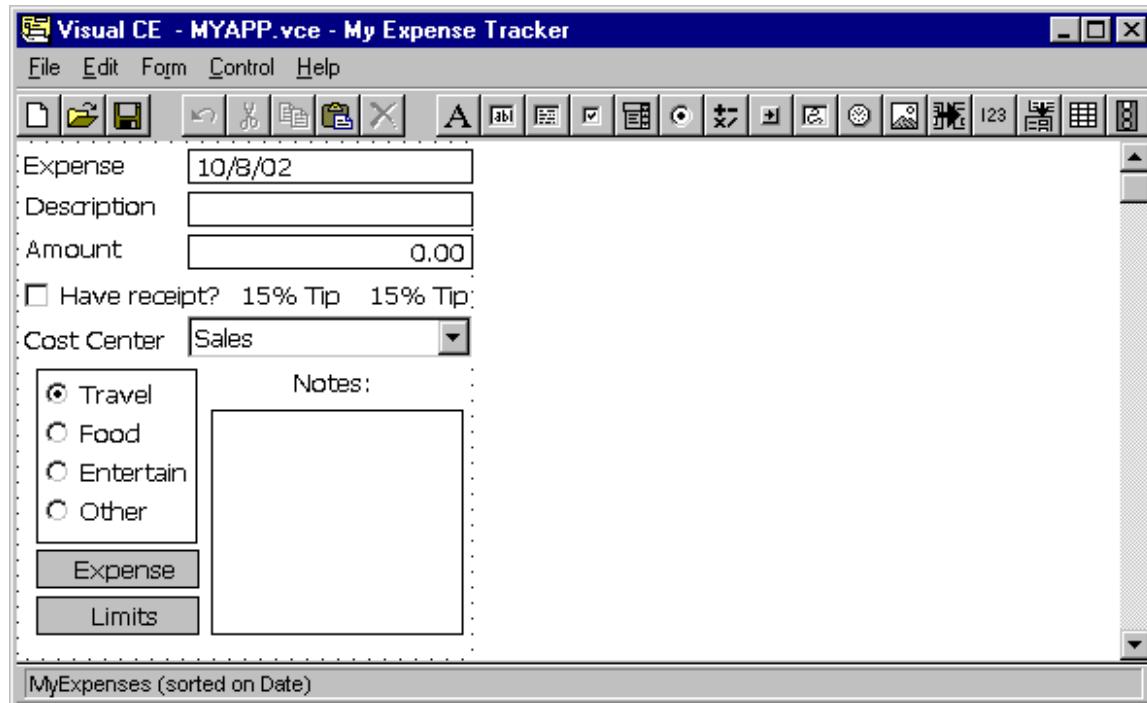
You can now begin to create the controls needed on the form.

As you create the application, use the pictures below as a reference for the placement of each control in the window.

H/PC format:



P/PC format:



Tip

If you make a mistake when creating a control, click the control to select it, then either:

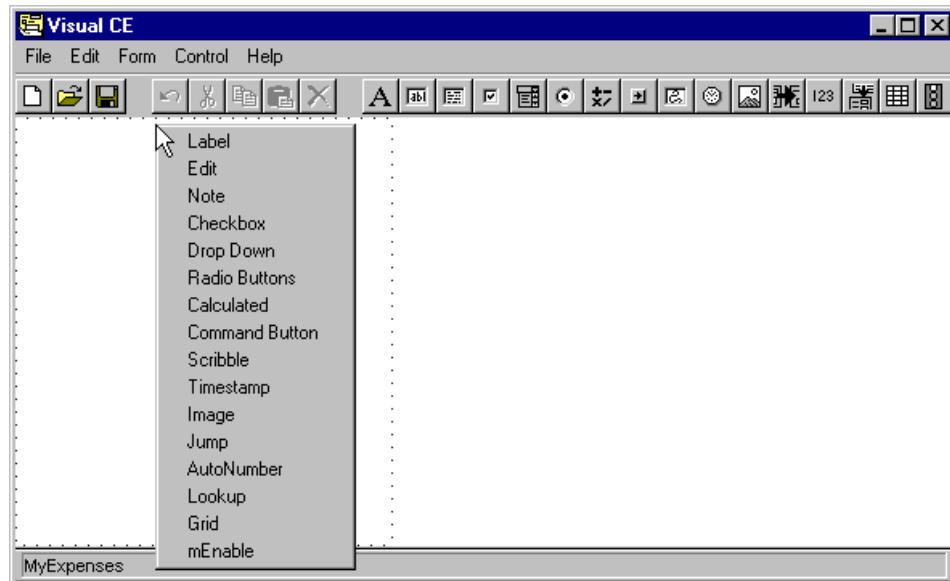
- Edit the control's properties (except the column or control type) by selecting **Edit–Property**, or
- Delete the control by selecting **Edit–Delete**, then recreate the control correctly

1 To begin creating the first edit control, do one of the following:

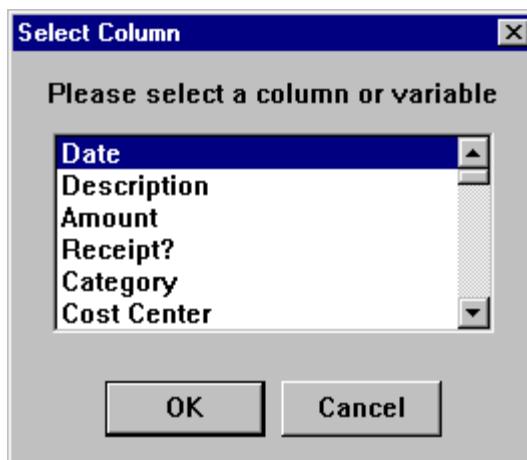
- Click the **Edit** button , or
- Select **Control–Edit**

Either of the above methods places the control in the upper left corner of the form, and you can reposition it later. Alternatively, you can position it now:

- Position the mouse pointer where you want to place the left edge of the control, right-click the mouse, and select **Edit** from the drop-down that appears.

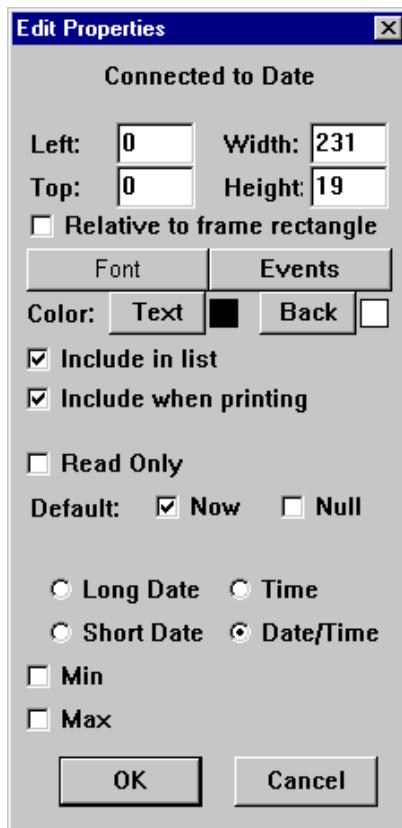


Visual CE displays the Select Column dialog box.



2 Click **Date**, then click **OK**. (Date is the table column to which the edit box relates.)

Visual CE displays the Edit Properties dialog box. Here, you can specify the edit box's placement relative to the top-left of the application window, its dimensions (width and height), as well as the control's font size, weight, and color. You can choose whether or not information stored in the column will appear in the list display and in print outs, and you can define the control as read-only, its default value (today, null, or a specific date), and the type of information it will collect and store (date and/or time). You can even specify a range of acceptable values for the control (minimum and maximum). Advanced options include the ability to fix the control in one place relative to the screen (frame rectangle) even when the user scrolls or moves to another region of the form; and to assign actions that will be triggered automatically when specific events happen to the control. (Each of the Edit properties are described in more detail on page 168).



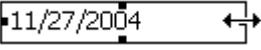
- Accept all of the defaults except one: click **Short Date**. Then, click **OK**.

Tip

The status bar in the lower-left corner of the Development Environment window displays the location and size of the currently selected control in pixels -- specifically, the horizontal position of the left side of the control, the vertical position of the top of the control, and the control's width and height. As you will see in the next step, these values are constantly updated as you move and resize the control.

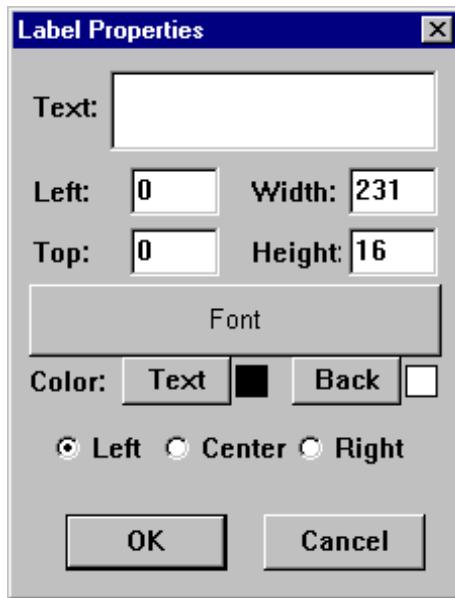
MyExpenses 0,0 231x19

- Click the edit box to make it the active control, then resize it to shorten its width:

- Move the pointer to the control's right "handle" (the small, black square), so that the pointer turns into a double-headed arrow 
- Click and drag the right handle to the left until the control width just exceeds the text area

- Release the mouse button
- 5 If necessary, drag the edit box to the desired location. Note that the box moves in 10 pixel increments. This "snap-to-grid" feature is designed to help you align the controls. If you want to position the box more freely, press and hold the Shift key while you drag the box.
- 6 To create a label for the control, first do one of the following:
- Click the **Label** button , or
 - Select **Control-Label**, or
 - Right-click where you want to place the label, then select **Label**.

Visual CE displays the Label Properties dialog box.



- 7 Enter **Expense** in the Text field, and then click **OK**.
- 8 If necessary, resize and drag the Expense label until it more or less aligns with the edit box. You don't need to be exact; you'll align the controls more precisely in Quick Lesson 6.

You have already learned the basics of creating controls in Visual CE.

Next: Creating the Remaining Edit Boxes and Labels at a faster pace

Quick Lesson 5: Creating the Remaining Edit Boxes and Labels

There are two other edit boxes and labels to be created. One edit box is initially blank for text entries, and the other is defined to accept money entries.

1 Create the "Description" edit control:

- Select **Control>Edit**, or click the **Edit** button , or right-click where you want the edit box to appear and select **Edit**.
- Click **Description**, then click **OK** in the Select Column dialog box.
- Click **OK** to accept the defaults in the Edit Properties dialog box.
- If necessary, drag the edit box to the approximate location.

2 Now make a label for the "Description" edit control:

- Select **Control-Label**, or click the **Label** button , or right-click where you want the label to appear and select **Label**.
- Enter **Description** in the Text field of the Label Properties dialog box, then click **OK**.
- If necessary, drag the label to the approximate position, then resize it.

3 Create the "Amount" edit control:

- Select **Control>Edit**, or click the **Edit** button , or right-click where you want the edit box to appear and select **Edit**.
- Click **Amount**, then click **OK** in the Select Column dialog box.
- Click **Money**, then click **OK** in the Edit Properties dialog box.
Note that the "Money" selection causes numbers to be displayed with two decimal places.
- If necessary, drag the edit box to the approximate location.

4 Create the label for the "Amount" edit control:

- Select **Control-Label**, or click the **Label** button , or right-click where you want the label to appear and select **Label**.
- Enter **Amount** in the Text field of the Label Properties dialog box, then click **OK**.
- If necessary, drag the label to the approximate position, then resize it.

5 Save your work:

- Select **File-Save** or click the **Save** button .
The Save As dialog box appears.
- If necessary, change to the folder in which you want the file saved.
- Enter **MYAPP.VCE** in the File Name field, then click **Save**.

Visual CE saves the application to the MYAPP file. The extension .vce indicates a Visual CE application.

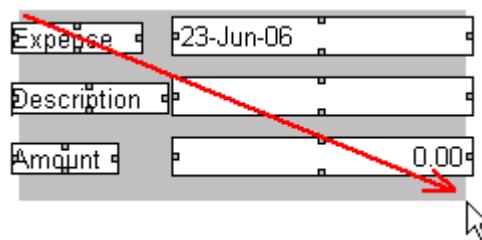
You have created all of the edit boxes required on the Expense form.

Next: Arranging Controls on the Form

Quick Lesson 6: Arranging Controls on the Form

Now that you've created a few controls, you may want to arrange them more precisely so that your form has a neat appearance. In this lesson, you'll learn about Visual CE's multiple-select and align features, as well as how to specify the exact position of a control in pixels.

- 1 Align each label with its edit box:
 - Click the date edit box
 - While pressing the Control key, click on the Expense Date label. Notice that the handles around the first box you selected are solid black, while those around the second box are empty. Visual CE aligns the controls to the first one you select, which is always indicated by the solid boxes.
 - Select **Edit-Align-Bottom**. The bottom edges of the boxes are now aligned.
- 2 Repeat this process for the remaining labels and edit boxes.
- 3 Specify a precise horizontal location for the Date edit box:
 - Right-click (or double-click) the date edit box. Visual CE displays the Edit Properties dialog box.
 - In the "Left" field, type **85** (for an H/PC) or **80** (for a P/PC). This is the distance in pixels that Visual CE will locate the control from the left edge of the form.
 - Click **OK**.
- 4 Align the three edit boxes vertically along their left edges:
 - Click the edit box you just positioned, so that Visual CE will use its location as the guide.
 - Select the other two edit boxes (click the edit boxes while pressing the Control key)
 - Select **Edit-Align-Left**.
- 5 Make all of the edit boxes a uniform width:
 - Click on the date edit box to make it "active"
 - Drag its right handle so that the box is the approximate width shown on the sample
 - While pressing the Control key, click the other two edit boxes.
 - Select **Edit-Align-Width**.
- 6 Drag the entire group of controls to a desired position:
 - First, select all of the controls using one of the following methods:
 - ◊ Select **Edit-Select All**, or
 - ◊ While pressing the Control key, click each of the edit boxes and labels, or
 - ◊ Left-click the mouse and drag over the controls. The selection area is indicated by a gray box. Release the mouse button. (This method works the same as a Windows marquee select.)



- Then, position the mouse pointer over any one of the selected controls, and press and hold the mouse button.
- Drag the group horizontally and vertically. Notice that the relative position of individual controls stays the same.

- To return the group to their original position (if desired), select **Edit-Undo**.
- To unselect the group, double-click in the development window away from the group.

Tip

If you have selected a group of controls, and want to deselect any individual control in the group, press the Control key and click the control.

As you create additional controls in the remainder of this tour, you can apply these techniques to quickly organize the layout of your form.

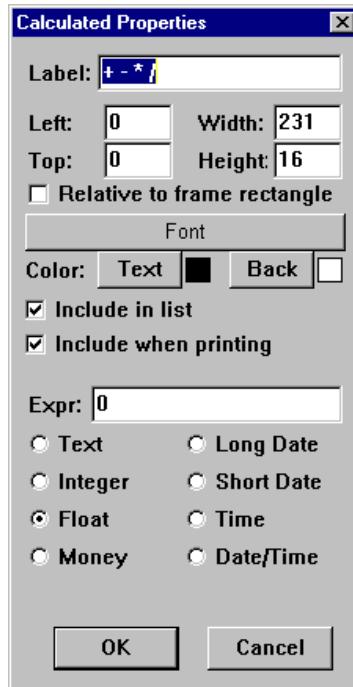
Next: Creating a Calculated Field

Quick Lesson 7: Creating a Calculated Field

Calculated fields are great time-savers for anyone who uses your application. They can automatically apply an operation to a field value and save the results. In this lesson, you will set up a control that automatically calculates a 15% tip on the amount of the expense.

- 1 Select **Control-Calculated**, or click the **Calculated** button , or right-click where you want the control to appear and select **Calculated**.
- 2 Visual CE asks if you want the result of the calculation to be saved in a column in the table. For the purposes of this tutorial, click **No**.

Visual CE displays the Calculated Properties dialog box.



- 3 In the Calculated Properties dialog box:
 - Highlight the default label (+-* /) and type a new one, **15% tip**
(Visual CE will use this text for the column heading in the record list display mode)
 - Enter **Amount * 15/100** in the Expr field
 - Select the **Money** option to specify the display format for the results
 - Click **OK**
- 4 If necessary, drag the field to the correct location.

- 5 Create a label for the field:

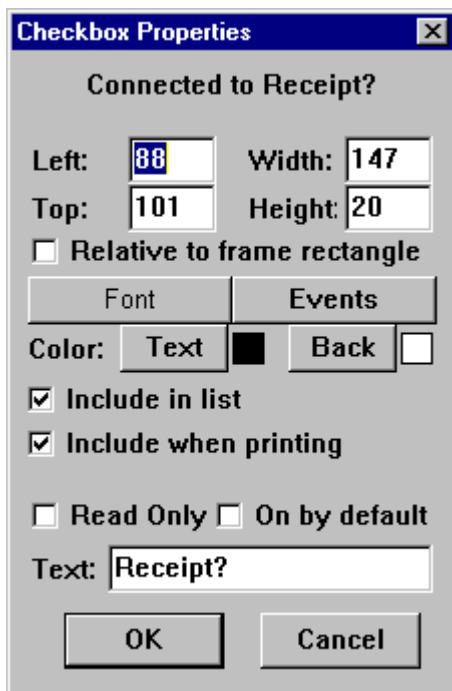
- Select **Control-Label**, or click the **Label** button , or right-click where you want the control to appear and select **Label**.
- Enter **15% Tip** in the Text field of the Label Properties dialog box
(Visual CE will use this text for the field label in the single record display mode)
- Click **OK**
- If necessary, drag the label to the correct position, then resize it

Next: Creating a Checkbox

Quick Lesson 8: Creating a Checkbox

Checkboxes take only seconds to create.

- 1 Select **Control–Checkbox**, or click the **Checkbox** button , or right-click where you want the control to appear and select **Checkbox**.
Visual CE displays the Select Column dialog box.
- 2 Click **Receipt**, then click **OK**.
Visual CE displays the Checkbox Properties dialog box.



- 3 Enter **Have Receipt?** in the Text field, then click **OK**.
- 4 If necessary, drag the checkbox to the correct position, then resize it.

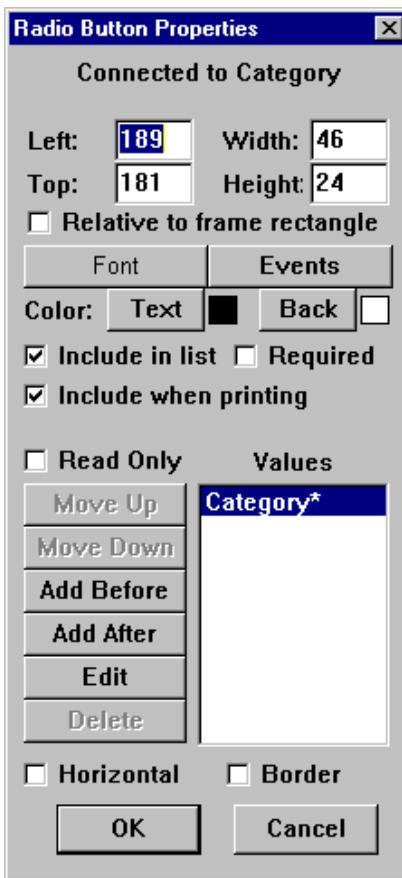
As you created the checkbox, you created the caption (Have Receipt?). You do not need to create labels for checkboxes.

Next: Creating Radio Buttons

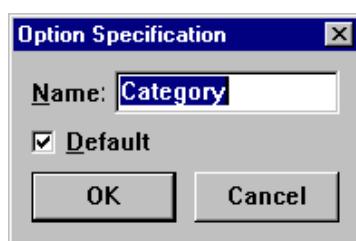
Quick Lesson 9: Creating Radio Buttons

Now you will create a group of four radio buttons, one of which is defined as the default.

- 1 Select **Control–Radio Buttons**, or click the **Radio Buttons** button , or right-click where you want the control to appear and select **Radio Buttons**.
Visual CE displays the Select Column dialog box.
- 2 Click **Category**, then click **OK**.
Visual CE displays the Radio Buttons Properties dialog box.



- 3 In the Radio Buttons Properties dialog box:
 - Click the **Edit** button near the bottom of the dialog box.
Visual CE displays the Option Specification dialog box.



- Remove the word **Category** from the Name field and enter **Travel** in its place. (Leave the **Default** box checked to indicate that **Travel** will be selected as the default.) Click **OK**. Visual CE displays the word **Travel** in the Values column – the asterisk (*) indicates

that it will be the default value in the form.

- Click **Travel** to highlight it, then click the **Add After** button.
 - Enter **Food** in the Name field, and click **OK**.
 - Click the **Add After** button.
 - Enter **Entertainment** in the Name field, and click **OK**.
 - Click the **Add After** button.
 - Enter **Other** in the Name field, and click **OK**.
 - Click the **Border** checkbox (near the bottom of the dialog box) to place a box around the radio buttons in the form.
- 4 Click **OK** in the Radio Button Properties dialog box to close it.
- 5 If necessary, drag the control to the correct location, and resize it so all the radio buttons and their names can be clearly read.
- 6 You may want to save your work again by selecting **File—Save** or clicking the **Save** button.

Because you specified the radio button captions (Travel, Food, etc.) as you created the buttons, you do not need to create labels for them.

Next: Creating a Note Box

Quick Lesson 10: Creating a Note Box

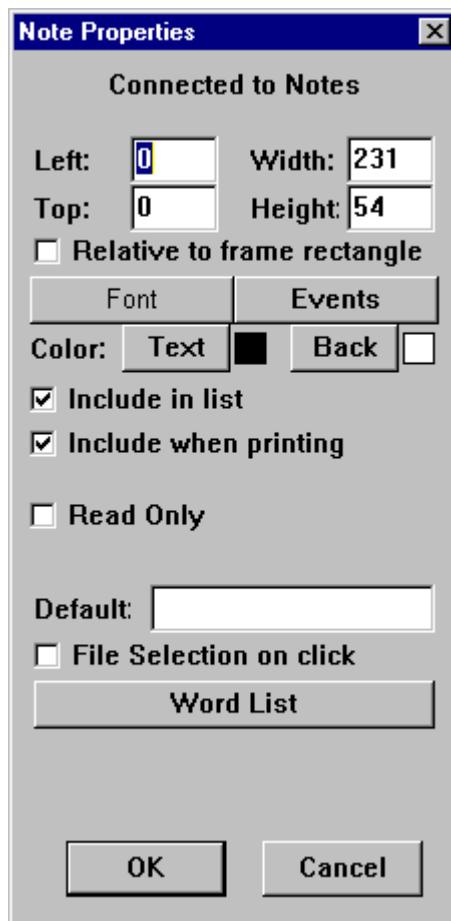
A note box is really just an edit box that allows multiple lines of text – an edit box allows only a single line of text.

- 1 Select **Control–Note**, or click the **Note** button , or right-click where you want the control to appear and select **Note**.

Visual CE displays the Select Column dialog box.

- 2 Click **Notes**, then click **OK**.

Visual CE displays the Note Properties dialog box.



- 3 Click **OK** to accept the default values.
- 4 If necessary, drag the note control to the correct location.
- 5 Now create a label for the note control:

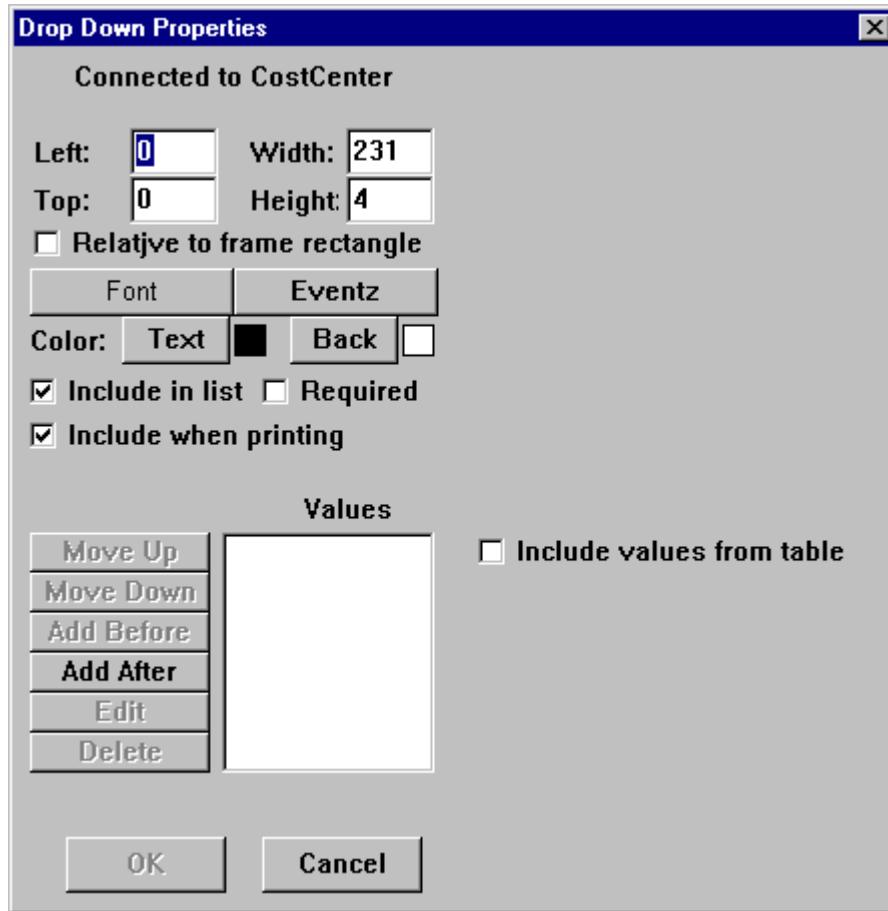
- Select **Control–Label**, or click the **Label** button , or right-click where you want the label to appear and select **Label**.
- Enter **Notes:** in the Text field, then click **OK**.
- If necessary, drag the label to the correct location.

Next: Creating a Drop-Down List

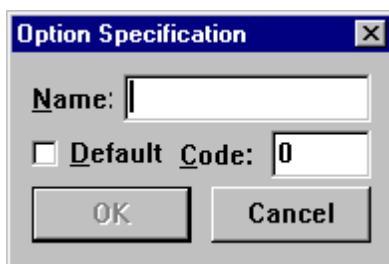
Quick Lesson 11: Creating a Drop-Down List

In this lesson, you will create a drop-down list containing six options. One option will be defined as the default.

- 1 Select **Control—Drop Down**, or click the **Drop Down** button , or right-click where you want the control to appear and select **Drop Down**.
Visual CE displays the Select Columns dialog box.
- 2 Click **CostCenter**, then click **OK**.
Visual CE displays the Drop Down Properties dialog box.



- 3 Click **Add After**.
Visual CE displays the Option Specification dialog box.



4 Do the following to create the first value in the drop-down list:

- Enter **Sales** in the Name field.
- Click **Default** to select the Sales option as the default of the drop-down list.
- Enter **10045** in the Code field.

Note

Codes are always/only used when the drop-down is connected to a numeric column in the table. Cost Center is a numeric column. Therefore, you need to provide a numeric code for each drop-down caption you enter. The user sees the name (in this case, Sales) on the form, but Visual CE stores the more compact numeric code (in this case 10045) in the table.

- Click **OK**.

5 Now create the remaining values in the drop-down list:

- Click **Sales** to highlight it, then click **Add After**.
- Enter **Marketing** in the Name field and **14578** in the Code field. Click **OK**.
- Click **Add After**.
- Enter **Production** in the Name field and **14902** in the Code field. Click **OK**.
- Click **Add After**.
- Enter **Admin** in the Name field and **18012** in the Code field. Click **OK**.
- Click **Add After**.
- Enter **R&D** in the Name field and **14013** in the Code field. Click **OK**.
- Click **Add After**.
- Enter **Other** in the Name field and **19999** in the Code field. Click **OK**.

6 Click **OK** in the Drop Down Properties dialog box to close it.

7 Drag the drop-down list control to the correct location.

8 Now create a label for the drop-down list:

- Select **Control-Label**, or click the **Label** button , or right-click where you want the label to appear and select **Label**.
- Enter **Cost Center** in the Text field of the Label Properties dialog box, then click **OK**.
- If necessary, drag the label to the correct position, then resize it.

9 To save your work, click the **Save** button.

Next: Creating Command Buttons

Quick Lesson 12: Creating Command Buttons

Now you will create two command buttons that will enable users to quickly jump to predefined regions of your form without having to use the scroll bars. You'll label the buttons "Expense" and "Limits."

- 1 Select **Form-Form Size**.

Visual CE displays the Form Size dialog box.

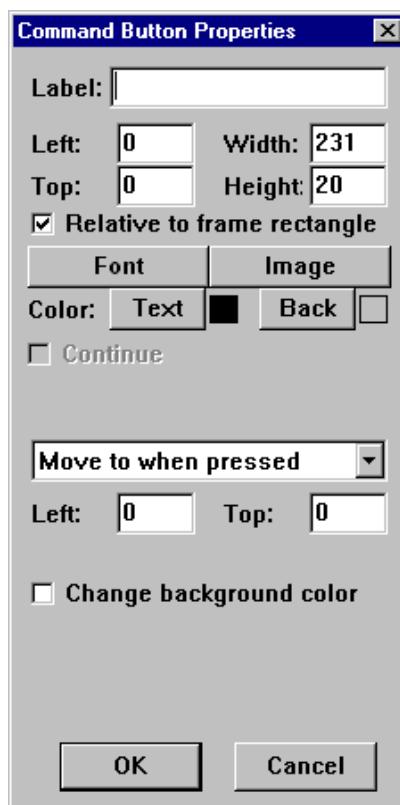
- 2 Double the value for height and click **OK**. (Note: the values shown here are not necessarily appropriate for your device.)



- 3 The position of the dotted-line frame specifies the region of the form that will be displayed when the form's user presses the first command button. It should be in upper left corner of the development window. If it is not already there, reposition it by pressing the Shift key on your keyboard and dragging it into place.

- 4 Create the first command button:

- Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command Button**.
Visual CE displays the Command Button Properties dialog box.



- In the Label field, enter **Expense**.

- Since you want the "Expense" button always to be visible regardless of which region of the form is being displayed, leave "Relative to frame rectangle" checked.
 - Accept the remaining defaults and click **OK**.
 - If necessary, drag the command button to the correct location (relative to the frame or display), and resize it.
- 5 Specify the region of the form to display when the user clicks the second command button:
- Scroll down to the bottom of the form.
 - While pressing the Shift key on your keyboard, drag the dotted-line frame to the lower left corner of the form.
- 6 Enter text to appear in this new region of the form:

- Select **Control-Label**, or click the **Label** button , or right-click where you want the label to appear and select **Label**.
- Enter the text (shown below) in the Text field. You can use spaces and carriage returns to format the text. Use the up and down arrow keys on the keyboard to move between lines without making changes. **Tip:** Each digit is the equivalent in width to two blank spaces. So, for example, to align the decimal points in the sample text, press the Space bar four times before typing "8.00 Breakfast," two times before "10.00 Lunch," and no times before "150.00 Hotel." Characters are variable widths.

Expense Limits

```

8.00 Breakfast
10.00 Lunch
35.00 Dinner
150.00 Hotel
30.00 Car rental (daily)
50.00 Entertainment (including client dinner)

```

- When you have completed the text, click **OK**.
 - Resize the box so that all of the text is visible, and drag it into position.
- 7 Create the "Limits" command button in the same way you created the "Expense" tab button in step 4 above.
- 8 Confirm that the command button displays the desired region:
- While pressing the Shift key on your keyboard, drag the dotted-line frame to its original position. Release the Shift key.
 - While pressing the Shift key on your keyboard, click the **Limits** tab button on the form display.
- The frame or display moves to the new position, indicating the area that will be displayed on the CE device. (You may have to scroll the development window's display to view the frame's new position.)
- 9 To save your work, click the **Save** button.

Tip

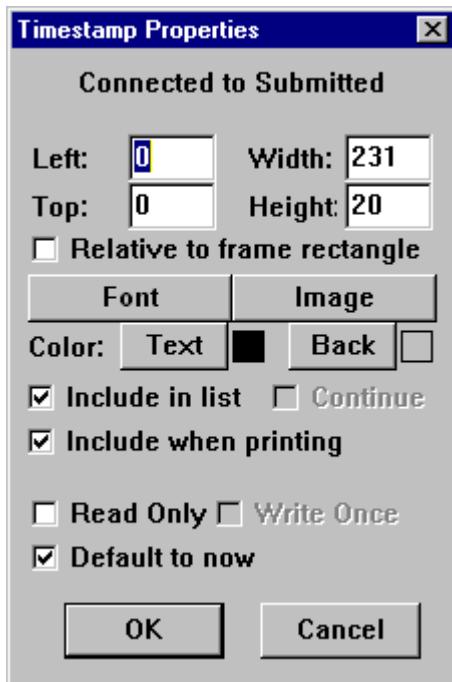
When creating your own applications, as an alternative to command buttons that jump to different regions of the form, you can divide your form into any number of tabbed pages. That technique is described in the topic "Adding Tabs", on page 164.

Next: Creating a Timestamp Button

Quick Lesson 13: Creating a Timestamp Button

In this lesson, you will create a button that the user clicks to "stamp" the record with a date and time.

- 1 Select **Control-Timestamp**, or click the **Timestamp** button , or right-click where you want the control to appear and select **Timestamp**.
Visual CE displays the Select Columns dialog box.
- 2 Click **Submitted**, then click **OK**.
Visual CE displays the Timestamp Properties dialog box.



- 3 Click the **Default to now** checkbox to remove the checkmark, and click **OK**.
- 4 If necessary, drag the button to the correct location. (If you have a P/PC, place it in the Limits region of the form.)
- 5 Create a label for the Timestamp button:
 - Select **Control-Label**, or click the **Label** button, or right-click where you want the label to appear and select **Label**.
 - Enter **Submitted** in the Text field, then click **OK**.
 - If necessary, drag the label to the correct location.
- 6 Save your work by clicking the **Save** button.

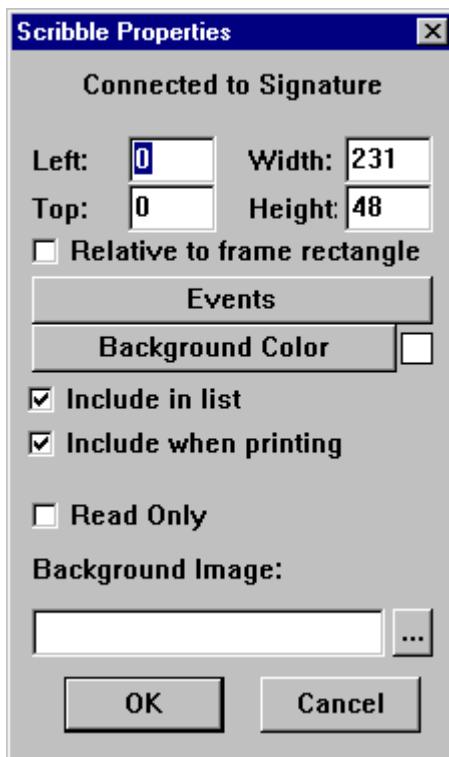
You have created all of the controls on the Expense form, but one – a Scribble box in which the user can sign the expense form.

Next: Creating a Scribble Box

Quick Lesson 14: Creating a Scribble Box

Scribble boxes let you draw images on the form -- just like using a pencil on a pad of paper. Visual CE saves the images in the table for later display and printing. In this lesson, you'll create a box in which users can sign their expense records.

- 1 Select **Control-Scribble**, or click the **Scribble** button , or right-click where you want the control to appear and select **Scribble**.
Visual CE displays the Select Column dialog box.
- 2 Click **Signature**, then click **OK**.
Visual CE displays the Scribble Properties dialog box.



- 3 Click **OK** to accept the default values.
- 4 If necessary, drag the scribble box to a location of your choice (probably in the Limits region of the form).
- 5 Now create a label for it:
 - Select **Control-Label**, or click the **Label** button, or right-click where you want the label to appear and select **Label**.
 - Enter **Signature** in the Text field, then click **OK**.
 - Position the label as desired.
- 6 Click **Save**.

Next: Creating a Title for the Application

Quick Lesson 15: Creating a Title for the Application

The application's title will display in the title bar in both the development environment window and the application window.

- 1 Select **Form—Title Bar**.

Visual CE displays the Title Bar dialog box.



- 2 Enter **My Expense Tracker** in the Title field, then click **OK**.

Visual CE displays your title in the title bar.

Next: Specifying an Initial Sort/Search Order

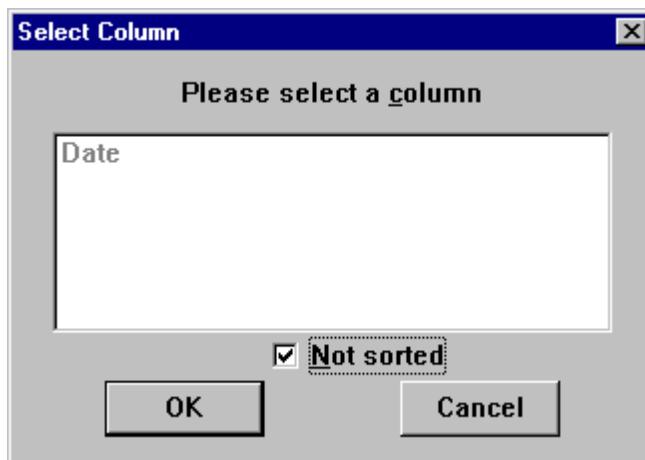
Quick Lesson 16: Specifying an Initial Sort/Search Order

You have the option to specify an initial sort/search order for record display (the default is unsorted records).

As you saw in the first Quick Tour, the records in the sample application are initially ordered based on the entry in the Date field.

1 Select Form-Sort On.

Visual CE displays the Select Column dialog box. Notice that the list includes the only column that you "indexed" in Quick Lesson 2. (Only indexed columns can be used for sorting.)



- 2 Click the **Not Sorted** checkbox to remove the checkmark
- 3 Click **Date** to select sorting based on the value in the Date column, then click **OK**.

That's all there is to specifying an initial sort/search order.

Next: Specifying the Tab and Column Order

Quick Lesson 17: Specifying the Tab and Column Order

Users may enter data into your application by clicking each field, or by pressing the Tab key to move from field to field. You can specify the order in which the cursor moves from field to field when the Tab key is pressed.

Note

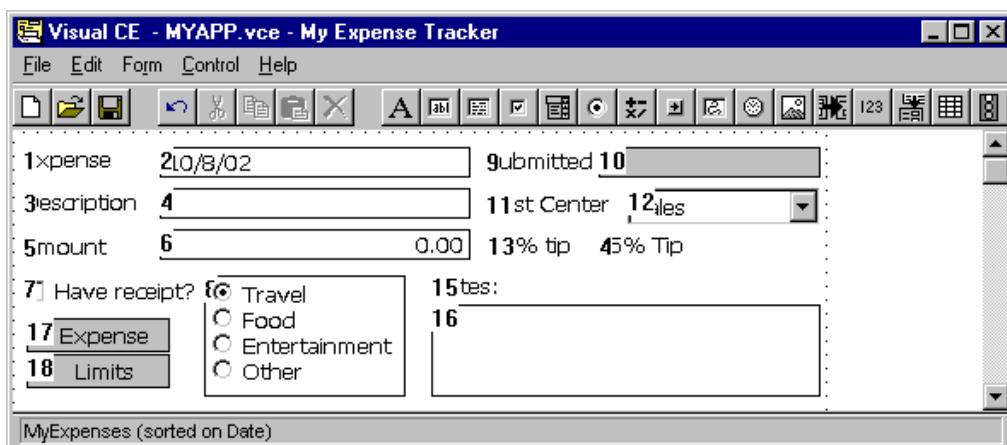
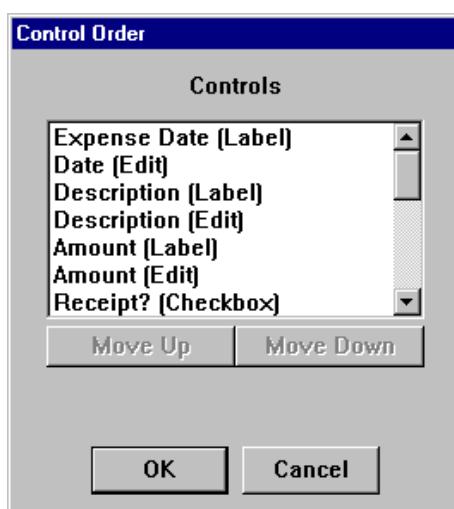
Visual CE applies the order you specify here to other places in the application, including the **order of the columns in the record list display** and the **order of record values in an import or export file**.

1 Select Form–Control order.

Visual CE displays the Control Order dialog box on top of the form. This dialog box lists the controls in the order you created them. You can drag the dialog box to the side to view the form.

On the form, Visual CE displays a number next to each control you created. The numbers on the form match the order in the dialog box.

You can change the control order using the dialog box, or the form, or both.



Note

All of the controls are numbered, but when the application is used, the cursor tabs only to those fields in which user entries are allowed.

- 2 In the Control Order dialog box, select one of the controls and click the Move Up or Move Down button as many times as you like.

Visual CE moves the selected control in the sequence. Notice how the numbering displayed in the form is automatically updated to match the sequence in the Control Order dialog box.

You can repeat this technique with the other controls until the entire list is reordered as desired.

- 3 An alternative way to reorder the controls is by clicking them one at a time in the form layout. Try it now:

- On the form, click the first control – the one in which you want the cursor to appear initially.
Visual CE changes that control's number to "1."
- Click the control you want the cursor to move to next.
Visual CE changes that control's number to "2."
- Continue to click each control to indicate the order you want.

Tips

- If you make a mistake, click the **Cancel** button in the Control Order dialog box.
Alternatively, select **Edit–Undo**.
 - Picture boxes are included in the tab order, but they do not appear in the list display.
- 4 When you have finished specifying the control order, click **OK** in the Control Order dialog box, or click anywhere in the environment window (except on a control).
 - 5 Select **File–Save** or click the **Save** button to save your work.

Congratulations! You have created the Expenses sample application, including most of the controls provided by Visual CE.

Next: Downloading Your Application

Quick Lesson 18: Downloading Your Application

Each time you save your application, Visual CE saves the .vce file on your desktop PC. When you have finished creating the application, you need to download the file to your CE device.

- 1 In Visual CE on your desktop PC, select **File–Download Form**.
Visual CE displays the Save dialog box.
- 2 Enter **MYAPP.VCE** in the Name field, then click **OK**.
Visual CE downloads the application to your CE device within seconds – the file size is very small.

Next: Testing Your Application

Quick Lesson 19: Testing Your Application

Test the application on the CE device to be sure that it works the way you intended.

- 1 On the CE device, click **Start-Programs-MYAPP**.
Visual CE displays your Expense Tracker in the application window.
- 2 Test your application to be sure that it works correctly. You can use the instructions given in the application use tutorial (Quick Tour I) to test it.
- 3 If your application works correctly, you have done a great job! If not, you have probably made a simple mistake or two.

To correct any mistakes, close the application on the CE device and return to the Visual CE development window. If you made a mistake with a control, delete the control and recreate it. If your mistake concerned the title bar, sort/search order, or tab order, respecify the option correctly. After making your changes, click the **Save** button to save your work, then download it to the CE device by selecting **File-Download Form**. Then repeat the test instructions above.

You have finished Quick Tour II. You are already a Visual CE expert!

Next:

You can take Quick Tour III (page 99) which goes step by step through the process of downloading a table from the desktop PC to the CE device and synchronizing the contents of the two tables, Quick Tour IV (page 106) which introduces macros and events, and Quick Tour V (page 116) which explains how to track down mistakes in a macro.

Or, you can begin now to create your own applications using Visual CE (page 124).

Quick Tour IIa: Creating a Visual CE Application for Smartphones

Overview of Quick Tour IIa

You can learn the basics of developing Visual CE applications for non-touchscreen Smartphones in under ten minutes. In this quick tour, you will create an application that is nearly identical to the sample Expense Tracker program that you explored in Quick Tour Ia.

Note

If you are creating applications for touchscreen devices, take Quick Tour II instead.

Creating the application involves the following steps:

- Quick Lesson 1a: Starting Visual CE's Development Environment
- Quick Lesson 2a: Creating a table
- Quick Lesson 3a: Selecting a table
- Quick Lesson 4a: Creating an edit
- Quick Lesson 5a: Creating a command button to sort records
- Quick Lesson 6a: Creating the remaining edit boxes and labels
- Quick Lesson 7a: Arranging controls on the form
- Quick Lesson 8a: Creating a calculated field
- Quick Lesson 9a: Creating a checkbox
- Quick Lesson 10a: Creating a drop-down list
- Quick Lesson 11a: Creating radio buttons
- Quick Lesson 12a: Creating a note box
- Quick Lesson 13a: Creating command buttons to move to different regions of the form
- Quick Lesson 14a: Creating a timestamp button
- Quick Lesson 15a: Creating a title for the application
- Quick Lesson 16a: Specifying an initial sort/search order
- Quick Lesson 17a: Specifying the tab and column order
- Quick Lesson 18a: Downloading the application
- Quick Lesson 19a: Testing the application

Note

To best understand the effect of your development choices, you should use the Expense Tracker application on the CE device before developing an identical program. See Quick Tour Ia.

Quick Lesson 1a: Starting Visual CE's Development Environment

In this lesson, you will start Visual CE's development environment on the desktop PC and get acquainted with its features.

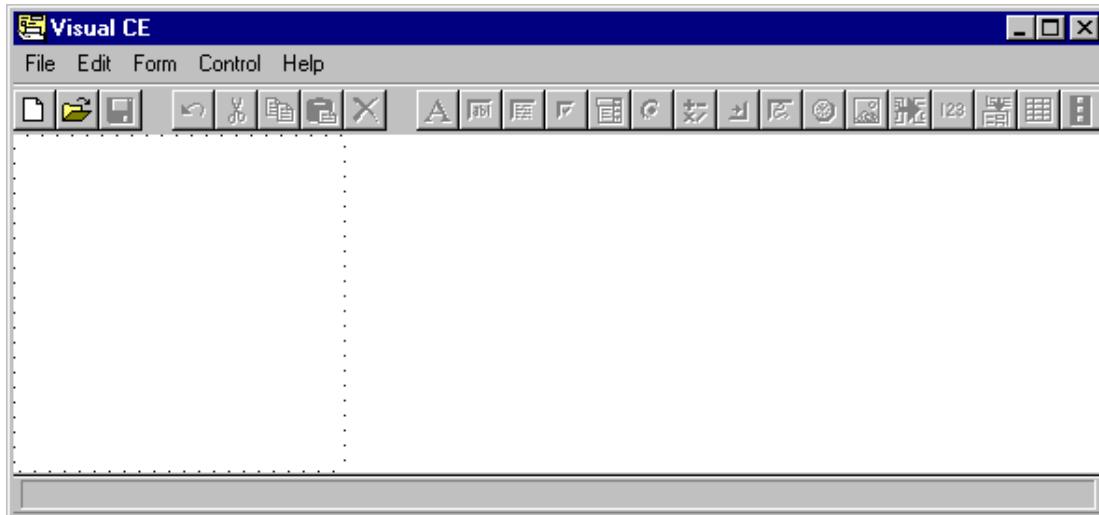
Note

You cannot build applications directly on the Smartphone. You must use Visual CE's development environment on the desktop PC.

To start Visual CE's development environment on the desktop PC:

- 1 Connect your CE device to your desktop PC.
- 2 On your desktop PC, click the Windows **Start** button, then **Programs–Visual CE–Visual CE**.

Visual CE displays its development environment window.



Visual CE's development environment window has five menus: File, Edit, Form, Control, and Help. You can select menu options by clicking them with the mouse pointer or by pressing the shortcut keys.

Its toolbar buttons provide a faster way to activate the most frequently used commands.

The dotted outline in the development window indicates the screen size for the CE device. (Visual CE automatically detects which type of CE device is connected, and displays the screen size for that type.) If you have a device with a larger or smaller display, you can adjust the default using the instructions in the topic "Device Screen Size" on page 128.

Visual CE's default form size matches the screen size. You can change this to any desired size, even a size larger than the CE device screen size. (See "Form Size" on page 158 for additional information.)

The status bar in the lower-left corner (empty in the illustration above) provides helpful feedback as you build the form.

Next: Creating a Table

Quick Lesson 2a: Creating a Table

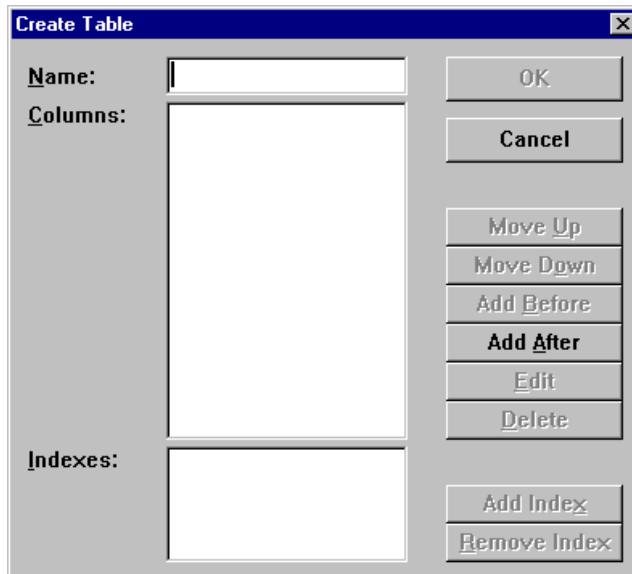
Your first step in creating a new application is to select an existing table or create a new one. For the purposes of this Quick Tour, you will create a new table that is very similar to the one belonging to the Expense Tracker.

You must create one column for each category of data that you want to collect and store in your application.

To create the sample table using Visual CE on the desktop PC:

- 1 Select **File—Create Table**.

Visual CE displays the Create Table dialog box.

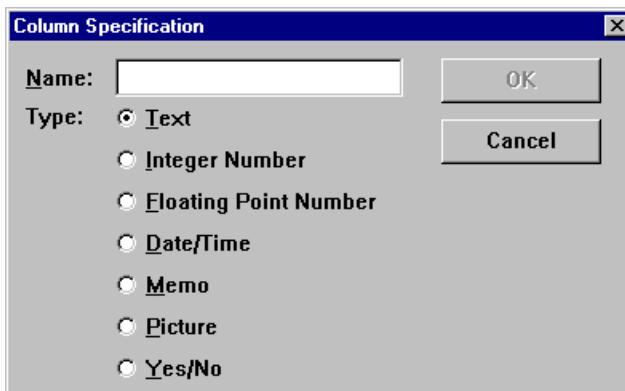


- 2 For Name, type **MyExpenses**.

- 3 Define the first column:

- Click the **Add After** button

Visual CE displays the Columns Specifications dialog box.



- For name, type **Date**

- For Type, click **Date/Time**

- Click **OK**

4 Define the next column:

- Click the **Add After** button
- For name, type **Description**
- For Type, click **Text**
- Click **OK**

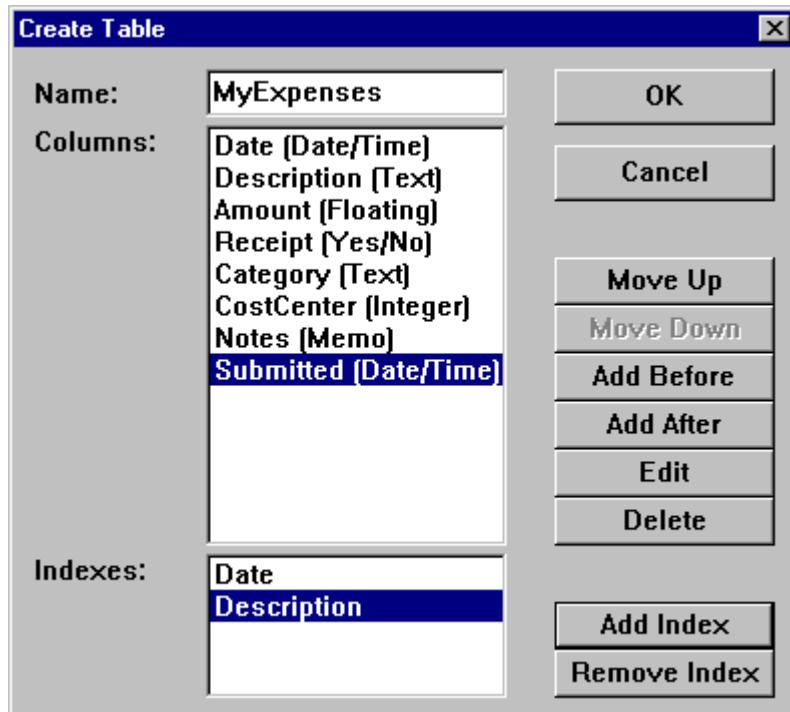
5 Define the remaining columns using the same method as described in Step 4:

Name	Type
Amount	Floating Point Number
Receipt	Yes/No
Category	Text
CostCenter	Integer Number
Notes	Memo
Submitted	Date/Time

6 "Indexing" a column enables Visual CE to sort the records in the table by values stored in that column:

- Click the **Add Index** button
- In the dialog that appears, select **Date**
- Click **OK**
- Click the **Add Index** button
- In the dialog that appears, select **Description**
- Click **OK**

Your table definition so far should look like this:



7 Click **OK** to save your table definition.

8 Visual CE asks if you would like the Form Wizard to create a form for you. (This convenient feature automatically creates edit boxes and labels for each column in the table. You can then modify and rearrange the labels and controls as desired.) However, to get the most practice now, click **No**.

Visual CE's status bar (at the bottom of the development environment window) now displays the name of your new table – indicating that the form you'll create in the following lessons will be built on that table.

You might want to take a break before starting your form design. If so, close Visual CE by selecting **File-Close**.

Next: Selecting a Table

Quick Lesson 3a: Selecting a Table

About Tables and Forms

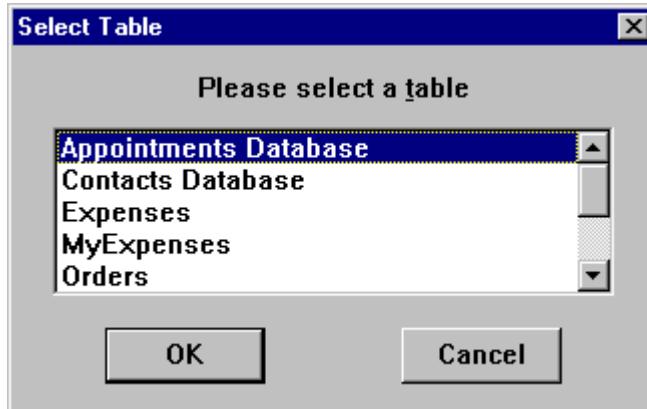
To create a Visual CE application, you typically build two components: a table and a form. A table is a structure that stores and organizes your data inside the computer. It is invisible to the user. You cannot view or modify data stored in a table—except via a form. A form is a program (.vce file) that makes it possible for you to view, add, modify, and delete data stored in the table. A form is the user's interface with the data—it's what you see on the screen.

In the previous lessons, you created a table. In this and the following lessons, you will build a form on that table.

Note

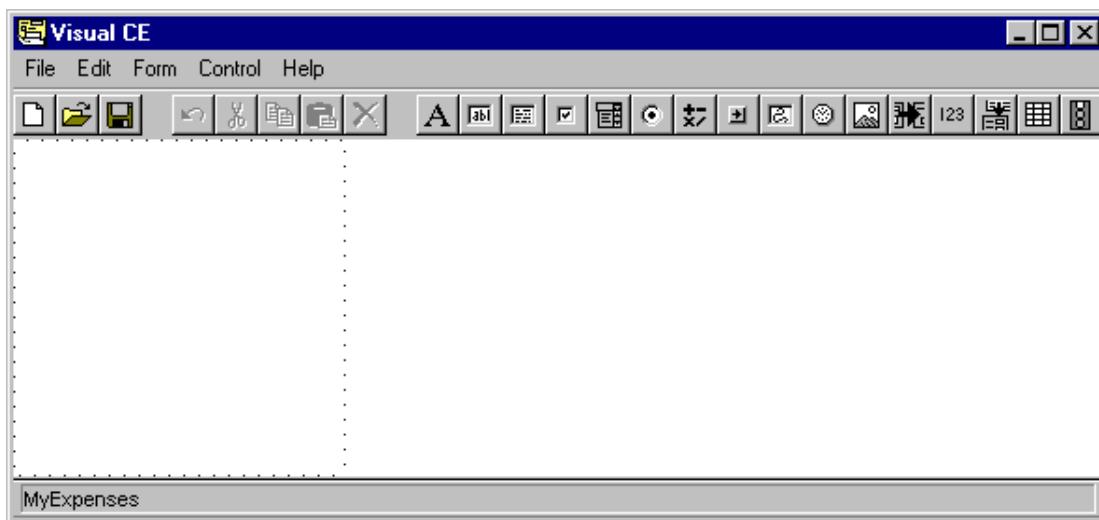
If Visual CE is still open from the previous lesson, the table you just created is already selected. You can tell whether or not a table is selected by looking at the development environment's status bar in the lower left corner – it should display the name of your table. **If so, skip ahead to Quick Lesson 4a.** Otherwise – if you closed Visual CE after the previous lesson – reopen Visual CE's development environment on the desktop PC by selecting **Start–Programs–Visual CE–Visual CE**, and continue below.

- 1 In Visual CE's development environment, select **File–New Form** or click the **New Form** button .
- 2 If Pocket Access or SQL Server CE databases are supported, Visual CE asks you to specify the location of the table. In that case, click **Windows CE Store**. The Select Table dialog box appears.



- 3 In the Select Table dialog box, click **MyExpenses** and **OK**.
- 4 Visual CE asks if you want the Form Wizard to create the form. For the purposes of this Quick Tour, click **No**.

Visual CE displays the name of the selected table in its status bar.



You are now ready to build your first form!

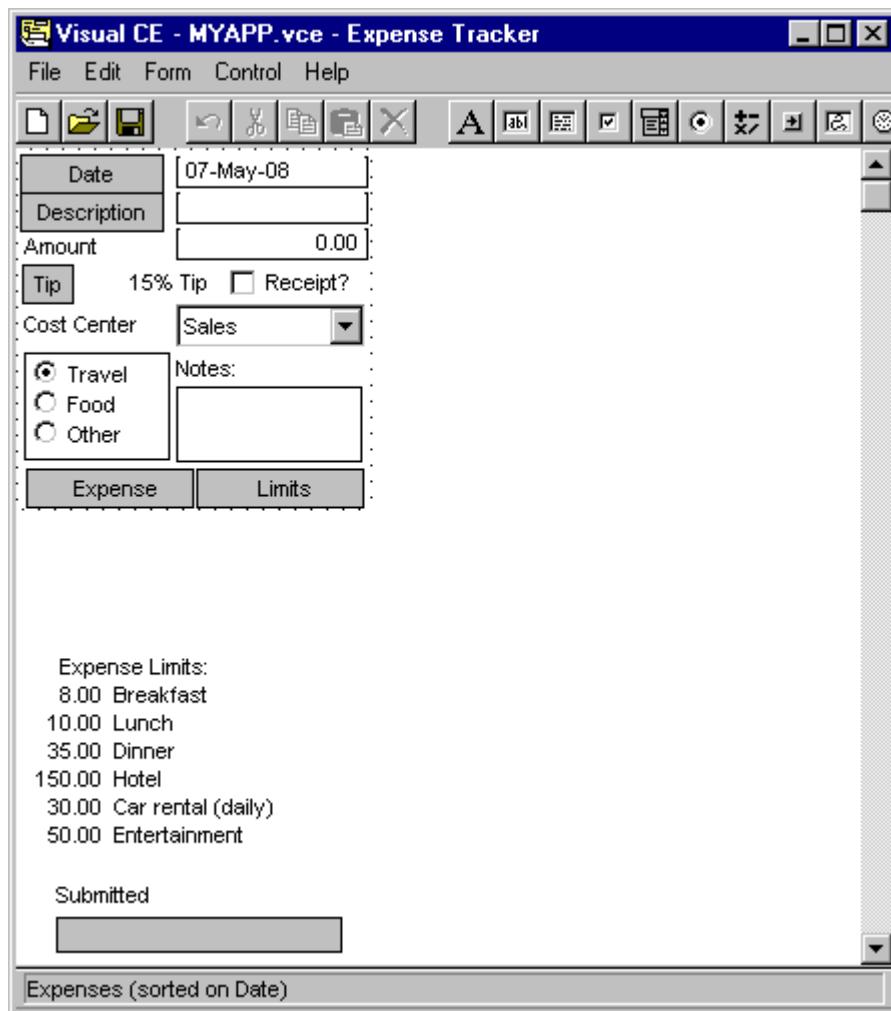
Next: Creating an Edit Box

Quick Lesson 4a: Creating an Edit Box

You can now begin to create the controls needed on the form.

As you create the application, use the pictures below as a reference for the placement of each control in the window.

Smartphone format:



Tip

If you make a mistake when creating a control, click the control to select it, then either:

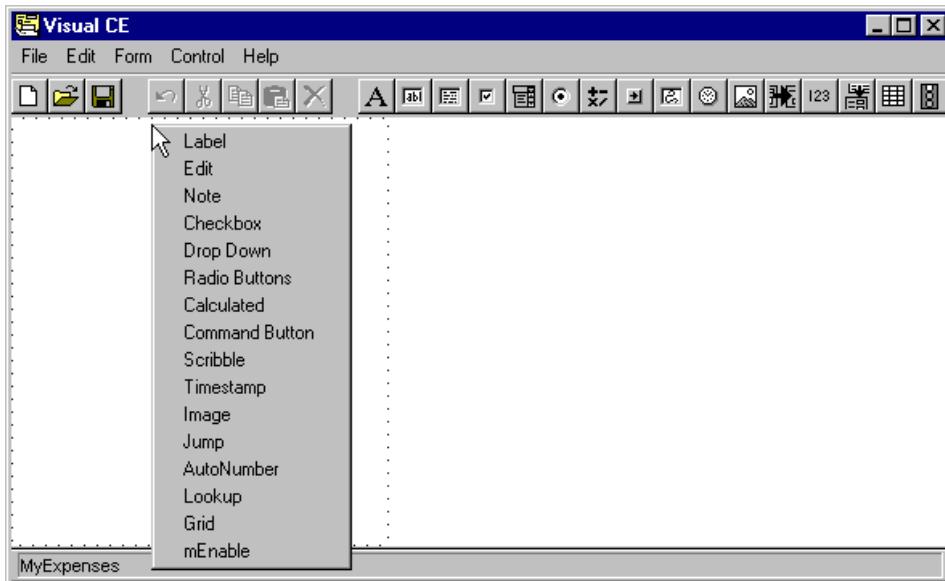
- Edit the control's properties (except the column or control type) by selecting **Edit–Property**, or
- Delete the control by selecting **Edit–Delete**, then recreate the control correctly

1 To begin creating the first edit control, do one of the following:

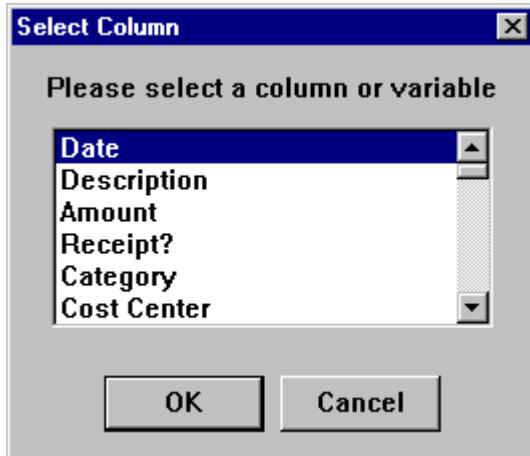
- Click the **Edit** button , or
- Select **Control–Edit**

Either of the above methods places the control in the upper left corner of the form, and you can reposition it later. Alternatively, you can position it now:

- Position the mouse pointer where you want to place the left edge of the control, right-click the mouse, and select **Edit** from the drop-down that appears.

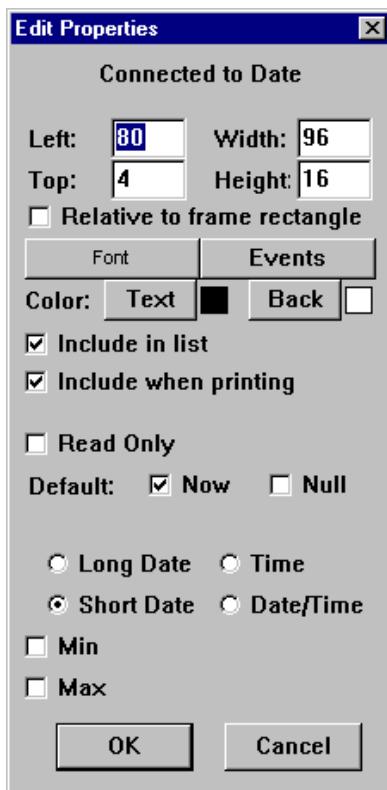


Visual CE displays the Select Column dialog box.



- 2 Click **Date**, then click **OK**. (Date is the table column to which the edit box relates.)

Visual CE displays the Edit Properties dialog box. Here, you can specify the edit box's placement relative to the top-left of the application window, its dimensions (width and height), as well as the control's font size, weight, and color. You can choose whether or not information stored in the column will appear in the list display and in print outs, and you can define the control as read-only, its default value (today, null, or a specific date), and the type of information it will collect and store (date and/or time). You can even specify a range of acceptable values for the control (minimum and maximum). Advanced options include the ability to fix the control in one place relative to the screen (frame rectangle) even when the user scrolls or moves to another region of the form; and to assign actions that will be triggered automatically when specific events happen to the control. (Each of the Edit properties are described in more detail on page 168).



- 3 Because non-touchscreen devices tend to have a smaller screen size, you may want to minimize the space that each control occupies on the screen. For that reason, you may want to assign a small font to the text that appears in the Edit box. Click **Font**, and enter **8** for size and deselect the **Bold** option.
- 4 Accept the defaults for all of the other properties except one: click **Short Date**. Click **OK**.

Tip

The status bar in the lower-left corner of the Development Environment window displays the location and size of the currently selected control in pixels -- specifically, the horizontal position of the left side of the control, the vertical position of the top of the control, and the control's width and height. As you will see in the next step, these values are constantly updated as you move and resize the control.

MyExpenses 0,0 231x19

- 5 Click the edit box to make it the active control, then resize it to shorten its width:
 - Move the pointer to the control's right "handle" (the small, black square), so that the pointer turns into a double-headed arrow 
 - Click and drag the right handle to the left until the control width just exceeds the text area
 - Release the mouse button
- 6 If necessary, drag the edit box to the desired location. Note that the box moves in 10 pixel increments. This "snap-to-grid" feature is designed to help you align the controls. If you want to position the box more freely, press and hold the Shift key while you drag the box.

You have already learned most of the basics of creating controls in Visual CE!

Next: Creating Command Buttons to Sort Records

Quick Lesson 5a: Creating a Command Button to Sort Records

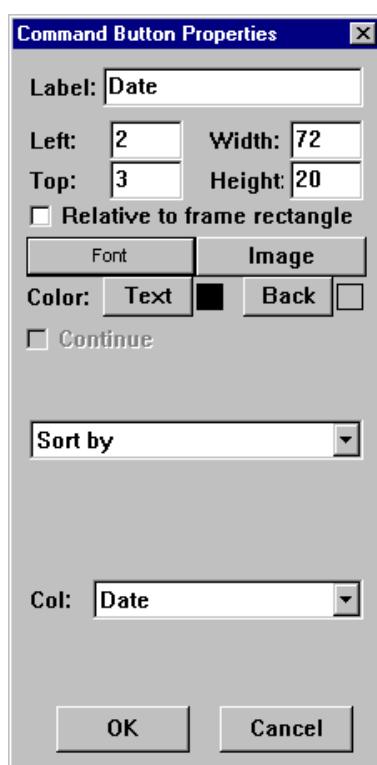
If you want to enable users to sort records, you must provide a command button or menu option for that purpose. In this lesson, you will create a command button that will enable users to sort records by date.

Note

The sort column must be indexed. If you followed the instructions in Quick Lesson 2a, both Date and Description are indexed.

- 1 Select **Control—Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.

Visual CE displays the Command Button Properties dialog box.



- 2 For Label, enter **Date**.
- 3 Make sure that the **Relative to frame rectangle** option is *not* selected.
- 4 To use less space on the screen, you may want to assign a small font to the label. Click **Font**, and enter **8** for size and deselect the **Bold** option.
- 5 In the drop-down menu, select **Sort by**.
- 6 For Col, enter **Date**.
- 7 Click **OK**.
- 8 Resize and drag your new Sort by Date command button to the upper-left corner of the screen.

Next: Creating More Edit Boxes and Labels

Quick Lesson 6a: Creating the Remaining Edit Boxes and Labels

There are two more edit boxes and a label to be created. One edit box is initially blank for text entries, and the other is defined to accept money entries. You also need a Sort by Description command button.

Tip

To minimize the space that each control occupies on the screen, use the Font button in the Properties dialog for each control to assign a font size smaller than the default (as you have done in the two previous lessons).

1 Create the "Description" edit control:

- Select **Control>Edit**, or click the **Edit** button , or right-click where you want the edit box to appear and select **Edit**.
- Click **Description**, then click **OK** in the Select Column dialog box.
- Click **OK** to accept the defaults in the Edit Properties dialog box.
- If necessary, drag the edit box to the approximate location.

2 Now create a command button to sort records on the Description column. Use the technique described in the previous lesson. Make sure "Relative to frame rectangle" is *not* checked. For the command button property Col, select **Description**. Size and drag the button to the approximate location.

3 Create the "Amount" edit control:

- Select **Control>Edit**, or click the **Edit** button , or right-click where you want the edit box to appear and select **Edit**.
- Click **Amount**, then click **OK** in the Select Column dialog box.
- Click **Money**, then click **OK** in the Edit Properties dialog box.
Note that the "Money" selection causes numbers to be displayed with two decimal places.
- If necessary, drag the edit box to the approximate location.

4 Create the label for the "Amount" edit control:

- Select **Control-Label**, or click the **Label** button , or right-click where you want the label to appear and select **Label**.
- Enter **Amount** in the Text field of the Label Properties dialog box, then click **OK**.
- If necessary, drag the label to the approximate position, then resize it.

5 Save your work:

- Select **File-Save** or click the **Save** button .
The Save As dialog box appears.
- If necessary, change to the folder in which you want the file saved.
- Enter **MYAPP.VCE** in the File Name field, then click **Save**.
Visual CE saves the application to the MYAPP file. The extension .vce indicates a Visual CE application.

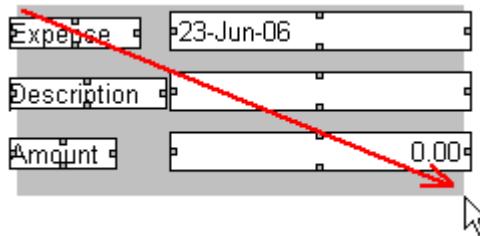
You have created all of the edit boxes required on the Expense form.

Next: Arranging Controls on the Form

Quick Lesson 7a: Arranging Controls on the Form

Now that you've created a few controls, you may want to arrange them more precisely so that your form has a neat appearance. In this lesson, you'll learn about Visual CE's multiple-select and align features, as well as how to specify the exact position of a control in pixels.

- 1 Align the controls horizontally:
 - Click the date edit box
 - While pressing the Control key, click on the Date button. Notice that the handles around the first box you selected are solid black, while those around the second box are empty. Visual CE aligns the controls to the first one you select, which is always indicated by the solid boxes.
 - Select **Edit-Align-Bottom**. The bottom edges of the boxes are now aligned.
- 2 Repeat this process for the Description command button and edit box, and the Amount label and edit box.
- 3 Specify a precise horizontal location for the Date edit box:
 - Right-click (or double-click) the date edit box. Visual CE displays the Edit Properties dialog box.
 - In the "Left" field, type **80**. This is the distance in pixels that Visual CE will locate the control from the left edge of the form.
 - Click **OK**.
- 4 Align the three edit boxes vertically along their left edges:
 - Click the edit box you just positioned, so that Visual CE will use its location as the guide.
 - Select the other two edit boxes (click the edit boxes while pressing the Control key)
 - Select **Edit-Align-Left**.
- 5 Make all of the edit boxes a uniform width:
 - Click on the date edit box to make it "active"
 - Drag its right handle so that the box is the approximate width shown on the sample
 - While pressing the Control key, click the other two edit boxes.
 - Select **Edit-Align-Width**.
- 6 Drag the entire group of controls to a desired position:
 - First, select all of the controls using one of the following methods:
 - ◊ Select **Edit-Select All**, or
 - ◊ While pressing the Control key, click each of the edit boxes and labels, or
 - ◊ Left-click the mouse and drag over the controls. The selection area is indicated by a gray box. Release the mouse button. (This method works the same as a Windows marquee select.)



- Then, position the mouse pointer over any one of the selected controls, and press and hold the mouse button.
- Drag the group horizontally and vertically. Notice that the relative position of individual controls stays the same.

- To return the group to their original position (if desired), select **Edit-Undo**.
- To unselect the group, double-click in the development window away from the group.

Tip

If you have selected a group of controls, and want to deselect any individual control in the group, press the Control key and click the control.

As you create additional controls in the remainder of this tour, you can apply these techniques to quickly organize the layout of your form.

Next: Creating a Calculated Field and Button

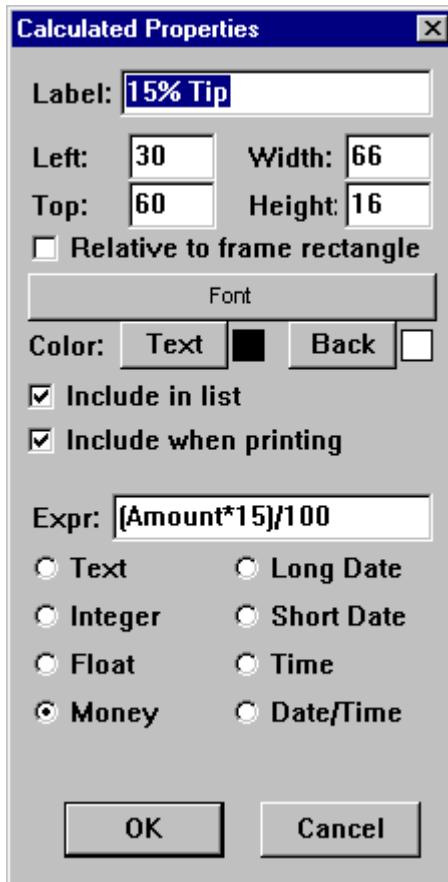
Quick Lesson 8a: Creating a Calculated Field and Button

Calculated fields are great time-savers for anyone who uses your application. They can automatically apply an operation to a field value and save the results. In this lesson, you will set up a control that automatically calculates a 15% tip on the amount of the expense.

You will also set up a command button that users click to apply the calculation.

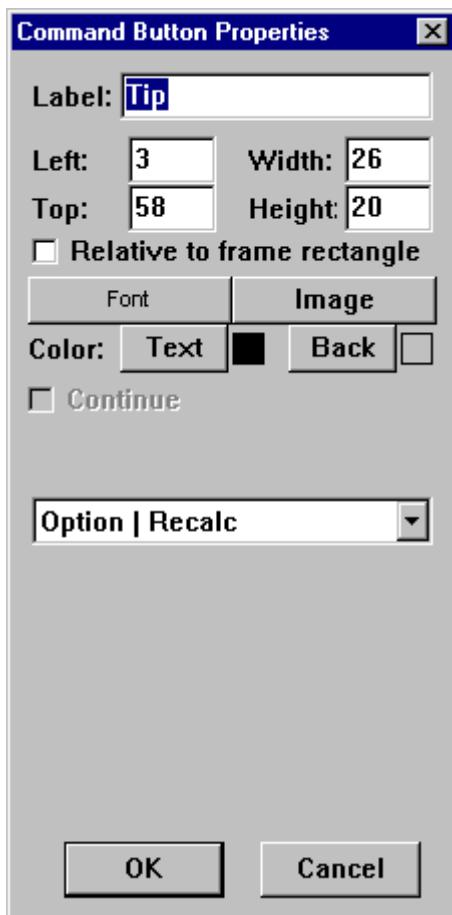
- 1 Select **Control–Calculated**, or click the **Calculated** button , or right-click where you want the control to appear and select **Calculated**.
- 2 Visual CE asks if you want the result of the calculation to be saved in a column in the table. For the purposes of this tutorial, click **No**.

Visual CE displays the Calculated Properties dialog box.



- 3 In the Calculated Properties dialog box:
 - Make sure that **Relative to frame rectangle** is *not* selected
 - Enter **Amount * 15/100** in the Expr field
 - Select the **Money** option to specify the display format for the results
 - Click **OK**
- 4 If necessary, drag the field to the correct location.
- 5 Create a Tip button that users click to do the recalculation:
 - Select **Control–Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.

The Command Button Properties dialog box opens.



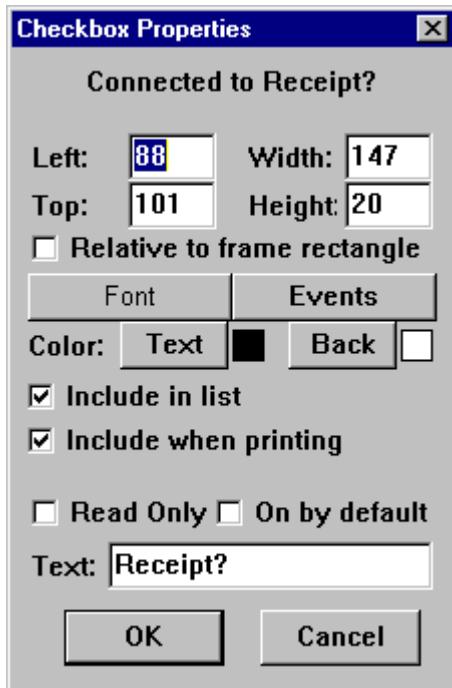
- For Label, enter **Tip**
- Make sure that **Relative to frame rectangle** is *not* selected
- In the drop-down menu, select **Option | Recalc**
- Click **OK**
- If necessary, drag the button to the correct position, then resize it

Next: Creating a Checkbox

Quick Lesson 9a: Creating a Checkbox

Checkboxes take only seconds to create.

- 1 Select **Control–Checkbox**, or click the **Checkbox** button , or right-click where you want the control to appear and select **Checkbox**.
Visual CE displays the Select Column dialog box.
- 2 Click **Receipt**, then click **OK**.
Visual CE displays the Checkbox Properties dialog box.



- 3 Enter **Receipt?** in the Text field, then click **OK**.
- 4 If necessary, drag the checkbox to the correct position, then resize it.

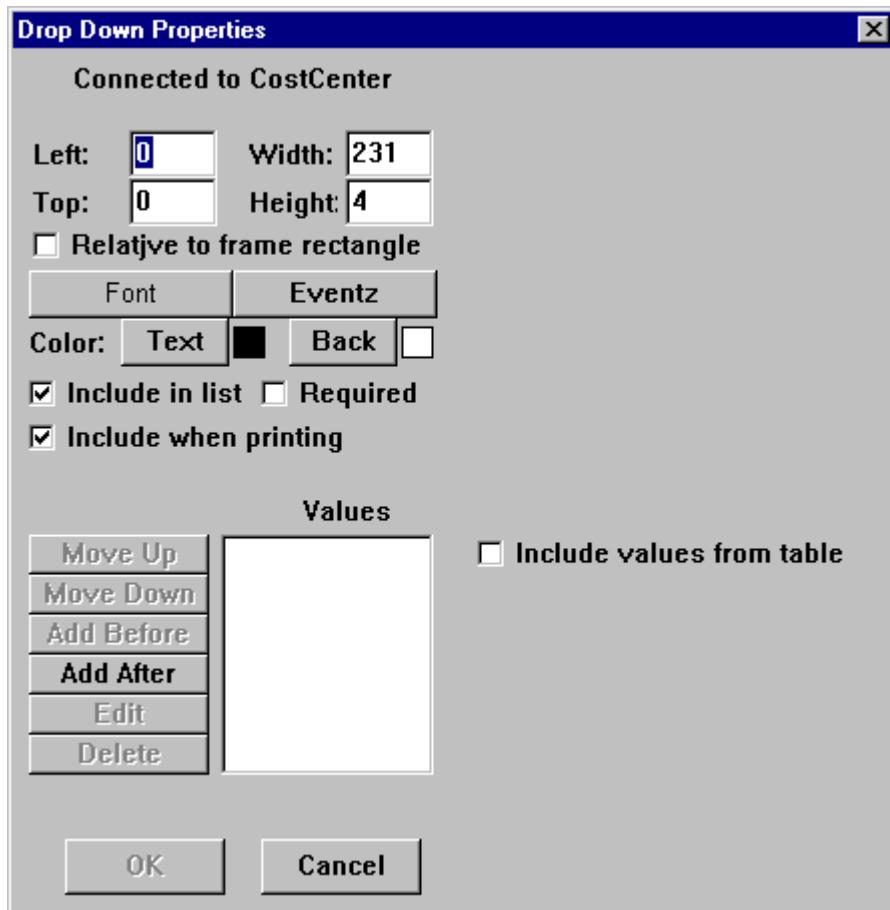
As you created the checkbox, you created the caption (Receipt?). You do not need to create labels for checkboxes.

Next: Creating a Drop-Down List

Quick Lesson 10a: Creating a Drop-Down List

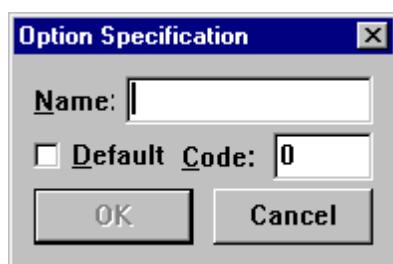
In this lesson, you will create a drop-down list containing six options. One option will be defined as the default.

- 1 Select **Control—Drop Down**, or click the **Drop Down** button , or right-click where you want the control to appear and select **Drop Down**.
Visual CE displays the Select Columns dialog box.
- 2 Click **CostCenter**, then click **OK**.
Visual CE displays the Drop Down Properties dialog box.



- 3 Click **Add After**.

Visual CE displays the Option Specification dialog box.



4 Do the following to create the first value in the drop-down list:

- Enter **Sales** in the Name field.
- Click **Default** to select the Sales option as the default of the drop-down list.
- Enter **10045** in the Code field.

Note

Codes are always/only used when the drop-down is connected to a numeric column in the table. Cost Center is a numeric column. Therefore, you need to provide a numeric code for each drop-down caption you enter. The user sees the name (in this case, Sales) on the form, but Visual CE stores the more compact numeric code (in this case 10045) in the table.

- Click **OK**.

5 Now create the remaining values in the drop-down list:

- Click **Sales** to highlight it, then click **Add After**.
- Enter **Marketing** in the Name field and **14578** in the Code field. Click **OK**.
- Click **Add After**.
- Enter **Production** in the Name field and **14902** in the Code field. Click **OK**.
- Click **Add After**.
- Enter **Admin** in the Name field and **18012** in the Code field. Click **OK**.
- Click **Add After**.
- Enter **R&D** in the Name field and **14013** in the Code field. Click **OK**.
- Click **Add After**.
- Enter **Other** in the Name field and **19999** in the Code field. Click **OK**.

6 Click **OK** in the Drop Down Properties dialog box to close it.

7 Drag the drop-down list control to the correct location.

8 Now create a label for the drop-down list:

- Select **Control-Label**, or click the **Label** button , or right-click where you want the label to appear and select **Label**.
- Enter **Cost Center** in the Text field of the Label Properties dialog box, then click **OK**.
- If necessary, drag the label to the correct position, then resize it.

9 To save your work, click the **Save** button.

Next: Creating a Drop-Down List

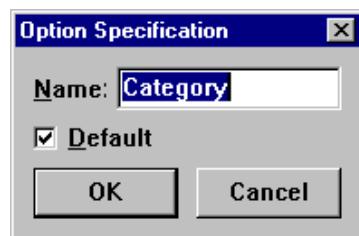
Quick Lesson 11a: Creating Radio Buttons

Now you will create a group of three radio buttons, one of which is defined as the default.

- 1 Select **Control–Radio Buttons**, or click the **Radio Buttons** button , or right-click where you want the control to appear and select **Radio Buttons**.
Visual CE displays the Radio Buttons Properties dialog box.
- 2 Click **Category**, then click **OK**.
Visual CE displays the Select Column dialog box.



- 3 In the Radio Buttons Properties dialog box:
 - Click the **Edit** button near the bottom of the dialog box.
Visual CE displays the Option Specification dialog box.



- Remove the word **Category** from the Name field and enter **Travel** in its place. (Leave the **Default** box checked to indicate that Travel will be selected as the default.) Click **OK**. Visual CE displays the word **Travel** in the Values column – the asterisk (*) indicates that it will be the default value in the form.

- Click **Travel** to highlight it, then click the **Add After** button.
 - Enter **Food** in the Name field, and click **OK**.
 - Click the **Add After** button.
 - Enter **Other** in the Name field, and click **OK**.
 - Click the **Border** checkbox (near the bottom of the dialog box) to place a box around the radio buttons in the form.
- 4 Click **OK** in the Radio Button Properties dialog box to close it.
- 5 If necessary, drag the control to the correct location, and resize it so all the radio buttons and their names can be clearly read.
- 6 You may want to save your work again by selecting **File—Save** or clicking the **Save** button.

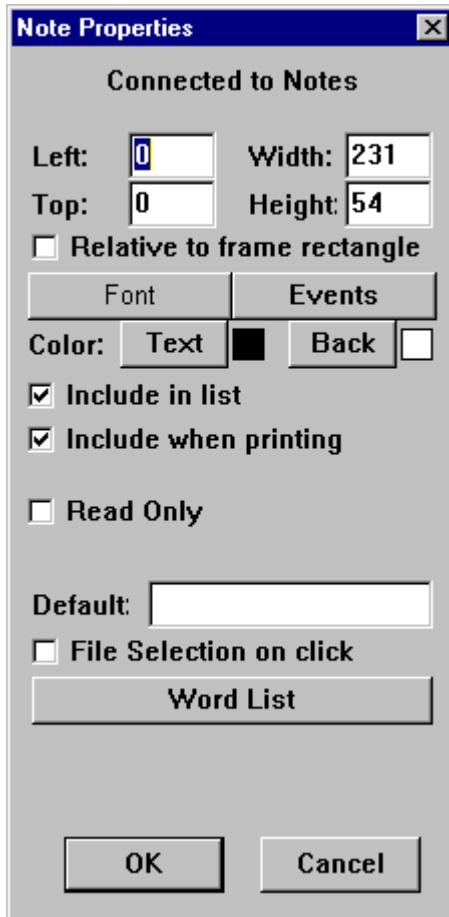
Because you specified the radio button captions (Travel, Food, etc.) as you created the buttons, you do not need to create labels for them.

Next: Creating a Note Box

Quick Lesson 12a: Creating a Note Box

A note box is really just an edit box that allows multiple lines of text – an edit box allows only a single line of text.

- 1 Select **Control–Note**, or click the **Note** button , or right-click where you want the control to appear and select **Note**.
Visual CE displays the Select Column dialog box.
- 2 Click **Notes**, then click **OK**.
Visual CE displays the Note Properties dialog box.



- 3 Click **OK** to accept the default values.
- 4 If necessary, drag the note control to the correct location.
- 5 Now create a label for the note control:
 - Select **Control–Label**, or click the **Label** button , or right-click where you want the label to appear and select **Label**.
 - Enter **Notes:** in the Text field, then click **OK**.
 - If necessary, drag the label to the correct location.

Next: Creating Command Buttons to Move to Different Regions of the Form

Quick Lesson 13a: Creating Command Buttons to Move to Different Regions of the Form

Now you will create two command buttons that will enable users to quickly jump to predefined regions of your form. You'll label the buttons "Expense" and "Limits."

1 Select **Form-Form Size**.

Visual CE displays the Form Size dialog box.

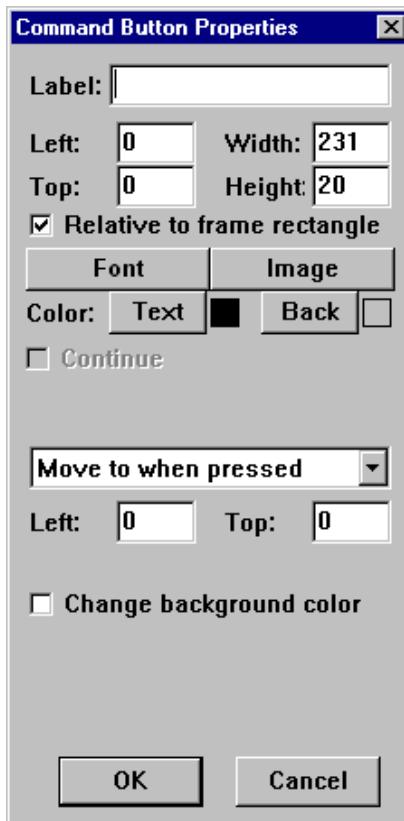
2 Double the value for width and click **OK**. (Note: the values shown here are not necessarily appropriate for your device.)



3 The position of the dotted-line frame specifies the region of the form that will be displayed when the form's user presses the first command button. It should be in upper left corner of the development window. If it is not already there, reposition it by pressing the Shift key on your keyboard and dragging it into place.

4 Create the first command button:

- Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command Button**. Visual CE displays the Command Button Properties dialog box.



- In the Label field, enter **Expense**.

- Since you want the "Expense" button always to be visible regardless of which region of the form is being displayed, leave "Relative to frame rectangle" checked.
 - Accept the remaining defaults and click **OK**.
 - If necessary, drag the command button to the correct location (relative to the frame or display), and resize it.
- 5 Specify the region of the form to display when the user clicks the second command button:
- Scroll down to the bottom of the form.
 - While pressing the Shift key on your keyboard, drag the dotted-line frame to the lower left corner of the form.
- 6 Enter text to appear in this new region of the form:

- Select **Control-Label**, or click the **Label** button , or right-click where you want the label to appear and select **Label**.
- Enter the text (shown below) in the Text field. You can use spaces and carriage returns to format the text. Use the up and down arrow keys on the keyboard to move between lines without making changes. **Tip:** Each digit is the equivalent in width to two blank spaces. So, for example, to align the decimal points in the sample text, press the Space bar four times before typing "8.00 Breakfast," two times before "10.00 Lunch," and no times before "150.00 Hotel." Characters are variable widths.

Expense Limits

```

8.00 Breakfast
10.00 Lunch
35.00 Dinner
150.00 Hotel
30.00 Car rental (daily)
50.00 Entertainment (including client dinner)

```

- When you have completed the text, click **OK**.
 - Resize the box so that all of the text is visible, and drag it into position.
- 7 Create the "Limits" command button in the same way you created the "Expense" tab button in step 4 above.
- 8 Confirm that the command button displays the desired region:
- While pressing the Shift key on your keyboard, drag the dotted-line frame to its original position. Release the Shift key.
 - While pressing the Shift key on your keyboard, click the **Limits** tab button on the form display.
- The frame or display moves to the new position, indicating the area that will be displayed on the CE device. (You may have to scroll the development window's display to view the frame's new position.)
- 9 To save your work, click the **Save** button.

Tip

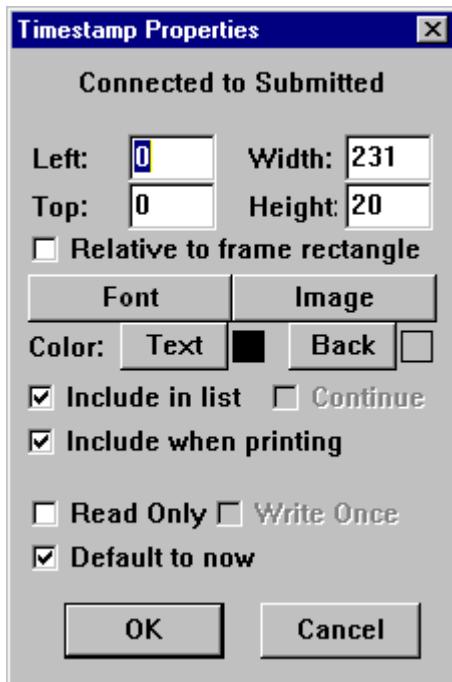
When creating your own applications, as an alternative to command buttons that jump to different regions of the form, you can divide your form into any number of tabbed pages. That technique is described in the topic "Adding Tabs", on page 164.

Next: Creating a Timestamp Button

Quick Lesson 14a: Creating a Timestamp Button

In this lesson, you will create a button that the user clicks to "stamp" the record with a date and time.

- 1 Select **Control-Timestamp**, or click the **Timestamp** button , or right-click where you want the control to appear and select **Timestamp**.
Visual CE displays the Select Columns dialog box.
- 2 Click **Submitted**, then click **OK**.
Visual CE displays the Timestamp Properties dialog box.



- 3 Click the **Default to now** checkbox to remove the checkmark, and click **OK**.
- 4 If necessary, drag the button to the correct location.
- 5 Save your work by clicking the **Save** button.

Next: Creating a Title for the Application

Quick Lesson 15a: Creating a Title for the Application

The application's title will display in the title bar in both the development environment window and the application window.

- 1 Select **Form–Title Bar**.

Visual CE displays the Title Bar dialog box.



- 2 Enter **My Expense Tracker** in the Title field, then click **OK**.

Visual CE displays your title in the title bar.

Next: Specifying an Initial Sort/Search Order

Quick Lesson 16a: Specifying an Initial Sort/Search Order

You have the option to specify an initial sort/search order for record display (the default is unsorted records).

As you saw in the first Quick Tour, the records in the sample application are initially ordered based on the entry in the Date field.

1 Select **Form-Sort On**.

Visual CE displays the Select Column dialog box. Notice that the list includes the columns that you "indexed" in Quick Lesson 2a. (Only indexed columns can be used for sorting.)

2 Make sure that the **Not Sorted** checkbox is unchecked.

3 Click **Date** to select sorting based on the value in the Date column, then click **OK**.

That's all there is to specifying an initial sort/search order.

Next: Specifying the Tab and Column Order

Quick Lesson 17a: Specifying the Tab and Column Order

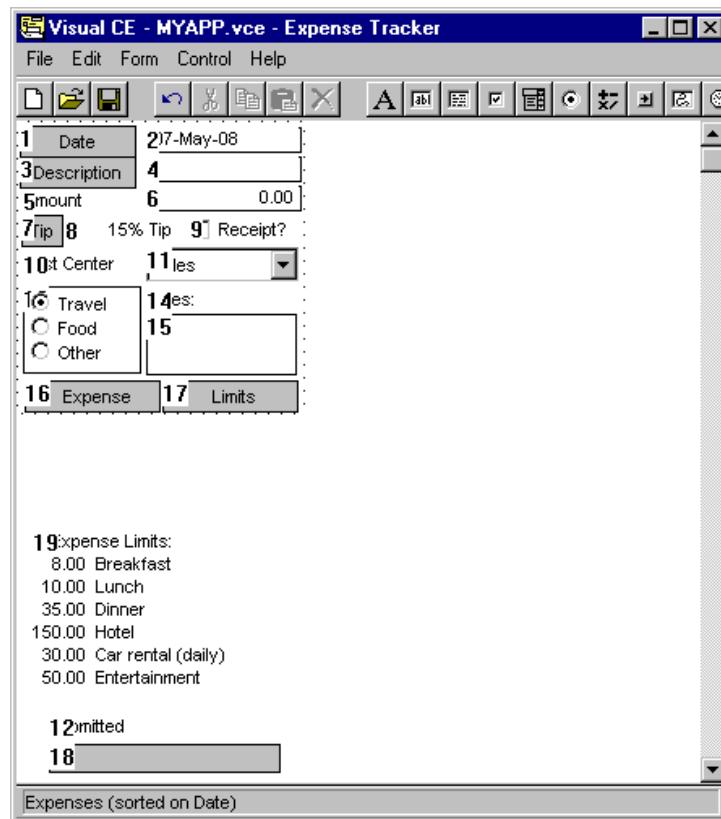
You can specify the order in which the cursor moves from field to field when the up/down rocker key is pressed.

1 Select Form–Control order.

Visual CE displays the Control Order dialog box on top of the form. This dialog box lists the controls in the order you created them. You can drag the dialog box to the side to view the form.

On the form, Visual CE displays a number next to each control you created. The numbers on the form match the order in the dialog box.

You can change the control order using the dialog box, or the form, or both.



Note

All of the controls are numbered, but when the application is used, the cursor tabs only to those fields in which user entries are allowed.

- 2 In the Control Order dialog box, select one of the controls and click the Move Up or Move Down button as many times as you like.

Visual CE moves the selected control in the sequence. Notice how the numbering displayed in the form is automatically updated to match the sequence in the Control Order dialog box.

You can repeat this technique with the other controls until the entire list is reordered as desired.

- 3 An alternative way to reorder the controls is by clicking them one at a time in the form layout. Try it now:

- On the form, click the first control – the one in which you want the cursor to appear initially.
Visual CE changes that control's number to "1."
- Click the control you want the cursor to move to next.
Visual CE changes that control's number to "2."
- Continue to click each control to indicate the order you want.

Tips

- If you make a mistake, click the **Cancel** button in the Control Order dialog box.
Alternatively, select **Edit–Undo**.
- 4 When you have finished specifying the control order, click **OK** in the Control Order dialog box, or click anywhere in the environment window (except on a control).
- 5 Select **File–Save** or click the **Save** button to save your work.

Congratulations! You have created the Expenses sample application, including most of the controls provided by Visual CE.

Next: Downloading Your Application

Quick Lesson 18a: Downloading Your Application

Each time you save your application, Visual CE saves the .vce file on your desktop PC. When you have finished creating the application, you need to download the file to your CE device.

- 1 In Visual CE on your desktop PC, select **File–Download Form**.
Visual CE displays the Save dialog box.
- 2 Enter **MYAPP.VCE** in the Name field, then click **OK**.
Visual CE downloads the application to your CE device within seconds – the file size is very small.

Next: Testing Your Application

Quick Lesson 19a: Testing Your Application

Test the application on the Smartphone to be sure that it works the way you intended.

- 1 On the Smartphone, click **Start–MYAPP**.

Visual CE displays your Expense Tracker in the application window.

- 2 Test your application to be sure that it works correctly. You can use the instructions given in the application use tutorial (Quick Tour Ia) to test it.

- 3 If your application works correctly, you have done a great job! If not, you have probably made a simple mistake or two.

To correct any mistakes, close the application on the Smartphone and return to the Visual CE development window. If you made a mistake with a control, delete the control and recreate it. If your mistake concerned the title bar, sort/search order, or tab order, respecify the option correctly. After making your changes, click the **Save** button to save your work, then download it to the Smartphone by selecting **File–Download Form**. Then repeat the test instructions above.

You have finished Quick Tour IIa. You are already a Visual CE expert!

Next:

Take Quick Tour III (page 99) which goes step by step through the process of downloading a table from the desktop PC to the CE device and synchronizing the contents of the two tables, Quick Tour IV (page 106) which introduces macros and events, and Quick Tour V (page 116) which explains how to track down mistakes in a macro.

Or, you can begin now to create your own applications using Visual CE (page 124).

Quick Tour III: Sharing Data with Desktop Database Applications

Overview of Quick Tour III

In this Quick Tour, you will learn how to create a Visual CE application that utilizes data from an existing Microsoft Access table on the desktop PC; as well as how to set up synchronization between the Access table on the desktop PC and the new table on the CE device.

Note

- In order to take this Quick Tour, you must have a 32-bit version of Microsoft Access installed on your desktop PC (Access 95 or higher).

This tour consists of two lessons:

- Quick Lesson 1: Downloading a Table to Create a New Application
- Quick Lesson 2: Looking at the Synchronization Options

Quick Lesson 1: Downloading a Table to Create a New Application

In this lesson, you will download an existing Microsoft Access table and data from the desktop PC to the CE device, and quickly create a Visual CE application for it. You will also set up synchronization for the two tables so that the contents of the table on the desktop PC stay current with those on the CE device, and vice versa.

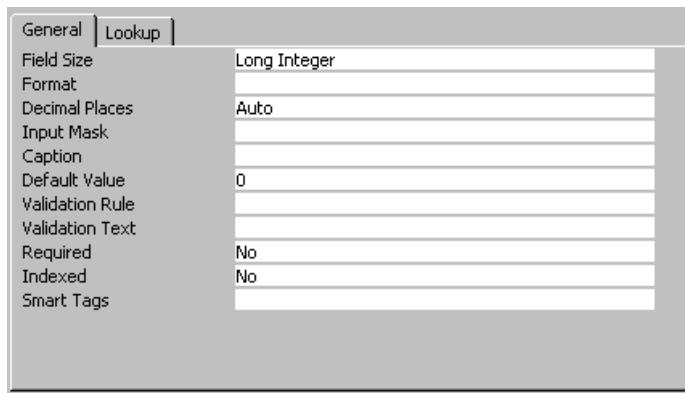
For the purposes of this lesson, SYWARE has provided a sample Access database, "Tutorial.mdb," that contains a table called "Phonebook."

1 Before you can apply synchronization to an Access table on the desktop PC for the first time, you must add an "OID" column to it. (Visual CE automatically adds a corresponding column to the Windows CE table on the CE device and uses the columns to keep records in the two tables in sync.) To do this:

- Start Microsoft Access on the desktop PC
- Open **Tutorial.mdb** (typically stored in \Program Files\Windows CE Services\SYWARE Visual CE)
- Select **Phonebook** table and go to the Design view
- Add a new field definition to the *top of the list*, with the following specifications:
Field Name: OID
Data Type: Number

	Field Name	Data Type
▶	OID	Number
	Name	Text
	Phone	Text

- ◊ In the General tab in the Access Design view, turn off indexing. Leave the remaining default values in place.



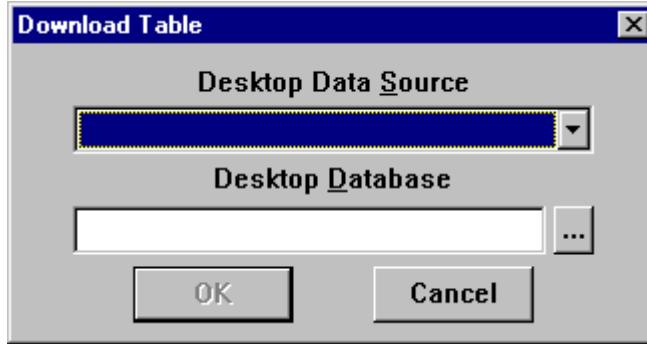
- ◊ The value stored in the OID column for the existing records must be zero or null (blank). (You can verify this by looking at the table in Spreadsheet view.)

	OID	Name	Phone
▶		John Smith	555-1234
		Mary Jones	555-4321
*	0		

- Save your changes and close Access.

- 2 Make sure that the CE device is connected to your desktop PC.
- 3 Start Visual CE's development environment on the desktop PC (**Start-Programs-Visual CE-Visual CE**).
- 4 Select **File-Download Table**.
- 5 Visual CE asks if you want to establish synchronization between the tables. For the purposes of this lesson, click **Yes**.

Visual CE displays the Download Table dialog box.



- 6 Establish a connection with the desktop PC database that contains the table you want to download to the CE device:
 - Click on the arrow to the right of the "Desktop Data Source" field, and select **MS Access Database**
 - Click the Browse button ..., and navigate to the folder that contains Tutorial.mdb. (By default, it's the same folder in which you installed the Visual CE program, typically \Program Files\Windows CE Services\SYWARE Visual CE.)
 - Highlight **Tutorial.mdb** and click **Open**
 - Click **OK** to close the Download Table dialog box

Visual CE displays the Select Table dialog box.



- 7 Highlight **Phonebook** and click **OK**.
- 8 Visual CE asks if the table will be synchronizing to multiple handhelds. For the purposes of this tutorial, click **No**.
Visual CE downloads the table, and all of the records it contains, from the desktop PC to the CE device.
- 9 Visual CE asks if you would like the Form Wizard to create a form for you. Click **Yes**.

- 10 Visual CE asks if you would like to download the form to your CE device. Click **Yes**.
The Save dialog box appears.
- 11 Enter **Phonebook.vce** (or another name of your choosing for the Visual CE application) in the Name field, then click **OK**.
- 12 Visual CE displays a suggestion that you customize the form. Click **OK** to acknowledge the message.
- 13 Close Visual CE by selecting **File-Close**.
Visual CE asks if you want to save your Visual CE application on the desktop PC. For the purposes of this tutorial, click **No**.

That's all there is to it! You have created a Visual CE application for the CE device that utilizes the same table structure and data as an Access table on your desktop PC, and set up synchronization for the two tables using Visual CE's default synchronization settings.

In the next lesson, you'll learn how to view the synchronization settings for the tables.

Next: Looking at the Synchronization Options

Quick Lesson 2: Looking at the Synchronization Options

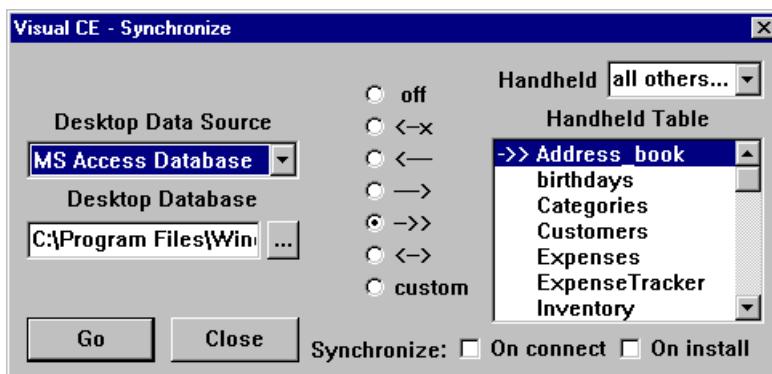
If you plan to use a table on both the desktop PC and the CE device, you'll want to take advantage of Visual CE's synchronization feature. Using this feature, changes made to the contents of one table are automatically (or on command) copied to the other.

In this lesson, you'll learn how to view the synchronization settings. These settings allow you to specify whether updates are applied in just one or both directions, and to fine-tune how Visual CE handles specific differences between records in the two tables. You can actually set up synchronization for any number of pairs of desktop/handheld tables and apply different synchronization settings to each pair. You can also set up synchronization between one desktop table and the corresponding table on multiple handheld devices.

1 Open Visual CE on your desktop PC.

2 Select **File-Synchronize**.

The Synchronize dialog box appears. It allows you to view all of the synchronization settings currently in place, as well as to modify them and launch synchronization.



The fields on the left side of the dialog box pertain to the desktop PC tables; those on the right, to the tables on the CE device.

- **Desktop Data Source** and **Desktop Database** together specify a connection to the ODBC-enabled data source on the desktop PC. All of the desktop PC tables you want to synchronize must be in this one data source, which is typically an Access database (.mdb or .accdb file). **Note:** Once you have completed the Quick Tour, you can specify a different data source for your own desktop PC tables. But you must keep all of the tables you want to synchronize in that one new data source from then on.
- **Handheld Table:** This listbox contains the names of all tables on the CE device that are eligible for synchronization. The symbol to the left of the table name in the listbox indicates the synchronization option currently specified for that table (Visual CE automatically applies the specified option to each table every time synchronization occurs). The symbols have the following meanings:

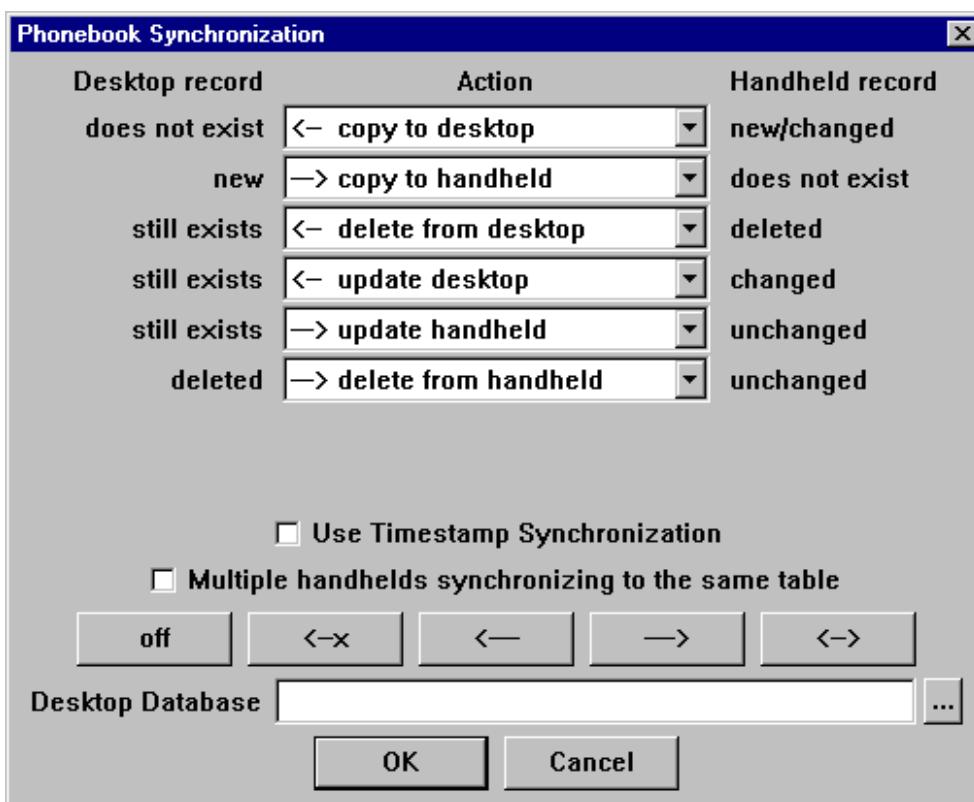
[blank]	Same as "off." Prevent synchronization of the Windows CE table with the desktop PC table. Any changes made to records in one table are NOT applied to the other.
<-x	Copy new or modified records from the Windows CE table to the desktop PC table, and delete them from the Windows CE table (e.g., move records from the handheld to the desktop PC).
<-	Apply all changes to records in the Windows CE table to the desktop PC table. Changes in the desktop PC table are NOT applied to the Windows CE table.
→	Apply all changes to records in the desktop PC table to the Windows CE table. Changes in the Windows CE table are NOT applied to the desktop PC table.
→>	Publish. Replace all records in the Windows CE table with all records from the desktop PC table.

- ↔ Apply all changes to records in the Windows CE table to the desktop PC table, and vice-versa (e.g., all changes appear in both tables.)
- Apply a custom option that you specify (explained in Step 3).

Notice that Visual CE automatically selected two-way synchronization (<->) for the Phonebook table that you downloaded in the previous lesson.

- **Synchronize on Connect:** This option lets you specify whether Visual CE automatically synchronizes the data in the tables according to the options specified in the Handheld Tables list whenever the CE device is connected, or just when you open this dialog and explicitly run synchronization. For the purposes of this lesson, leave the box unchecked.
 - **Synchronize on Install:** (Pro and Mobile 360 Editions Only) When this option is checked, the runtime installer automatically performs a synchronization after installing the Visual CE application on the CE device--so that records from the database on the desktop PC are automatically copied into the table on the CE device.
- 3 You can use the radio buttons in the middle of the dialog box to apply a different option to a table highlighted in the Handheld Table list box. The "Custom" button in particular lets you view specific actions associated with each of the basic options, and to fine-tune them to your specific requirements. Try it out:
- Highlight **Phonebook** in the Handheld Table list box, and click the **Custom** radio button (alternatively, you could have double-clicked **Phonebook**)

Visual CE displays the Phonebook Synchronization dialog box, which lists the specific actions currently specified for the Phonebook table (in this case, the actions associated with two-way synchronization). Experiment with the drop-downs in the Action column to view the different options available.



- Click **Cancel** to close the dialog box without saving any modifications you may have made. You are back at the Synchronize dialog box.
- 4 Click **Close** to close the Synchronize dialog box without starting synchronization.

5 Close Visual CE by selecting **File-Close**.

You've now explored the synchronization options offered by Visual CE. The section, "Synchronizing Tables" (page 434) provides additional details for further fine-tuning them to your specific circumstances.

Next, Quick Tour IV introduces two of Visual CE's most powerful features for making your applications dynamic and easy-to-use: macros and events.

Quick Tour IV: Getting Acquainted with Macros and Events

Overview of Quick Tour IV

This Quick Tour introduces two of Visual CE's most powerful advanced features: macros and events. Macros are scripts you can write and add to your forms to have virtually any sort of custom processing happen automatically "behind the scenes" – such as multi-step tasks and input validation. Events are actions that can trigger macros – for example, the form opens or the user changes a value in an input field. (Macros can also be launched with command buttons, menu picks, or hardware buttons.)

Macros give you the power and flexibility of a programming language, but tailored specifically to the Visual CE environment. If you are not a programmer, don't worry – the Visual CE Macro Builder guides you through the process of creating, editing, and managing macros.

In this Quick Tour, you will first try out and then build a Visual CE application that includes a macro triggered by an event. The sample application uses a macro to evaluate the input's input for one control, then display a second control only if the option is relevant to the user's choice for the first.

Quick Tour IV consists of the following lessons:

- Quick Lesson 1: Exploring the Sample Application
- Quick Lesson 2: Creating the Form
- Quick Lesson 3: Planning the Macro
- Quick Lesson 4: Opening the Macro Builder
- Quick Lesson 5: Writing a Show/Hide Control Command
- Quick Lesson 6: Adding a Skip Command with an "If" Expression
- Quick Lesson 7: Attaching the Macro to a Control Event

Quick Lesson 1: Exploring the Sample Application

In this lesson, you will open the sample application, Events.vce, simply to see *what* the demonstration macro does. In the following lessons, you will learn *how* the macro works by creating an identical application.

To run the sample application:

1 If you have not already downloaded the sample applications to a CE device, do so now using the following instructions:

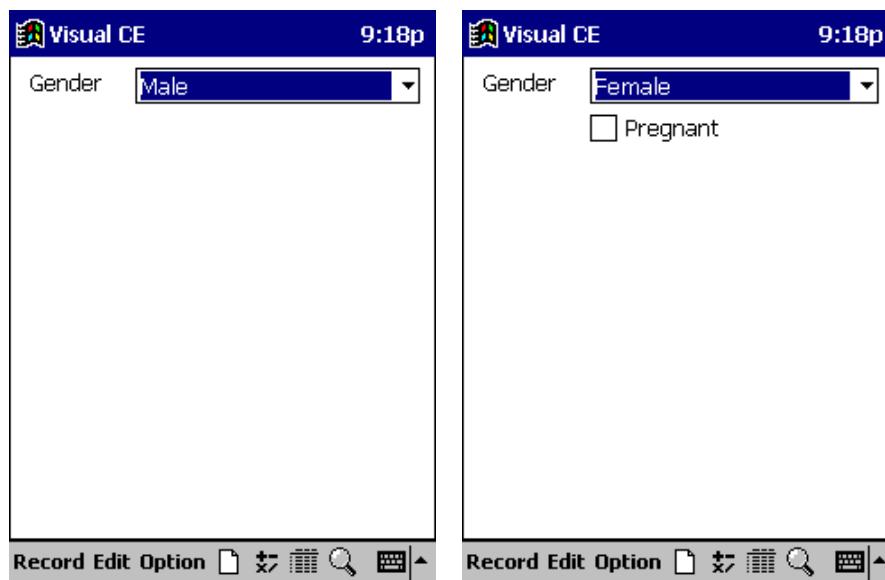
- Connect your CE device to your desktop PC.
 - On the desktop PC, start Visual CE by clicking **Start-Programs-Visual CE-Visual CE**.
 - Select **File-Open**.
 - Navigate to the folder in which you installed Visual CE (by default, it's \Program Files\Windows CE Services\SYWARE Visual CE).
 - Open the .VCE file you wish to download and explore, in this case **Event.vce**.
 - Select **File-Download Form**.
- Visual CE displays the Save dialog box.
- Navigate to the root (top level) directory on the CE device and click **OK**.
- Normally, you will save forms in the default location, which is the Microsoft ActiveSync synchronize folder. The root folder is recommended in this one unique case to ensure that all of the samples stay together.

2 Open Event.vce on the CE device by clicking the Windows **Start** button, then selecting **Programs**, then **Event.vce**.

Visual CE displays the Event application form in its application window.

3 Alternately select the **Male** and **Female** value from the Gender drop-down list.

Note that the Pregnant checkbox is available only when Female is selected. This form behavior is accomplished with a macro script that is triggered by an event.



4 Select **Record-Close**.

Next: Creating the Form

Quick Lesson 2: Creating the Form

Before you build a macro, you should create the form as you would for any other Visual CE application. In this lesson, you will duplicate the form you explored in the previous lesson.

Notes

- Strictly for simplicity's sake, the sample application does not store records and therefore does not have a table. It's most likely that your applications would be built on a table.
- The Macro Builder is available only in the desktop PC version of Visual CE.
- The following instructions assume that you know how to create form controls. If not, be sure to take Quick Tour II or IIA first.

To create the sample form:

- 1 If you have not already done so, connect the CE device to the desktop PC.
- 2 On the desktop PC, select **Start–Programs–Visual CE–Visual CE**.
- 3 Select **File–New Form**.
- 4 When Visual CE asks which database contains the table, for the purposes of this tutorial select **No underlying database**. (Again, this has nothing to do with macros...this is just to keep the sample as simple as possible.)
- 5 Create the Gender Drop-Down List control:
 - Select **Control–Drop Down**.
 - In the Select Column dialog box, select **@var(0)** and **Text**. Click **OK**.

Notes

- `@var(n)` is a global variable you can use to hold a value for processing when you don't want to write the value to a table.
- We selected Text as the datatype because we want to store text values -- either 'Male' or 'Female'.
- In the Drop Down Properties dialog box, click **Add After**.
 - In the Option Specification dialog box, enter **Female** in the Name field and select the **Default** option so that it is checked. Click **OK**.
 - Click **Add After** and enter **Male** in the Name field. Click **OK**.
 - Leave the remaining values in place and click **OK** to close the Drop Down Properties dialog box.
 - Resize and drag the control into place.

- 6 Add a label:
 - Select **Control–Label**.
 - Enter **Gender** in the Text field, and click **OK**.
 - Resize and drag the label into place.
- 7 Create the Pregnant checkbox:
 - Select **Control–Checkbox**.
 - In the Select Column dialog, select **@var(1)** and click **OK**.
 - For Text, replace the default value with **Pregnant**.
 - Resize and drag the control into place.
- 8 Save the application as myEvents.vce, or other name of your choosing.

Next: Planning the Macro

Quick Lesson 3: Planning the Macro

Now you're ready to add the macro to the form. But first, it's always a good idea to plan what you want your macro to accomplish, and how, before you actually build it.

In this example, we want to hide the "Pregnant" checkbox if the user selects Male for Gender, and show it if the user selects Female. We can express these logical steps on paper using language similar to the Visual CE commands that make up a real macro, as follows:

- Step 1: Check the user's selection in the Gender drop-down. If it's Male, skip 0 steps (i.e., go to the next step). If it's Female, skip Step 2 and Step 3.
- Step 2: Hide the Pregnant checkbox.
- Step 3: Return from the macro – that is, stop here.
- Step 4: Show the Pregnant checkbox.
- Step 5: Return from the macro.

In the next lessons, you will use the Macro Builder to express these steps in commands that Visual CE understands.

Next: Opening the Macro Builder

Quick Lesson 4: Opening the Macro Builder

The Macro Builder is a convenient development environment that helps guide you through the process of writing, editing, and managing macros.

To open the Macro Builder:

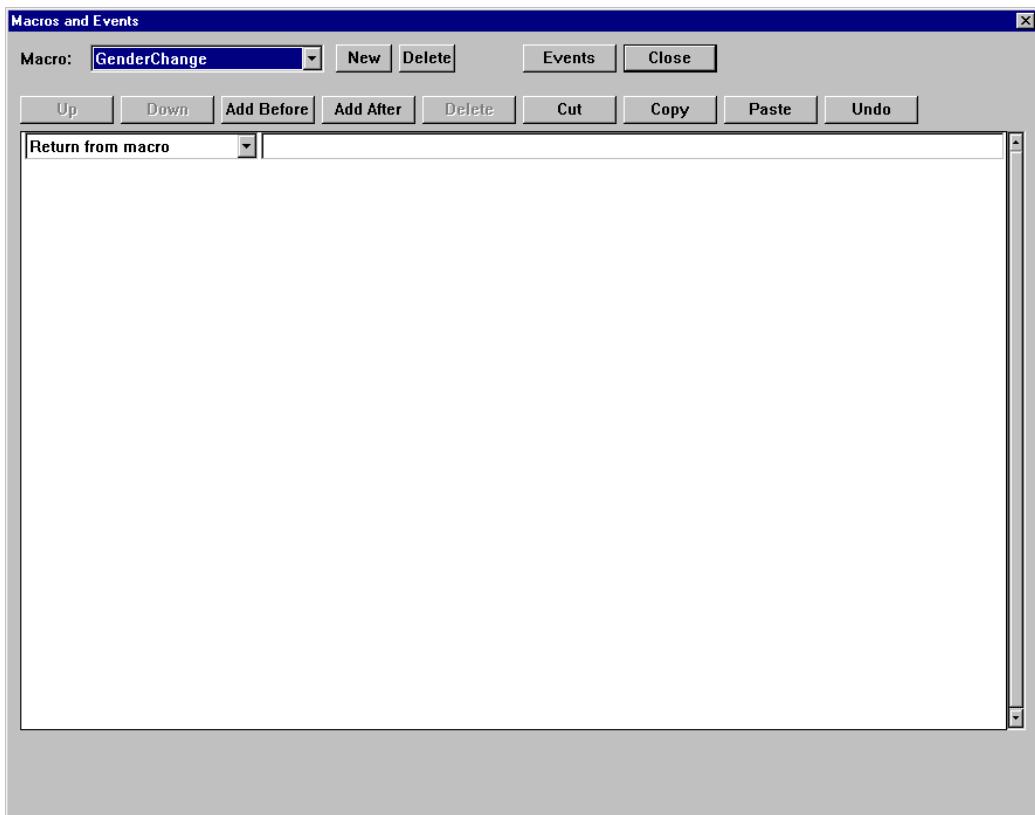
- 1 Select **Edit–Macros / Events**.

Visual CE displays the Macro Name dialog box.



- 2 Enter **GenderChange** and click **OK**.

Visual CE opens the Macro Builder window.



The Macro Builder has two buttons for managing macros: **New** and **Delete**. The **Events** button lets you associate a macro with some types of events. **Close** closes the Macro Editor. The remaining buttons let you organize the individual statements (commands) that make up a macro script. The scripting window (the large white area) is where you write the individual statements and put them in the desired order.

Next: Writing a Show/Hide Control Command

Quick Lesson 5: Writing a Show/Hide Control Command

A macro is a series of commands that execute automatically in order. Visual CE offers more than 50 pre-defined commands that you can use as the building blocks of your macros.

For each step in your macro, all you need to do is select one of the pre-defined command types and fill out the options specific to that command.

In this lesson, you will create two simple “Show/hide control” commands for the Pregnant checkbox. (These commands correspond to Step 2 and Step 4 in the script’s “pseudocode” version on page 109.)

To create a Show/hide control command:

- 1 In the Macro Builder, click **Add Before**.

Visual CE inserts a new statement at the beginning of the script.

Note

Visual CE automatically supplies a “Return from macro” statement. It’s a good practice to always end your macro with one of these statements.

- 2 Select **Show/hide control** from the drop-down list of available commands. (You may have to scroll the list to bring it into view.)

At the bottom of the Macro Builder window, Visual CE displays options for the Show/hide commands.

- 3 For control, select **@var(1)(Checkbox)**.

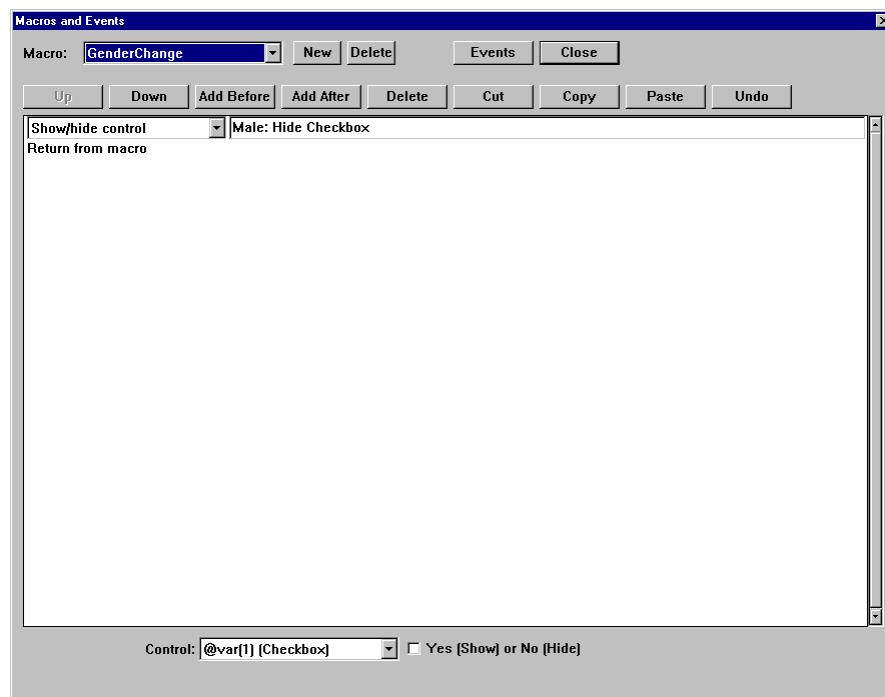
- 4 Leave the **Yes>Show or No Hide** option unchecked.

- 5 In the comments field to the right of the command, enter **Male: Hide Checkbox**.

Note

Comments do not affect how the command runs. Use them as a reminder to yourself of what each command does, in case you need to modify or debug the script later.

You’ve just created a command that, when executed, hides the Pregnant checkbox.



Now, create a second Show/Hide command; this one to show the checkbox. This time we will use the copy/paste feature.

1 With the ‘Show/hide control’ command you just created still selected, click **Copy**.

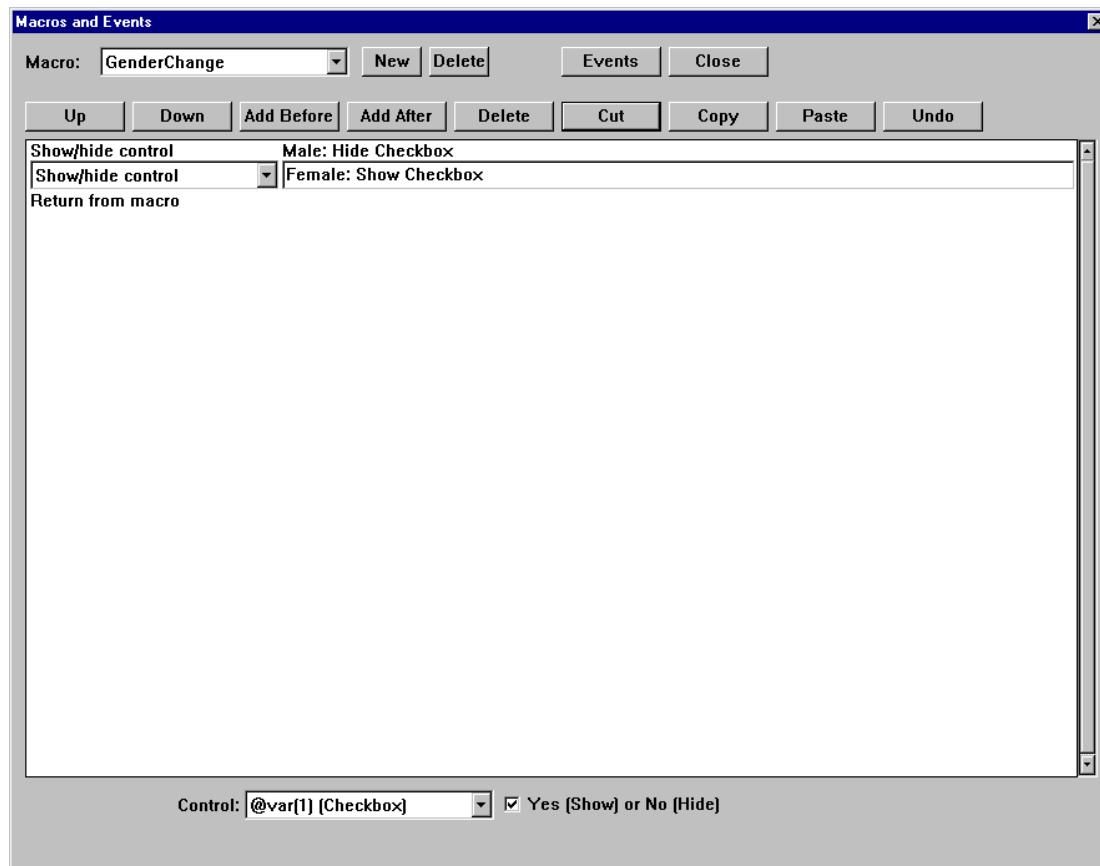
2 Click **Paste**.

The Macro Editor inserts the new command below the original.

3 Update the options for the new command. For Control, select **@var(1)(Checkbox)**. But this time, check the **Yes>Show or No>Hide** option.

4 In the comments field to the right of the command, enter **Female: Show Checkbox**.

At this point, your script should look similar to the one shown below:



Next: Adding a Skip Command with an “If” Expression

Quick Lesson 6: Adding a Skip Command with an “If” Expression

In the simplest macros, the commands execute one after the other in the order they appear in the script. But sometimes you want a more complicated flow, such as conditional branching in which the macro executes one set of commands if a condition is met and another if it is not.

In Visual CE macros, you can accomplish this by using an “if” expression in a Skip command.

In this lesson, you’ll create commands that correspond to Step 1 and Step 3 in the “pseudocode” version of the script on page 109.

To create a Skip command with an “if” expression and a Return from Macro command:

- 1 With the second Show/hide command still selected, click **Add Before**.
- 2 Click **Up** to move the new command to the beginning of the script.
- 3 Select **Skip** from the drop-down list of available commands. (You’ll probably have to scroll the list to bring it into view.)

Visual CE displays the Num field at the bottom of the Macro Builder window.

- 4 In the Num field enter the following expression:

if @var(0) = ‘Male’ then 0 else 2

This expression instructs Visual CE to execute the next command in the script if the value in the global variable @var(0) is ‘Male’; otherwise, to skip two commands.

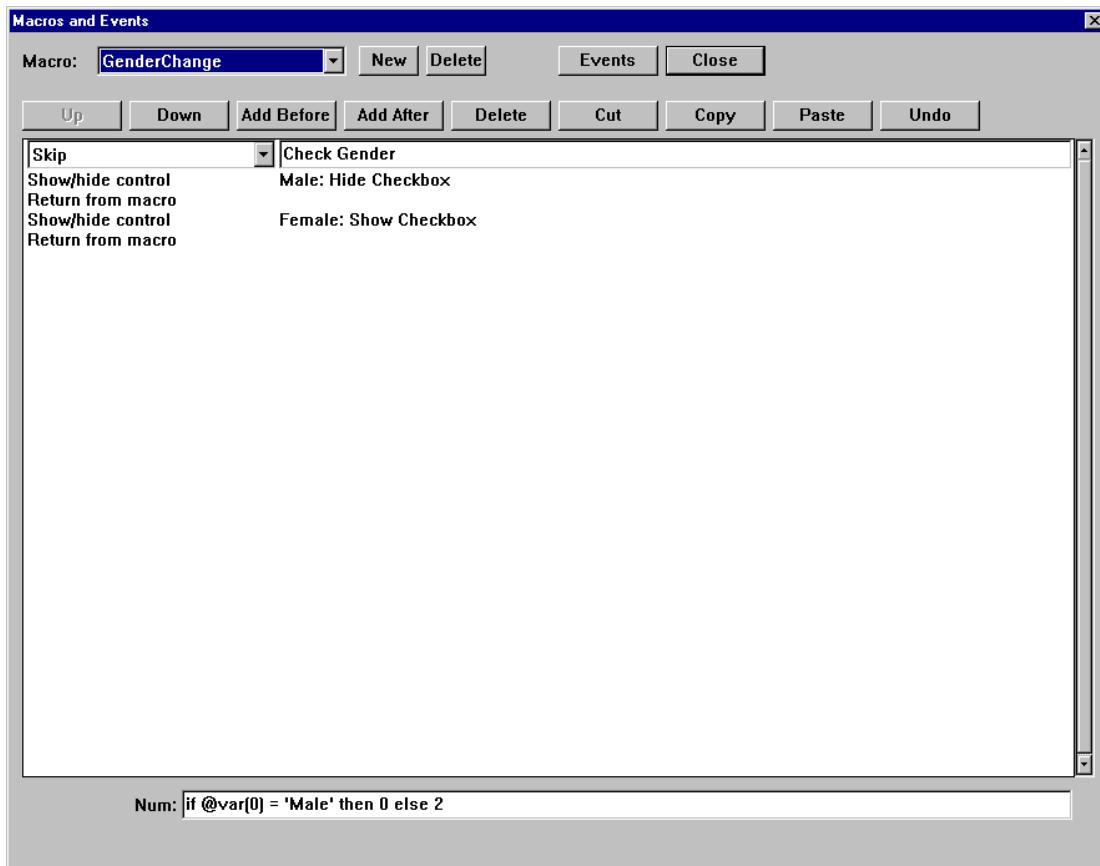
Tip

For fields that accept expressions, you can double-click the field to open a larger area in which to enter or edit the text. This is not particularly useful in this instance, but it can be very helpful when working with long expressions since it enables you to see all of the text at once.

- 5 In the comments field to the right of the Skip command, enter **Check Gender**.
- 6 Select the first Show/Hide control, and click **Add After**.
- 7 Select **Return from macro** from the command drop-down list.

This command stops the script after it hides the checkbox. Otherwise, it would continue running and display the checkbox.

You've now completed all of the commands in the sample script. It should look like the one shown below.



- 8 To close the Macro Builder, click the **Close** button near the top right side of the window.
- 9 To save the macro, you must save the application. Select **File–Save**.

Important!

Macros are saved as part of the application (.vce file). If you close the Visual CE development environment without saving the application, your macro will be lost.

Next: Attaching the Macro to an Event

Quick Lesson 7: Attaching the Macro to a Control Event

There are several ways to run a macro. For example, you can create a command button that the user clicks to start the macro; or you can assign the macro to one of the hardware buttons on the CE device. It's also possible have the macro fire automatically when the user interacts with one of the form controls; that is, when a "control event" occurs.

In this lesson, you'll learn how to have an event trigger the macro automatically. More specifically, you will set it up so that the GenderChange macro runs automatically when the user changes the value in the Gender drop-down list on the form.

To attach a macro to a control event:

- 1 In the form, right-click or double-click the Gender drop-down list control.
The Drop-Down Properties dialog box opens.
- 2 Click the **Eventz** button.
The Events dialog box opens.



- 3 For Changed, select **GenderChange**. Click **OK**.
- 4 Click **OK** to close the Drop-Down Properties dialog box.
- 5 Select **File-Save** to save the form and the macro.

You've now completed the sample macro. Be sure to download and test it on the CE device.

Next:

This quick tour has provided just a small glimpse of the capabilities of macros and events. The possibilities are described in more detail in the reference section beginning on page 303. In addition, be sure to check out the SYWARE website, in particular the Tip of the Month section, <http://www.syware.com/totm>, for sample macros and ideas.

The next Quick Tour explores the Visual CE Macro Debugger, a very helpful tool for tracking down errors in misbehaving macros.

Quick Tour V: Using the Macro Debugger

Overview of Quick Tour V

Once you've created a macro and begun to test it, you may find that it doesn't perform as expected – maybe it returns an erroneous value or it simply stops processing and displays an error message. Especially if your macro consists of many steps and/or sub-macros, identifying the source of the problem can be a difficult and time-consuming process.

Enter the Visual CE Macro Debugger. This visual tool enables you to more quickly find mistakes in your script by letting you "see into" your macro as it runs on the CE device. Working on the desktop PC, it lets you pause the execution at predefined "breakpoints" in the script, as well as advance the execution one step at a time, so that you can examine the values in table columns and variables at various points. Once you have identified the macro step that is causing the error, you can close the Debugger and correct it in the Macro Builder.

In this Quick Tour, you will learn about the Macro Debugger by using it to debug a simple tip calculator that returns erroneous values. The macro is included in a sample application called 'TipCalc.vce,' which was provided by SYWARE as part of the Visual CE installation package.

Note:

The Macro Debugger runs only on the desktop PC.

Quick Tour V consists of the following lessons:

- Quick Lesson 1: Testing the Sample Application
- Quick Lesson 2: Opening the Macro in the Debugger
- Quick Lesson 3: Inserting Breakpoints
- Quick Lesson 4: Running the Macro with the Debugger

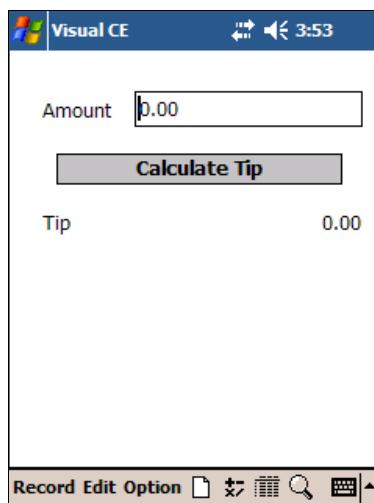
Quick Lesson 1: Testing the Sample Application

In this lesson, you will open and run the sample application, TipCalc.vce, on the CE device, much as you would open and test any new application you have created. This lesson is intended to get you familiar with TipCalc.vce before you begin to debug its macro in the next lesson.

The sample application includes a macro that is intended to calculate a 15% tip on a user-supplied value. The macro is launched by clicking a command button labeled "Calculate Tip."

To run the sample application and its macro:

- 1 If you have not already downloaded the sample applications to a CE device, do so now using the instructions on page 15.
- 2 Open TipCalc.vce on the CE device by clicking Windows **Start–Programs–TipCalc.vce**. Visual CE displays the TipCalc form in the application window.



- 3 Enter an amount in the **Amount** box, and click **Calculate Tip**.

Note that result displayed on the form is not 15% of the amount, as intended. Unless you are a very generous tipper, you'll want to track down and fix the error.

- 4 Select **Record–Close** to close the application on the CE device..

About the Sample Macro

Before getting started with the Debugger, it is useful to understand the macro that you will be debugging. "Under the hood," the sample macro, which is named "Calculate Tip," should consist of the following steps:

Step	Command	Command parameters	Comment
1	Assign	Value: 15/100 Col: @var(2)	Store the desired tip percentage in the global variable @var(2).
2	Assign	Value: @var(0) * @var(2) Col: @var(1)	Multiply the amount entered by the user--which the Amount control stores in @var(0)--by the tip percentage stored in @var(2), and store the result in @var(1).
3	Return from macro		Return processing control to the form, which displays the result.

Next: Opening the Macro in the Debugger

Quick Lesson 2: Opening the Macro in the Debugger

In this lesson, you will open the macro in the Macro Debugger.

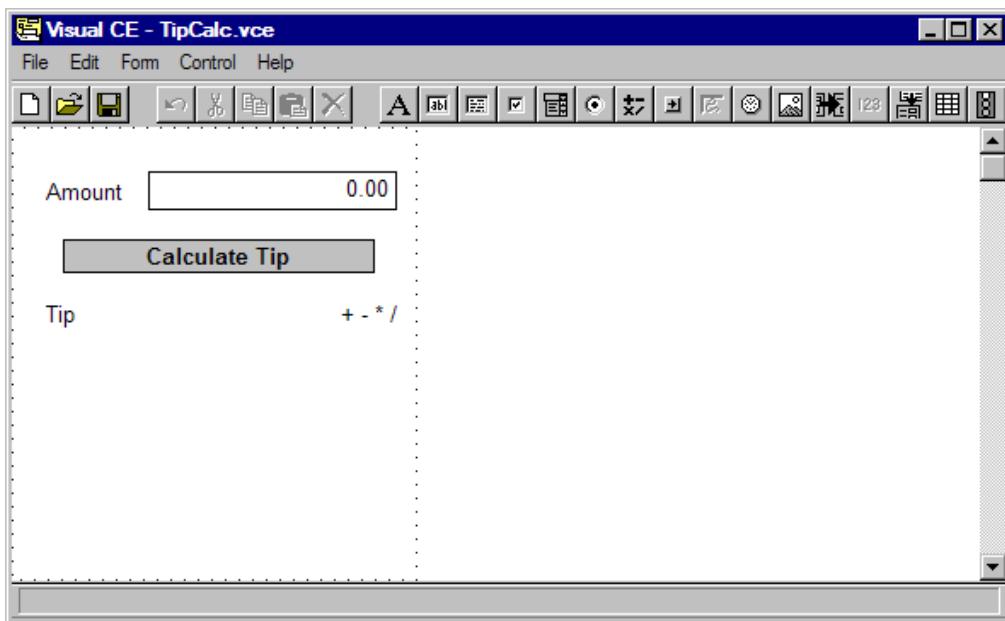
Note:

The Macro Debugger runs only on the desktop PC. The CE device must be attached to the desktop PC throughout the debugging session.

To open the macro in the Debugger:

- 1 If you have not already done so, connect the CE device to the desktop PC.
- 2 On the desktop PC, select **Start–Programs–Visual CE–Visual CE**.
- 3 Open the TipCalc.vce application in the Visual CE development window:
 - Select **File–Open**.
 - Navigate to the folder in which you installed Visual CE (typically,\Program Files\Windows CE Services\SYWARE Visual CE)
 - Open **TipCalc.vce**.

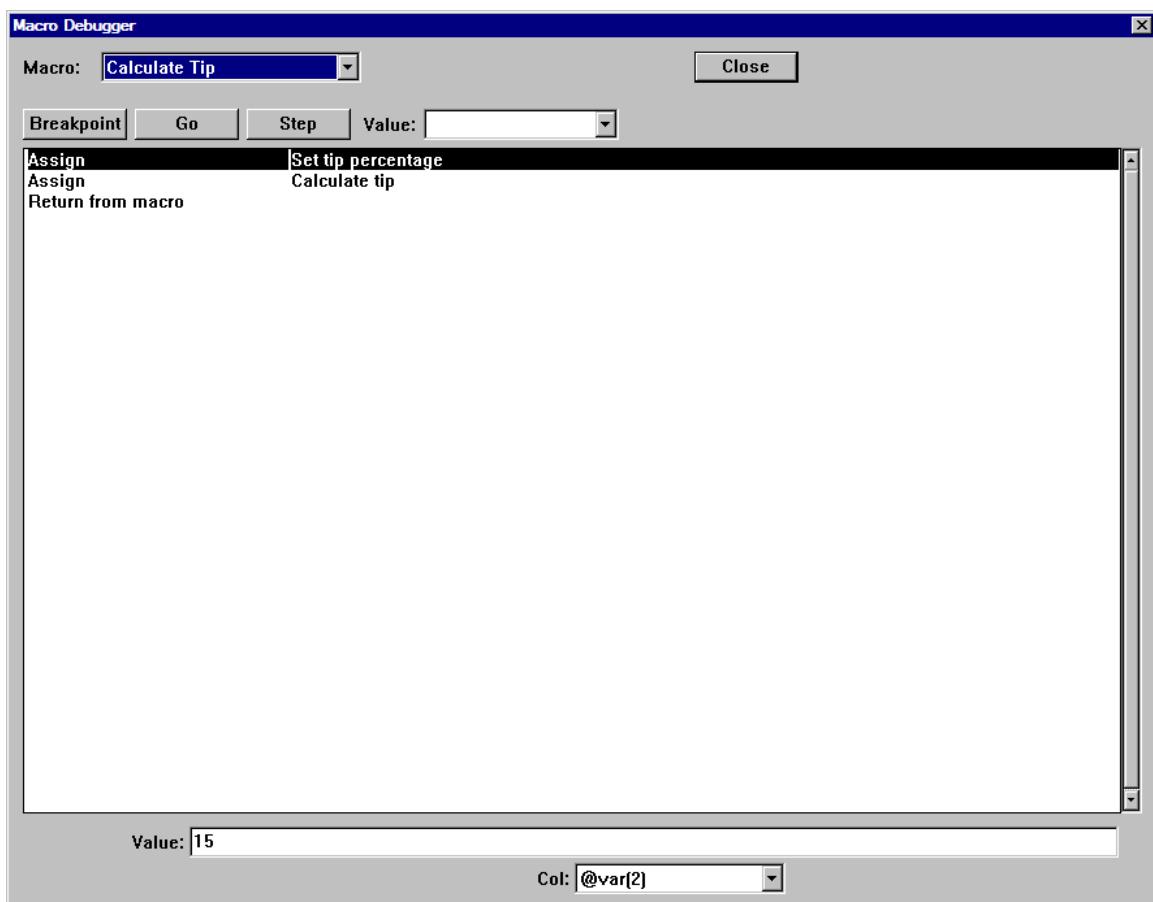
Visual CE displays the form in the development window.



- 4 Select **File–Debug**.

Visual CE displays the Save dialog box. (Prior to a debugging session, Visual CE always asks you to download the current version of the form to the CE device; this ensures that the form whose macro you are debugging on the CE device is consistent with the version with which you are working on the desktop PC.)

- 5 Navigate to the location on the CE device where you want to download the application and click **OK**.
- 6 When Visual CE asks if you want to overwrite the existing file, click **OK**.
Visual CE opens the Macro Debugger on the desktop PC.
Since Calculate Tip is the only macro associated with TipCalc.vce, it is automatically displayed in the Debugger window.



The Macro Debugger displays the commands (steps) that make up the macro in the script window (the large white area).

The Breakpoint, Go and Step buttons enable you to start and pause execution of the macro on the CE device. More specifically, the **Breakpoint** button enables you to insert points in the script where the Debugger will automatically pause execution when the macro runs. The **Go** button initiates a debugging session by starting the form on the CE device, or, if a macro is running, advances the execution to the next breakpoint. The **Step** button advances execution one step.

The **Value** drop-down menu lets you select a variable or table column whose value you wish to view while the macro execution is paused.

The fields at the bottom of the window display the command arguments for the step highlighted in the script window (just as in the Macro Builder).

The **Close** button closes the Macro Debugger, saving any breakpoints you have inserted for future debugging.

Next: Inserting Breakpoints

Quick Lesson 3: Inserting Breakpoints

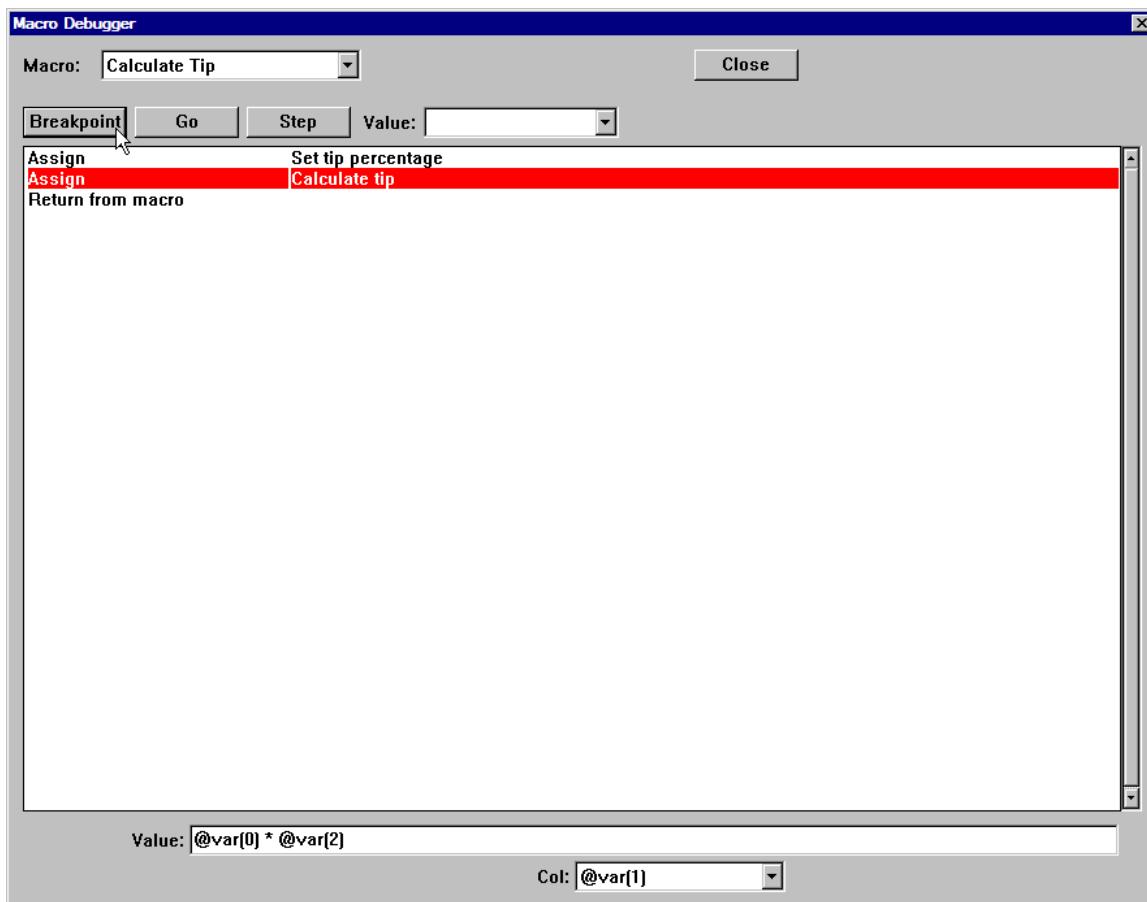
A breakpoint lets you specify a place in the macro script where the execution will pause, so that you can look at the values contained in variables and table columns at that point.

In this example, we want to see the values assigned to the variables by the first step, so we will set the breakpoint between the first and second steps.

To insert a breakpoint:

- 1 Click (highlight) the second step in the script window.
- 2 Click the **Breakpoint** button.

Note that the second step has turned red. The macro will pause immediately before the step that is highlighted in red.



Tip

If you accidentally put a breakpoint in the wrong place and want to remove it, simply highlight the 'red' step and click **Breakpoint**.

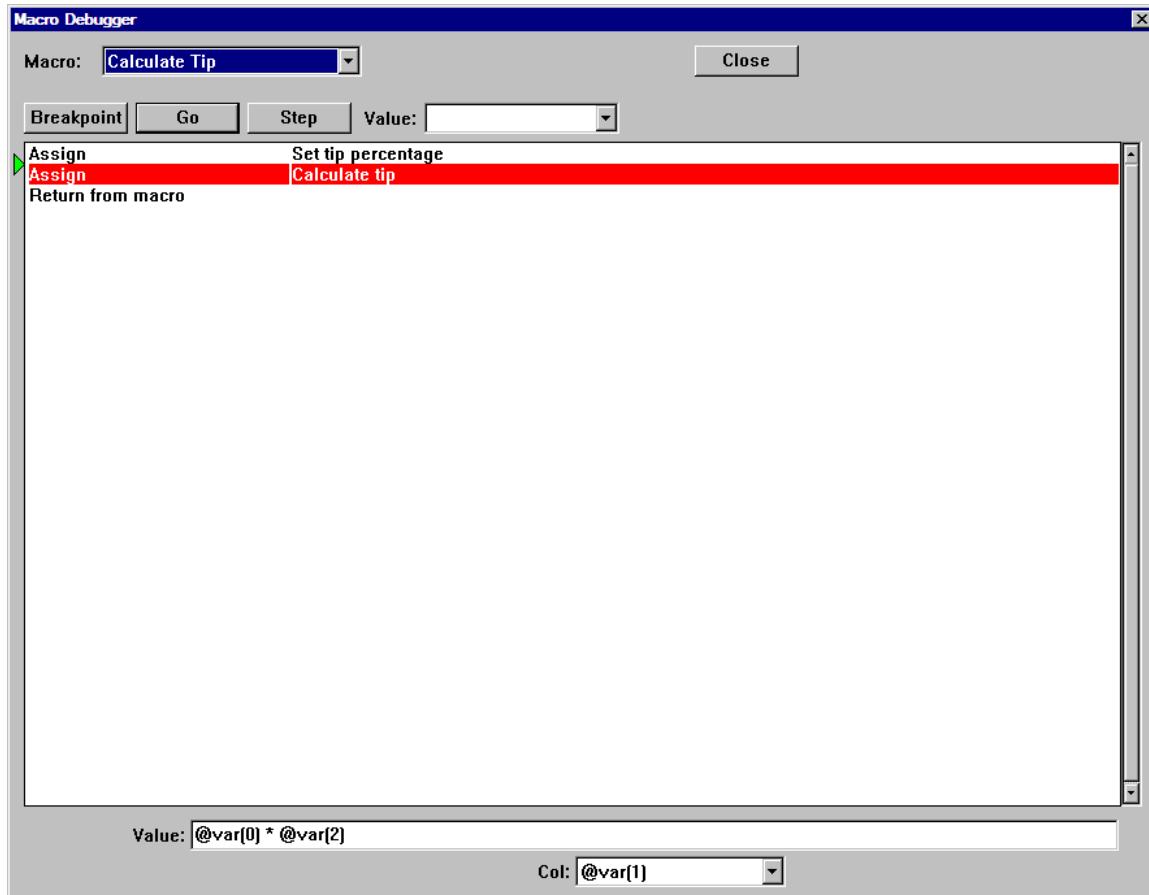
Next: Running the Macro with the Debugger

Quick Lesson 4: Running the Macro with the Debugger

Now that you've set at least one breakpoint, you are ready to run the macro with the Debugger. The macro actually runs on the CE device, while you control it from Debugger on the desktop PC.

To run the macro with the debugger:

- 1 In the Macro Debugger window on the desktop PC, click the **Go** button.
Visual CE starts TipCalc.vce on the CE device.
- 2 In the form on the CE device, enter **100.00** in the **Amount** field.
- 3 Still working on the CE device, start the macro by clicking the **Calculate Tip** button.
The macro begins to run on the CE device, but the Debugger on the desktop PC stops its execution at the breakpoint you set in the previous lesson. Note the green arrow displayed in the Debugger window, indicating the point at which the macro execution is paused. In the screen below, execution has been paused after the first step and before the second.



The fields at the bottom of the Debugger window display the command parameters for the currently selected step (in this case, Step 2). So, you can tell that the next step to execute after the breakpoint will multiply the values contained in the global variables @var(0) and @var(2). Recall from the summary on page 117 that @var(0) should contain the amount entered by the user into the Amount field, while @var(2) should contain the tip percentage.

assigned by Step 1 in this macro (15/100).

- 4 To view the value currently assigned to @var(0) – that is, the value held in @var(0) at the breakpoint -- select **@var(0)** from the **Value** dropdown menu near the top of the Macro Debugger window.

Note that the datatype and value currently held by the selected variable is displayed to the right of the dropdown. Integer and 100 are correct.

- 5 Now select **@var(2)** from the **Value** dropdown menu. Note that this variable contains 15, rather than 15/100. This likely means that the first Assign statement in the macro is incorrect. To verify this, click the first Assign statement in the script window. Looking at the fields at the bottom of the Debugger window, we see that the value for the Assign statement was incorrectly entered as 15, when it should have been 15/100. We've found the source of the problem.

- 6 For practice sake, click the **Step** button in the Macro Debugger. The Step button advances execution one step.

If you select @var(1) from the Value drop-down menu, you'll see the value assigned to it by the second macro step.

- 7 Click **Close** to close the Debugger, or select **Record–Close** on the CE device. Either way, both the Debugger and the .vce application close. Also, Visual CE automatically saves any breakpoints you've set, so that they are available the next time you open the macro in the Debugger. (Behind the scenes, Visual CE saves the breakpoints in a file on the CE device with the same name and in the same folder as the application, but with the extension .dbg; in this case, TipCalc.dbg.)

- 8 Open the macro in the Macro Builder (**Edit–Macro / Events**) and correct the value set by the first Assign statement (change 15 to 15/100). Close the Macro Builder and save your changes on the desktop PC by selecting **File–Save**.

You can now either download your revised application to the CE device for further testing, or you can reopen it in the Macro Debugger (**File–Debug**) for further investigation.

- 9 Once you have finished debugging the macro, go to the CE device and remove the file that stores the breakpoints, in this case, TipCalc.dbg. Otherwise, you will not be able to run TipCalc.vce on the CE device, except via the Debugger.

Next:

This Quick Tour introduced you to most of the features of the Macro Debugger. Additional information about the Debugger, including a quick summary of the steps for using it, are provided in the Macro Debugger reference section, which begins on page 370.

Part II:

Creating Your Own Visual CE Applications

Creating Visual CE Applications

Overview of Part II

If you created the sample application in Quick Tour II or IIa, you've already mastered the basics of creating Visual CE applications. This section contains a more detailed description of each step involved, and presents the many additional features and controls not covered in the Quick Tour.

The following topics are discussed:

- Visual CE Development Environment
 - ◊ Starting and Exiting
 - ◊ Device screen size
- Getting Started
 - ◊ Overview of the steps required to create an application
 - ◊ Planning an application
- Creating and Managing Tables
 - ◊ Creating a table
 - ◊ Modifying a table
 - ◊ Deleting tables
 - ◊ Changing a form's table
- Global Variables
 - ◊ About using global variables to store temporary values
- Beginning a New Form
 - ◊ Specifying a table
 - ◊ Specifying the form's size and color
 - ◊ Creating an application title
 - ◊ Removing or customizing the title bar
 - ◊ Adding tabs
- Adding Controls to the Form
 - ◊ Edit boxes
 - ◊ Note boxes
 - ◊ Labels
 - ◊ Checkboxes
 - ◊ Radio buttons
 - ◊ Drop-down lists
 - ◊ Calculated fields
 - ◊ Command buttons
 - ◊ Scribble boxes
 - ◊ Timestamp buttons
 - ◊ Image controls
 - ◊ Jump buttons

- ◊ AutoNumbering
- ◊ Lookups
- ◊ Grid controls
- ◊ mEnable stoplight
- Modifying Control Properties
 - ◊ Changing control properties
 - ◊ Specifying control font properties
 - ◊ Fixing a control's position relative to the frame (device screen)
- Arranging Controls on the Form
 - ◊ Turning off snap-to-grid
 - ◊ Aligning and sizing controls
 - ◊ Cutting, copying, and pasting controls
 - ◊ Deleting controls
- Customizing the Form
 - ◊ Customizing or removing the menu bar
 - ◊ Specifying control order
 - ◊ Preventing duplicate values in a column
 - ◊ Enabling auto recalc
 - ◊ Enabling bar code scanners
 - ◊ Creating "About Box" text
- Defining Record Sorts and Searches
 - ◊ Specifying an initial sort/search ordering
 - ◊ Indexing columns
 - ◊ Creating pre-defined filters
- Security
 - ◊ Password-protecting an application
 - ◊ Allowing inserts and deletes
 - ◊ Locking the form design
- Downloading Testing and Managing Applications
 - ◊ Saving an application
 - ◊ Downloading an application
 - ◊ Testing an application
 - ◊ Opening an existing application
 - ◊ Modifying an existing application
 - ◊ Organizing Visual CE files for Use on a Desktop PC
- Distributing an application (Pro and Mobile 360 Editions Only)
 - ◊ Creating distribution files

Visual CE Development Environment

Starting Visual CE's Development Environment

A Visual CE application consists of a table* and a form. You create the table first, then the form.

Typically, you will create a **table** using Visual CE's development environment on the desktop PC or CE device. Alternatively, you can use Microsoft Pocket Access or SQL Server CE on the CE device. Tables created with third-party desktop ODBC-enabled applications (for example, Access and Excel) can also be downloaded and used.

You create the **form** using Visual CE's development environment on either a desktop PC or a CE device with a touchscreen. It is not possible to create forms directly on Smartphones.

If you intend to use a table created with an application other than Visual CE, create the table before proceeding here. Refer to the topics, "Creating a Table with Pocket Access" (page 143), "Creating a Table with SQL Server CE" (page 144), "Downloading a Table from an ODBC-Enabled Application" (page 146), or "Copying an Access Database to the CE Device" (page 145).

*Note

In a few instances, you may want to create a form that reads and writes *temporary* values to global variables only. In that case, you do not need to create a table to store records. Open the development environment as explained below, then refer to the topic, "Creating a Form without a Table," on page 157 for instructions.

To start Visual CE's development environment on a desktop PC:

- 1 Connect the CE device to your desktop PC.

You can perform some development activities on the desktop PC without connecting the CE device. In "disconnected" mode, you can create a new table and form, but you cannot download an application; modify, clone, or delete an existing table; or create jumps, lookups, or grid controls between tables.

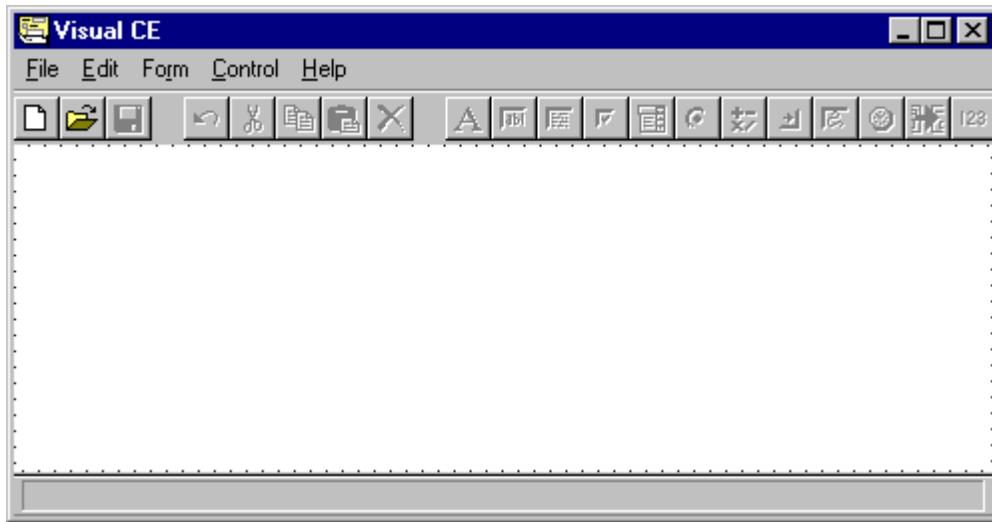
- 2 On the desktop PC, double-click the Visual CE icon  to start Visual CE. Or, click the Windows **Start** button, then select **Programs-Visual CE-Visual CE**.

To start Visual CE's development environment on a CE device:

- Click the Windows **Start** button, then select **Programs-Visual CE**.

Visual CE displays its development environment window. All of your application development work is carried out here.

Note: The desktop PC version is shown below; the Pocket PC version has the menu bar but not the toolbar.



Visual CE's development environment window has five menus: File, Edit, Form, Control, and Help. Select menu options by clicking them with the mouse pointer or by pressing the shortcut keys.

Its toolbar buttons (available on desktop PC version) provide a faster way to activate the most frequently used commands.

If you are running Visual CE's development environment on the desktop PC, the dotted line indicates Visual CE's screen size for the type of CE device connected. If your device's actual screen is a different size, specify its dimensions using the instructions in the topic, "Changing the Device Screen Size," on page 128.

Visual CE's default form size is the same as the screen size. You can change this to any desired size, even one larger than the CE device screen size. (See "Specifying Form Size" on page 158 for additional information.)

Changing the Device Screen Size

Visual CE automatically determines the screen size for the type of CE device connected to the desktop PC. Visual CE indicates this screen size by the dotted-line frame in its development window.

If your CE device has a non-standard screen size (such as a Smartphone), you can manually specify the dimensions of the screen using the instructions below.

To specify a different screen size:

- 1 Copy **VISUALCE.INI** (typically located in \Program Files\Windows CE Services\SYWARE Visual CE on your desktop PC) to the Windows directory on your desktop PC.
- 2 Using Notepad or other editor, open the copy of **VISUALCE.INI** in the Windows directory. It contains the following information:
[SCREEN]
Width=nnn
Height=nnn
where nnn is a number of pixels.
- 3 Replace each "nnn" for width and height with the actual pixel size of your CE device screen. If the appropriate value is already specified in the file, you can simply remove the leading ; character that had commented out the line.

Tip

The physical screen size of a Smartphone is typically 178 pixels wide by 205 pixels high.

- 4 Save the file with your changes.
- 5 If you are currently running Visual CE, select **File—Close**.
- 6 Start Visual CE. The new screen size is now indicated by the dotted lines in the development environment window.

This change remains in effect for all future Visual CE sessions, unless you change the size again.

Tip

Visual CE's default form size is the same as the screen size. You can make the form any size, even one larger than the CE device screen size. (See "Specifying Form Size" on page 158 for additional information.)

File Menu Commands

Command	Button	Shortcut	Function
New Form		Ctrl+N	Begins the creation of a new application.
Open		Ctrl+O	Opens an existing application for viewing and/or modifying.
Save		Ctrl+S	Saves the application currently displayed in the development environment window.
Save As	(none)	(none)	Allows you to save the currently displayed application under another name or in another location.
Run*	(none)	Ctrl+U	Runs the application currently displayed in the development window, for testing and debugging.
Download Form **	(none)	(none)	Downloads the current application to the connected CE device.
Debug**	(none)	(none)	Opens the Macro Debugger, a visual tool for tracking down errors in macros.
Create Table	(none)	(none)	Allows you to create a new table to be used with an application.
Modify Table	(none)	(none)	Allows you to add, delete, reorder, and index columns in an existing table.
Clone Table**	(none)	(none)	Creates an exact duplicate of an existing table structure (without the data).
Delete Table**	(none)	(none)	Allows you to delete one or more tables.
Download Table**	(none)	(none)	Downloads a desktop PC table to the CE device and creates a Visual CE application for it.
Synchronize**	(none)	(none)	Allows you to move data changes between tables on the desktop PC and CE device.
Create Distribution Files***	(none)	(none)	Starts wizard to create installation files for distributing your application to other users.
Close		Ctrl+W	Closes the Visual CE development environment, saving any changes you have made to the current application.

*Available only on handheld version of development environment

**Available only on desktop version of development environment

***Available only in Pro and Mobile 360 Editions

Edit Menu Commands

Command	Button	Shortcut	Function
Undo		Ctrl+Z	Undoes last edit.
Cut		Ctrl+X	Cuts the currently selected control and saves it to the clipboard.
Copy		Ctrl+C	Copies the currently selected control and saves the copy to the clipboard.
Paste		Ctrl+V	Pastes the contents of the clipboard onto the development area. You can copy or cut controls from one application and paste them into another.
Delete		Delete	Deletes all currently selected controls.
Select All	(none)	Ctrl+A	Selects all controls in the development area.
Property	(none)	(none)	Displays the Properties dialog box for the currently selected control(s) for modification. Note: Double-click or right-click any control to display and/or modify its properties.
Filters	(none)	(none)	Enables you to create and save record filters as part of the form design.
Alarm	(none)	(none)	Enables you to define an alarm message and defer time.
Macros/ Events*	(none)	(none)	Opens the Macro Builder where you can specify a sequence of commands to be executed upon a single command or event.
Custom Colors	(none)	(none)	Add up to 16 custom colors to the standard palette. You can apply these colors to the form and list display.
Align	(none)	(none)	Enables you to align selected controls with one another along their right, left, top, or bottom edges or their horizontal or vertical centers; or to make a group of selected controls the same height or width. See "Aligning and Sizing Controls on the Form," on page 269.

*Available only on desktop version of development environment

Form Menu Commands

Command	Function
Title Bar	Allows you to enter text to appear in the application's title bar, to remove the record count from the title bar, or to remove the title bar altogether.
	Allows you to disable the Windows CE Start flag, effectively restricting use of the device to the Visual CE application.
Menu Bar	Allows you to specify a custom set of options for the application's menu bar, or to remove the menu bar altogether.
Tabs	Allows you to divide your form into tabbed pages.
Sort On	Allows you to specify an initial sort/search column that Visual CE uses to organize records and to enable record searches.
Unique	Allows you to prevent duplicate values in a selected column.
Control Order	Allows you to specify the field-to-field order for cursor movement when the Tab key is used; the order of columns in the record list display; the order of record fields in a printed copy; the order of record fields in an imported or exported text file; and (discouraged) the order of buttons in a macro program.
Form Size	Allows you to change the size of the form.
Scroll Bars*	Allows you to hide the scroll bars in the development window.
Password	Allows you to specify a password that a user must enter before opening the application on the CE device.
Lock	Allows you to lock the form design so that no one can alter it.
Color	Allows you to specify a background color for the form.
List Mode Color	Allows you to specify the color of text that appears in list mode.
About Text	Allows you to specify the text that will be displayed in the About box.
Table	Allows you to change the table associated with the form.
Allow Inserts	Allows you to specify whether or not the user can add new records to the table. If checked, the user can add records; if unchecked, the user cannot add records.
Allow Deletes	Allows you to specify whether or not the user can delete records from the table. If checked, the user can delete records; if unchecked, the user cannot delete records.
Auto Recalc	Allows you to specify when Visual CE updates values for the Calculated controls. If this option is checked, Visual CE automatically recalculates the values whenever the user presses a timestamp button or moves the input focus from an edit control. If this option is unchecked, the user must explicitly select the Option-Recalc command.

*Available only on handheld version of development environment

Control Menu Commands

Command	Button	Function
Label		Creates a label control.
Edit		Creates a single-line edit control.
Note		Creates a multi-line edit control.
Checkbox		Creates a checkbox control.
Drop Down		Creates a drop-down list control.
Radio Buttons		Creates a group of radio button controls.
Calculated		Creates a calculated field control.
Command button		Creates a command button.
Scribble		Creates a scribble box.
Timestamp		Creates a timestamp button.
Image		Creates an image control.
Jump		Creates a jump button.
AutoNumber		Creates an autonumber control.
Lookup		Creates a lookup control.
Grid		Creates a grid control.
mEnable		Creates an mEnable button.

Development Window Toolbar Buttons

Button	Name	Function
	New Form	Begins the creation of a new application.
	Open	Opens an existing application for viewing and/or modifying.
	Save	Saves the application currently displayed in the development environment window.
	Undo	Undoes the last edit.
	Cut	Cuts the currently selected control and saves it to the clipboard.
	Copy	Copies the currently selected control and saves the copy to the clipboard.
	Paste	Pastes the contents of the clipboard onto the development area. You can copy or cut controls from one application and paste them into another.
	Delete	Deletes the currently selected controls.
	Label	Creates a label control.
	Edit	Creates a single-line edit control.
	Note	Creates a multi-line edit control.
	Checkbox	Creates a checkbox control.
	Drop Down	Creates a drop-down list control.
	Radio Buttons	Creates a group of radio button controls.
	Calculated	Creates a calculated field control.
	Command button	Creates a command button.
	Scribble	Creates a scribble box.
	Timestamp	Creates a timestamp button.
	Image	Creates an image control.
	Jump	Creates a jump button.
	AutoNumber	Creates an autonumber control.
	Lookup	Creates a lookup control.
	Grid	Creates a grid control.
	mEnable	Creates an mEnable button.

Development Window Shortcut Keys

File Functions

To do...	Press
Begin creation of a new application.	Ctrl+N
Open an existing application to view or modify.	Ctrl+O
Save the application currently displayed in the development window.	Ctrl+S
Run the application currently displayed in the development window, for testing and debugging.*	Ctrl+U
Close the Visual CE development environment.	Ctrl+W

Edit Functions

To do...	Press
Undo last edit.	Ctrl+Z
Cut the currently selected control and save it to the clipboard.	Ctrl+X
Copy the currently selected control to the Windows clipboard.	Ctrl+C
Paste information from the Windows clipboard into the development area.	Ctrl+V
Delete all currently selected controls.	Delete
Select all controls in the development area.	Ctrl+A

Selecting and Arranging Controls

To do...	Press
Select multiple controls.	Ctrl while selecting; or left-mouse click and drag cursor across controls
Deselect a single control in a selected group.	Ctrl while selecting
Turn off snap-to-grid.	Shift while dragging

*Available only on handheld version of development environment

Exiting Visual CE's Development Environment

You can exit the Visual CE development environment window when you have finished developing an application, or at any point during application development.

- Select **File—Close** or click the **Exit** button  on the title bar to exit Visual CE.

Each time you exit, Visual CE saves all of the entries and changes you have made.

Getting Started

Steps in Creating an Application

The sequence for creating Visual CE applications is:

- 1 Plan the application (page 137)
- 2 Create the table (page 138)
- 3 Begin the form:
 - ◊ Specify the table that the form will use; could be no table (page 155)
 - ◊ Optional: Specify form size (page 158) and background color (page 160)
 - ◊ Optional: Add a custom title or hide the title bar (page 163)
 - ◊ Optional: Create and name tabbed pages (page 164)
- 4 Create the controls (page 166)
- 5 Arrange the controls on the form (page 268)
- 6 Customize the form:
 - ◊ Optional: Customize or hide the menu bar (page 272)
 - ◊ Optional: Specify a control ordering. Determines order of: columns in record list display, record values in printed copy, field-to-field cursor movement when user tabs, order of fields in an imported or exported file, and (discouraged) execution sequence in macro programs (page 276)
 - ◊ Optional: Allow only unique values in a column (page 278)
 - ◊ Optional: Enable Auto Recalc for calculated field values (page 279)
 - ◊ Optional: Specify text for the About box (page 281)
- 7 Define Record Sorts and Searches:
 - ◊ Optional: Specify an initial sort/search ordering for record display (page 282)
 - ◊ Optional: Create filters for record display (page 284)
- 8 Optional: Add a macro program (page 303)
- 9 Apply security features:
 - ◊ Optional: Password-protect the application (page 286)
 - ◊ Optional: Allow/prevent users from inserting/deleting records (page 288)
 - ◊ Optional: Prevent modification to the form design (page 289)
- 10 Save the application (page 290)
- 11 Download the application (if developing on desktop PC) (page 292)
- 12 Test the application (page 293)
- 13 Optional, to run the application on a desktop PC: Synchronize files and tables (page 296)

Each step is described in detail in the following topics.

Note

You can use Visual CE's development environment on either a desktop PC or a CE device with a touchscreen. For the most part the process is the same on either type of computer; any differences are noted in this documentation.

Tips for Planning Your Application

If you want to create a Visual CE application that can be used far into the future, spend a little time planning your form before using Visual CE to create it.

Be sure to consider:

- The types of controls you will use and where you will place them in the application window

Note

The form can exceed the size of the CE device screen. See "Form Size" on page 158 for further information. Also, if you are developing your form on the desktop PC, you can specify the size of the CE device screen which is indicated by the dotted lines in the Visual CE development environment window. See "Device Screen Size," on page 128 for further information.

- Whether the labels and captions you plan to use are self-explanatory (meaningful to anyone beside yourself)
- Whether a particular control will be read-only (display only - no user input) or read/write (allowing user input)
- Whether a particular control will have a default value
- Whether input will be required for a particular control
- Whether you want records organized based on a particular control value (sorting)

In general, it is better to keep edit boxes and note boxes to a minimum, because the text entries to these controls consume more space on the CE device. Wherever possible, use drop-down lists, radio buttons, and checkboxes, because they use less space and are easier to use with the stylus.

Creating and Managing Tables

About Creating Tables

You can think of a table as a data storage structure, internal to the computer, that organizes data in rows and columns.

Rows store the data records that users create when they work with your Visual CE application. For example, if your application was designed to track expenses, each row would store a single expense record.

Each column in the table stores all of the values for one category of information, or in other words, for one field in the records. For example, one column might store all of the expense report dates, while another stores all of the amounts, and another all of the descriptions.

When you create the table for your application, you specify a column for each record field whose values you want to store. You specify a name for the column and the type of data it will store, which must be one of the following: text, integer number, floating point number, date/time, memo, picture (OLE object in Pocket Access or SQL Server CE), or yes/no.

When you eventually create the application's form (the user interface), you create one control on the form for each column in the table. The control makes it possible for users to view, add, modify, and delete values in the column.

There are many ways to create a table to be used by a Visual CE application:

- Create the table using Visual CE (page 139)
- Clone an existing Visual CE Table (page 142)
- Create the table using Pocket Access (page 143) or SQL Server CE (page 144) on the CE device
- Drag an Access table from the desktop PC to the CE device using ActiveSync/MobileDevices (page 145)
- Download the table and data from an ODBC-enabled application on the desktop PC (page 146)

Notes

- If you plan to run a Visual CE application on a desktop PC, do not use Pocket Access or SQL Server CE to create the table.
- You can create controls that read and write values to global variables, rather than to table columns. This is useful if you have designed a process that requires temporary values. In a few instances, you may even want to create a form that consists entirely of controls that read and write values to variables, in which case you would not create a table for the form at all. Refer to the topic, "About Using Global Variables to Store Temporary Values," on page 153 for more information.

Creating a Table Using Visual CE

Using Visual CE's development environment on a desktop PC or CE device, you can create a table that is initially empty. You can then use a Visual CE application you design to create records in the table.

Note

Before you begin creating a new table, put some time into planning it carefully. After records exist in the table, you can modify its structure, but doing so can cause synchronization problems and invalidate any forms that use it.

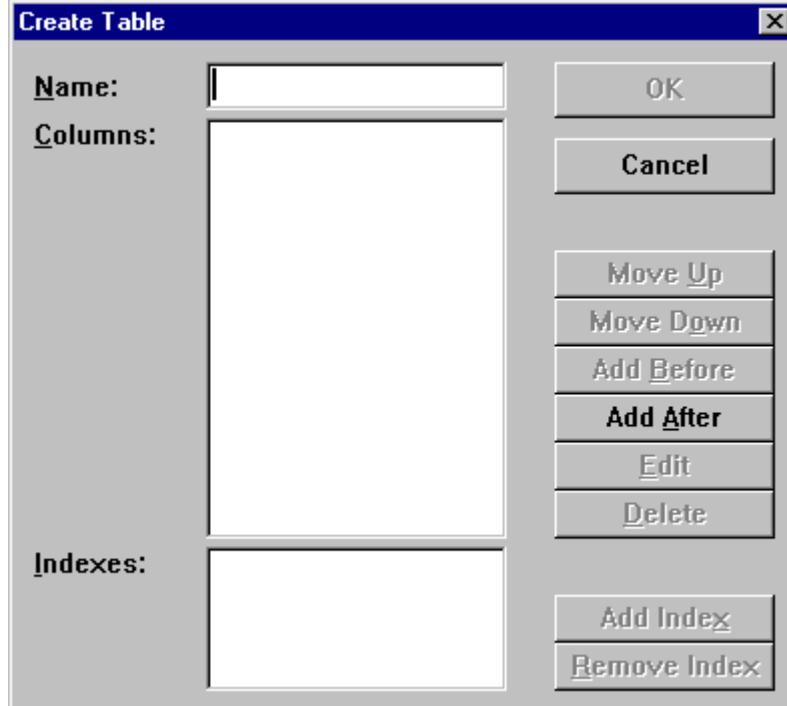
On your desktop PC or CE device:

1 Start Visual CE:

- **On a desktop PC:** Double-click the Visual CE icon  or select **Start-Programs-Visual CE-Visual CE**.
- **On a CE device:** Select **Start-Programs**, then click the Visual CE icon .

2 Select **File—Create Table**.

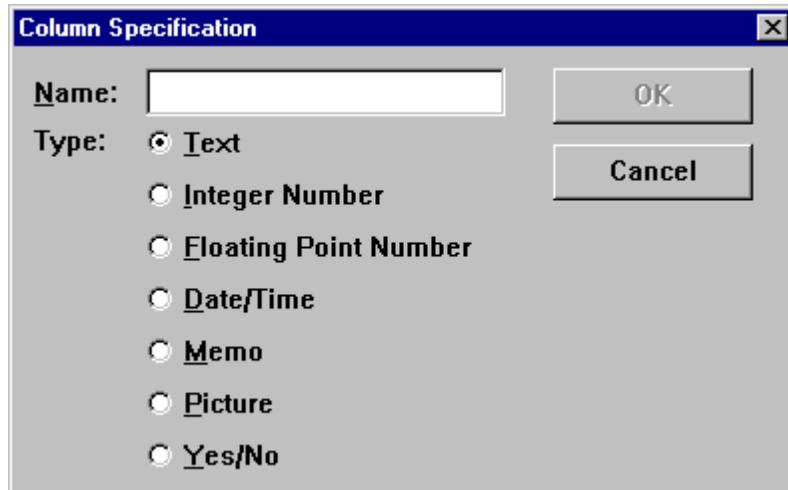
Visual CE displays the Create Table dialog box.



3 Enter the name of the table in the Name field.

4 Click the **Add After** button.

Visual CE displays the Column Specification dialog box.



- 5 Enter the name of the column in the Name field.

Note

To facilitate synchronization, avoid names that contain special characters such as spaces, commas, or dashes, or that begin with a digit. (A message box appears if the name you choose could cause problems.) Also, since Visual CE will use this name as the column heading in the record list display on the CE device, try to use a name that is both meaningful and short.

- 6 Specify the type of data that the column will store by clicking the corresponding radio button:
 - **Text** Small amounts of text (less than 256 characters); if you'll have larger amounts of text, use Memo instead
 - **Integer Number** Whole numbers
 - **Floating Point Number** Numbers with digits to the right of the decimal point
 - **Date/Time** Dates, times, or both
 - **Memo** Large amounts of text (more than 255 characters)
 - **Picture** Scribbles or other graphics
 - **Yes/No** Also referred to as "Boolean"
- 7 Click **OK** to save your entries and return to the Create Table dialog box.
- 8 Click the **Add After** button (or Add Before, if you want this column to precede the column you already entered).
- 9 Continue specifying columns as described in Steps 4 through 8 until you have defined all of the table's columns.

Note

If you need to change your entries for a column, click the column (to highlight it), then click the **Edit** button. To remove a column, click the column, then click the **Delete** button. If you want to change the order of the columns, click a column, then click the **Move Up** or **Move Down** button.

- 10 Indexes are used for record sorting and searching. If you are using Visual CE's development environment on a desktop PC and you intend to use a column in the application for record sorts, searches, lookups, grid controls, or jumping to another Visual CE table, you should index the column now:
 - Click the **Add Index** button
 - Select the column to index, and click **OK**.

Notes

- It is not necessary to index a column unless you want to specify an initial sort/search ordering (see "Specifying an Initial Sort/Search Ordering," on page 282), create a jump button (see "Creating a Jump to Another Table," on page 248), use the column as a lookup in another table (see "Creating a Lookup," on page 254), or create a grid control (see "Creating a Grid Control," on page 257). If you do not specify the indexes now, users can select their own sort orderings. Visual CE supports up to four indexes per table at a time.
- If you are working on the CE device, you'll index columns in the Visual CE application window, rather than here in the Create table dialog box. Complete your table and form design following the remaining steps in this section. Open the application in the Visual CE application window and follow the instructions in the topic, "Indexing Columns," on page 283.

- 11 When you have finished specifying columns, click **OK** on the Create Table dialog box.
- 12 Visual CE asks if you want the Form Wizard to create a form for this table:
 - Click **Yes** if you want the Form Wizard to automatically create a control for each column in the table, including labels that correspond to the columns' names. The Wizard arranges the controls along the left side of the development window. You can then quickly rearrange the layout and change the properties of any of the controls to meet your needs, if desired.
 - Click **No** if you want to create each control from scratch (as you did in Quick Tour II).
- 13 If you are working on a desktop PC and responded Yes to the previous message, Visual CE asks if you want to download the form it just created to your CE device:
 - Click **Yes** if you want to begin using the application immediately, as is.
 - Click **No** if you want to customize the form before downloading it.

Visual CE displays the name of the specified table in the status bar at the bottom of the development environment window. You can now create or modify the form's controls using the instructions in later topics.

Note

If you created the table on the CE device--and you intend to synchronize it with a table on the desktop PC--you must create a form and open it in Visual CE's development environment on the desktop PC. This will register the table with the Visual CE software that controls the synchronization process. You don't need to do this if you intend to use the application and its underlying table strictly on the CE device.

Cloning a Table

Working in Visual CE's development environment on the desktop PC, you can create a new table by cloning an existing table on the CE device. Visual CE copies the structure, but not the contents of the original.

You start the process by opening the form associated with the table you want to clone. Visual CE asks if you want to switch the form's association from the old table to the new—in that way creating a new, empty table for the form while keeping the original data intact.

Notes

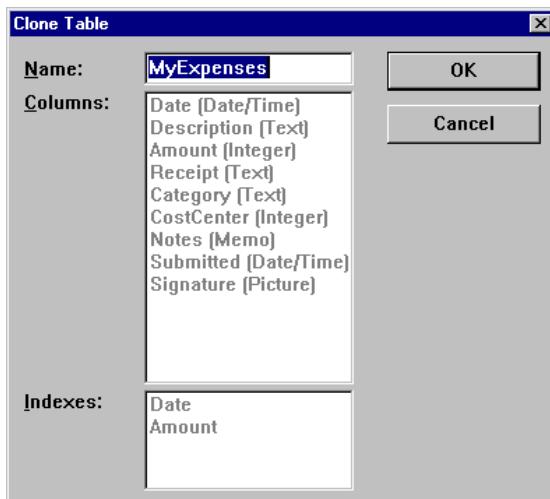
- This feature is available only in the desktop PC version of the Visual CE development environment – not in the handheld version.
- The existing table must have been created using Visual CE – you cannot clone a Pocket Access or SQL Server CE table.

On your desktop PC:

1 Open an existing .VCE application file that uses the table you want to clone.

2 Select **File—Clone Table**.

Visual CE displays the Clone Table dialog box.



3 Replace the entry in the Name field with a name for the new table. (You cannot modify any of the other entries.)

4 Click **OK**.

Visual CE asks if you want to use the cloned table as the underlying table of the current form.

- Click **Yes** if you want to switch the form's association from the old table to the new. The old table and its contents remain intact.
- Click **No** if you want to leave the original table associated with the current form. In this case you will need to create a new form for the cloned table.

The new table is now available on the CE device.

Creating a Table with Pocket Access

Note

Visual CE continues to support files created with Pocket Access (.cdb files), even though Windows Mobile no longer supports Pocket Access.

If you have Pocket Access on your CE device, you can use it to create a table for use with a Visual CE application.

Create and save the table using Pocket Access on the CE device as you would normally, keeping in mind the following:

- If you create a table with Pocket Access, you will **not** be able to run the associated form on a desktop PC. Use Pocket Access to create the table only if you intend to run the Visual CE application strictly on CE devices.
- Several features of Visual CE applications—such as sorting and searching records—require indexed columns. SYWARE provides a free utility that enables you to add/delete indexes to Pocket Access tables. Refer to the topic, "Visual CE Add-on Tools," on page 17 for instructions on how to obtain a copy.
- Visual CE works with Pocket Access tables that are in either the Windows CE Store, or in database volumes (.cdb files) anywhere on the CE device's file system. Keep in mind that if you store the table in a .cdb file, you will need to use Microsoft's Database Synchronization software to synchronize with a database on the desktop PC (you cannot apply the Visual CE synchronization software to tables stored in a .cdb file).
- If you create a .cdb file, you will need to save the form (*.vce file) in the same folder on the CE device.
- "OLE Object" is the Pocket Access equivalent of Visual CE's "picture" data type.

Refer to your Pocket Access documentation for additional information.

Once you have created the table, go to the topic, "Specifying the Table to be Used by a Form," on page 155.

Tips

- You can create a Visual CE form for a pre-existing Pocket Access table--even if it already has data in it.
- Using an older version of Microsoft ActiveSync, you can copy an Access database (.mdb) on the desktop PC to create a new Pocket Access database (.cdb file) on the CE device. Refer to the topic, "Copying an Access Database to the CE Device," on page 145.

Creating a Table with SQL Server CE

Create and save the SQL Server CE table (.sdf file) on the CE device as you would normally, keeping in mind the following:

- If you create a table with SQL Server CE, you will **not** be able to run the associated form on a desktop PC. Use SQL Server CE to create the table only if you intend to run the Visual CE application strictly on CE devices.
- Several features of Visual CE applications—such as sorting and searching records—require indexed columns. SYWARE provides a free utility called SSCE that enables you to run SQL statements to add/delete indexes to SQL Server CE tables. Refer to the topic, "Visual CE Add-on Tools," on page 17 for instructions on how to obtain a copy. You must purchase sqlceEnable, an optional add-on module from SYWARE, to use the SSCE utility.
- You cannot use the Visual CE synchronization software to synchronize .sdf files with a database on the desktop PC. You must use Microsoft's Database Synchronization software instead.
- You must save the SQL Server CE table (*.sdf file) and its associated form (*.vce file) in the same folder on the CE device.
- "OLE Object" is the SQL Server CE equivalent of Visual CE's "picture" data type.

Refer to your SQL Server CE documentation for additional information.

Once you have created the table, go to the topic, "Specifying the Table to be Used by a Form," on page 155.

Note

Use of SQL Server CE tables with Visual CE requires SYWARE's sqlceEnable, a separate software product.

Tip

You can create a Visual CE form for a pre-existing SQL Server CE table--even if it already has data in it.

Copying an Access Database to the CE Device

Notes

- Visual CE continues to support files created with Pocket Access (.cdb files), even though Windows Mobile no longer supports Pocket Access.
- Rather than creating a .cdb file as described here, you may wish to download a table from an ODBC-enabled application on the PC to the Windows CE Data Store on the CE device. That method is described in the topic, "Downloading a Table from an ODBC-Enabled Application on the Desktop PC", on the following page .

You can use an older version of Microsoft's ActiveSync/Mobile Devices to simply drag an Access database (.mdb file) from the desktop PC to the CE device. The file is automatically copied onto the CE device as a Pocket Access database volume (.cdb file)—even if you do not have Pocket Access installed on your CE device.

You can build a form for the new table just as you would any other table, keeping in mind the following:

- If you create a table using this method, you will **not** be able to run the associated form on a desktop PC. Use Pocket Access tables only for Visual CE applications that will be run strictly on CE devices.
- Several features of Visual CE applications—such as sorting and searching records—require indexed columns. SYWARE provides a free utility that enables you to add/delete indexes to Pocket Access tables. Refer to the topic, "Visual CE Add-on Tools," on page 17 for instructions on how to obtain a copy.
- You must save the form (*.vce file) and the .cdb file in the same folder on the CE device.
- "OLE Object" is the Pocket Access equivalent of Visual CE's "picture" data type.
- Keep in mind that if you store the table in a .cdb file, you will need to use Microsoft's Database Synchronization software to synchronize with a database on the desktop PC (you cannot apply the Visual CE synchronization software to tables stored in a .cdb file).

Once you have created the table, go to the topic, "Specifying the Table to be Used by a Form," on page 155.

Downloading a Table from an ODBC-Enabled Application on the Desktop PC

If you have a table on the desktop PC that was created with an ODBC-enabled application—such as Microsoft Access, FoxPro, or Excel—you can download it (and any records it contains) to the CE device, and quickly create a Visual CE application for it.

For your convenience, the download procedure also includes optional steps for establishing synchronization between the two tables.

Notes

- The CE device must be connected to the desktop PC throughout the download procedure.
- **If you intend to use Visual CE's synchronization feature for more than one table on a CE device:** Visual CE usually synchronizes all of the eligible tables on a CE device with one desktop PC database at a time. For that reason, it is most efficient (though not required) to keep all of the desktop PC tables you intend to synchronize with tables on a given CE device in a single desktop PC database -- such as a single .mdb or .accdb file. This is true even if the tables on the CE device are unrelated to one another. Desktop PC tables to be synchronized with tables on different CE devices can be kept in different desktop PC databases.

- 1 (Perform this step if you plan to synchronize the tables. Otherwise, go to step 2.) In order for synchronization to work, the first column of the desktop table must be an "OID" column. (Visual CE automatically adds a corresponding column to the Windows CE table on the CE device and uses the columns to keep records in the two tables in sync.) Before starting the download procedure, open the table in the desktop PC application that was used to create it (e.g., Microsoft Access). Add a new first column to the table with the following specifications:

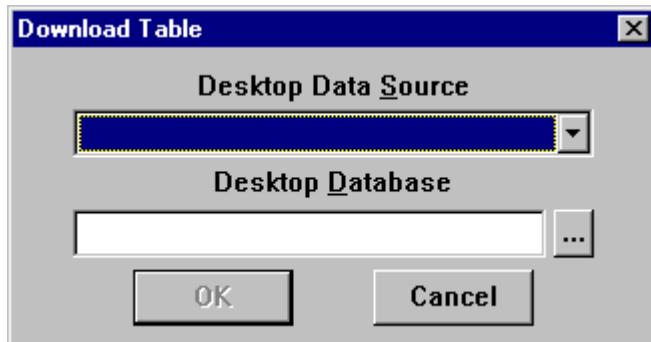
- Field name: OID
- Data Type: Number
- Indexed: No

The value stored in the OID column for each existing record must be zero or null (blank). Save your changes and close the application.

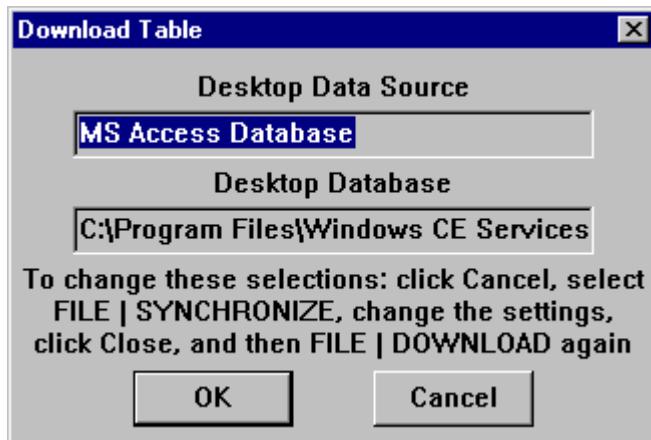


- 2 Start Visual CE on the desktop PC by double-clicking the Visual CE icon or by selecting **Start-Programs-Visual CE-Visual CE**.
- 3 Select **File-Download Table**. Visual CE asks if you want to establish synchronization between the tables.
 - Click **Yes** if you plan to maintain data in both the CE- and desktop-versions of the table and want to share updates between them.
 - Click **No** if you simply want to download the table to the CE device (if you click No now, you can always set up synchronization later).
 Visual CE displays the Download Table dialog box.
- 4 The options available in the Download Table dialog box depend upon whether or not you have already download a table to the CE device:

- If you have not downloaded a table before, select the database that contains the table you want to download:



- Click on the arrow to the right of the "Desktop Data Source" field, and select the data source type.
- Click the Browse button ..., and navigate to the folder that contains the database.
- Select the database and click **Open**.
- Click **OK** to close the Download Table dialog box. Visual CE displays the Select Table dialog box. Go to step 5.
- If you have downloaded a table before, Visual CE assumes that you want to download the table from the same desktop PC database you used before. If that is the case, click **OK** and go to step 5.



- If you have downloaded a table before, but you want to download the current table from a different desktop PC database, click **Cancel** to close the Download Table dialog box. Then:
 - Select **File-Synchronize**. Visual CE opens the Visual CE Synchronize dialog box.
 - Select the new database and other settings as desired (refer to page 436 for additional details). Click **Close** to save your new settings in the Visual CE Synchronize dialog box.
 - Select **File-Download Table**. Visual CE displays the Download Table dialog box again, this time with the new database selected. Click **OK**.

Visual CE displays the Select Table dialog box.



- 5 Select the table you want to download, then click **OK**.
- 6 Visual CE asks if you intend to synchronize the table with more than one handheld. Respond as desired.
- 7 Visual CE asks if you want the Form Wizard to create a form for this table:
 - Click **Yes** if you want the Form Wizard to automatically create a control for each column in the table, including labels that correspond to the columns' names. The Wizard arranges the controls along the left side of the development window. You can then quickly rearrange the layout and change the properties of any of the controls to meet your needs, if desired.
 - Click **No** if you want to create each control from scratch (as you did in Quick Tour II).
- 8 If you responded Yes to the previous message, Visual CE asks if you want to download the form it just created to your CE device:
 - Click **Yes** if you want to begin using the application immediately, as is.
 - Click **No** if you want to customize the form before downloading it.

Visual CE displays the name of the specified table in the status bar at the bottom of the development environment window. You can now create or modify the application's controls using the instructions in later sections.

You can also customize the synchronization options to fit your needs (refer to the topic, "Selecting Table Synchronization Options," on page 435).

Adding, Deleting, or Modifying Columns in an Existing Table

Using Visual CE's development environment on a desktop PC, you can add, delete, or modify columns in an existing table. If you are working on a CE device, you can add columns (but not delete or modify them).

Notes

- If you are working on a desktop PC, the CE device must be connected to it for this procedure.
- Keep in mind that if you change the table structure, you may have to modify any forms that use it. Additionally, any changes to its structure can cause synchronization problems.
- The following applies only to Windows CE tables created with Visual CE. If the table is in a Pocket Access or SQL Server CE database (.cdb or .sdf file), you must use Pocket Access or SQL Server CE to modify it.
- If you are working on a CE device, you can add columns (but not delete or modify them).

To modify a table:

- 1 Select **File—Modify Table** and click Yes in the message box that appears.

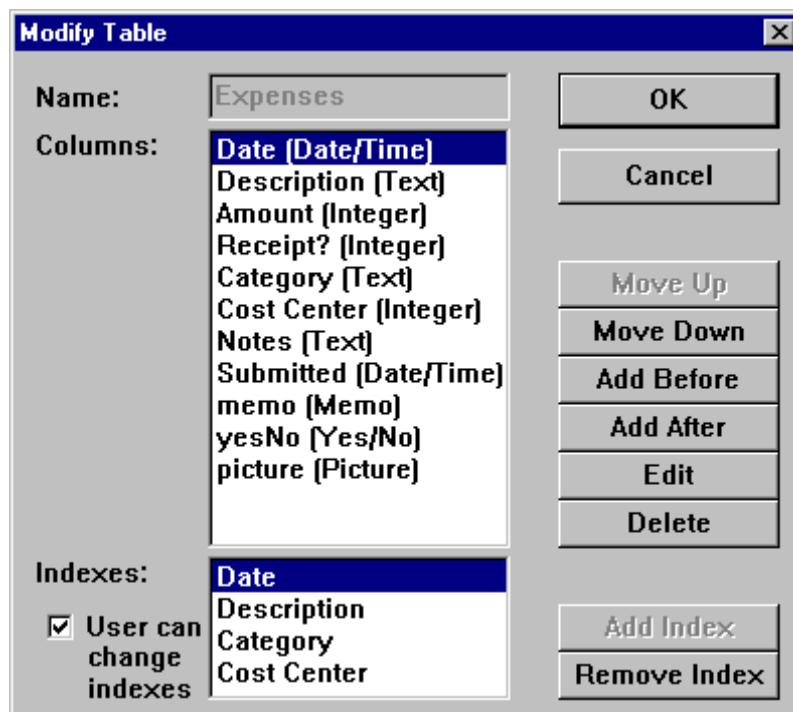
If you have not yet opened a form (.VCE application) in the development environment, Visual CE asks you to select the table you want to modify. Select the table and click OK.

Alternatively, if a form (.VCE application) is already open, Visual CE automatically opens the Modify Table dialog box for the table upon which that form is built.

Tip

If a form is already open but you want to modify a table other than the one upon which the form is built (e.g., a lookup or other support table), hold down the **Shift key** while selecting **File—Modify Table**. After you click Yes in the message box that appears, Visual CE displays the Select Table dialog box. Select the table and click OK.

Visual CE displays the Modify Table dialog box.



- 2 You can make changes to the table structure as desired (except you cannot change the table name). Refer to the section, "Creating a Table in Visual CE," on page 139 for more information.
- 3 If you are working on the desktop PC, you can also use this dialog box to index columns for sort and search orders. See "Indexing Columns," on page 283.
- 4 If the table is associated with the currently open form, you can "freeze" the indexes so that the user cannot change them. (This setting is a property of the form, not the table.) To do this, deselect the **User can change indexes** checkbox. This is useful if you have a macro that depends upon indexed columns, for example. Users will be unable to add or delete indexes in the application.

Deleting Tables

Deleting Visual CE tables from the CE device is easy. You can delete any number of tables at one time.

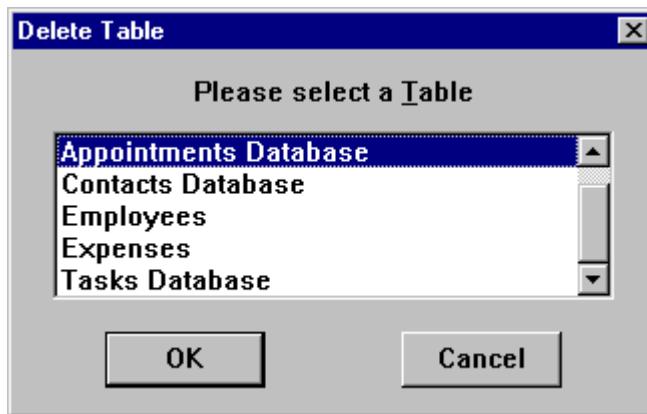
Notes

- You must be working on the desktop PC, and the CE device must be connected to it.
- If the table is in a Pocket Access database (.cdb file), you must use Pocket Access to delete it.
- If the table is in a SQL Server CE database (.sdf file), you must use SQL Server CE to delete it.

On your desktop PC:

- 1 Select **File–Delete Table**.

Visual CE displays the Delete Table dialog box.



- 2 Click the names of the tables you want to delete, then click **OK**.

Visual CE asks you to confirm each selection, then deletes the specified table from the CE device.

Changing a Form's Table

You can change the table associated with an existing form, as long as the new table has the same number of columns as the old one, and the column names and data types associated with each column are identical in both tables.

To change the table associated with a form:

- 1 With the form open in the Visual CE development environment, select **Form-Table**. After a few moments, Visual CE displays a list of available tables in the Select Table dialog box.
- 2 Highlight the name of the new table that you want to associate with the form, and click **OK**.

Tip

You can create a new, empty table and associate it with an existing form in a single operation. See "Cloning a Table," on page 142 for more information.

Using Global Variables

About Using Global Variables to Store Temporary Values

Most Visual CE applications consist of a table that stores records, and a form made up of controls that read and write data to individual record fields (table columns).

It is also possible to associate a control with a Visual CE global variable, so that the control reads and writes a value to the variable rather than to a table column. This is useful when the value is required temporarily—to facilitate a task such as jumping to another table, for example.

You can even create a form that consists entirely of controls that read and/or write values to variables, so that that the form has no underlying table at all.

Global variables are also useful for passing values between macro steps.

Visual CE stores values for up to 100 different global variables at one time: @var(0), @var(1)...@var(99). The variables are global and persistent — meaning that assigned values are available to all Visual CE and Report CE applications running on the device, and that they persist until overwritten by new values — even when the application is closed and the device is turned off.

Variables can store any data type except Picture.

To associate a control with a variable, simply select the variable when you create the control (refer to the instructions for creating particular types of controls, beginning on page 166).

To use a global variable in a macro command, select the variable when specifying the command parameters. Refer to the command reference beginning on page 328.

To create a form without a table, refer to the topic, "Creating a Form without a Table," on page 157.

What Controls and Form Features Use Global Variables

The following list summarizes controls and other form features that read and write values to global variables.

Writes values to global variables:

- Controls
 - ◊ Most controls; only AutoNumber and Scribble controls write values to a table column but not to a global variable.
- Commands (Macros and Buttons)
 - ◊ Assign
 - ◊ GPS
 - ◊ Communicate | Receive
 - ◊ File | Read
 - ◊ SMS Receive message

Reads values from global variables:

- Controls
 - ◊ Dependent Drop-Down lists (filter value)
 - ◊ Grid control (key value in application table)
 - ◊ Jump (key value in "jump-from" table)
 - ◊ Lookup (key value in application table)
 - ◊ Calculated field control

Reads values from global variables:

- Commands (Macros and Buttons)
 - ◊ Assign
 - ◊ Communicate | Send
 - ◊ Dial
 - ◊ File | Open
 - ◊ File | Write
 - ◊ File | Delete
 - ◊ SMS Send message
 - ◊ Skip
 - ◊ Sleep
 - ◊ Message box
 - ◊ Timer
 - ◊ Run external
 - ◊ Record | Search (search value)
 - ◊ Option | Import (pathname of text file to import)
 - ◊ Option | Export (pathname of text file to export records to)
 - ◊ mEnable Put File
 - ◊ mEnable Get File
- Pre-defined record filter
- Alarm message

Beginning a New Form

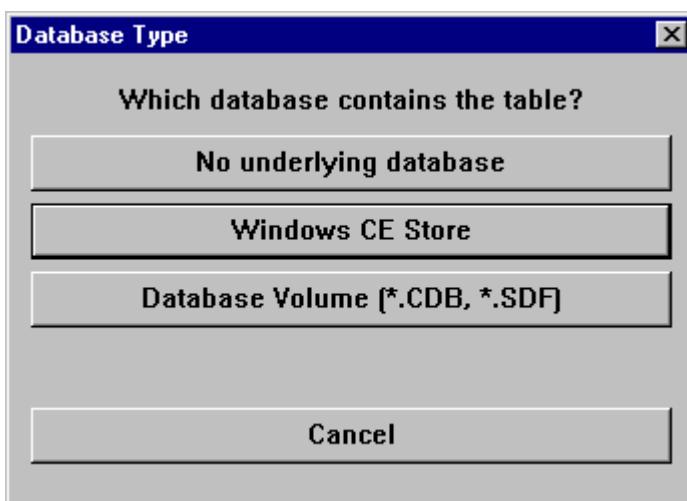
Specifying the Table to be Used by a Form

Before you can build a new Visual CE form, you select a table to store the application's records.

Note

In almost all cases, you will create forms that read and write records to a table. In a few instances, you may need to create a form that reads and writes temporary values to global variables only, in which case you do not associate the form with a table. Go to the topic, "Creating a Form without a Table," on page 157.

- 1 In Visual CE's development environment window, select **File–New Form** or click the **New Form** button .
- 2 Visual CE asks you to specify the location of the table.

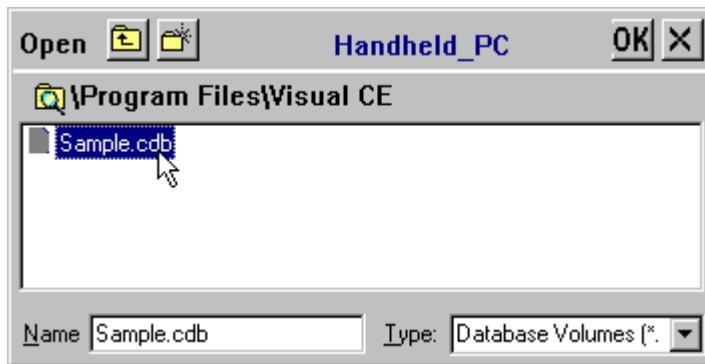


- Select **No underlying database** if you will use global variables to store data associated with all controls on the form. If you select this option, Visual CE opens the development environment window. You have completed this topic. You can now create your application using the instructions in the following sections, and you can add any type of controls to your form except AutoNumbers and Scribbles.
- Select **Windows CE Store** if you created the table using Visual CE or if you created a Pocket Access table in the Windows CE store. Go to step 4.
- Select **Database Volume** if the table is stored in a Pocket Access database (a .cdb file) or an SQL Server CE database (an .sdf file). You must have sqlceEnable, a separate SYWARE product, to use SQL Server CE databases with Visual CE.

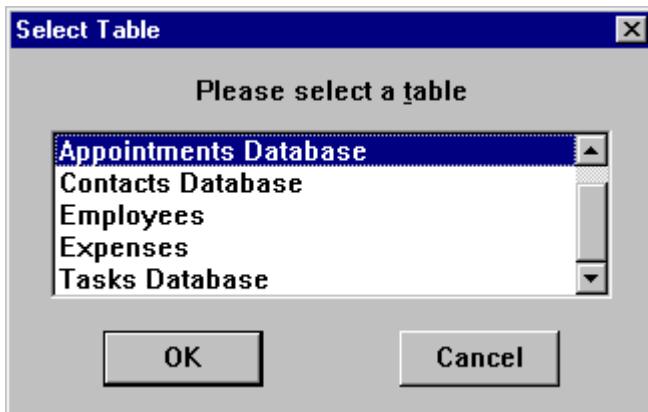
Note

If you want to run your form on the desktop PC as well as the CE device, you must use a table stored in the Windows CE Store, **not** in a .cdb or .sdf file.

- 3 If you selected the "Database Volume" option above, Visual CE displays an Open File dialog box. Navigate to the file that contains the desired table, select it, and click **OK**.



- 4 Visual CE displays the Select Table dialog box.



- 5 Click the table you want, then click **OK**.

- 6 Visual CE asks if you want the Form Wizard to create a form for this table:

- Click **Yes** if you want the Form Wizard to automatically create a control for each column in the table, including labels that correspond to the columns' names. The Wizard arranges the controls along the left side of the development window. You can then quickly rearrange the layout and change the properties of any of the controls to meet your needs, if desired.
- Click **No** if you want to create each control from scratch (as you did in Quick Tour II).

- 7 If you are working on a desktop PC and responded Yes to the previous message, Visual CE asks if you want to download the form it just created to your CE device:

- Click **Yes** if you want to begin using the application immediately, as is.
- Click **No** if you want to customize the form before downloading it.

Visual CE displays the name of the specified table in the status bar at the bottom of the development environment window. You can now create or modify the application's controls using the instructions in the following sections.

Tip

As you create your application, you should periodically save your work to avoid losing it if your computer crashes. To save the application, select **File—Save** or click the **Save** button



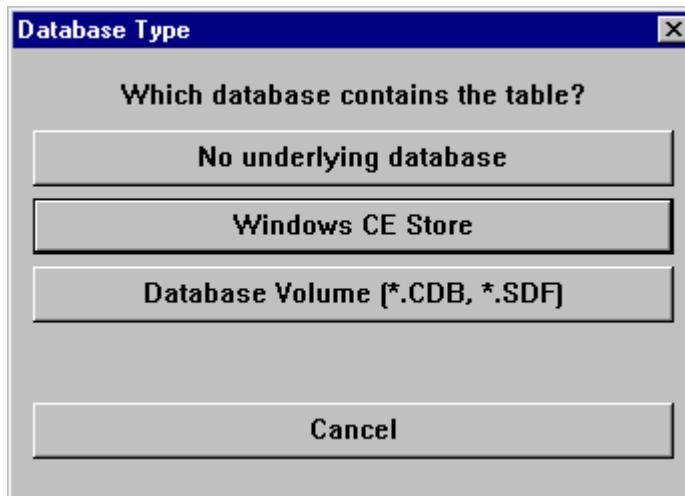
Creating a Form Without a Table

You can create a form that consists entirely of controls that read and write temporary values to global variables. In that case, there is no table associated with the form.

To start building a form that doesn't use a table:

- 1 In Visual CE's development environment window, select **File–New Form** or click the **New Form** button .

Visual CE displays the Database Type box, which asks you to specify where your application will store data.



- 2 Select **No underlying database**.

Visual CE opens the development environment window. You can now create your application using the instructions in the following sections. You can add any type of control to your form, except Scribbles and AutoNumbers.

Specifying the Form Size

By default, the form size is the same as the device screen size. You can specify a form size that is larger than the CE device's screen size.

Note

The device screen size is indicated by the dotted-line frame in the development window. If this outline does not match the size and shape of your CE device screen, you can change it using the instructions on page 128.

To change the form size:

- 1 Open the application's .VCE file or select **File—New Form** to begin designing a new application.
- 2 Select **Form—Form Size**.

Visual CE displays the Form Size dialog box.



- 3 Enter the height and width (in pixels) of the form, then click **OK**.

When a form that is larger than the device screen size is run on the CE device, Visual CE adds scroll bars to the application window to enable its users to access the entire form. Alternatively, you can add Command buttons to the form so that users can more easily move from one region of the form to another (refer to page 223 for instructions).

Tips

- If you find that you are short on space as you lay out the controls, you can remove the application's title bar. See "Customizing the Title Bar," on page 163.
- (Desktop version of development environment) You can verify how much of the form can be displayed on the CE device screen at any one time. To do this, click inside the dotted-line frame while pressing the Shift key, and drag the frame around the development environment window. Any controls that appear within the frame will appear together on the CE device screen.
- (Desktop version of development environment) After you change the form size, you may want to change the size of the development window so you can see more of the form at one time. To do this, click the lower right corner of the window so that the pointer changes to a double-headed arrow. Drag the corner outward and release the mouse button.

Hiding Scroll Bars in the Development Window (Handheld Version of Development Environment Only)

If you are creating your form directly on the CE device and the form is larger than your screen size, Visual CE automatically provides scroll bars in the development window. You can hide these scroll bars so that the entire development area is visible.

This option is available only in development mode; Visual CE automatically displays the scroll bars in the application if they are needed (that is, if the form size is larger than the device screen size and you do not provide command buttons for moving to another region of the form.)

- 1 Select **Form-Scroll Bars**.

The scroll bars are no longer visible.

- 2 To toggle the bars back on, select **Form-Scroll Bars** again.

Specifying Form Background Color

If your CE device has a color display, you can add interest and clarity to your form by applying color.

Note

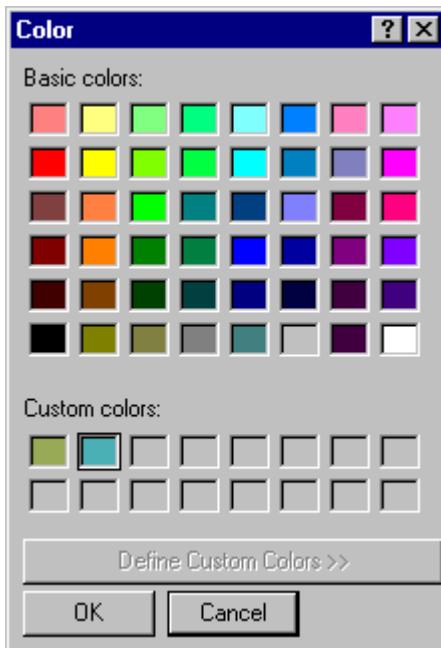
You can choose from the basic palette of 48 pre-defined colors, or you can add up to 16 custom colors to the selection. Refer to the topic, "Adding Custom Colors to the Color Palette," on page 162.

To specify the form's background color:

1 Open the application's .VCE file or select **File—New Form** to begin designing a new application.

2 Select **Form—Color**.

Visual CE displays the Color palette.



3 Click one of the color swatches, then click **OK**.

Visual CE displays the form with your selection in the background.

Tips

- If you specified a dark background color, you may want to apply a light color to the list mode text so that it is legible. Refer to the topic, "Specifying Color for List Mode Text," on page 161.
- As you create controls, you can specify their colors as well (refer to the instructions for the particular control type). You can also apply a color to several controls at once (refer to the topic, "Changing Properties for Multiple Controls," on page 267).

Specifying Color for List Mode Text

Black is the default color for text in list mode. If you have given your application a dark background color, you may want to assign a light color to the list mode text for readability.

Note

You can choose from the basic palette of 48 pre-defined colors, or you can add up to 16 custom colors to the selection. Refer to the topic, "Adding Custom Colors to the Color Palette," on page 162.

To specify list mode text color:

- 1 Select **Form–List mode color**.
Visual CE displays the Color dialog box.
- 2 Click one of the color swatches, then click **OK**.

Adding Custom Colors to the Color Palette

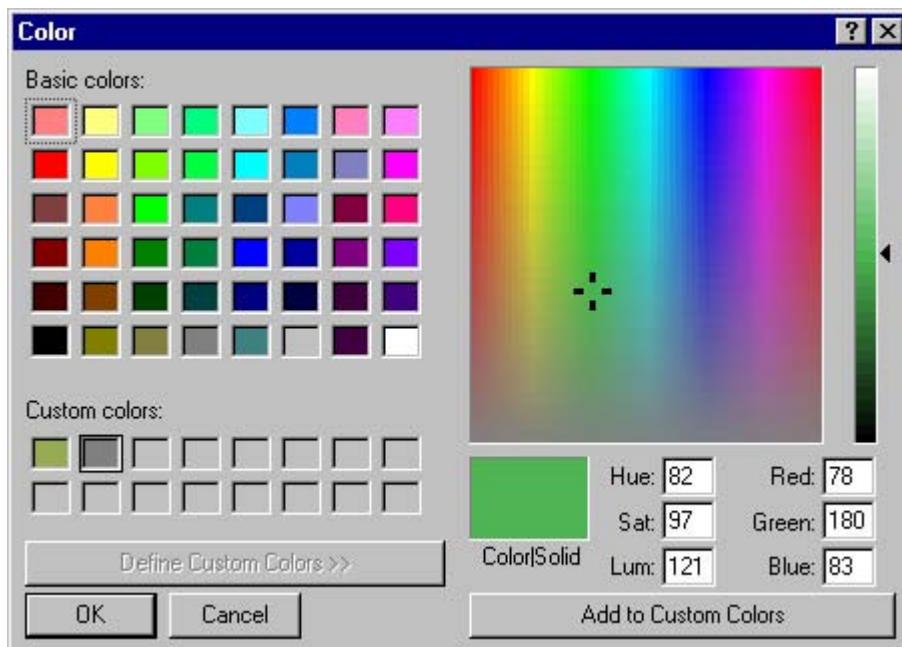
You can add up to 16 custom color swatches to the basic pre-defined palette of 48. The colors are then available to apply to any element of the form and list display (background, controls, fonts, etc.).

You must save the application to save the custom palette. Also, custom color swatches are saved with just the current .vce application.

To add a custom color swatch:

- 1 Select **Edit–Custom Colors**.

Visual CE displays the Windows Color dialog box.



- 2 Select one of the empty boxes under "Custom colors" in the lower-left portion of the dialog box. (The box you select is where the new color swatch will be placed.)
- 3 Define a new color using the controls on the right side of the dialog box, for example:
 - Click anywhere in the color matrix (the cross-hairs identify the area you clicked). The resulting color appears in the "Color/Solid" box.
 - Alternatively, you can define a color numerically by typing values for Hue, Saturation, and Luminosity; or Red, Green, and Blue. You can also change the luminosity (lightness/darkness) using the vertical slider.
- 4 Once you are satisfied with the color as it appears in the "Color/Solid" box, click **Add to Custom Colors**.
The new color swatch appears under "Custom colors."
- 5 Click **OK** to save your new color palette.

To replace an existing custom color swatch:

- 1 Click the custom color swatch you want to replace.
- 2 Define the new color as described in step 3 above.
- 3 Click **Add to Custom Colors**, and click **OK** to save the changes.

Customizing or Removing the Title Bar and Start Flag

You can specify a title for your application. The title displays in the title bar in both the development environment window and the application window. If the title is long, you may want to remove the record count that is normally displayed at the right side of the application title bar.

On the other hand, if you find that space is tight as you are laying out the controls on the form, you can gain a bit of extra room by hiding the application title bar altogether. (Note: This option does not apply to Pocket PCs.)

You also have the option of suppressing the Windows Start flag that normally provides access to other applications on the CE device, effectively limiting use of the device to the Visual CE application.

- 1 Select **Form–Title Bar**.

Visual CE displays the Title Bar dialog box.



- 2 Enter the title in the **Title** field.

- 3 Select **No title bar** to remove the title bar from the application. (If you specified a title in Step 2, it will still appear at the top of the development window.) Do not select this option if you are creating a form for a Pocket PC.

- 4 Select **No record count** to disable the record count display.

If you leave this option unchecked, your Visual CE application will automatically display the current number of records in the associated table at the right side of the title bar.

- 5 Select **No start flag** to suppress display of the Windows Start flag on the CE device.

- 6 Click **OK**.

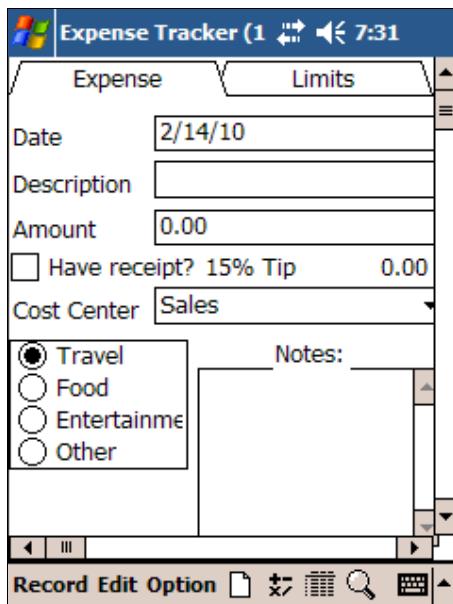
Visual CE displays your title at the top of the development window. If you selected "No title bar," the vertical dimension of the dotted-line frame indicating the device screen size in the development window expands accordingly.

Tip

You can also hide or customize the application menu bar. Refer to "Customizing or Removing the Menu Bar" on page 272 for details.

Adding Tabs

Tabs enable you to group the controls on your form into categories and give your users quick access to each group.



Tips

- It's generally easier to create the tabs first, then create the controls on each tabbed page. If you create the controls first, and then the tabs, Visual CE automatically puts all of the controls on the first page, in which case you'll have to cut and paste some of the controls onto the other pages.
- The tabs are displayed at the top of the application window. Visual CE automatically sets the width of all tabs to the width of the longest word in any of the tab names. If you create more tabs than fit in the screen's width, Visual CE automatically adds left and right scrolling arrows.

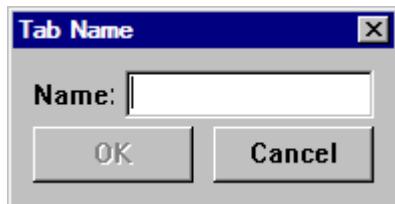
To add tabs to your application:

- 1 Select **Form-Tabs**.
Visual CE displays the Tabs dialog box.



- 2 Select **Add Before**.

Visual CE displays the Tab Name dialog box.



- 3 In the **Name** field, enter the text you want to appear on the first tab in the application window, and click **OK**.
- 4 If desired, click the **Text** button to assign a color to the tab text, and click the **Back** button to assign a color to the tab background. The default is black text on a white background.
- 5 Specify the remaining tabs:
 - Click an existing tab name in the **Tabs** window to highlight it.
 - Click the **Add After** button if the new tab is to follow the highlighted tab, or **Add Before** if the new tab is to precede it. (The tabs will appear from left to right across the top of the application window in the order that they appear in the Tabs window.)
 - When the Tab Name dialog opens, enter the tab text in the **Name** field and click **OK**.
 - If you want to change the order of the tabs as they will appear in the application window, click a tab name to highlight it, then click the **Move Up** or **Move Down** button.
- 6 When you have finished defining the tabs, click **OK** on the Tabs dialog box.
Visual CE displays the tabs in the development window as they will appear on your form.
- 7 To add controls to each tabbed page, simply select the tab in the development window and add controls as you normally would.

Adding Controls to the Form

About Controls

You can create any of the following types of controls in your Visual CE application:

- Edit boxes – see page 167
- Note boxes – see page 177
- Labels – see page 179
- Checkboxes – see page 180
- Radio buttons – see page 182
- Drop-down lists – see page 184
- Calculated fields – see page 201
- Command buttons – see page 213
- Scribble boxes – see page 237
- Timestamp buttons – see page 239
- Image controls – see page 241
- Jump buttons – see page 248
- AutoNumber – see page 252
- Lookups – see page 254
- Grid controls – see page 257
- mEnable stoplight – see page 262

Creating Edit Boxes

An edit box is a single-line text box. It can display the value of a table column or global variable, and/or accept the user's entry of a value for a table column or global variable.

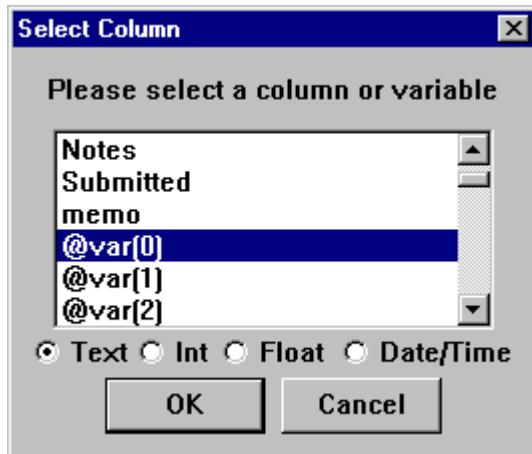
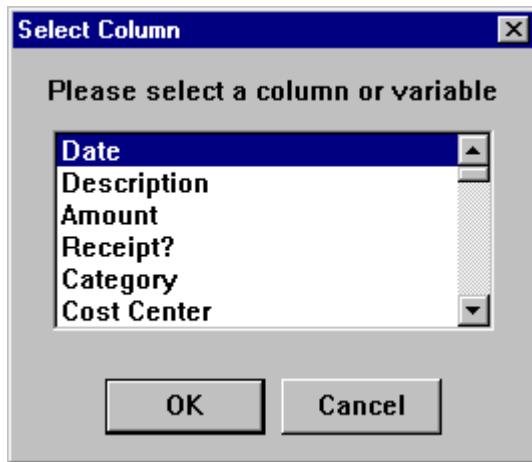
Edit boxes can be read-only or read/write. An edit box can be used to display data from any type of column in the table, except Pictures and Yes/No. An edit box can be associated with a global variable that stores text, integers, floating point numbers, or date/times.

Visual CE displays any edit box value as a character string. If an edit box is associated with a column or variable that stores integers, Visual CE can store and display the values as either whole numbers or as money values with two decimal places. If an edit box is associated with a column or variable that stores numeric data (integer or floating point) or date/time values, and the user leaves the field blank, your Visual CE application stores a null value.

You can also apply the Password property to create a password box that hides user-entered characters with bullets.

To create an edit box:

- 1 Select **Control–Edit**, or click the **Edit** button , or right-click where you want the control to appear and select **Edit**.
Visual CE displays the Select Column dialog box.
- 2 Select the table column or global variable to which the edit box relates. If you select a variable, you must also choose the datatype that the control will read/write to the variable.



3 Click **OK**.

Visual CE displays the Edit Properties dialog box, which contains the following fields:

Field	Description
Connected to [column or variable]	The table column or global variable to which the control relates – cannot be changed.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control is positioned relative to the frame; i.e., the device screen. The control remains in the same place on the screen, even if the user scrolls the display or moves to another region of the form. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Font	The font, size and style of the type that appears in the edit box. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.
Events	Optional. Select macros to be triggered when the user puts the focus on and/or moves the focus away from the control. Refer to "Triggering Macros with Events," on page 320.
Text	The color of the type that appears in the edit box (ignored if your CE device's display is black and white).
Back	The color of the background in the edit box (ignored if your CE device's display is black and white).
Include in list	If checked, data from this column or variable appears in list display mode. If unchecked, it's excluded from the list, but still appears in the single record display.
Include when printing	If checked, data from this box appears in the printed version of the record.
Read-only	If checked, users cannot change the control's value.
Password	Text Boxes Only. If checked, characters entered by the user are hidden by bullets.
Required	Lite Edition Only. If the column you specified is associated with text, the Required checkbox appears in the Edit Properties dialog box. If checked, users must enter data before saving the record. (Personal and Pro Edition Users, refer to the topic, "Using Min/Max to Make an Edit Box Required," on page 175.)
Use edit mask	Text Boxes Only. Allows you to specify an input structure for data that has a consistent format. For example, you might use (###) ###-#### for US telephone numbers. Refer to the topic, "Edit Box Input Masks," on page 170 for details.

Default	The default value for the field. This option is unavailable for text fields when "Use edit mask" is checked. Refer to the topic, "Edit Box Default Values," on page 171 for details.
File Selection on Click	For text columns or variables intended to store pathnames of files to be opened or saved (you choose which using the radio buttons that appear when you enable the option). If checked, a File Open (or File Save) dialog automatically displays when the user selects the control. This feature is especially useful when the form has a command button that launches another program requiring a file pathname as an argument. Refer to the topic, "Creating a Command Button to Start Another Program," on page 226 for details. Note: If you intend to run your Visual CE application on a desktop PC, please see the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," on page 296.
Word List	Available if the column or variable you specified is associated with text or memos. This feature enables you to specify a list of words or phrases that users can pick from to include in their input. Visual CE draws the list from a column in the form's current table or another table you specify. Refer to the topic, "Edit Box Word Lists," on page 176.
Money	If the column you specified is associated with either integers or floating point numbers, the Money checkbox displays. If Money is checked, entries are displayed with two decimal places. Refer to the topic, "Edit Box Money Values," on page 173 for details.
[Date/time options]	If the column or variable is associated with dates/times, four options are available for the type of data that can be entered and displayed: Long Date, Short Date, Time, or Date/Time. Note: The display format associated with each option is determined by the Regional Settings on the computer running the form. Refer to the topic, "Edit Box Date/Time Formats," on page 174 for details.
Min, Max	Check one or both of these boxes to specify a range of allowable inputs. (For text boxes, Min and Max sets the number of characters allowed.) If you select the Min or Max checkbox, Visual CE displays a button to click to enter a new minimum or maximum value. Note: You can use the Min/Max options to make a field required. Refer to the topic, "Using Min/Max to Make an Edit Box Required," on page 175 for details.

- 4 Make any changes required to the edit box properties, then click **OK**.
- 5 If necessary, drag the edit box to the correct location.
- 6 If necessary, create a label to identify the edit box.

Edit Box Input Masks

Input masks are useful for data that can or should be entered in a consistent format, such as telephone numbers or social security numbers. You can incorporate these masks in edit boxes to ensure that users enter data in the desired format, as well as to save them the time and effort of entering characters, spaces, or punctuation that appear in every instance.

For example, a mask for U.S. telephone numbers might be: (###) ###-####. When the user types 6175551212 into the edit box on the form, Visual CE displays (617) 555-1212.

To create an input mask:

- 1 In the Edit Properties dialog, click the **Use edit mask (#?A&)** checkbox to check it.
- 2 In the **Mask** field that appears, specify the mask using the following guidelines:
 - You must supply a character for each position in the mask. If the mask consists of 14 character positions as in the telephone number example above, you would type a total of 14 characters.
 - Each character that you supply can be one of two types: either a "literal" or a "placeholder."
 - Literals appear in the application's edit box exactly as you type them, and the user cannot change them. In the telephone example, the literals are the parentheses, space, and dash that appear in the first, fifth, sixth, and tenth positions of the mask, respectively.
 - Placeholders stand for the characters that the user supplies. To signify a placeholder for a position, choose from the following:
 - # allows the user to enter a digit (0 to 9)
 - ? allows the user to enter a character (a to z and A to Z)
 - A allows the user to enter a digit (0 to 9) or a character (a to z and A to Z)
 - & allows the user to enter any character (a to z and A to Z; plus any special characters such as dashes, commas, etc.)
 - If you would like to use one of the placeholders (#, ?, A, or &) as a literal, put a backslash (\) in front of it. For example, say you want your mask to appear to the user this way: PART _____. (The word PART with spaces for inputting four digits). Normally, Visual CE would interpret the character "A" as a placeholder. So if the user entered 1234, it would appear on the device this way: P1RT 234_. To get the result you want, create the mask this way: P\ART ####. When the user types 1234, the result would be PART 1234.

Edit Box Default Values

In the Default field of the Edit Properties dialog box, your options depend on whether the table column or global variable associated with the control is defined as text, numeric, or date/time.

Text Edit Controls

For text edit controls, enter the default text, if any. **Note:** The option to define a default is available only when "Use edit mask" is unchecked.

Numeric Edit Controls (Integer or Floating Point Number)

For numeric edit controls, 0 is the initial default value. Alternatively, you can specify that the default be no value (null) or a particular value.

To specify that the default value is an empty field, click the **Null** checkbox.

To specify a particular value as the default:

- 1 If the numeric column is a money column, click the **Money** checkbox first. (If the numeric is **not** a money column, go to Step 2.)

Visual CE inserts the decimal point in your default entry.

0

- 2 Click the Default button

Visual CE displays a number pad dialog box. (If you specified money or floating point, the number pad includes a decimal point key.)



- 3 Click the buttons to enter the default amount – the number you enter displays in the title bar of the dialog box (if you make a mistake, click the **Clear** button, then enter the number again).
- 4 To confirm your entries, click **OK**.

Date/Time Edit Controls

The initial default is Now (the current date and/or time). Alternatively, you can specify that the default be no date/time (null) or a particular date and/or time.

To specify that there is no default date/time, click the **Null** checkbox.

To specify a particular date and/or time as the default:

- Select an option for the amount of information stored in the date/time column:

Option	Sample
Long Date	Thursday, April 22, 1999
Short Date	4/22/99
Time	2:28:00 PM
Date/Time	4/22/99 2:28:00 PM

Note

The Windows' Regional Settings on the CE device determine the format that Visual CE uses to display the selected option in the single record display mode. Refer to the topic, "Edit Box Date/Time Formats," on page 174 for details.

- Click the **Now** or **Null** checkbox so that neither box is checked.

Visual CE displays a button that identifies the date and/or time to be used as the default in the application.

4/22/99 2:28:00 PM

- Click the date/time button.

Visual CE displays a calendar dialog box. The elements displayed correspond to the format you specified in Step 1. For example, if you specified Time, only the time selectors are available.



- Specify the default date and/or time (again, only the options that correspond to the option you selected in Step 1 are available):

- Click the month (from the drop-down list)
- To change the year, enter it directly, or click the right-/left-arrow to increase/decrease the year
- Click the day in the calendar area
- To change the time, type hours and minutes directly or select from the drop-down lists; choose either am or pm
- To change the default back to the current date/time, click **Now**
- To confirm your entries, click **OK**

Edit Box Money Values

For columns or variables that contain numeric entries, the Edit Properties dialog box displays the Money checkbox. The Money checkbox is not checked initially.

To identify numeric entries as money values:

- 1 Click the **Money** checkbox to check it.

Visual CE updates the **Default** button to 0.00 (showing two decimal places).



- 2 To enter a specific money value as the default:

- Click the **Default** button (with the 0.00 label)
Visual CE displays a number pad dialog box, which includes a decimal point key for money entries.
- Click the number buttons to enter the default amount – the number you enter displays in the title bar of the dialog box
(If you make a mistake, click the **Clear** button, then enter the number again.)
- To confirm your entries, click **OK**.

Edit Box Date/Time Formats

For date/time columns or variables, the Edit Properties dialog box enables you to choose between four options for the amount of information stored: **Long Date**, **Short Date**, **Time**, and **Date/Time**.

Option	Sample Display
Long Date	Monday, April 26, 1999
Short Date	4/26/99
Time	9:30:37 AM
Date/Time	4/26/99 9:30:37 AM

In list mode, Visual CE automatically uses the most compact display format for the type of date/time information stored.

In the single record mode, on the other hand, Visual CE uses the display format specified in the Windows CE Regional Settings. For example, possible display formats for long dates include Monday, April 26, 1999 and 26 April, 1999; short dates include 4/26/99 and 1999-04-26. Many other formats are available for each option.

If you are using Visual CE development environment on the desktop PC: It's recommended that you match the Regional Settings on the desktop PC to those on the CE device when you create the form, so that you can size the controls to accommodate the length of the dates/times to be displayed.

To modify the regional settings (this procedure is the same for either a CE device or desktop PC):

- 1 Select **Start-Settings-Control Panel-Regional Settings**.
- 2 Click the **Date** tab to specify a date format. Click the **Time** tab to specify a time format.

Using Min/Max to Make an Edit Box Required

If you make an edit box required, users must enter data in that box before they can save the record.

You can make a text, numeric, or date/time edit box required by applying the Min and Max options in the Edit Properties dialog box. Simply specify a legal range, and set the field's default value outside that range.

For text edit boxes:

- 1 Leave the Default field empty.
- 2 Click the **Min** checkbox.
Visual CE displays a button to the right of the checkbox. The number it displays is the current minimum number of characters that must be entered into the field.
- 3 Click the button.
Visual CE displays the number pad dialog box.
- 4 Enter any non-zero number, then click **OK**.

For numeric edit boxes:

Use the following steps if the number can be any positive value:

- 1 Set the **Default** value to any negative number.
 - Click the Default button
 - Use the number pad dialog box to enter a negative value

Tip

To specify a negative number using the number pad dialog box, click the desired number, then click the "+/-" button and OK.

- 2 Click the **Min** checkbox.
Visual CE displays a button to the right of the checkbox. The number it displays is the current default for the minimum value allowed. You can leave the zero value in place.

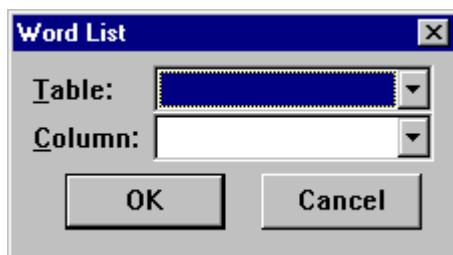
Edit Box Word Lists

Word lists are convenient pop-up lists of words and phrases that users can select to complete an edit or note box. Users display the list by double-clicking the edit or note box. They can edit the selection once entered into the box, and they can intermingle selections with typed input. Users can even save their own words in the word list.

Visual CE dynamically draws the list from a text or memo column in the form's current table or any other table you specify.

To specify a word list for an edit box or note box:

- 1 In the Edit Properties dialog box, click the Word List button . Visual CE searches the connected CE device for eligible tables. This can take several moments. Visual CE then displays the Word List dialog box. It is initially empty.



- 2 Select a table and a column; then click **OK**.

Note

If you intend to run your Visual CE application on a desktop PC, please see the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," on page 296.

Creating Note Boxes

A note box is a multi-line text box. A note box can display the value of a table column or global variable, and/or accept the user's entry of a value for a table column or global variable.

Note boxes can be read-only or read/write. A note box can be used to display any column in the table, except Pictures or Yes/No. A note box can be associated with a global variable that stores text, integers, floating point numbers, or date/times.

Visual CE displays any note box value as a character string. If a note box is associated with an integer value, Visual CE can display it as a whole number or as a money value with two decimal places.

To create a note box:

- 1 Select **Control–Note**, or click the **Note** button , or right-click where you want the control to appear and select **Note**.
Visual CE displays the Select Column dialog box.
- 2 Select the table column or global variable to which the note box relates. If you select a variable, you must also chose the datatype that the control will read/write to the variable.
- 3 Click **OK**.
Visual CE displays the Note Properties dialog box, which contains the following fields:

Field	Description
Connected to [column or variable]	The table column or global variable to which the control relates – cannot be changed.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control is positioned relative to the frame; i.e., the device screen. The control remains in the same place on the screen, even if the user scrolls the display or moves to another region of the form. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Font	The font, size and style of the type that appears in the note box. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.
Events	Optional. Select macros to be triggered when the user puts the focus on and/or moves the focus away from the control. Refer to "Triggering Macros with Events," on page 320.
Text	The color of the type that appears in the note box (ignored if your CE device's display is black and white).
Back	The color of the background in the note box (ignored if your CE device's display is black and white).

Include in list	If checked, data from this column or variable appears in list display mode. If unchecked, it's excluded from the list, but still appears in the single record display.
Include when printing	If checked, data from this box appears in the printed version of the record.
Read-only	If checked, users cannot change the control's value.
Default	The default value for the field.
File Selection on Click	Refer to the topic, "Edit Box Default Values," on page 171 for details.
	For text columns or variables intended to store pathnames of files to be opened or saved (you choose which using the radio buttons that appear when you enable the option). If checked, a File Open (or File Save) dialog automatically displays when the user selects the control. This feature is especially useful when the form has a command button that launches another program requiring a file pathname as an argument.
	Refer to the topic, "Creating a Command Button to Start Another Program," on page 226 for details.
	Note: If you intend to run your Visual CE application on a desktop PC, please see the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," on page 296.
Word List	Available if the column or variable you specified is associated with text or memos. This feature enables you to specify a pop-up list of words or phrases that users can pick from to include in their input. Visual CE draws the list from a column in the form's current table or another table you specify. Refer to the topic, "Edit Box Word Lists," on page 176.
Money	If the column or variable you specified is numeric (either integers or floating point numbers), the Money checkbox displays. If Money is checked, entries are automatically displayed with two decimal places.
	Refer to the topic, "Edit Box Money Values," on page 173 for details.
[Date/time options]	If the column or variable is associated with dates/times, four options are available for the type of data that can be entered and displayed: Long Date, Short Date, Time, or Date/Time. Note: The display format associated with each option is determined by the Regional Settings on the CE device. Refer to the topic, "Edit Box Date/Time Formats," on page 174 for details.

- 4 Make any changes required to note box properties, then click **OK**.
- 5 If necessary, drag the note box to the correct location.
- 6 If necessary, create a label to identify the note box.

Creating Labels

Labels display text that the user cannot modify. A label can be used to identify other controls, such as edit boxes and note boxes, that do not have their own captions. Visual CE always displays the same value, regardless of the current record.

Labels are always read-only. Unlike other Visual CE controls, labels are not associated with table columns or global variables.

To create a label:

- 1 Select **Control–Label**, or click the **Label** button , or right-click where you want the control to appear and select **Label**.

Visual CE displays the Label Properties dialog box, which contains the following fields:

Field	Description
Text	Enter the text to be displayed as a constant value. You can use spaces and carriage returns to format the text.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control is positioned relative to the frame; i.e., the device screen. The control remains in the same place on the screen, even if the user scrolls the display or moves to another region of the form. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Font	The font, size and style of the type for the label. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.
Text	The color of the label's type (ignored if your CE device's display is black and white).
Back	The color of the label's background (ignored if your CE device's display is black and white).
Left / Center / Right	Alignment of text within the box that contains it: left aligned, centered, or right aligned.

- 2 Make any changes required to label properties, then click **OK**.
- 3 Drag the label to the correct location.
- 4 If necessary, change the label's width and/or height.

Creating Checkboxes

A checkbox enables the user to turn an option off or on like a switch. An "X" appears in the checkbox to show a choice is selected.

Checkboxes can be read-only or read/write. Checkboxes can be associated with any text, integer, floating point number, memo, or yes/no column in the table; they cannot be associated with date/time or picture columns. A checkbox can be associated with a global variable.

If the checkbox is associated with text or memos, Visual CE checks the box for strings whose first value is either 1, Y, y, T, or t, and clears the box for strings that do not begin with any of those values.

If the checkbox is associated with an integer or floating point number, Visual CE checks the box for any non-zero value, and clears the box for any zero value.

To create a checkbox:

- 1 Select **Control–Checkbox**, or click the **Checkbox** button , or right-click where you want the control to appear and select **Checkbox**.
Visual CE displays the Select Column dialog box.
- 2 Select the table column or global variable to which the checkbox relates, then click **OK**.
Visual CE displays the Checkbox Properties dialog box, which contains the following fields:

Field	Description
Connected to [column or variable]	The table column or global variable to which the control relates – cannot be changed.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control is positioned relative to the frame; i.e., the device screen. The control remains in the same place on the screen, even if the user scrolls the display or moves to another region of the form. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Font	The font, size and style of the caption that appears to the right of the checkbox. Click to modify. Refer to "Specifying Font Properties for a Control" on page 265 for details.
Events	Optional. Select macros to be triggered when the user puts the focus on or away from the control, or changes the value in the control. Refer to "Triggering Macros with Events," on page 320.
Text	The color of the caption displayed to the right of the checkbox (ignored if your CE device's display is black and white).
Back	The color behind the checkbox and caption (ignored if your CE device's display is black and white).
Include in list	If checked, data from this box appears in list display mode. If unchecked, it's excluded from the list, but still appears in the single record display.

Include when printing	If checked, data from this box appears in the printed version of the record.
Read-only	If checked, users cannot change the control's value.
On by default	If checked, the checkbox is initially checked when a new record is created. If not checked, the checkbox is initially clear when a new record is created.
Text	The caption to be displayed to the right of the checkbox.

- 3 Make any changes required to checkbox properties, then click **OK**.
- 4 Drag the checkbox to the correct position, then resize it if necessary.

Creating Radio Buttons

Radio buttons offer a set of choices from which the user can select one option. You can require the user to explicitly select an option, or you can provide a default value.

Radio buttons can be read-only or read/write. They can be associated with any text, integer, floating point number, or memo column in the table; they cannot be associated with date/time, picture or yes/no columns. Radio buttons can be associated with global variables that store text, integers, or floating point numbers.

If radio buttons are associated with text or memos, the caption of the button selected is the value stored in the table or variable.

If the radio buttons are associated with an integer or floating point number, you must specify a caption and a unique numeric code for each option. While the user sees the caption, it is the number code that is stored.

Visual CE creates a radio button for each caption you specify. You can specify up to eight radio buttons per set of radio buttons.

You can arrange the radio buttons in a vertical stack or horizontal line, and you can choose whether or not a border appears around them.

To create radio buttons:

- 1 Select **Control–Radio buttons**, or click the **Radio Buttons** button , or right-click where you want the control to appear and select **Radio Buttons**.
Visual CE displays the Select Column dialog box.
- 2 Select the table column or global variable to which the radio buttons relate. If you select a variable, you must also choose the datatype that the control will read/write to the variable. Click **OK**.
Visual CE displays the Radio Buttons Properties dialog box, which contains the following fields:

Field	Description
Connected to [column or variable]	The table column or global variable to which the control relates – cannot be changed.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control is positioned relative to the frame; i.e., the device screen. The control remains in the same place on the screen, even if the user scrolls the display or moves to another region of the form. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Font	The font, size, and style of the caption that appears to the right of the radio button. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.

Events	Optional. Select macros to be triggered when the user puts the focus on or away from the control, or changes the value in the control. Refer to “Triggering Macros with Events,” on page 320.
Text	The color of the radio buttons' captions (ignored if your CE device's display is black and white).
Back	The color behind the radio buttons and captions (ignored if your CE device's display is black and white).
Include in list	If checked, data from this column or variable appears in list display mode. If unchecked, it's excluded from the list, but still appears in the single record display.
Required	If checked, the user must select one of the buttons before he/she can save the record. Note: If you wish to use this feature, do not specify a default value in Step 3.
Include when printing	If checked, data from this column or variable appears in the printed version of the record. If unchecked, it's excluded.
Read-only	If checked, users cannot change the control's value.
Horizontal	If checked, the radio buttons are arranged in a row on the form. If unchecked, they are stacked one above the other.
Border	If checked, a border appears around the set of radio buttons. If unchecked, no border appears.

- 3 After making any changes necessary to radio button properties, click the **Edit** button to specify the properties of the first button in the group.

Visual CE displays the Option Specification dialog box, which contains the following fields:

Field	Description
Name	The caption to be displayed to the right of the radio button.
Default	If checked, this radio button is initially selected when a new record is created. Only one radio button can be the default.
Code	Only shown if the control is associated with a column or variable that stores integers or floating point numbers. Enter a unique numeric code for each radio button specified.

- 4 Make the entries required for the first radio button caption, then click **OK**.

Note

Visual CE displays an asterisk (*) after the caption of the default radio button.

- 5 Specify the remaining buttons in the group:

- Click an existing radio button caption to highlight it.
- Click the **Add After** button if the new caption is to follow the highlighted caption, or **Add Before** if the new caption is to precede the highlighted caption.
- Enter the new caption in the Name field, (click Default if applicable).
- If you want to change the order of the radio buttons as they will appear on the form, click a radio button caption to highlight it, then click the **Move Up** or **Move Down** button.
- When you have finished specifying the properties, click **OK**.

- 6 If necessary, drag the control to the correct location, and resize it so all the radio buttons and their names can be clearly read.

Creating Drop-Down Lists

A drop-down is a scrollable list of choices from which the user can choose one or a number of options, or (for text or memos) enter a new option. Drop-down lists are especially useful for enabling users to pick from a group of frequently used values without having to type, or for restricting input to a set of allowable values. You can require the user to explicitly select an option, or you can provide a default value.

Drop-downs are similar to radio buttons, but they offer three additional advantages: 1) The list of options can be any length (not restricted to the space available on the form); 2) You can allow users to type in new options which become available for future selection; and 3) You can allow users to choose more than one option.

Drop-downs can be associated with any text, number, or memo column in the table; they cannot be associated with date/time, yes/no, or picture columns. They can be associated with global variables that store text, integers, or floating point numbers.

If the drop-down list is associated with text or memos, the name that the user picks from the drop-down list is the value stored in the table or variable. If the drop-down list is associated with a numeric column or variable, you must specify a name and a unique numeric code for each option. While the user always sees the name, it is the number code that is stored. Saving codes rather than text strings conserves space on the CE device.

User-defined entries: For text and memos, you can allow users to type new values (rather than just pick from the list). As the user types new values, the option is "auto-completed" with values already in the list. In addition, you can specify whether or not the new values appear in the drop-down for future selection. You cannot allow user-defined entries for integer or floating point columns or variables.

Multi-value select: For text and memos, you can allow users to select more than one option in the list. When stored in the table or variable, multiple values are separated by semi-colons.

Tip

You can also create "navigational" drop-downs that enable users to find existing records by picking from the list. Refer to "Creating Navigational Drop-Down Lists" on page 199.

To create a drop-down list:

- 1 Select **Control–Drop Down**, or click the **Drop Down** button , or right-click where you want the control to appear and select **Drop Down**.

Visual CE displays the Select Columns dialog box.

- 2 Select the table column or global variable to which the drop-down list relates. If you select a variable, you must also choose the datatype that the control will read/write to the variable. Click **OK**.

Visual CE displays the Drop Down Properties dialog box, which contains the following fields:

Field	Description
Connected to [column or variable]	The table column or global variable to which the control relates – cannot be changed.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The number of entries displayed at one time in the scrollable list box.

Relative to Frame Rectangle	If checked, the control is positioned relative to the frame; i.e., the device screen. The control remains in the same place on the screen, even if the user scrolls the display or moves to another region of the form. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Font	The font, size and style of the type in the list. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.
Events	Optional. Select macros to be triggered when the user puts the focus on or away from the control, or changes the value in the control. Refer to "Triggering Macros with Events," on page 320.
Text	The color of the type in the list (ignored if your CE device's display is black and white).
Back	The color of the background inside the drop-down list box (ignored if your CE device's display is black and white).
Include in list	If checked, data from this column or variable appears in list display mode. If unchecked, it's excluded from the list, but still appears in the single record display.
Required	If checked, the user must select or provide a value before he/she can save the record. Note: If you wish to use this feature, do not specify a default value in Step 3.
Include when printing	If checked, data from this column or variable appears in the printed version of the record. If unchecked, it's excluded.
User can add	(Available only for text or memos.) If checked, users can type new values, rather than just pick from the list.
Multi-value select	(Available only for text or memos. Unavailable if "User can add" option is checked.) If checked, the user can select more than one entry in the list. When stored in the table or variable, multiple values are separated by semi-colons.
Add new values to list	(Available only if "User can add" option is checked.) If checked, all user-supplied options automatically appear in the drop-down list.
Navigational	(Available only if control is connected to an indexed column.) Check to create a drop-down list control used to find existing records rather than to store values. (Refer to "Creating Navigational Drop-Down Lists" on page 199.)
Include values from table	Check to specify that Visual CE create the list from the contents of a column in another table. (Refer to "Using Values from Another Table as a Drop-Down List" on page 187 for details.)

- 3 After making any changes necessary to drop-down list properties, click the **Add Before** button.

Visual CE displays the Option Specification dialog box, which contains the following fields:

Field	Description
Name	The text to be displayed in the drop-down list for this option.
Default	If checked, this option is initially selected when a new record is created. Only one option in the drop-down list can be the default.
Code	If the control is associated with an integer or floating point number column or variable, this is the numeric code for the option.

- 4 Make the entries required for the first drop-down list caption, then click **OK**.

Note

Visual CE displays an asterisk (*) after the list's default option.

- 5 Specify the remaining drop-down list options:

- Click an existing caption to highlight it.
- Click the **Add After** button if the new caption is to follow the highlighted caption, or **Add Before** if the new caption is to precede the highlighted caption.
- Enter the option's caption in the Name field.
- Click **Default** if this option is to be the default.
- If the column or variable is numeric, enter a unique value for the option in the Code field.
- Click **OK**.

Tip

If you want to change the order of the drop down list options as they will appear on the form, click a caption to highlight it, then click the **Move Up** or **Move Down** button.

- 6 Click **OK** on the Drop Down Properties dialog box.

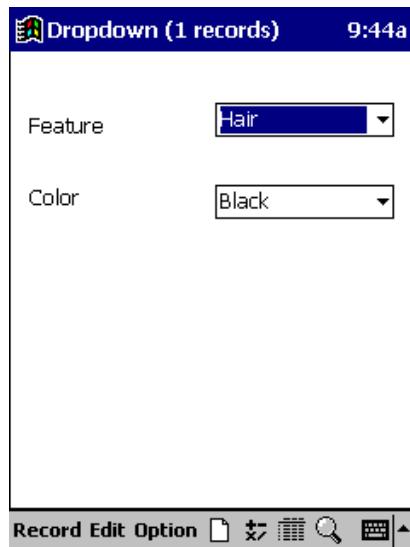
- 7 Drag the drop-down list to the correct location, and resize it if necessary.

Using Values from Another Table as a Drop-Down List

You can make it possible for users to complete a field by selecting from items stored in another table. More specifically, you can specify a drop-down list that Visual CE dynamically composes from the current contents of a column in another table.

Tip

You can see this feature in action by running the SYWARE-supplied sample application, Dropdown.vce, on your CE device. (For instructions on installing the samples, please refer to the topic, "Installing Visual CE," on page 15). The samples work only on the CE device, not the desktop PC.

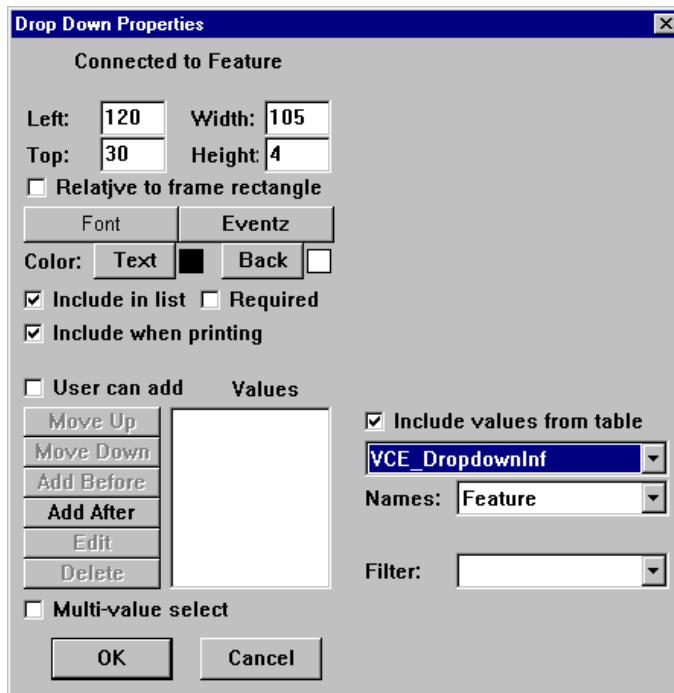


This sample form has two drop-down list controls: Feature and Color. The selections available to the user in both lists are stored in another table, the contents of which you can view using the SYWARE-supplied sample, DropdownInf.vce. (For instructions on installing the samples, please refer to the topic, "Installing Visual CE," on page 15). The samples work only on the CE device, not the desktop PC.

VCE_DropdownInf (7 records) 9:45a	
Feature	Color
Eyes	Brown
Eyes	Blue
Eyes	Green
Hair	Blond
Hair	Black
Hair	Brown
Hair	Red

To create a drop-down list using values from another table:

- 1 Select **Control–Drop Down**, or click the **Drop Down** button , or right-click where you want the control to appear and select **Drop Down**.
Visual CE displays the Select Columns dialog box.
- 2 Select the table column or global variable that will store the data. If you select a variable, you must also choose the datatype that the control will read/write to the variable. Click **OK**.
Visual CE displays the Drop Down Properties dialog box.
- 3 In the Drop-Down Properties dialog, select the **Include values from table** checkbox.
(Complete the other options as you normally would for drop-down lists. See the topic, "Creating Drop-Down Lists," on page 184.)
Visual CE searches the connected CE device to determine eligible tables. This can take several moments. Visual CE then displays additional fields in the lower right corner of the Drop Down Properties dialog.



- 4 Complete the additional fields:
 - In the first new field (directly under "Include values from table"), select the table that contains the values you want to use for the drop-down list.
 - In the **Names** field, select the column that contains the text you want to appear as options in the drop-down list.
 - If you selected a numeric column or variable in Step 2, Visual CE displays a third field, **Codes**. Select the column that contains the associated code values to be stored instead of the names.
 - **Filter:** Leave blank unless you are creating a dependent drop-down list. Refer to the topic, "Creating Dependant Drop-Down Lists," on page 189 for details.
- 5 Click **OK** on the Drop Down Properties dialog box.
- 6 Drag the drop-down list to the correct location, and resize it if necessary.

Creating Dependent Drop-Down Lists

If your application contains a field whose set of allowable values is dependent upon a value in another field, you may want to incorporate a "dependent drop-down list" in your design.

The options in a dependent drop-down list are automatically "filtered" according to the value that the user selects in another control on the form (call this value the "filter").

Consider the following example. Say your table has two columns: Feature and Color. Not all colors are appropriate choices for all features—for example, blue is a typical color for eyes but not for hair. You could set up your form so that the names displayed in the Color drop-down list depend on the user's selection for Feature. Feature is the filter, and Color is the dependent. If the user selects the feature Hair, the dependent drop-down list would offer the choices Black, Blond, Brown, and Red. But if the user selects Eyes, the dependent drop-down list would offer Blue, Brown, and Green.

Tip

You can try out this example of a dependent drop-down yourself by running the SYWARE-supplied sample application, Dropdown.vce, on your CE device. (For instructions on installing the samples, please refer to the topic, "Installing Visual CE," on page 15). The samples work only on the CE device, not the desktop PC.

All of the rules and instructions described in the previous topic, "Using Values from Another Table as a Drop-down List," on page 187 apply to dependent drop-downs.

The control for entering the filter value in the application form can be a drop-down list, but it doesn't have to be. You can associate the control with a table column if you want to store the user's entry for the filter value, or you can associate it with a global variable if you don't.

Prerequisite: Supporting Table to Store the Drop-Down Options

To incorporate dependent drop-downs in your new application, you must first create a separate table and fill it with the filter and dependent drop-down options. (You would typically build another Visual CE application to create and maintain this support information. For example, DropdownInf.vce is the supporting application for the SYWARE-supplied sample Dropdown.vce.)

The support table and its contents must meet the following requirements:

- There must be at least two columns in the support table: one to store the filter values and one to store the corresponding dependent option Names.

Note

Optionally, you can utilize space-saving numeric codes for the dependent and/or filter values, just as you would for any other drop-down list control. For example, you could create two columns for the dependent values: a text column to store the names that appear on the form in the list display, and a numeric column to store the invisible codes associated with each name. The user sees the name (e.g., black), but it is the more compact numeric code that is stored in the application table (e.g., 01).

- There must be matching columns for storing the filter values in both the support table and in the application table—that is, both columns must have the **identical column name** (e.g., Feature) and **data type** (e.g. text). Alternatively, you can associate the filter value in the application with a global variable. In that case, the variable's datatype must match that of the column in the support table.
- The column that stores the filter values in the support table must be indexed.
- Once you've created the support table, you must enter one unique row (record) for each filter/dependent pair. So for example, if you have two filter options--one with four dependent options and the other with three--you would have to create a total of seven records: $(1 \times 4) + (1 \times 3) = 7$.

The supporting table for the Feature/Color example might be organized like this:

Feature* (Filter)	Color (Dependent Name)	ColorID** (Dependent Code)
Hair	Black	01
Hair	Blond	02
Hair	Brown	03
Hair	Red	04
Eyes	Blue	05
Eyes	Brown	06
Eyes	Green	07
*Indexed		**Optional

Once you have created and filled the support table and created the corresponding columns in the application table, you are ready to create the filter and dependent drop-down list controls in your form:

- 1 First, create a control to allow the user to select the filter value (in our example, Feature). Usually this would be a drop-down control, but it does not have to be. If it is a drop-down control, use the same steps you would for any drop-down that utilizes values from another table:

- Select **Control–Drop Down**, or click the **Drop Down** button , or right-click where you want the control to appear and select **Drop Down**.
- In the Select Columns dialog, select the application table column or global variable that will store the selected filter value (in our example, Feature). If you select a variable, you must also choose the datatype that the control will read/write to the variable. Click **OK**.
- In the Drop-Down Properties dialog, select the **Include values from table** checkbox. (Complete the other options as you normally would for drop-down lists. See the topic, "Creating Drop-Down Lists," on page 184.) After a delay of several moments, Visual CE displays additional fields in the lower right corner at the bottom of the Drop Down Properties dialog.
- In the first new field (directly under "Include values from table"), select the supporting table that stores the drop-down options (in our example, VCE_DropdownInf).
- In the **Names** field, select the column that stores the text you want to appear as options in this drop-down list (in our example, Feature).
- **Filter:** Leave blank.
- Click **OK** to close the Drop Down Properties dialog box.
- Drag the drop-down list to the correct location, and resize it if necessary. Create a label to identify it.

- 2 Now, create a second control—this one is the dependent drop-down list control that will enable the user to select the dependent value. (Remember, the contents of this drop-down list will be filtered by the user's selection in the control you created in Step 1). To create the dependent drop-down:

- Select **Control–Drop Down** or click the **Drop Down** button .
- In the Select Columns dialog, select the application table column or global variable that will store the user's selection for this control (in our example, Color). If you select a variable, you must also choose the datatype that the control will read/write to the variable. Click **OK**.

- In the Drop-Down Properties dialog, select the **Include values from table** checkbox. (Complete the other options as you normally would for drop-down lists. See the topic, "Creating Drop-Down Lists," on page 184.) After a delay of several moments, Visual CE displays additional fields in the lower right corner at the bottom of the Drop Down Properties dialog.
 - In the first new field (directly under "Include values from table"), select the supporting table that stores the drop-down options (in our example, VCE_DropdownInf).
 - In the **Names** field, select the column that stores the text you want to appear as options in the dependent drop-down list (in our example, Color).
 - In the **Filter** field, select the table column or global variable that stores the filter values--the same table column name or global variable that you selected in Step 1. If you pick a variable, Visual CE asks you to specify the Key; that is, the column that stores the filter values in the support table (in our example, Feature).
 - Click **OK** to close the Drop Down Properties dialog box.
 - Visual CE displays a message asking if you want Auto Recalc turned on. Click **Yes**. Auto Recalc insures that the options displayed in the form for the dependent drop-down correspond to the user's selection in the control used to set the filter value. If this feature is turned off, and a user makes first one selection then another for the filter value, the dependent options won't adjust to the revised selection. (You would typically turn Auto Recalc off only if you have calculated fields in your form that you want to update on command, or if your CE device has performance issues. Refer to the topic, "Enabling the Auto Recalc Feature," on page 279 for more information.)
 - Drag the drop-down list to the correct location, and resize it if necessary. Create a label to identify it.
- 3 Be sure to test your application to confirm that the list of options in the dependent drop-down appear as expected. If not, confirm that the support table meets the prerequisites defined at the beginning of this topic.

Tip

You can create a series of dependent drop-downs, in which the options presented in each list are determined by the combination of the user's preceding selections, much like a decision tree. Refer to the topic, "Multi-Tiered Dependent Drop-Down Lists," on page 192 for details.

Multi-Tiered Dependent Drop-Down Lists

In the previous topic, you learned how to create a drop-down that automatically displays a list of valid options given the user's selection in another control. You can take this a step further by creating a series of dependent drop-downs in which the options available in each succeeding control are determined by the accumulation of the user's preceding selections in other controls. You can think of the sequence of options as a decision tree—each selection leads the user down a different branch. You can have any number of dependent drop-downs in your design.

Multi-tiered dependent drop-downs streamline data entry and help ensure that users make valid selections.

Successful implementation of multi-tiered dependent drop-downs requires careful planning and design. To illustrate this, we will step through the process of building a sample application.

Note

The rules for creating multi-tiered dependent drop-down are the same as those defined in the previous topic, "Creating Dependent Drop-Down Lists," on page 189 with one important addition: The filter value for each dependent drop-down must uniquely identify the combination of selections that preceded it. We will explain this concept fully in the following example.

The Scenario

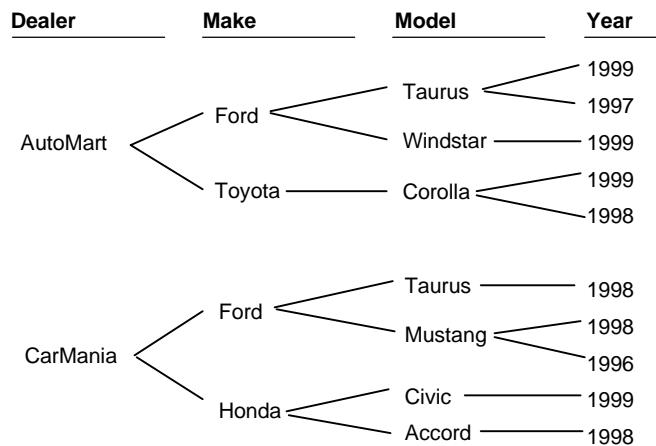
Say that you are a car salesperson and want to build an order form that enables you to pick a car by Dealer, Make, Model, and Year. Your initial application might look something like this:

The image shows two windows side-by-side. On the left is a screenshot of an application form titled 'Order Form'. It contains five text input fields labeled 'CustName', 'OrderNo', 'Dealer', 'Make', and 'Model'. At the bottom are standard Windows menu items: 'Record', 'Edit', 'Option', and icons for 'File', 'Edit', 'Search', and 'Help'. The time '8:25p' is displayed in the top right corner of the application window. On the right is a 'Create Table' dialog box. In the 'Name:' field, 'Orders' is entered. In the 'Columns:' section, there is a list box containing six entries: 'CustName [Text]', 'OrderNo [Integer]', 'Dealer [Text]', 'Make [Text]', 'Model [Text]', and 'Year [Text]'. The 'CustName [Text]' entry is currently selected. On the right side of the dialog are buttons for 'OK', 'Cancel', and several movement and edit buttons: 'Move Up', 'Move Down', 'Add Before', 'Add After', 'Edit', and 'Delete'. Below these are 'Add Index' and 'Remove Index' buttons. The status bar at the bottom of the dialog box reads: 'Definition of Application Table: Orders (Table will store records entered via OrderForm.vce)'.

Application Form: OrderForm.vce

Definition of Application Table: Orders
(Table will store records entered via OrderForm.vce)

Your form would be much more valuable if it allowed you to pick only cars that are actually in the dealers' inventories. In fact, you know that the range of options is as follows:



You decide to modify your form so that the controls for Dealer, Make, Model, and Year are intelligent drop-downs that display only the options available.

Design the Support Table

You know from the previous topic that your first step in creating dependent drop-downs is to create a support table that stores one row for each possible combination of options. Let's call the sample support table, "Inventory." Before you build the table on the computer, it's best to sketch it out on paper. Your initial plan might look like this. Note that every possible branch in the tree is captured.

Initial Inventory Table Worksheet

Dealer	Make	Model	Year
AutoMart	Ford	Taurus	1999
AutoMart	Ford	Taurus	1997
AutoMart	Ford	Windstar	1999
AutoMart	Toyota	Corolla	1999
AutoMart	Toyota	Corolla	1998
CarMania	Ford	Taurus	1998
CarMania	Ford	Mustang	1998
CarMania	Ford	Mustang	1996
CarMania	Honda	Civic	1999
CarMania	Honda	Accord	1998

You're not done, though. You need to add another level of information in order for the program to accurately filter the options beyond the Make of the car.

In order for multi-tier drop-downs to work, the options in each drop-down must be filtered by a value that indicates the combination of the user's selections in the previous drop-downs. For example, Make is filtered by Dealer; Model is filtered by Dealer and Make; and Year by Dealer, Make, and Model.

To create the filter values, do the following:

- 1 In each column except the last, assign a different id number to each unique entry in the column.

Dealer	Make	Model	Year
AutoMart 01	Ford 01	Taurus 01	1999
AutoMart 01	Ford 01	Taurus 01	1997
AutoMart 01	Ford 01	Windstar 02	1999
AutoMart 01	Toyota 02	Corolla 03	1999
AutoMart 01	Toyota 02	Corolla 03	1998
CarMania 02	Ford 01	Taurus 01	1998
CarMania 02	Ford 01	Mustang 04	1998
CarMania 02	Ford 01	Mustang 04	1996
CarMania 02	Honda 03	Civic 05	1999
CarMania 02	Honda 03	Accord 06	1998

- 2 Create new **integer** columns to hold the filter values; that is, insert one new column after every column in your worksheet, *except* after the first and last columns.

Dealer	Make		Model		Year
AutoMart 01	Ford 01		Taurus 01		1999
AutoMart 01	Ford 01		Taurus 01		1997
AutoMart 01	Ford 01		Windstar 02		1999
AutoMart 01	Toyota 02		Corolla 03		1999
AutoMart 01	Toyota 02		Corolla 03		1998
CarMania 02	Ford 01		Taurus 01		1998
CarMania 02	Ford 01		Mustang 04		1998
CarMania 02	Ford 01		Mustang 04		1996
CarMania 02	Honda 03		Civic 05		1999
CarMania 02	Honda 03		Accord 06		1998

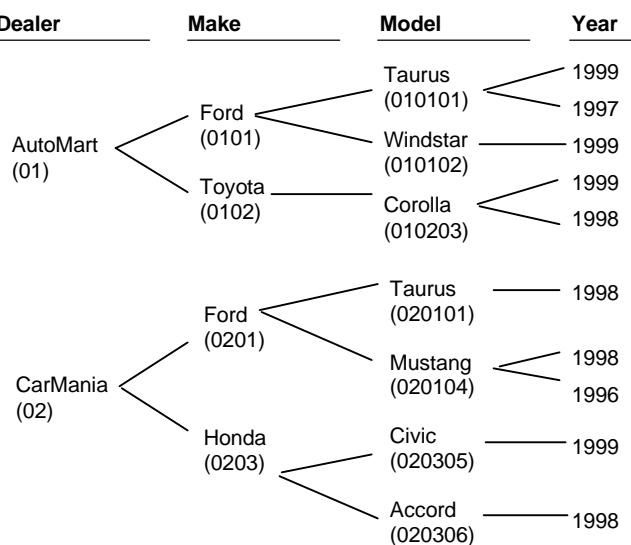
- 3 Name the new columns. We've chosen the names DealerMake and DealerMakeModel to convey the concept that the filter values identify the combination of the preceding selections, but you can use any name of your choosing.

Dealer	Make	DealerMake	Model	DealerMakeModel	Year
AutoMart 01	Ford 01		Taurus 01		1999
AutoMart 01	Ford 01		Taurus 01		1997
AutoMart 01	Ford 01		Windstar 02		1999
AutoMart 01	Toyota 02		Corolla 03		1999
AutoMart 01	Toyota 02		Corolla 03		1998
CarMania 02	Ford 01		Taurus 01		1998
CarMania 02	Ford 01		Mustang 04		1998
CarMania 02	Ford 01		Mustang 04		1996
CarMania 02	Honda 03		Civic 05		1999
CarMania 02	Honda 03		Accord 06		1998

- 4 Now, in each empty cell, combine the id numbers in the cells to the left to create a unique code representing the combination. For example, the DealerMake code in the first row would be 0101; the DealerMakeModel code, 010101.

Dealer	Make	DealerMake	Model	DealerMakeModel	Year
AutoMart 01	Ford 01	0101	Taurus 01	010101	1999
AutoMart 01	Ford 01	0101	Taurus 01	010101	1997
AutoMart 01	Ford 01	0101	Windstar 02	010102	1999
AutoMart 01	Toyota 02	0102	Corolla 03	010203	1999
AutoMart 01	Toyota 02	0102	Corolla 03	010203	1998
CarMania 02	Ford 01	0201	Taurus 01	020101	1998
CarMania 02	Ford 01	0201	Mustang 04	020104	1998
CarMania 02	Ford 01	0201	Mustang 04	020104	1996
CarMania 02	Honda 03	0203	Civic 05	020305	1999
CarMania 02	Honda 03	0203	Accord 06	020306	1998

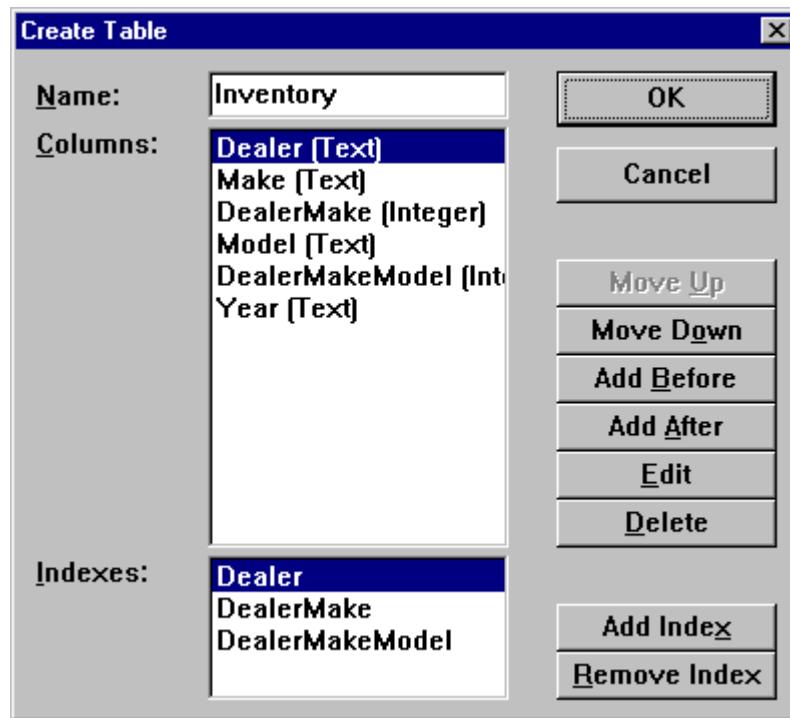
The coding scheme you choose to create the filter values is unimportant, as long as it uniquely identifies the combinations (decision paths) and can work as an unambiguous filter, as illustrated below.



Now that you have designed your scheme, you need to put it in place.

Build the Support Table on the Desktop PC

Build the support table, "Inventory," as illustrated below. Remember, the columns for Dealer, DealerMake, and DealerMakeModel store filter values, so you must index them. Remember also that the columns DealerMake and DealerMakeModel store numeric codes, so they must be **integer** columns.



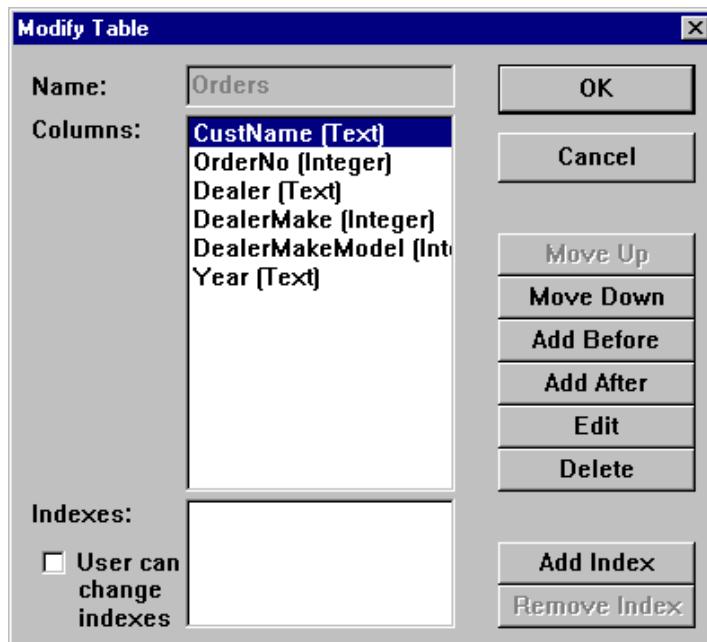
Download the Inventory support table to the CE device and create a form for it. Enter the records from your worksheet. Note: Don't enter the id numbers that you sketched in the Dealer, Make, or Model columns. **Do** enter the codes you created in the filter columns (DealerMake and DealerMakeModel), as illustrated below.

Inventory Table

Dealer	Make	DealerMake	Model	DealerMakeModel	Year
AutoMart	Ford	0101	Taurus	010101	1999
AutoMart	Ford	0101	Taurus	010101	1997
AutoMart	Ford	0101	Windstar	010102	1999
AutoMart	Toyota	0102	Corolla	010203	1999
AutoMart	Toyota	0102	Corolla	010203	1998
CarMania	Ford	0201	Taurus	020101	1998
CarMania	Ford	0201	Mustang	020104	1998
CarMania	Ford	0201	Mustang	020104	1996
CarMania	Honda	0203	Civic	020305	1999
CarMania	Honda	0203	Accord	020306	1998

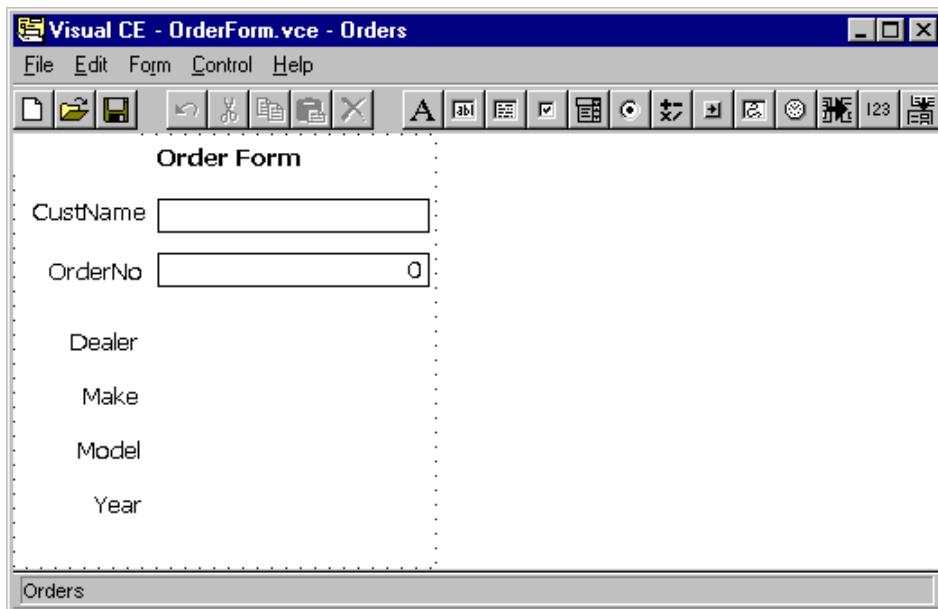
Modify the Application Table

In order for dependent drop-downs to work, there must be a matching column in the application table to store the filter values. It must have the same column name and data type as the corresponding column in the support table. That means you must add DealerMake and DealerMakeModel to the design of your Orders table, as illustrated below: Since the old Make and Model text column no longer serve any purpose in the application table, delete them (they are replaced by the more meaningful DealerMake and DealerMakeModel, respectively).



Specify the Drop-Down Controls in the Application Form

Remember, our goal is to create intelligent drop-downs. If your original form design for the application used simple edit controls for these fields, delete them from the form.

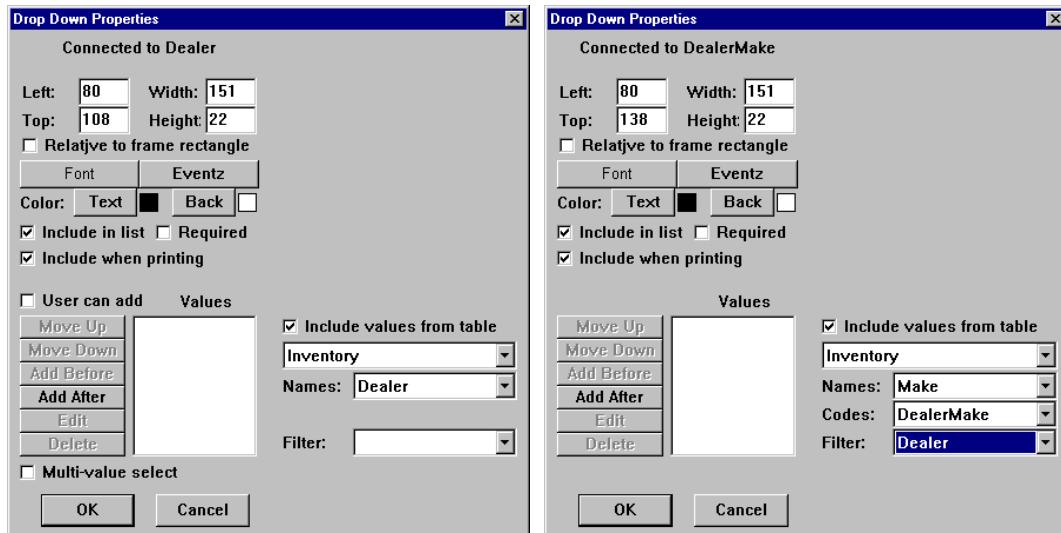


Before recreating the controls as drop-downs, you may find it helpful to plan your strategy on paper first, using a chart similar to the one shown below. Keep in mind that this is not a table to enter into the computer – it's just a chart you can refer to when specifying the drop-down control properties for the application form. Each line in this chart represents one drop-down control you will create in the form.

Control Label	Connected to	Include values from table	Names	Codes	Filter
Dealer	Dealer	Inventory	Dealer		
Make	DealerMake	Inventory	Make	DealerMake	Dealer
Model	DealerMakeModel	Inventory	Model	DealerMakeModel	DealerMake
Year	Year	Inventory	Year		DealerMakeModel

The controls are listed in the order you create them (Dealer, Make, Model, Year). Note that a control's Filter column is the same as the previously created control's "Connected to" column.

Create the drop-down controls as you would normally, using the specifications listed in the chart for the control properties. The two screen shots below illustrate how the specifications in the first two lines in the chart are applied to the properties for the first two controls.



Download your new version of the application form to the CE device, and test it to make sure that the drop-downs display the correct options. The illustration below shows how the finished sample order form might look on the CE device.

Creating Navigational Drop-Down Lists

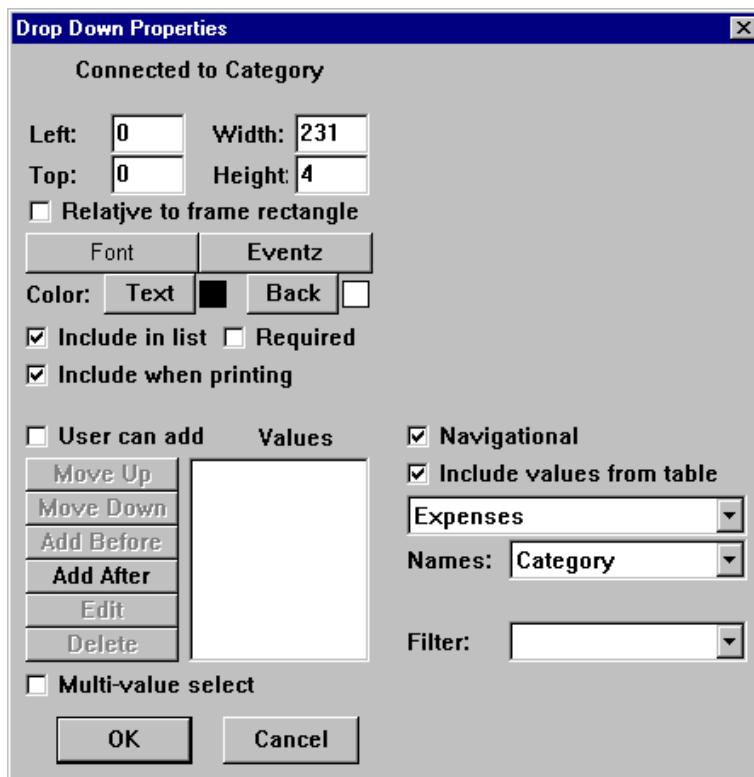
The previous topics explained how to create drop-down lists that enable users to select a value to store in a new or modified record. This topic explains how to create drop-down lists that enable users to quickly find existing records by a desired value.

Navigational drop-down lists are dynamically composed of values stored in a column in the application table. When the user selects a value from the list, the form displays the first record in the table that has the selected value.

Navigational drop-downs can be associated with any text, memo, or numeric column that has been indexed. They cannot be associated with date/time, yes/no, or picture columns; nor with global variables.

To create a navigational drop-down list:

- 1 Select **Control–Drop Down**, or click the **Drop Down** button , or right-click where you want the control to appear and select **Drop Down**.
Visual CE displays the Select Columns dialog box.
- 2 Select the table column that contains the values you want users to search by. Remember, this must be an indexed text, memo, numeric, or date/time column. Click **OK**.
- 3 In the Drop-Down Properties dialog, select the **Navigational** checkbox.
Visual CE then displays additional fields in the lower right corner of the Drop Down Properties dialog.



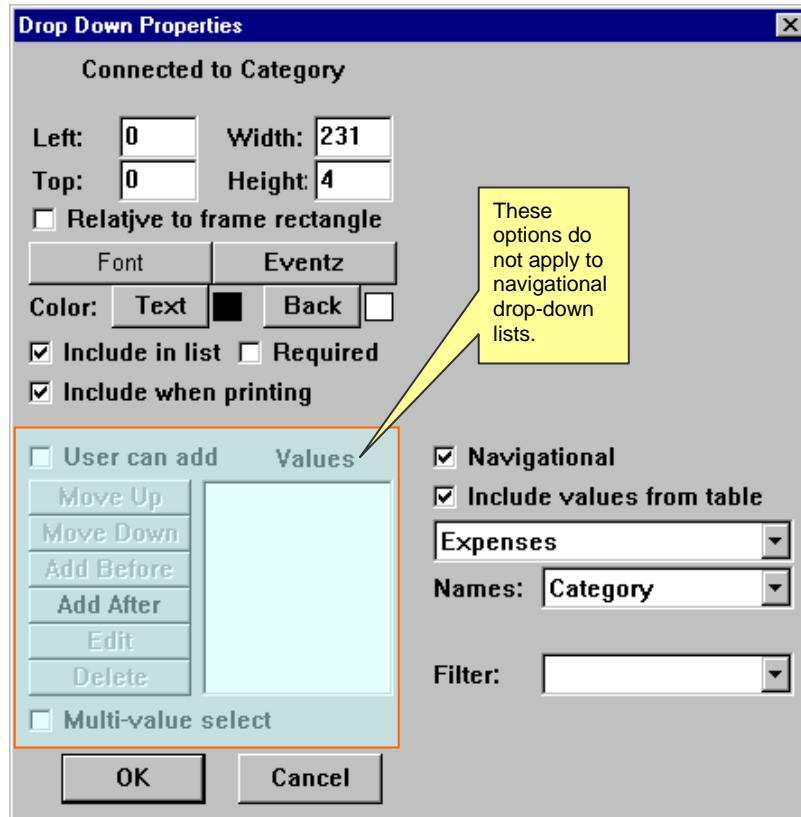
- 4 Complete the additional fields:
 - In the first two new fields (directly under "Include values from table"), leave the default values in place. The first field displays the name of the table that the form is built on; the second ("Names") displays the name of the table column you selected in Step 2.

- **Filter:** Leave blank, unless you want the options displayed in the list filtered by the user's selection for another form control. Refer to the topic, "Creating Dependent Drop-Down Lists," on page 189 for details.

- 5 Complete the options in the upper left corner of the dialog box to position and style the appearance of the control as desired.

Note

The options in the lower-left corner of the Drop Down Properties dialog box do not apply to navigational drop-down lists. If you select one of these options, Visual CE hides the options that do apply. To redisplay the relevant options, simply unselect the option that does not apply.



- 6 Click **OK** on the Drop Down Properties dialog box.
- 7 Drag the drop-down list to the correct location, and resize it if necessary.

Creating Calculated Fields

You can incorporate arithmetic or text operations—and even advanced functions—right into your form. When a user creates or displays a record, Visual CE automatically applies the embedded formula (called an **expression**) to the current record's values and displays the results in the calculated field. You specify whether Visual CE performs the calculations automatically whenever the record is displayed, on command, or just once when the record is created. You also specify whether or not Visual CE saves the results in the table or a global variable. The results can be numeric, text, or date/time.

For example, the sample Expense Tracker application that you worked with in the Quick Tours included a calculated field control that automatically calculated a 15% tip on the amount of the expense. It used the following expression:

Amount *15/100

Here are a few other examples of calculated field expressions:

This Expression	Returns
UnitPrice * Quantity	3.00 when the value in the Price column is 1.50 and the value in the Quantity column is 2
"First Name" & " " & "Last Name"	'Jane Smith' when the value in the First Name column is 'Jane' and the Last Name column is 'Smith'
@now + (5 * @days)	The date five days from today

About Constants, Column Names, Functions, and Operators

An expression can include any combination of constants, column names, functions, global variables, and operators:

- **Constants**

A constant is a value that does not change. Constants can be text, numeric, or date/time.

- **Text constants:** Text constants *must always* be enclosed in single quote marks; e.g., 'client'.
- **Date/Time constants:** Date/time constants *must always* be enclosed in single quote marks and use the following formats:

Date:	'yyyy-mm-dd'	'2003-02-14' for February 14, 2003
Time:	'hh:mm'	'13:41' for 1:41 p.m.
Date/time:	'yyyy-mm-dd hh:mm'	'2003-02-14 13:41' for 1:41 p.m. on February 14, 2003

- **Column Names**

If you include a reference to a column in the expression, Visual CE uses the value stored in the column for the current record. You can refer to columns that store text, numbers, yes/no, or dates/times. If the name includes blanks or special characters, you must enclose it in double quotes; e.g. "Employees First Name".

- **Functions**

Functions are pre-defined, named formulas included with Visual CE. They perform a wide variety of useful calculations. You specify the function by name, and Visual CE performs the calculation.

Functions are identified with a leading @ sign, and use this syntax:

`@functionName(parameter, parameter...)`, where *functionName* is the name recognized by Visual CE for the calculation, and *parameter* is the value to be operated on. A parameter can be any expression (including one that contains another function). Some functions require you to specify more than one parameter, in which case you must separate individual parameters with a comma.

Visual CE's most frequently used functions are described in the topic, "Using Basic Functions," on page 207. Visual CE offers other, more advanced functions, which are described in the topic, "Using Advanced Functions," on page 208.

▪ Global Variables

Global variables make it possible to store intermediate values so that they are available for further processing, without having to write them to a table. The Assign, GPS, Communicate Receive, SMS Receive, and File Read commands write values to global variables, as do many controls. You can access these values in any expression using the @var() function.

Visual CE can stores values for up to 100 different global variables at one time -- @var(0), @var(1)...@var(99). The variables are global and persistent, meaning that assigned values are available to all Visual CE forms running on the device and that they persist until overwritten by new values (even when the program is closed and the device turned off).

For more information about global variables, refer to the topic, "About Using Global Variables to Store Temporary Values," on page 153.

▪ Operators

Operators are symbols that specify the operation to perform.

▪ Mathematical operators (use with numeric arguments):

- + addition
- subtraction
- * multiplication
- / division

If you use more than one operator, Visual CE evaluates the expressions from left to right, and performs multiplication and division before addition and subtraction. You can change the precedence by inserting parentheses: Visual CE performs the operations enclosed in parentheses first. For example, the result of 500 – 300 – 100 is 100. You could insert parentheses to change the expression to 500 – (300 – 100), which returns 300 instead.

▪ Text operator (use with text arguments)

- & concatenation

It is important to match the operator to the data type of the arguments. Visual CE automatically interprets arguments on either side of a mathematical operator as numbers, and those on either side of a concatenation operator as text. Consider the following expression:

`(3 + 9) & (1 + 3)`

It returns '124', a text string.

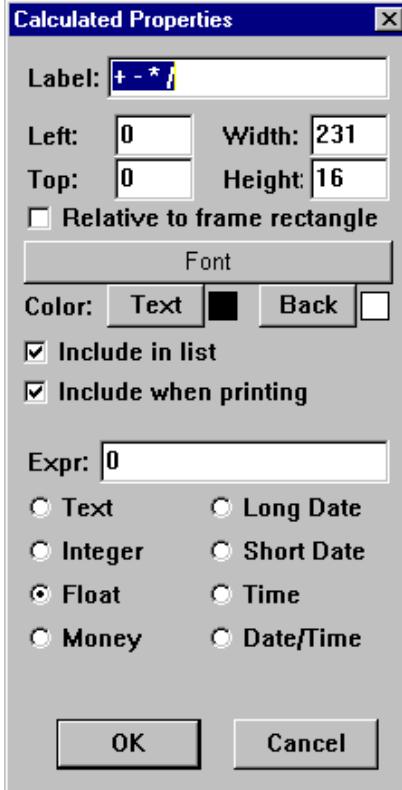
About IF-THEN-ELSE

You can also create special expressions that return a different result depending upon a comparison. These "IF-THEN-ELSE" expressions are described in the topic, "Using If-Then-Else in a Calculated Field," on page 206.

To create a calculated field:

- 1 Select **Control–Calculated**, or click the **Calculated** button , or right-click where you want the control to appear and select **Calculated**.
- 2 Visual CE asks if you want to store the results of the calculation.
 - If you click **Yes**, Visual CE asks you to select the table column or global variable where the results will be stored.
 - If you click **No**, Visual CE will display the results of the calculation in the application's form and record list display, but will not store them.

Visual CE displays the Calculated Properties dialog box.



The Calculated Properties dialog box contains the following fields:

Field	Description
Connected to [table column or global variable]	The table column or global variable to which the control relates – cannot be changed. (If you did not specify a column or variable to store the results, the Label field appears in the Calculated Properties dialog box instead.)
Label	The text used for the column heading in the record list display. "+-*" is the default. (If you associated the control with a table column or global variable, Visual CE uses the column name or variable name for the heading and this option is not available on the Calculated Properties dialog.)
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.

Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control is positioned relative to the frame; i.e., the device screen. The control remains in the same place on the screen, even if the user scrolls the display or moves to another region of the form. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Font	The font, size and style of the type in the calculated field. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.
Text	The color of the type in the calculated field (ignored if your CE device's display is black and white).
Back	The color of the background in the calculated field (ignored if your CE device's display is black and white).
Include in list	If checked, this data appears in list display mode. If unchecked, it's excluded from the list, but still appears in the single record display.
Include when printing	If checked, this data appears in the printed version of the record. If unchecked, it's excluded.
[Result options]	Specify the display format for the results. (If you selected a column or variable in which to store the results, only formats suitable for that datatype are available.) If you check Money, the resulting value is displayed with two decimal places, e.g., 15.28 Note: The display format associated with each date/time option is determined by the Regional Settings on the computer that runs the form. Refer to the topic, "Edit Box Date/Time Formats," on page 174 for details.
Expr	The expression, including constants, functions, and any column name whose value is to be used in calculating the result. Refer to the topics, "Using Date/Time Values in a Calculated Field Expression" on page 205; "Using If-Then-Else in a Calculated Field" on page 206; "Using Basic Functions" on page 207; and "Using Advanced Functions" on page 208 for more information.) Tip: Double-click the Expr field to display a larger text area in which to enter the expression.
Recalculate	If checked, the results are recalculated whenever the record is displayed or the user selects Option-Recalc (see page 279). If unchecked, the result is "write-once;" i.e., it is calculated at the time the record is first created and never updated.

- 3 Enter the expression in the Expr field of the dialog box, make any other changes required in the Calculated Properties dialog box, then click the **OK** button.
- 4 Drag the calculated field to the correct location on the form, and resize it if necessary.
- 5 If necessary, create a label to identify the calculated field.
- 6 If you want Visual CE to perform the calculation automatically, turn on Auto Recalc by selecting **Form-Auto Recalc** so that a checkmark appears next to the option. (Refer to the topic, "Enabling the Auto Recalc Feature," on page 279 for details.)

Using Date/Time Values in a Calculated Field Expression

You can use values in Date/Time columns or variables as arguments in a calculated field expression. This is especially useful for determining duration (the number of minutes that elapsed between two date/times) or to find a date that is some number of days before or after a given date.

For example, if your billing rate were \$50.00 per hour, the following expression would calculate the charge for the time spent on a job, assuming that you stored the start- and end-times for jobs in columns named START_TIME and END_TIME:

$$((\text{END_TIME} - \text{START_TIME}) / 60) * 50$$

Note: If you use this example, be sure to click the "Money" checkbox when specifying the calculated field control's properties so that Visual CE will display the results with two decimal places.

When building an expression that refers to date/time values, you can use the same kinds of references that you use for other data types; that is, date/time constants, the names of table columns or variables that contain date/time values, and functions that return date/time values (such as @now).

- **Date/Time Constants and Column Names:** For instructions on formatting date/time constants and column names, refer to the section, "About Constants, Column Names, Functions, and Operators," in the topic, "Creating Calculated Fields," on page 201.
- **Functions That Return Date/Time Values:** Visual CE provides many functions for working with dates/times. Three of the most useful are @now, @today, and @days:
 - To use the current date/time in an expression, use the function **@now**.
 - To use the current date in an expression, use the function **@today**.
 - To find a date some number of days before or after a given date, use the function **@days**.

For details about @now, @today and @days, refer to the topic, "Using Basic Functions in a Calculated Field Expression," on page 207. For information about all other date/time functions, refer to the topic, "Using Advanced Functions in a Calculated Field," on page 208.

Note

The internal representation for date/time values in Visual CE is the number of minutes from an arbitrary point in time, established as part of the software design. You can therefore add, multiply, and divide these values, but it is really only meaningful to subtract one value from another to determine the duration.

Using If-Then-Else in a Calculated Field

You can create a calculated field control in which Visual CE applies a different calculation depending upon whether or not the current record meets a specified condition. This type of calculated field expression is called "If-Then-Else."

For instance, say you want to apply a 5% discount to sales orders over \$500. You could use the following If-Then-Else expression to compute the discount:

```
IF SUBTOTAL > 500 THEN SUBTOTAL * 5/100 ELSE 0
```

Syntax

Create If-Then-Else expressions using the following syntax:

```
If <boolean-expression> then <expression1> else <expression2>
```

Where:

- <boolean-expression> is the condition that Visual CE uses to evaluate the current record. This expression compares the result of two numeric expressions using any of the following relational operators:

- < less than
- <= less than or equal to
- = equals
- > greater than
- >= greater than or equal to
- <> not equal to

You can use the logical connectors AND, OR, and NOT to join nested boolean expressions.

- <expression1> Specify the expression that Visual CE applies when the boolean expression is true for the current record.
- <expression2> Specify the expression that Visual CE applies when the boolean expression is false for the current record.

Tip - Testing for empty record fields:

You can test whether or not the current record contains a null value (is empty) by comparing the table column to an empty string. This works for columns of any data type, including pictures.

The syntax of the boolean expression must be:

```
<column_name> = ''
```

That is, the name of the table column, equals sign, then two single quote marks without an intervening space.

Example:

```
picture = ''
```

Using Basic Functions in a Calculated Field Expression

Visual CE provides seven basic functions that you can use in calculated field expressions: @now, @today, @days, @count, @sum(columnName), @prev(columnName), and @devicename. In addition to these basic functions which are described below, Visual CE offers many advanced functions which are described in the topic, "Using Advanced Functions in a Calculated Field," on page 208.

Note

If the user applies a filter to the records in the application, the functions @count and @sum(columnName) automatically find the correct value for the records that satisfy the filter. If the Filter option is turned off in the application, the functions find values for all records in the table. Refer to the topic, "Filtering the Record Display," on page 395.

Function	Returns...
@now	Current date/time. Useful for calculating the elapsed time between a date/time of interest and now.
@today	Current date, without time. Useful for displaying the current date or calculating the number of elapsed days between a date of interest and today.
@days	Number of minutes in a day (24 * 60). Useful for finding a date some number of days before or after a given date.
@count	Number of records currently stored in the entire table or a filtered list.
@sum(columnName)	Sum of all values stored in the specified column (applies to either the entire table or a filtered list).
@prev(columnName)	Value stored in the specified column for the previous record. This function can be used only in calculated controls for which you have turned off the recalculate option in the Calculated Properties dialog box.
@devicename	Name of the handheld as specified in its Control Panel. Useful for audit trails (e.g., who created the record). Note: To find name of a Pocket PC, point to Start – Settings – System – About – Device ID – Device Name . To find the name of an H/PC, go to Start – Settings – Control panel – Communications – Device Name .

Examples:

- To find the due date of a bill:

```
"Invoice Date" + (30 * @days)
```

If the invoice date stored in the record was November 18, 2002, the result would be December 18, 2002.

- To find someone's age in years (assuming that your application stores their birth date in a column named "birthday"):

```
(@today - birthday) / (365.25 * @days)
```

- To find the average expense amount in the sample Expense Tracker application (total of all values in the Amounts column divided by the total number of records currently stored):

```
@sum(Amount) / @count
```

- To find the starting balance in a checkbook application (assuming that the previous ending balance is stored in a column named EndingBalance):

```
@prev(EndingBalance)
```

- To display an audit trail for the record:

```
'This record created by ' & @devicename & ' on ' & @now
```

A sample result: This record created by CEdevice on 8/23/01 10:47:49AM

Using Advanced Functions in a Calculated Field

Visual CE offers a large collection of advanced functions that perform a variety of useful calculations. They fall into the following categories, according to the type of data on which they operate: **Text**, **Numeric**, **Date/Time**, **Global Variables**, **GPS**, and **System**. (In addition to the specialized functions described in this topic, Visual CE offers seven other functions, which are described in the topic, "Using Basic Functions in a Calculated Field," on page 207. For general rules about using functions in a calculated field expression, refer to the topic, "Creating Calculated Fields, on page 201.)

Here are a few examples of advanced functions:

This function	Returns
<code>@repeat('abc', 3)</code>	'abcabcbcabc'
<code>@sqrt(144)</code>	12
<code>@year(Employees.Birthday)</code>	1980 when the value stored in Birthday is 1/1/1980

Text Functions

Text functions manipulate text — they count, combine, search for, insert, replace, and convert character strings.

Function	Description
<code>@ascii(text)</code>	Returns the ASCII code value (integer) of the leftmost character in <i>text</i> .
<code>@char(ASCII_code_value)</code>	Returns the character that has the specified ASCII code value. The value must be between 0 and 255, otherwise, the return value is undefined.
<code>@insert(text, start, length, insert_text)</code>	Returns a character string, where number of characters specified by <i>length</i> have been deleted from <i>text</i> , beginning with the character position indicated by the value of <i>start</i> , and where <i>insert_text</i> has been inserted into <i>text</i> , beginning at <i>start</i> . For example, <code>@insert('ABCDF', 3, 2, 'XYZ')</code> returns 'ABXYZF'
<code>@lcase(text)</code>	Returns <i>text</i> as all lowercase.
<code>@left(text, count)</code>	Returns the leftmost <i>count</i> characters in <i>text</i> .
<code>@length(text)</code>	Returns the number of characters in <i>text</i> , excluding trailing blanks.
<code>@locate(search_string, text, start)</code>	Returns the starting position of the first occurrence of <i>search_string</i> within <i>text</i> . The search begins with the character position indicated by the value of <i>start</i> (1 indicates the first character position). If the search string is not found, the function returns 0.
<code>@ltrim(text)</code>	Returns the characters in <i>text</i> , without leading blanks.
<code>@repeat(text, count)</code>	Returns the characters in <i>text</i> , repeated <i>count</i> times.

<code>@replace(text, search_string, replacement_text)</code>	Search <i>text</i> for <i>search_string</i> , and replace with <i>replacement_text</i> in all occurrences.
<code>@right(text, count)</code>	Returns the rightmost <i>count</i> characters in <i>text</i> .
<code>@rtrim(text)</code>	Returns the characters in <i>text</i> , without trailing blanks.
<code>@space(count)</code>	Returns a character string consisting of <i>count</i> blank spaces.
<code>@substring(text, start, length)</code>	Returns characters in <i>text</i> , beginning at the character position specified by <i>start</i> , through the number of characters specified by <i>length</i> .
<code>@ucase(text)</code>	Returns <i>text</i> as all uppercase.

Numeric Functions

Number functions manipulate numeric data.

Function	Description
<code>@abs(number)</code>	Returns the absolute value of <i>number</i> .
<code>@acos(number)</code>	Returns the arccosine of <i>number</i> as an angle, expressed in radians.
<code>@asin(number)</code>	Returns the arcsine of <i>number</i> as an angle, expressed in radians.
<code>@atan(number)</code>	Returns the arctangent of <i>number</i> as an angle, expressed in radians.
<code>@atan2(number1, number2)</code>	Returns the arctangent of the x and y coordinates, specified by <i>number1</i> and <i>number2</i> , respectively, as an angle, expressed in radians.
<code>@ceiling(number)</code>	Returns the smallest integer greater than or equal to <i>number</i> .
<code>@cos(number)</code>	Returns the cosine of <i>number</i> , where <i>number</i> is an angle expressed in radians.
<code>@cot(number)</code>	Returns the cotangent of <i>number</i> , where <i>number</i> is an angle expressed in radians.
<code>@degrees(number)</code>	Returns the number of degrees converted from <i>number</i> radians. This is especially useful for converting the results of other functions to degrees.
<code>@exp(number)</code>	Returns the exponential value of <i>number</i> .
<code>@floor(number)</code>	Returns the largest integer less than or equal to <i>number</i> .
<code>@log(number)</code>	Returns the natural logarithm of <i>number</i> .
<code>@log10(number)</code>	Returns the base 10 logarithm of <i>number</i> .

<code>@mod(number, divisor)</code>	Returns the remainder (modulus) of <i>number</i> divided by <i>divisor</i> .
<code>@pi</code>	Returns the constant value of pi as a floating-point value.
<code>@power(number1, number2)</code>	Returns the value of <i>number1</i> to the power of <i>number2</i> .
<code>@radians(number)</code>	Returns the number of radians converted from <i>number</i> degrees.
<code>@rand</code>	Returns a random floating-point value.
<code>@round(number, precision)</code>	Returns <i>number</i> rounded to the number of decimal places in <i>precision</i> . If <i>precision</i> is negative, <i>number</i> is rounded to <code>@abs(precision)</code> places to the left of the decimal point; e.g., -1 rounds <i>number</i> to the nearest ten, -2 to the nearest 100, and so on.
<code>@sign(number)</code>	Returns an indicator of the sign of <i>number</i> : -1 when <i>number</i> is negative; 0 when <i>number</i> equals zero; and 1 when <i>number</i> is positive.
<code>@sin(number)</code>	Returns the sine of <i>number</i> , where <i>number</i> is an angle expressed in radians.
<code>@sqrt(number)</code>	Returns the square root of <i>number</i> .
<code>@tan(number)</code>	Returns the tangent of <i>number</i> , where <i>number</i> is an angle expressed in radians.
<code>@truncate(number, precision)</code>	Returns <i>number</i> truncated to the number of decimal places in <i>precision</i> . If <i>precision</i> is negative, <i>number</i> is truncated to <code>@abs(precision)</code> places left of the decimal point; for example, <code>@truncate(12654.6, -3)</code> returns 12000

Date/Time Functions

Date/Time functions manipulate and calculate dates and times.

<u>Function</u>	<u>Description</u>
<code>@dayname(datetime)</code>	Returns the name of the day for <i>datetime</i> . For example, <code>@dayname('2002-02-14')</code> returns Thursday.
<code>@dayofmonth(datetime)</code>	Returns the day of the month for <i>datetime</i> as a integer in the range 1-31.
<code>@dayofweek(datetime)</code>	Returns the day of the week for <i>datetime</i> as an integer in the range 1-7, where 1 represents Sunday.
<code>@dayofyear(datetime)</code>	Returns the day of the year for <i>datetime</i> as an integer in the range 1-366 For example, <code>@dayofyear('2005-02-02')</code> returns 33.
<code>@hour(datetime)</code>	Returns the hour in <i>datetime</i> as an integer in the range 0-23.
<code>@minute(datetime)</code>	Returns the minute in <i>datetime</i> as an integer in the range 0-59.
<code>@month(datetime)</code>	Returns the month in <i>datetime</i> as an integer in the range 1-12.
<code>@monthname(datetime)</code>	Returns the name of the month for <i>datetime</i> . For example, <code>@monthname('2002-02-14')</code> returns February.
<code>@quarter(datetime)</code>	Returns the quarter for <i>datetime</i> as an integer value in the range 1-4, where 1 represents January 1 through March 31.
<code>@second(datetime)</code>	Returns the second in <i>datetime</i> as an integer in the range 0-59.
<code>@year(datetime)</code>	Returns the year in <i>datetime</i> .

Global Variable Function

<u>Function</u>	<u>Description</u>
<code>@var(n)</code>	<p>Returns the value that has been assigned to the variable. The value can be a text string, date/time, or number; but not a picture.</p> <p><i>n</i> is the identifier of the variable, which must be either a constant between 0 and 99 or an expression that returns a value between 0 and 99. You can use up to 100 global variables: <code>@var(0), @var(1)....@var(99)</code>. A value assigned to a variable is available for all Visual CE forms on the device, and persists until a new value is assigned (even after the device is turned off).</p> <p>For more information, refer to the topic, "About Using Global Variables to Store Temporary Values," on page 153.</p>

GPS Functions

GPS functions return specified data from GPS (Global Positioning System) sentences. These functions are typically applied to text strings returned by a Communicate | GPS command button (see page 230).

To see an example of how these functions can be used in a macro to capture GPS data, look at the SYWARE-supplied sample application, GPS.VCE, which was provided with your installation package. (For instructions on installing the samples, please refer to the topic, "Installing Visual CE," on page 15). The samples work only on the CE device, not the desktop PC.

Function	Description
<code>@alt(gpssentence)</code>	Returns the value for altitude from the <i>gpssentence*</i> .
<code>@diffgps(gpssentence)</code>	Returns the differential GPS (RTCM SC-104) data age, the number of seconds since last valid RTCM transmission (null if non-DGPS), from <i>gpssentence*</i> .
<code>@diffref(gpssentence)</code>	Returns the differential reference station id (1 to 1023; null if non-DGPS) from the <i>gpssentence*</i> .
<code>@fixtype(gpssentence)</code>	Returns the GPS quality indication from the <i>gpssentence*</i> : 0 = fix not available 1 = Non-differential GPS fix available 2 = Differential GPS (DGPS) fix available
<code>@geoheight(gpssentence)</code>	Returns the geoidal height (-9999.9 to 9999.9 meters) from <i>gpssentence*</i> .
<code>@hdop(gpssentence)</code>	Returns the horizontal dilution of precision (0.5 to 99.9) from <i>gpssentence*</i> .
<code>@lat(gpssentence)</code>	Returns the value for latitude from <i>gpssentence*</i> in decimal degrees.
<code>@long(gpssentence)</code>	Returns the value for longitude from the <i>gpssentence*</i> in decimal degrees.
<code>@sat(gpssentence)</code>	Returns the number of satellites in use (0 to 12) from <i>gpssentence*</i> .

**gpssentence* is either the name of a table column that holds the GPS data or a global variable that has been assigned GPS data. For example, `@alt(gps)` or `@alt(@var(0))`

System Functions

System functions return values indicating the status of some aspect of the system.

Function	Description
@backupbattery	Returns an integer that indicates how much charge the backup battery has (0 to 100 percent).
@battery	Returns an integer that indicates how much charge the battery has (0 to 100 percent).
@currenttab	Returns index of currently selected tab (0 for first tab, 1 for second tab, 2 for third tab, etc.); returns -1 if there are no tabs on the form.
@endoffile(<i>file#</i>)	Returns a value indicating whether or not the end of the file identified by <i>file#</i> has been reached: 0 for No, 1 for Yes.
@fileexists(<i>file_pathname</i>)	Searches CE device for <i>file_pathname</i> . Returns non-zero value if file is found; returns 0 if file is not found.
@landscape	Returns a value indicating the current orientation of the device window: 0 for No (not in landscape therefore in portrait), 1 for Yes (in landscape), or 2 for square.
@listview	Returns a value indicating which display mode the Visual CE application is in: 0 for Record view, 1 for List view.
@serialnumber	Returns the serial number of the CE device. If not supported by the device, returns an empty string.

Creating Command Buttons

A command button is a button on a form that users click to execute a task. For example, you can create a command button that moves the display to a pre-defined region of the form (as illustrated in Quick Tour I, Lesson 6), duplicates the current record, or launches a macro or another program.

All of the commands in the Visual CE application window menu bar can be assigned to a command button. (This is useful if you want to hide the standard menu bar and restrict user access to just a customized subset of the menu commands.) In addition, Visual CE offers a large library of special tasks that you can assign to command buttons. A list of these tasks, grouped by category, is provided at the end of this topic.

If desired, you can display an image on the button's face—with or without a text label on top of it. The image must be a .bmp or .2bp file that is stored in the file system on the CE device.

General instructions for creating command buttons are provided below. Some buttons require you to specify additional options—they are covered in following topics.

About Button Macros

In previous versions of Visual CE, macro programs were created by assembling a sequence of command buttons – one button for each step in the macro – and hiding them outside the visible area of the form. While this approach is still possible, the Macro Builder enables you to compose steps more directly by organizing commands in a scripting window. (Refer to the section, "Macros and Events," beginning on page 302.) Some of the Command button options that are related to the earlier method of macro programming, such as the Continue flag, are still available for backward compatibility, but their use is discouraged.

To create a command button:

- 1 Select **Control–Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.

Visual CE displays the Command Button Properties dialog box. Complete the first 11 properties as described below.

Field	Description
Label	The label to appear on the button.
Left (Position and size)	The initial horizontal position of the left side of the button.
Top (Position and size)	The initial vertical position of the top of the button.
Width	The initial width of the button.
Height	The initial height of the button.
Relative to frame rectangle	<p>If checked, the command button remains in the same place on the device screen, even if the user scrolls the display or moves to another region of the form. The button is always visible in the same place on the screen. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details.</p> <p>If unchecked, the command button maintains its position relative to the form, so that it moves as the user scrolls or moves the display. The button can disappear from view.</p>

Font	The font, size and style of the type used as the label on the button. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.
Image	Optional. The path and filename of an image to be displayed on the face of the button. The image must be a .bmp or .2bp stored in the file system on the CE device. Tip: You can gang together a number of buttons on a form to create an apparently single image with hotspots.
Text	The color of the type that appears on the button.
Back	The color of the button.
Continue	<i>For creating a macro program with buttons; this option is available to maintain backward compatibility with previous versions of Visual CE, but its use is discouraged.</i> If checked, Visual CE automatically executes the next button in the Control Order. This option is only available when the next control in the Control Order is a button. You can also set the Continue option using Form-Control Order ; refer to the topic, "Specifying Control Order," on page 272.

- 2 From the command drop-down list, select the action that Visual CE will perform when the user clicks your command button. A brief description of each command in the list is provided in the charts below.

Depending upon your selection, Visual CE may display additional options at the bottom of the Command button Properties dialog box. The charts below direct you to instructions for completing these options.

Note

The following charts list the commands you are likely to make available to users as buttons on the form. Other commands are intended primarily as steps in a macro. For a complete list of *all* commands available in Visual CE, refer to page 328.

- 3 Make any necessary changes to the additional command button properties, then click **OK**.
 4 Drag the command button to the correct position, then resize it if necessary.

Command Buttons, Organized by Category

Visual CE Application Menu Commands (and Duplicate Record)

For additional tips about incorporating these command buttons in your form, see the topic, "Creating Command Buttons as Shortcuts for Menu Picks," on page 219. For details about what individual commands do, refer to the section, "Using Visual CE Applications," beginning on page 378.

Command	Does this
Record First	Displays the first record
Record Previous	Displays the previous record
Record Next	Displays the next record
Record Last	Displays the last record
Record Search	Locates a record based on the current sort/search order. Search can be either for exact match or closest (if no match, next greatest). Control can prompt user for search value, or form designer can provide via global variable. See page 220.
Record Print	Prints the current record
Record Save	Saves the current record
Record Insert	Creates a new record
Record Duplicate	Duplicates the current record (Note: this command is not available in the application menu bar)
Record Delete	Deletes the current record
Record Close	Closes the application
Edit Undo	Undoes last record edit
Edit IR Beam	Beams current record to another CE device
Edit IR Beam All	Beams all records in current table or filtered list to another CE device
Edit Receive IR Beam	Receives records beamed from another CE device
Option List	Toggles the display between list / single record mode
Option Filter Off	Displays all records in table
Option Filter Custom	Opens the Define Filter dialog box so that users can filter the record display by desired criteria
Option Filter Predefined	Runs a pre-defined filter.
Option Import	Imports contents of an ASCII text or XML file into the current table. File can be identified by a file pathname stored in a global variable or table column. See page 221.
Option Export	Exports records from the current table or filtered list to an ASCII text or XML file. File can be identified by a file pathname stored in a global variable or table column. See page 221.
Option Clear	Deletes all records in the table
Option Recalc	Recalculates calculated field values in the current record
Option About	Displays information in the About box

Display		
Command	Does this	For more information ...
Move to When Pressed	Moves the screen display to a pre-defined region of the form	Creating a Command Button to Move to Another Region of the Form, page 223

Run External Programs		
Command	Does this	For more information ...
Run macro	Runs a macro	Creating a Command Button to Run a Macro, page 225
Run external	Launches another Visual CE application or other executable program and (optionally) passes a value from the current record to the second program	Creating a Command Button to Start Another Program, page 226
Report CE Report	Displays the current record in a Report CE layout	Creating a Command Button to Open Report CE, page 228

mEnable Connection Commands	
<i>These commands make it possible to control the mEnable connection using command buttons instead of the mEnable stoplight control. For details, refer to the topic, "Creating Command Buttons as mEnable Controls," on page 234.</i>	
Command	Does this
mEnable off	Disconnect from the server database (equivalent of the Red Stoplight)
mEnable synchronize	Execute full synchronization between all eligible local tables and the server database (equivalent of the Yellow Stoplight).
mEnable on	Establish an mEnable connection to the server database and synchronize individual records as they are read or written to the local table with the server database (equivalent of the Green Stoplight).

mEnable File Transfer Commands

These commands make it possible upload or download any type of file from the CE device to the mEnable Server. For details, refer to the topic, "Creating Command Buttons to Put or Get Files via mEnable," on page 235.

Command	Does this
mEnable put file	Upload file, whose pathname is identified in the current Visual CE record or global variable, from the CE device to the mEnable server via an mEnable connection. The file can be of any type.
mEnable get file	Download file, whose pathname is identified in the current Visual CE record or global variable, from the mEnable server to the CE device via an mEnable connection. The file can be of any type.

Miscellaneous Commands

Command	Does this	For more information ...
Set alarm	Sets an alarm for the current record	Creating a Command Button to Set an Alarm, page 232
Sort by	Sorts records by values in a column pre-selected by the form designer	Creating a Command Button to Sort Records, page 229
Dial	Dials a phone number stored in the current record or global variable	Creating a Command Button to Dial a Phone Number, page 231
SMS Send message	Sends an SMS instant message from the application	Creating a Command Button to Send an SMS Message, page 230

The following topics cover individual types of command buttons in detail.

Note

To maintain backward compatibility, the following command buttons are available. However, they were intended primarily as steps in a macro sequence, and using command buttons to create macros is now discouraged. Instead, use the commands as steps in a script macro created with the Macro Builder. See “Macros and Events,” page 302.

- Stop macro
- Return from macro
- Message box
- Play sound
- Assign
- Skip
- Timer
- Sleep
- Set focus
- Show/Hide control
- Communicate
- SMS Receive message
- File I/O (Open, Read, Write, and Close files)
- Start of macro (used only to name a button macro; not used in buttonless, script macros)
- Grid
- Select tab
- Macro step label

Creating Command Buttons as Shortcuts for Menu Picks

This topic explains how to create a command button that replaces a series of Visual CE application window menu picks with a single click. For example, you might put a "Save" button on your form that the user clicks instead of selecting "Record," then "Save" from the menu bar. You can also create a command button to duplicate the current record.

Shortcut buttons give you greater control over how users interact with your form by making it possible to remove the standard menu bar and provide buttons for just those commands you want to make available.

Note

An alternative way of giving users access to a customized set of commands is to modify the Menu Bar. Refer to the topic, "Customizing or Removing the Menu Bar," on page 272.

Tip

The Duplicate Record command is an especially good candidate for a shortcut button if your application will run on a P/PC. That's because the command normally requires using the Shift key, which can be awkward for P/PC users.

All of the commands in Visual CE's application window menu bar, as well as Duplicate Record, can be assigned to command buttons. For a complete list of the commands available, please refer to the chart, "Visual CE Application Menu Commands (and Duplicate Record)," provided on page 216.

Tip

You can also use any of these commands as steps in a macro program. Refer to the section, "Macros and Events," beginning on page 302 for details.

To create a shortcut button:

- 1 Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.
Visual CE displays the Command Button Properties dialog box.
- 2 Specify the first 11 properties (through "Continue") as you would for any other command button. Refer to step 1 in "Creating Command Buttons" on page 213, for details.
- 3 From the command drop-down list, select the action to occur when the user clicks your command button.
- 4 Click **OK**.
- 5 Drag the command button to the desired position, and resize it if necessary.

Creating Command Buttons to Search Records

You can create a command button that moves the display to the first record whose value in a specified column matches the search value. You can have the form prompt the user for the search value, or you can have the form get the search value from a global variable. You can also specify whether the search is for an exact match or the closest value (e.g., equal to or next greater than).

Note

The search will be performed on whichever column the table is currently sorted on.

Tip

You can also use the Search command as a step in a macro program. Refer to the section, "Macros and Events," beginning on page 302 for details.

To create a Search command button:

- 1 Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.
Visual CE displays the Command Button Properties dialog box.
- 2 Specify the first 11 properties (through "Continue") as you would for any other command button. Refer to step 1 in "Creating Command Buttons," on page 213 for details.
- 3 From the drop-down list of commands, select **Record | Search**.
- 4 The dialog displays two additional options:
 - **Exact matches only.** Select this option to find records with values that exactly match the search value. For example, if **Fred** were the search value, the result would be the first record in the table that stored the value **Fred** in the sort column. Records containing **Fred Smith** or **Freddy** would not match. If the table does not contain a record with the exact match to Fred in the sort column, the result would be the message "Record not found." Case is ignored.
If you do not select this option, the command finds the first record that begins with the search value, or the next greatest value if no exact matches exist. For example, searching for **Fred** would find **Fred Smith**, or **George Jones** if Fred did not exist and George Jones was the first record greater than Fred. Case is ignored.
 - **Arg: Optional.** Select the global variable that stores the search value. If you do not select a variable, the form will prompt the user for the search value.
- 5 Click **OK**.

Creating Command Buttons to Import/Export Text or XML Files

You can create command buttons that import the contents of a comma delimited ASCII text or XML file into the current table, or export the contents of the current table or filtered list to a comma delimited ASCII text or XML file. For either import or export, the file may be identified by a file pathname stored in a global variable or table column.

For text files, you can choose to import or export data for all columns in the table, or just those columns that are associated with controls on the form. You can chose whether or not to include the table column names.

For XML files, import or export is automatically for all table columns.

Tips

- You can also use these commands as steps in a macro program. Refer to the section, "Macros and Events," beginning on page 302 for details.
- You may want to create an edit control that enables users to pick file pathnames from the file system (see the "File Selection on Click" option in the topics, "Creating Edit Boxes" on page 167 or "Creating Note Boxes" on page 177.)

Notes about Importing

- To import an XML file, the full file pathname -- including the .xml extension -- must be specified. If the extension is not .xml, the file is assumed to be a comma delimited text file.
- If no file pathname is specified, Visual CE automatically looks in the folder that contains the Visual CE application (.vce file) for a text file with the same name as the table associated with the application. For example, if MyForm.vce uses MyTable, Visual CE will look for a text file named 'MyTable.txt' in the same folder as MyForm.vce.
- To determine the proper format for an import file of either text or XML, including field order, export the current contents of the table to the desired file type and examine the results. (To export a file, open the form on the CE device and select **Options-Export**.)
- Visual CE considers the following date/time formats valid for either text or XML files, and imports values that conform to them, as is:
 - ◊ YYYY-MM-DD
 - ◊ YYYY-MM-DD HH:MM
 - ◊ YYYY-MM-DD HH:MM:SS
 - ◊ YYYY-MM-DDTHH:MM:SS
 - ◊ Formats specified via the Windows Regional Settings.
- Pictures can be imported from an XML file; they cannot be imported from a text file.

Notes about Exporting

- To export an XML file, the full file pathname -- including the .xml extension -- must be specified. If the extension is not .xml, a comma delimited text file is created.
- If no file pathname is specified, Visual CE automatically creates a text file, assigns it the name of the table from which it was exported, and puts it in the same folder as the application. For example if the current application, MyForm.vce, uses MyTable, Visual CE will export the contents of the table to MyTable.txt and put it in the same folder as MyForm.vce.
- If you are exporting all table columns, columns are exported in the order they appear in the table.
- If the 'All table columns' export option is turned off, field values are exported in the same order as the control order on the form. (Refer to "Specifying Control Order," page 276.)

- All date/time values are automatically exported in the following format:
 - ◊ *Text:* YYYY-MM-DD HH:MM:SS
 - ◊ *XML:* YYYY-MM-DDTHH:MM:SS
- Pictures can be exported to an XML file; they cannot be exported to a text file.
- For text files, you can choose whether or not to export the table column names (i.e., headers).

Refer to the topics, “Export Table” and “Import an ASCII or XML File” on pages 414 and 415 respectively for more general information about exporting and importing records.

To create an Import or Export command button:

- 1 Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.
Visual CE displays the Command Button Properties dialog box.
- 2 Specify the first 11 properties (through "Continue") as you would for any other command button. Refer to step 1 in "Creating Command Buttons," on page 213 for details.
- 3 From the drop-down list of commands, select either **Option | Import** or **Option | Export**.
- 4 *Applicable to text files, not XML:* If you wish to suppress import/export of the table column names, select **No headers**. When exporting, choose this option if you do *not* want the table column names placed in the first line of the file. When importing, choose this option if the file contains no headers; that is, the first line of the file does not contain the column names.
- 5 If you wish to import/export data to/from all columns in the table, select the **All Table Columns** checkbox. If you do not select this option, Visual CE imports/exports data to/from only columns that are associated with a form control
- 6 *Optional.* In the drop-down list that appears, select the table column or global variable that will store the pathname of the file to import from or export to.
- 7 Click **OK**.
- 8 Drag the command button to the correct position, then resize it if necessary.

Creating Command Buttons to Move to Another Region of the Form

This type of command button is a handy control that lets users move the display to a pre-defined region of the form without having to scroll.

Tips

- In order to apply this feature, the form must be larger than the device screen. Refer to the topic, "Specifying the Form Size," on page 158.
- To help orient users to different sections of your form, you can assign a different color background to the destination region. To see how this can work, run the SYWARE-supplied sample application, Bigform.vce, on your CE device. (For instructions on installing the samples, please refer to the topic, "Installing Visual CE," on page 15). The samples work only on the CE device, not the desktop PC.
- You can also use the "Move to when pressed" command as a step in a macro program. Refer to the section, "Macros and Events," beginning on page 302 for details.
- Alternatively, you can divide your form into tabbed pages. That approach is described on page 164.

To create a command button to move to another region of the form:

- 1 Specify the region of the form that you want the display to move to when the user presses the command button (do one of the following):
 - (Desktop version of development environment) While pressing the Shift key, drag the dotted-line frame to the desired region.
 - (Handheld version of development environment) Scroll the display to the desired region.
- 2 Select **Control–Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**. Visual CE displays the Command Button Properties dialog box.
- 3 Specify the first 11 properties (through "Continue") as you would for any other command button. Refer to step 1 in "Creating Command Buttons," on page 213 for details.

Tip

The "Relative to Frame Rectangle" option is useful if you want the command button to remain visible when the user moves to another region of the form. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details.

- 4 From the drop-down list of commands, select **Move to when pressed**.

Visual CE displays additional properties associated with this type of command:

Field	Description
Left (Move to)	The horizontal position of the left side of the region to move to.
Top (Move to)	The vertical position of the top of the region to move to.
Change background color	If checked, the background of the move-to region automatically inherits the background color of the command button. For example, if the background of the command button is red, the region of the form displayed when the user clicks the command button will also be red.

- 5 Make any necessary changes to these command button properties, then click **OK**.
- 6 Drag the command button to the correct position, then resize it if necessary.
- 7 You can confirm that the command button displays the desired region. First, press the Shift key and drag the dotted-line frame (desktop PC) or scroll the display (handheld) to its original position. Then, while pressing the Shift key on your keyboard, click the command button on the form. The frame or display moves to its new position.

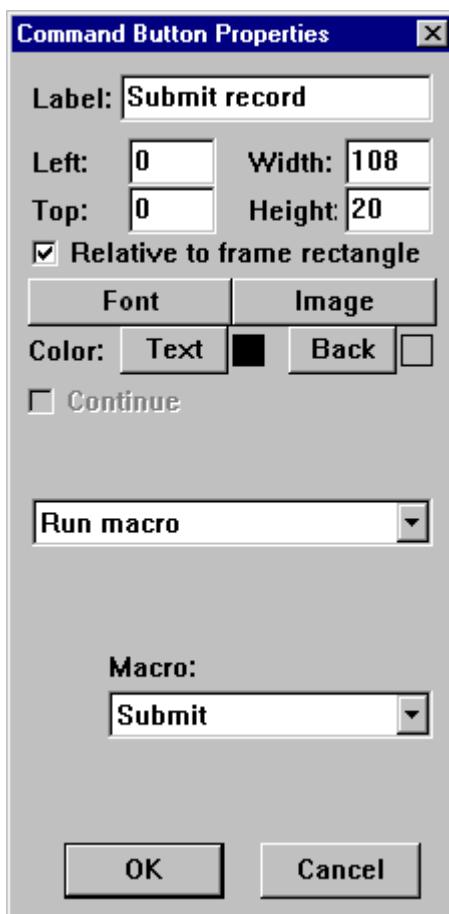
Creating Command Buttons to Run Macros

You can create a command button that users click to run a macro.

You must have already created and named the macro as part of the current Visual CE application. Refer to the section, "Macros and Events," beginning on page 302 for complete instructions on creating macros.

To create a command button to run a macro:

- 1 Select **Control–Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.
Visual CE displays the Command Button Properties dialog box.



- 2 Specify the first 11 properties as you would for any other command button. Refer to step 1 in "Creating Command Buttons" on page 213, for details.
- 3 From the command drop-down list, select **Run macro**.
- 4 For **Macro**, select the name of the macro you want to run when the user clicks the command button.
- 5 Click **OK**.
- 6 If necessary, resize and drag the button to the desired position on the form.

Creating Command Buttons to Start External Programs

You can create command buttons that start other programs when clicked. A button can launch another Visual CE application (.vce file) or any other executable program.

Tips

- You can set up the command button to not only launch an external program, but also to pass a value from the current record or global variable to the program as a command line argument. You could apply this feature to have the launched program automatically open a file whose pathname the user has stored in the current record. For example, you could create a "Picture Viewer" with a command button that launches Internet Explorer and passes the file pathname of a .bmp to the program for display.

To see how this might work, run the SYWARE-supplied sample application, Picture.vce, on your CE device. (For instructions on installing the samples, please refer to the topic, "Installing Visual CE," on page 15). The samples work only on the CE device, not the desktop PC.

When using this feature, you may want to create an edit control that enables users to pick file names from the file system rather than having to type them from scratch (see the "File Selection on Click" option in the topics, "Creating Edit Boxes" on page 167 or "Creating Note Boxes" on page 177.)

- You can also use the "Run External" command as a step in a macro program. Refer to the section, "Macros and Events," beginning on page 302 for details.

To create a command button to start another program:

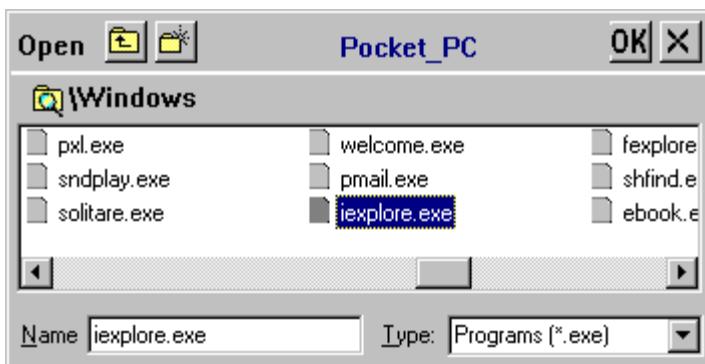
- 1 Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.
Visual CE displays the Command Button Properties dialog box.
- 2 Specify the first 11 properties (through "Continue") as you would for any other command button. Refer to step 1 in "Creating Command Buttons," on page 213 for details.
- 3 From the drop-down list of commands, select **Run external**.
- 4 If you want the current Visual CE application to close when the user clicks the command button, select the **Close Form** checkbox. If you want the application to remain open, make sure this box is clear.
- 5 In the **Command** field, identify the program to run. You can either type its name, or click the Browse button  to locate it in the CE device's file system.
 - If you type it, be sure to specify the full file pathname; for example, \Program Files\Visual CE\Projects.vce. Go to step 6.
 - If you click the browse button , the **Open** file dialog appears. Navigate to the program in the Windows CE file system, select it, and click **OK**.

Tip about Report CE switches

If you want to supply one or more switches to a Report CE layout, you can simply type them before the pathname; for example:

/f13345 \Program Files\Report CE\MyReport.rce

Refer to the topic, "Running a Report from a Visual Basic or C/C++ Application" in the *Report CE User's Guide* for information about switches.



- 6 Optional--if you want the launched application to automatically receive a value stored in the current record or a global variable as a command line argument: In the **arg** field, select the name of the column or variable that stores the value to be passed to the program.
- 7 Drag the command button to the correct position, and resize it if necessary.

Note

If you intend to run your Visual CE application on a desktop PC, please see the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," on page 296.

Creating Command Buttons to Open Report CE

You can create a command button to launch a Report CE report (.rce). When the user clicks the button, the current record is automatically displayed in the Report CE layout.

Prerequisites

Create the report template (.rce file) according to the instructions in the Report CE documentation. The application table for the Visual CE form and the top level table for the Report CE layout must be the same table. Tip: If the tables are not the same, you can alternatively set up a command button to start another program, and specify the Report CE template (.rce file) as the command. Refer to the topic, "Creating a Command Button to Start An External Program," on page 226.

You can run the report on either the CE device or the desktop PC. Regardless of which computer you use, Report CE uses data from tables on the CE device. Therefore, if you run Report CE from the desktop PC, the CE device must be connected.

Tip

You can also use the "Report CE report" command as a step in a macro program. Refer to the section, "Macros and Events," beginning on page 302 for details.

To create a command button to open Report CE:

- 1 Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.
Visual CE displays the Command Button Properties dialog box.
- 2 Specify the first 11 properties (through "Continue") as you would for any other command button. Refer to step 1 in "Creating Command Buttons," on page 213 for details.
- 3 From the drop-down list of commands, select **Report CE report**.
- 4 If you want the current Visual CE application to close when the user clicks the command button, click the **Close Form** checkbox. If you want the application to remain open, make sure this box is clear.
- 5 In the **Report** field, identify the .rce file to run. You can either type its name, or click the  Browse button  to locate it in the CE device's file system.
 - If you type it, be sure to specify the full file pathname; for example, \Program Files\Report CE\MyReport.rce. Go to step 6.
 - If you click the browse button , the **Open** file dialog appears. Navigate to the program in the Windows CE file system, select it, and click **OK**.

Tip about Report CE switches

If you want to supply one or more switches to a Report CE layout, you can simply type them before the pathname; for example:
`/f13345 \Program Files\Report CE\MyReport.rce`

Refer to the topic, "Running a Report from a Visual Basic or C/C++ Application" in the *Report CE User's Guide* for information about switches.

- 6 Drag the command button to the correct position, then resize it if necessary.

Note

If you intend to run your Visual CE application on a desktop PC, please see the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," 296.

Creating Command Buttons to Sort Records

You can create a command button that automatically sorts the current records by a specified column.

Note

The selected column must be indexed. Refer to the topic, "Indexing Columns," on page 283 for instructions.

Tip

You can also use the "Sort by" command as a step in a macro program. Refer to the section, "Macros and Events," beginning on page 302 for details.

To create a Sort command button:

- 1 Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.
Visual CE displays the Command Button Properties dialog box.
- 2 Specify the first 11 properties (through "Continue") as you would for any other command button. Refer to step 1 in "Creating Command Buttons," on page 213 for details.
- 3 From the drop-down list of commands, select **Sort by**.
- 4 In the **Col** field that appears, select the column to sort by.
- 5 Click **OK**.

Creating Command Buttons to Send SMS Messages

You can create a command button that sends an SMS instant text message from the Visual CE application. To send the message, the Visual CE application must be running on a Smartphone, PocketPC Phone Edition or a CE device that is connected to a mobile phone via infrared, Bluetooth, or serial cable.

You specify the message as a text string or an expression that returns a text string. For example, to create a message template for notifying customers of package shipments, you could specify the following expression (assuming that `firstName` is a table column):

```
'Dear ' & firstName & ', ' & 'Your package was mailed on ' & @now & '.'
```

You can specify that the recipient's phone number be read from a table column or a global variable.

Tip

You can also use the "Send SMS message" command as a step in a macro program. Refer to the section, "Macros and Events," beginning on page 302 for details.

To create a SMS Send Message command button:

- 1 Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.
Visual CE displays the Command Button Properties dialog box.
- 2 Specify the first 11 properties (through "Continue") as you would for any other command button. Refer to step 1 in "Creating Command Buttons," on page 213 for details.
- 3 From the drop-down list of commands, select **SMS Send message**.
- 4 Complete the remaining fields as follows:

Field	Description
Msg	The message to be sent, specified as a text string or an expression that returns a text string. Text constants must be enclosed in single quote marks; e.g., 'Hello World'. Refer to the topic, "Creating Calculated Fields," on page 201 for detailed instructions on creating expressions. Tip: Double-click the <code>Msg</code> field to display a larger text area in which you can enter the message.
Port	Leave blank if the Visual CE application will run on a Smartphone or PocketPC Phone Edition. Otherwise, select the COM port on the CE device that will be used to connect with the phone's modem.
Phon	The phone number to be dialed, specified as a table column or global variable.
[Flow control, Baud, Data, Parity, Stop]	If you selected a COM Port: Communications settings for the phone modem (refer to the documentation for that device)

- 5 Click **OK**.

Creating Command Buttons to Dial a Phone Number

You can create a command button that dials a phone number.

You can specify that the phone number be read from a table column or a global variable. If you do not specify a table column, Visual CE uses the first phone number it finds in the current record, starting its search with the control that has focus.

Tip

You can also use the "Dial" command as a step in a macro program. Refer to the section, "Macros and Events," beginning on page 302 for details.

To create a Dial command button:

- 1 Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.
Visual CE displays the Command Button Properties dialog box.
- 2 Specify the first 11 properties (through "Continue") as you would for any other command button. Refer to step 1 in "Creating Command Buttons," on page 213 for details.
- 3 From the drop-down list of commands, select **Dial**.
- 4 In the **Col** field, select the table column or global variable that stores the phone numbers to be dialed. Alternatively, if you leave this field empty, Visual CE will search the values in the current record until it finds a phone number, beginning its search with the control that has focus
- 5 Click **OK**.

Creating Command Buttons to Set Alarms

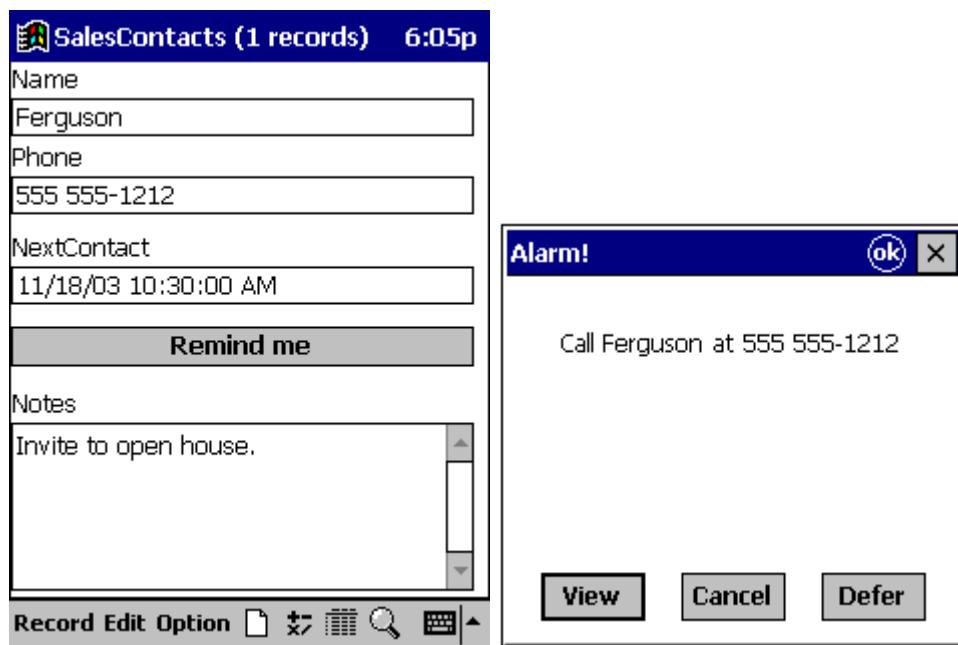
You can create a command button that users click to set an alarm for individual records.

The date and time that the alarm goes off is set by the user for each record. You can have Visual CE prompt the user for the date/time value when he/she presses the alarm button, or you can specify that Visual CE get the value from a table column or global variable associated with another control on the form.

When the alarm goes off, Visual CE beeps and displays an alarm message. You define the message as part of the form design. It can consist of a text string or an expression that returns a text string. You also specify the number of minutes, hours or days before the message redisplays if the user selects a Defer button (similar to "snooze").

In use, the form need not be running for the alarm to go off. As long as any Visual CE application is running, the alarm will display at the specified time. The alarm window contains a View button that the user can click to open the record that triggered the alarm.

The first illustration below shows a form with an Alarm command button ("Remind me"). The alarm is tied to the date/time value stored in NextContact. The second illustration shows the Alarm window that opened when the alarm went off.



Implementing the alarm feature in your form design is a two-step process: first, create the alarm message; second, add an Alarm command button.

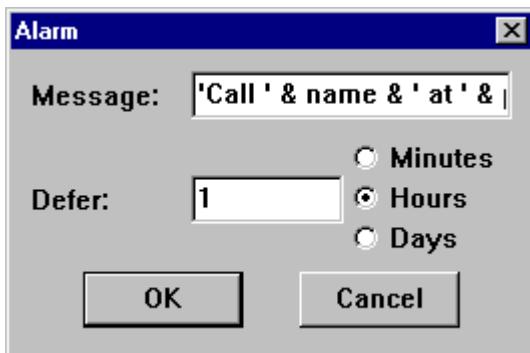
Tips

- You can also use an Alarm command as a step in a macro program. Refer to the section, "Macros and Events," beginning on page 302 for details.
- The Timer command is also available for use as a macro step. The Timer command launches a macro some number of seconds after the command is executed; the Alarm command launches a form at a given date/time.

To create the alarm message and set the defer time:

- 1 Select **Edit-Alarm**.

Visual CE displays the Alarm dialog box.



- 2 In the **Message** field, specify a text string or an expression that returns a text string. You can double-click the field to open a larger text box in which to enter the message. Text constants must be enclosed in single quote marks; e.g., 'Hello World'. Refer to the topic, "Creating Calculated Fields," on page 201 for detailed instructions on creating expressions.
- 3 In the **Defer** field, specify the amount of time to elapse before the message redisplays if the user selects a Defer button in the Alarm window (similar to "snooze").
- 4 Click **OK**.

To create an Alarm command button:

- 1 Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.
Visual CE displays the Command Button Properties dialog box.
- 2 Specify the first 11 properties (through "Continue") as you would for any other command button. Refer to step 1 in "Creating Command Buttons," on page 213 for details.
- 3 From the drop-down list of commands, select **Set alarm**.
- 4 In the **Col** field, select the table column or global variable that stores the date/time trigger values for the alarm. If you select none (leave field empty), the user will be asked to provide a date/time when setting the alarm.
- 5 Click **OK**.

Creating Command Buttons to Control mEnable Connections

You can supply command buttons to control mEnable connections.

Notes

- mEnable is a separate SYWARE product that makes it possible for Visual CE users to access data on remote servers – interactively and in real time—via wired or wireless connections including the Internet. Please refer to the *mEnable User's Guide* for details about the setup and use of mEnable.
- The mEnable Stoplight is an alternative control that performs the same function as the command buttons described here. It is compact and has the added advantage of displaying the status of the mEnable connection and the strength of the WI-FI signal. (Refer to "Creating an mEnable Stoplight Control," on page 262 for details.) You would use the command buttons described here instead of the Stoplight in three situations: 1) You want to incorporate the mEnable commands in a macro program, 2) You want to allow the user to do one type of synchronization but not the other (for example, full but not continuous), or 3) You want to suppress the log on dialog box that normally appears when the user starts an mEnable session (this dialog is pictured on page 419).

The three mEnable connection command buttons are:

Command	Does this
mEnable off	Disconnect from the server database (equivalent of the Red Stoplight)
mEnable synchronize	Establish an mEnable connection to the server database (if one is not already open) and execute full synchronization between all eligible local tables and the server database (equivalent of the Yellow Stoplight)
mEnable on	Establish an mEnable connection to the server database and synchronize individual records as they are read or written to the local table with the server database (continuous synchronization--equivalent of the Green Stoplight)

Tip

You can also use these commands as steps in a macro program. Refer to the section, "Macros and Events," beginning on page 302 for details.

To create an mEnable connection command button:

- 1 Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**.
Visual CE displays the Command Button Properties dialog box.
- 2 Specify the first 11 properties (through "Continue") as you would for any other command button. Refer to step 1 in "Creating Command Buttons," on page 213 for details.
- 3 From the drop-down list of commands, select one of the mEnable commands. Click **OK**.
- 4 If you selected either **mEnable on** or **mEnable synchronize**, the dialog includes an optional **Quiet** checkbox. Select this checkbox to suppress the logon box that would otherwise appear when a user starts an mEnable session.

Note

To allow the user to supply the connection settings, the logon box will always appear when the user initiates the first mEnable session on the CE device. In subsequent sessions, mEnable will use those settings and the logon box will no longer appear if the Quiet switch is selected.

- 5 Drag the command button to the correct position, then resize it if necessary.

Creating Command Buttons to Put or Get Files via mEnable

Two mEnable command buttons make it easy for users to transfer files between the mEnable server and the CE device using an mEnable connection:

Command	Does this
mEnable Put	Uploads a file of any type from the CE device to the mEnable server
mEnable Get	Downloads a file of any type from the mEnable server to the CE device

When one of these buttons is activated, mEnable opens a connection to the mEnable Server (unless one is already open). Once the connection is made, the Command button gets the pathname of the file to upload/download from the current record or global variable.

About Directory Structures on the CE Device and the mEnable Server

Note

The following describes the default mEnable directory structures. If mEnable Manager is in use, the system instead utilizes the custom directory structures specified by the mEnable administrator. Refer to the *mEnable User's Guide* for details.

The path to the folders that contain (or will receive) the transfer files on the CE device are mirrored on the mEnable server, with a subfolder of the folder containing the mEnable server software being the equivalent of the root on the CE device. mEnable automatically gives that subfolder the name of the CE device. For example, say that the mEnable server software is installed at C:\Program Files\ SYWARE mEnable and the CE device is named My Handheld. In that case, the following two locations map to one another:

On the CE device (named My Handheld):

\My Documents\myPhoto.jpg

On the mEnable server:

C:\Program Files\SYWARE mEnable\My Handheld\My Documents\myPhoto.jpg

When the user puts a file to the server, the file pathname must completely identify the file on the CE device (this is true regardless of whether or not mEnable Manager is in use). mEnable automatically creates the CE device folder structure on the server if the folders do not already exist there. In the example above, mEnable would create the subfolder My Handheld in the SYWARE mEnable folder, then the subfolder My Documents in My Handheld, and would finally put myPhoto.jpg into My Documents.

Conversely, when the user gets a file from the server, the file pathname must identify the destination of the file on the CE device. mEnable automatically creates the server folder structure on the CE device if the folders do not already exist there. In the example above, mEnable would create the folder My Documents on the CE device and put myPhoto.jpg in it.

Tips

- You can also use these commands as steps in a macro program. Refer to the section, "Macros and Events," beginning on page 302 for details.
- The function @fileexists (page 208) checks whether or not a file exists on the CE device. This could be a useful step in an upload or download macro.

To create an mEnable Put or mEnable Get command button:

- 1 Select **Control-Command Button**, or click the **Command** button , or right-click where you want the control to appear and select **Command**. Visual CE displays the Command Button Properties dialog box.

- 2 Specify the first 11 properties (through "Continue") as you would for any other command button. Refer to step 1 in "Creating Command Buttons," on page 213 for details.
- 3 From the drop-down list of commands, select either **mEnable get** or **mEnable put**.
- 4 Select the **Quiet** checkbox to suppress the logon box that would otherwise appear when a user starts an mEnable session. (When the user initiates the very first mEnable session on the client device, the logon appears so that the user can specify the connection settings. mEnable uses those settings for subsequent sessions.) If the Quiet checkbox is unchecked, the logon box appears every time a connection is initiated
- 5 In the drop-down list that appears, select the table column or global variable that will contain the pathname of the file to put (upload to server) or get (download from server).
Reminder: The pathname must identify the file location on the CE device, not the server

Creating Scribble Boxes

A scribble box lets users draw, save, and print freehand images such as simple sketches and signatures.

A scribble box can be read-only or read/write. It must be associated with a column defined for the "Picture" (Visual CE) or "OLE Object" (Pocket Access or SQL Server CE) data type.

Optionally, you can specify a "backing" that always appears in the scribble box. For example, you might supply a diagram that users can mark up and annotate. (For example, if you were creating an application for auto insurance adjusters, you might supply a simple outline of a car that the adjusters could mark to record areas requiring repair.) The backing image must be a .bmp, .gif, or .jpg file.

To create a scribble box:

- 1 Select **Control–Scribble** or click the **Scribble** button , or right-click where you want the control to appear and select **Scribble**.

Visual CE displays the Select Column dialog box.

- 2 Click the column in which the scribbles will be stored, then click **OK**.

Visual CE displays the Scribble Properties dialog box, which contains the following fields:

Field	Description
Connected to [column]	The table column to which the control relates – cannot be changed.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control remains in the same place on the device screen, even if the user scrolls the display or moves to another region of the form. The control is always visible in the same place on the screen. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Events	Optional. Select macros to be triggered when the user puts the focus on and/or moves the focus away from the control. Refer to "Triggering Macros with Events" on page 320.
Background color	The color of the background in the scribble box (ignored if your CE device's display is black and white).
Include in list	If checked, data from this column appears in list display mode. If unchecked, it's excluded from the list, but still appears in the single record display.
Include when printing	If checked, data from this column appears in the printed version of the record.
Read Only	If checked, users cannot draw in the scribble box. (The read-only versions do not have a box around them when displayed on the CE device.)

Background Image Optional. The file pathname of the image displayed as background for the users' scribbles (e.g., \windows\cubes.2bp.)

- 3 Make any necessary changes to the scribble box properties, then click **OK**.
- 4 Drag the box to the correct position, then resize it if necessary.
- 5 Create a label to identify the scribble box, if desired.

Note

If you intend to run your Visual CE application on a desktop PC, please refer to the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," 296.

Creating Timestamps

Timestamp buttons let users "stamp" the record with the current date and time by simply clicking the button. Alternatively, you can specify that the control automatically records the date/time when the record is created. In either case, you can lock the value so that, once the date/time is recorded, it will not be overwritten.

If desired, you can display an image on the button's face—with or without a label on top of it. The image must be a .bmp or .2bp file that is stored in the file system on the CE device.

Timestamps can be associated with Date/Time columns or global variables only.

To create a timestamp button:

- 1 Select **Control-Timestamp**, or click the **Timestamp** button  , or right-click where you want the control to appear and select **Timestamp**.
Visual CE displays the Select Columns dialog box.
- 2 Select the table column or global variable to which the timestamp button relates, then click **OK**.
Visual CE displays the Timestamp Properties dialog box, which contains the following fields:

Field	Description
Connected to [column or variable]	The table column or global variable to which the control relates – cannot be changed.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control remains in the same place on the device screen, even if the user scrolls the display or moves to another region of the form. The control is always visible in the same place on the screen. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Font	The font, size and style of the type used as the label on the button. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.
Image	Optional. The path and filename of an image to be displayed on the face of the button. The image must be a .bmp or .2bp stored in the file system on the CE device. Tip: You can gang together a number of buttons on a form to create an apparently single image with hotspots.
Text	The color of the type that appears on the button.
Back	The color of the button.
Include in list	If checked, data from this column appears in list display mode. If unchecked, it's excluded from the list, but still appears in the single record display.

Include when printing	If checked, data from this column appears in the printed version of the record.
Read Only	If checked, users cannot change the date/time recorded by the control.
Default to now	(Available only if "Read Only" is unchecked.) For the default, Visual CE uses the date and time that the record was created.
Continue	<i>For creating a macro program with buttons; this option is available to maintain backward compatibility with previous versions of Visual CE, but its use is discouraged.</i> If checked, Visual CE automatically executes the next button in the Control Order. This option is only available when the next control in the Control Order is a button. You can also set the Continue option using Form-Control Order ; refer to the topic, "Specifying Control Order," on page 272.
Write Once	If checked, Visual CE writes the date/time value once (the value is "locked" and will not be overwritten, even if the user presses the button again). Note: If the default is set to "now," this option is unavailable.

- 3 Make any necessary changes to the timestamp properties, then click **OK**.
- 4 Drag the box to the correct position, then resize it if necessary.
- 5 Create a label to identify the timestamp button, if desired.

Tip

Visual CE conforms to the settings for date/time and number formats that are specified on your desktop PC and CE device. You can change these formats using "Regional Settings" in the Windows Control Panel.

Creating Image Controls

An Image control is a box that displays pictures (.bmp, .gif, or .jpg files). You can use Image controls in two basic ways:

- to enable users to view and input (or just view) pictures in Visual CE records
- to place a static design element on the form, such as a company logo

Tip

In addition, there are two special applications for Image Controls, which are detailed in separate topics:

- Enable users to capture, file, and view images taken with a digital camera attached to the CE device; refer to “Using Image Controls with Digital Cameras” on page 244.
- Enable users to view web pages within a form; refer to “Using Image Controls to Display Web Pages” on page 246.

Images as Record Values

When you use an image control to read/write images as record values, you have several options for how the images are stored and referenced:

- **Text:** You can associate the image control with a text column or a global variable that stores text. In this case, Visual CE stores the pathname of the original image file; for example, \Windows\car.bmp. When the user opens a record that references this image, Visual CE locates car.bmp in the Windows folder and displays the picture in the form.

By saving just the link (file pathname), you conserve space. However, the user must maintain the original image file in the specified location in the device's file system, or Visual CE will be unable to locate and display the picture.

- **Picture:** You can associate the image control with a picture column. In this case, Visual CE stores a copy of the original image file in the table.

By storing copies of the images in the table, picture columns eliminate the need to maintain the original image files separately. However, since every record that stores an image includes an entire copy of it, this approach can require a great deal of space on the device.

When considering whether to use a text or picture column to store images, it's useful to consider the advantages and drawbacks of each approach in terms of storage space, maintainability, and flexibility:

The text approach may be best when images are used in multiple records. For example, if car.bmp is used in 100 records, you can store just the original image file in a folder, rather than 100 copies in the table. This approach is also preferable in cases where users will update images. For instance, if car.bmp is replaced with a new version (also named car.bmp) in the Windows folder, Visual CE automatically displays the new image in every record that references \Windows\car.bmp.

The Picture approach may be best when each image is stored in one or just a few records, or when it is important to protect the images against loss or inadvertent substitution.

Image controls can be read-only or read/write. If you allow user input, users tap the Image control on the form and pick the desired image. Visual CE saves either a copy of the selected image or a link to the original (as described above).

Static Images

Images that you place as part of the form design (such as a company logo) are always referenced by their file pathname. You do not associate them with a table column or variable, and they are not stored in the form. You must maintain them in a folder in the device's file system.

To create an image control:

- 1 Select **Control–Image**, or click the **Image** button , or right-click where you want the control to appear and select **Command**.
Visual CE displays a message asking whether the image depends on a value in a table column or global variable.
 - If form users will employ the Image control to display and/or input pictures as record values or global variables, select **Yes**. Go to step 2
 - If you intend to use the Image control to display a static image as part of the form design (e.g., a company logo), select **No**. Go to step 3
- 2 Visual CE displays the **Select Column** dialog box, listing available text and picture columns, and global variables. Select a column or variable to store the picture information, and click **OK**. (Select a text column or variable to store the pathname of the image files, which must be maintained in folders elsewhere on the device running the form. Select a picture column to store copies of the image files.)
Visual CE displays the **Image Properties** dialog box.
- 3 The properties displayed in this box depend upon whether or not you associated the control with a table column in step 1.

Field	Description
Connected to [column or variable]	If the Image control is associated with a table column or global variable (that is, if the images will be stored as record values): The column or variable that stores the picture information – cannot be changed.
Image	If the Image control is not associated with a table column or global variable (that is, if the image will be a static element that is part of the form design, such as a company logo): Specify the picture you want to appear on the form. You can either type the file pathname, or click the Browse  button to locate it in the CE device's file system.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control remains in the same place on the device screen, even if the user scrolls the display or moves to another region of the form. The control is always visible in the same place on the screen. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Events	Optional. Select macros to be triggered when the user puts the focus on and/or moves the focus away from the control. Refer to "Triggering Macros with Events," on page 320.
Image Selection on Click	If the Image control is associated with a table column or global variable (that is, if the images will be stored as record values): If checked, users will be able to tap on the control (or press the Space bar) and pick an image file to store with the record. If unchecked, the Image control will be read-only.

HTML Browser	This option must be cleared.
Border	If the Image control is not associated with a table column or global variable (that is, if the image will be a static element that is part of the form design, such as a company logo): If checked, Visual CE displays a border around the image. The border is the same size and shape as the Image control box.

- 4 Make any changes necessary to the Image control properties, then click **OK**.
- 5 Drag the box to the correct position, then resize it if necessary.
- 6 Create a label to identify the image control, if desired.

Note

If you intend to run your Visual CE application on a desktop PC, please refer to the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," 296.

Creating Image Controls to Take Pictures with Digital Cameras

If a camera is connected or integrated into the CE device, an Image control can be used to capture pictures.

The user would click the Image control or press the Space bar, aim the camera (using the Image control as the viewfinder), and click the control again to capture the picture. If you associate the control with a picture column, Visual CE puts the picture file in the record. When the record is synchronized to the desktop, it will be an OLE Object (bitmap) in the desktop database. If you associate the control with a text column or global variable, Visual CE instead puts the picture in a file in the same folder as the .vce file on the CE device and stores the picture's file pathname in the record. Visual CE automatically assigns the filename using the convention: *deviceName_dateTimeTaken*.

Cameras Currently Supported

As of this writing, the following digital cameras are supported:

- HP SDIO 1.3MP digital camera
- LifeView FlyCAM 1.3MP digital camera
- HP built-in cameras
- Cameras that support the Windows CE Camera API

Refer to the SYWARE website (www.syware.com) for the most up-to-date list.

To create an image control to take and save pictures:

- 1 Select **Control–Image**, or click the **Image** button , or right-click where you want the control to appear and select **Image**.
- 2 Visual CE displays a message asking whether the image depends on a value in a table column or global variable. Select **Yes**.
- 3 Visual CE displays the Select Column dialog box, listing available text and picture columns, and global variables. Select a column or variable to store the picture information. The type of column you choose depends upon where you wish the image files to be stored, either:
 - Select a picture column to store the image files in the table, or
 - Select a text column or a global variable to store the pathname of the image files. The image files themselves will be automatically stored in the same folder as the .vce file.
 When you have made your selection, Click **OK**. Visual CE displays the Image Properties dialog box.

Field	Description
Connected to [column or variable]	The column or variable that stores the picture information – cannot be changed.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control remains in the same place on the device screen, even if the user scrolls the display or moves to another region of the form. The control is always visible in the same place on the screen. Refer

	to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details.
	If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Events	Optional. Select macros to be triggered when the user changes a value in the control. Refer to "Triggering Macros with Events," on page 320.
Image Selection on Click	This option must be selected in order to use a camera with the control.
4 Make any changes necessary to the Image control properties.	
Note	
Be sure that the "Image Selection on Click" option is checked.	
5 Click OK .	
6 Drag the box to the correct position, then resize it if necessary.	
7 Create a label to identify the image control, if desired.	

Additional camera-specific instructions

You must install software on the CE device:

- **HP SDIO 1.3MP digital camera:** Install the manufacturer's software. Also, download <http://www.syware.com/download/wce/sdiocam.zip> and, using Active Sync, put it into the \Windows folder on the CE device.
- **LifeView FlyCAM 1.3MP digital camera:** Install the manufacturer's software.
- **HP built-in cameras:** No additional drivers needed. In particular, do **NOT** install software from <http://www.syware.com/download/wce/sdiocam.zip> for these cameras.
- **Cameras that support the Windows CE Camera API:** No additional drivers needed.

Note

If you intend to run your Visual CE application on a desktop PC, please refer to the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," 296.

Creating Image Controls to Display Web Pages

You can create an Image control that enables users to view web pages within the form. The Image control acts like a mini-browser window embedded in the form, but without the browser menu bar and toolbar.

To create an image control to display web pages:

- 1 Select **Control-Image**, or click the **Image** button , or right-click where you want the control to appear and select **Image**.
Visual CE displays a message asking whether the image depends on a value in a table column or global variable.
 - If the web page URL is a variable that the Image control will get from a table column or global variable, select **Yes**. Go to step 2
 - If the web page URL is a constant that you will “hardcode” into the control, select **No**. Go to step 3
 - 2 Visual CE displays the Select Column dialog box, listing all available table columns and global variables. Select a *text column* or *global variable* to store the URL, and click **OK**.
- Note**
You cannot select a Picture column for this application of Image controls.
- 3 Visual CE displays the Image Properties dialog box. The properties displayed in this box depend upon whether or not you associated the control with a table column in step 1.

Field	Description
Connected to [column or variable]	If the Image control is associated with a table column or global variable : The column or variable that stores the URL – cannot be changed.
Image	If the Image control is not associated with a table column or global variable : Specify the URL you want to appear display in the control. You can either type the URL (for example, ) http://www.syware.com), or click the Browse  button to locate a browser-compatible file in the CE device's file system.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control remains in the same place on the device screen, even if the user scrolls the display or moves to another region of the form. The control is always visible in the same place on the screen. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Events	Optional. Select macros to be triggered when the user puts the focus on and/or moves the focus away from the control. Refer to "Triggering Macros with Events," on page 320.

Image Selection on Click	If the Image control is associated with a table column or global variable: You must clear this checkbox. When you do, the dialog box displays the HTML Browser option.
HTML Browser	You must select this option to display a web page.
Border	If the Image control is not associated with a table column or global variable: If checked, Visual CE displays a border around the web page. The border is the same size and shape as the Image control box.

- 4 Make any changes necessary to the Image control properties, then click **OK**.
- 5 Drag the box to the correct position, then resize it if necessary.
- 6 Create a label to identify the image control, if desired.

Note

If you intend to run your Visual CE application on a desktop PC, please refer to the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," 296.

Creating a Jump to Another Table

A jump button links one Visual CE application to another. It enables the user to "jump" from the currently-displayed record in one Visual CE application to a related record with additional information in another.

The two applications must have a common key that Visual CE uses to recognize related records. Typically, the applications have a matching table column with the same name and datatype. Matching values in these columns indicate related records. Alternatively, the key value in the "jump from" application may be the value assigned to a global variable, while the "jump to" application stores its key values in a table column.

When the user clicks a jump button, Visual CE reads the current record's value in the key column or global variable. It opens the "jump to" form and displays the first record in the other table with a matching key value. If a matching key value does not exist in the other table, you can specify that Visual CE create a new record with only the key value filled in or display a message that no related records exist. Alternatively, you can specify that Visual CE always create a new, blank record in the "jump to" table with just the key value filled in, regardless of whether or not matching records exist.

Say you have two applications: ORDERS and CUSTOMERS. Both ORDERS and CUSTOMERS have a column for customer numbers (CUSTNUM). CUSTNUM is the key column. You could include a jump button on the CUSTOMERS form that would enable users to jump to the ORDERS form. Referring to the tables below, if custnum 2000 were currently displayed on the form, order A1652 would appear when the user clicked the Jump button. On the other hand, if a user were on Custnum 3000 and clicked the jump button, there would be no matching records to jump to...so Visual CE would create a new order record with Custnum 3000 already supplied.

CUSTOMERS			
CUSTNUM	LASTNAME	FIRSTNAME	PHONE
1000	Smith	John	(555) 555-0000
2000	Jones	Mary	(555) 555-9988
3000	Green	Bob	(555) 555-7540

ORDERS	
CUSTNUM	ORDERNUM
1000	A1701
1000	A1351
2000	A1652

You can see these jumps in action by running the SYWARE-supplied sample applications, JumpOrders.vce and JumpCustomers.vce. Another interesting illustration of jumps is provided by the samples, Select.vce and SelectInf.vce. (For instructions on installing these samples, please refer to the topic, "Installing Visual CE," on page 15). The samples work only on the CE device (not the desktop PC), and they must have been installed in the root directory on the CE device in order to work properly.

Tips

- You can incorporate Jumps in a macro program. Refer to the section, "Macros and Events" beginning on page 302 for details.
- **Cascade Deletes:** If you enable the jump button's "Cascade deletes" property, then the

Visual CE application automatically deletes not only the current record but also any related records in the "jump to" table when the user selects Record-Delete.

Prerequisites

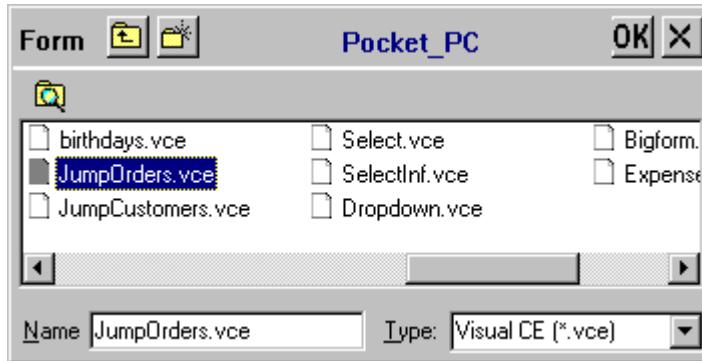
The key column in the destination table must be indexed. (In the example above, you would have to index the CUSTNUM column in the ORDERS table before adding the jump button to the Customers form.) See "Indexing Columns," on page 283 for instructions.

Note

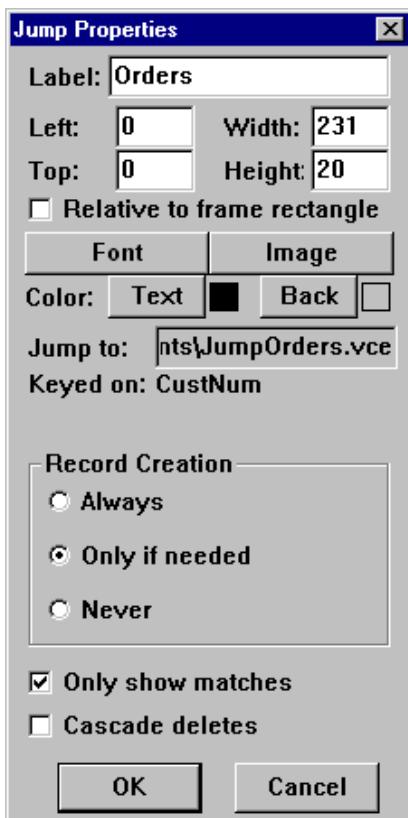
If you are working on a desktop PC, the CE device must be connected to it.

To create a jump button:

- 1 Select **Control-Jump**, or click the **Jump** button , or right-click where you want the control to appear and select **Jump**.
Visual CE displays the Select Columns dialog box.
- 2 Select the key column or global variable for the current application, then click **OK**.
Visual CE displays the Form to Jump to box.



- 3 Navigate to the application on the CE device that will be the destination for the jump button, select it, and click **OK**.
- 4 If you selected a global variable in step 2, the Select Column dialog box appears. Select the key column for the "jump to" application, and click **OK**.
Visual CE displays the Jump Button Properties dialog box.



The Jump Properties dialog box contains the following fields:

Field	Description
Label	The text that appears on the button in the form.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the button remains in the same place on the device screen, even if the user scrolls the display or moves to another region of the form. The button is always visible in the same place on the screen. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Font	The font, size and style of the type for the label on the button. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.
Image	Optional. The path and filename of an image to be displayed on the face of the button. The image must be a .bmp or .2bp stored in the file system on the CE device. Tip: You can gang together a number of buttons on a form to create an apparently single image with hotspots.
Text	The color of the type that appears on the button.
Back	The color of the button.

Jump To	The application that Visual CE opens when the user clicks the jump button (cannot be changed).
Keyed on [variable/column]	The matching columns, or variable/column combination, that jump is keyed on (cannot be changed).
Record Creation	Action that Visual CE takes when opening second application: <ul style="list-style-type: none"> ▪ Always - Always create a new record. If you select the "Only show matches" checkbox, Visual CE automatically fills in the value from the key column or variable; otherwise, the new record is completely empty. ▪ Only if Needed - Create a new record only if there are no existing records with a matching value in the key column. If you select the "Only show matches" checkbox, Visual CE automatically fills in the value from the key column or variable; otherwise, the new record is completely empty. ▪ Never - Do not create a new record. Display first record in "jumped to" application that has a matching value in the key column. If no matches are found, display messages, "Record not found."
Only show matches	If checked, Visual CE filters records in the "jumped to" application so that only records that have a matching value in the key column are retrieved for display. If unchecked, all records in the "jumped to" application are retrieved for display.
Cascade deletes	If checked, Visual CE deletes the current record in the application table and all related records in the "jumped to" table if the user selects Record-Delete when the form is running.

- 5 In the Label field, replace the column name with a label for the button (if desired). Make any other changes required in the jump properties dialog box, and click the **OK** button.
- 6 Drag the button to the desired location on the form.

Note

If you intend to run your Visual CE application on a desktop PC, please refer to the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," 296.

Creating an AutoNumber Control

You can create a control on the Visual CE form that automatically numbers each new record as it is created. Visual CE numbers the records sequentially, so that the latest record has the highest value. The values are incremented by 1, and begin with 1 or a value you specify. You can also specify the highest number that will be assigned.

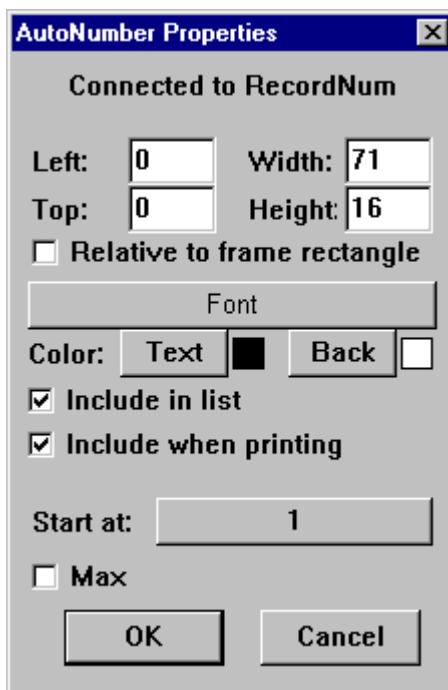
AutoNumber controls are read-only. They can be associated only with Integer or Floating Point columns. AutoNumbers cannot be associated with global variables.

Note

If all records in the table are deleted, the next new record is automatically assigned the next number in the sequence, as though the old records still existed. The autonumbering does not reset. In that way, all records have a unique identifier. For example, if the table had seven records and they were all deleted, the first record added to the table would be assigned the number 8.

To create an AutoNumber control:

- 1 Select **Control-AutoNumber**, or click the **AutoNumber** button , or right-click where you want the control to appear and select **AutoNumber**.
Visual CE displays the Select Columns dialog box.
- 2 Click the column to store the AutoNumber values, then click **OK**.
Visual CE displays the AutoNumber Properties dialog box.



The AutoNumber Properties dialog box contains the following fields:

Field	Description
Connected to [column]	The table column to which the control relates – cannot be changed.

Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	<p>If checked, the control remains in the same place on the device screen, even if the user scrolls the display or moves to another region of the form. The control is always visible in the same place on the screen. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details.</p> <p>If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.</p>
Font	The font, size and style of the type that appears in the AutoNumber field. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.
Text	The color of the type that appears in the AutoNumber field (ignored if your CE device's display is black and white).
Back	The color of the background in the AutoNumber field (ignored if your CE device's display is black and white).
Include in list	If checked, data from this column appears in list display mode. If unchecked, it's excluded from the list, but still appears in the single record display.
Include when printing	If checked, data from this column appears in the printed version of the record.
Start at [number]	Number assigned to first new record. To specify a different number, click the button and select a number using the number-picker that appears.
Max	If checked, you can specify a maximum value. Once that number is reached, any additional records that users create are assigned the same number.

3 Make any necessary changes to the AutoNumber properties, then click **OK**.

4 Drag the box to the correct position, then resize it if necessary.

Creating a Lookup

A lookup control retrieves and displays values from related records in another table (the "lookup table"). For instance, you could design an order form in which Visual CE looks up the price when the user enters a part-number.

Tip

To see first-hand how this can work, run the SYWARE-supplied sample application, *Lookup.vce*, on a CE device. (For instructions on installing the samples, please refer to the topic, "Installing Visual CE," on page 15). The samples work only on the CE device, not the desktop PC.

You specify whether the lookup value is saved with the record in a table column, written temporarily to a global variable, or merely displayed on screen. If saved in a column or variable, the value is available for printing and use in calculated fields.

Prerequisites

Visual CE matches the lookup record to the application's current record using a key value that is common to both. Typically the application table and the lookup table each have a column with the same name and datatype that store the key values.

Alternatively, a global variable may be used to store the key value. For example, you might have a calculated field in the application form that writes a temporary value for the current record to a global variable. Visual CE would compare that computed value to values stored in the lookup table's key column to identify related records.

In all cases, the key column in the lookup table must be indexed. (See the topic, "Indexing Columns," on page 283 for instructions.)

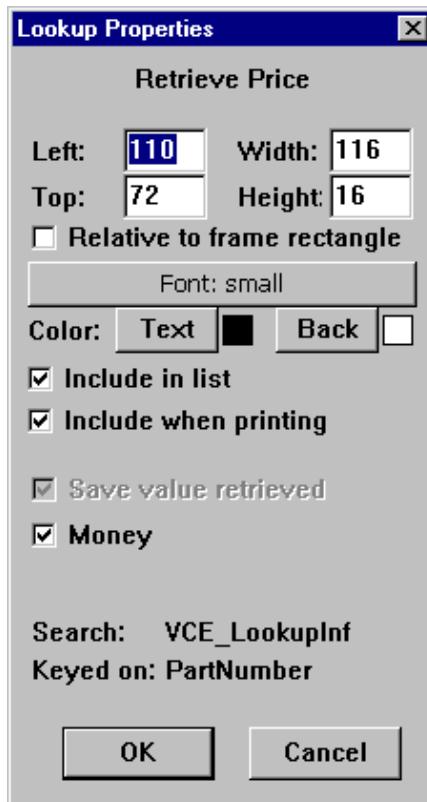
Note

If you are working on a desktop PC, the CE device must be connected to it throughout the following procedure.

To create a lookup:

- 1 Select **Control-Lookup**, or click the **Lookup** button , or right-click where you want the control to appear and select **Lookup**.
Visual CE displays the Select Table dialog box.
- 2 Select the table that stores the lookup values.
Visual CE displays the Select Column dialog box.
- 3 Select the application "key" column or global variable.
 - If you selected a column, go to Step 4.
 - If you selected a global variable, Visual CE displays the Select Column box. Pick the key column in the lookup table. Go to Step 4.
- 4 Visual CE asks if you want the lookup value saved:
 - Respond **No** if you want the lookup value displayed in the form but not saved. Go to Step 5.
 - Respond **Yes** if you want the lookup value saved when the user saves the record. Visual CE asks where to store the lookup value. Select a table column or global variable.
 - ◊ If you selected a column to store the lookup value, go to Step 6.
 - ◊ If you selected a global variable to store the lookup value, go to Step 5.
- 5 Select the column that contains the values you want displayed in the lookup field.

6 Visual CE displays the Lookup Properties dialog box.



The Lookup Properties dialog box contains the following fields:

Field	Description
Retrieve [column/variable]	The table column or global variable that contains the lookup value to be displayed by the control – cannot be changed.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control remains in the same place on the device screen, even if the user scrolls the display or moves to another region of the form. The control is always visible in the same place on the screen. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Font	The font, size and style of the type that appears in the lookup field. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.
Text	The color of the type that appears in the lookup field (ignored if your CE device's display is black and white).

Back	The color of the background in the lookup field (ignored if your CE device's display is black and white).
Include in list	If checked, data from the lookup field appears in list display mode. If unchecked, it's excluded from the list, but still appears in the single record display.
Include when printing	If checked, data from the lookup field appears in the printed version of the record.
Save value retrieved	If checked, value in lookup field will be saved with the record. If unchecked, value will be displayed but not saved. (Whether or not this option is checked depends upon your response in Step 4 above. You cannot change this property here.).
Money	If the lookup value you specified is stored as either an integer or floating point number, the Money checkbox displays. If Money is checked, entries are displayed with two decimal places.
[Date/time options]	If the column is associated with dates/times, four options are available for the type of data that can be displayed: Long Date, Short Date, Time, or Date/Time. Note: The display format associated with each option is determined by the Regional Settings on the computer running the form. Refer to the topic, "Edit Box Date/Time Formats," on page 174 for details.
Search: [tablename]	The name of the lookup table – cannot be changed.
Keyed on: [columnname/variable]	Name(s) of the keys that associate the records in the two tables – cannot be changed

7 Make any changes required to the lookup properties, then click **OK**.

8 Drag the lookup control to the desired location on the form.

If necessary, create a label to identify the lookup field.

Note

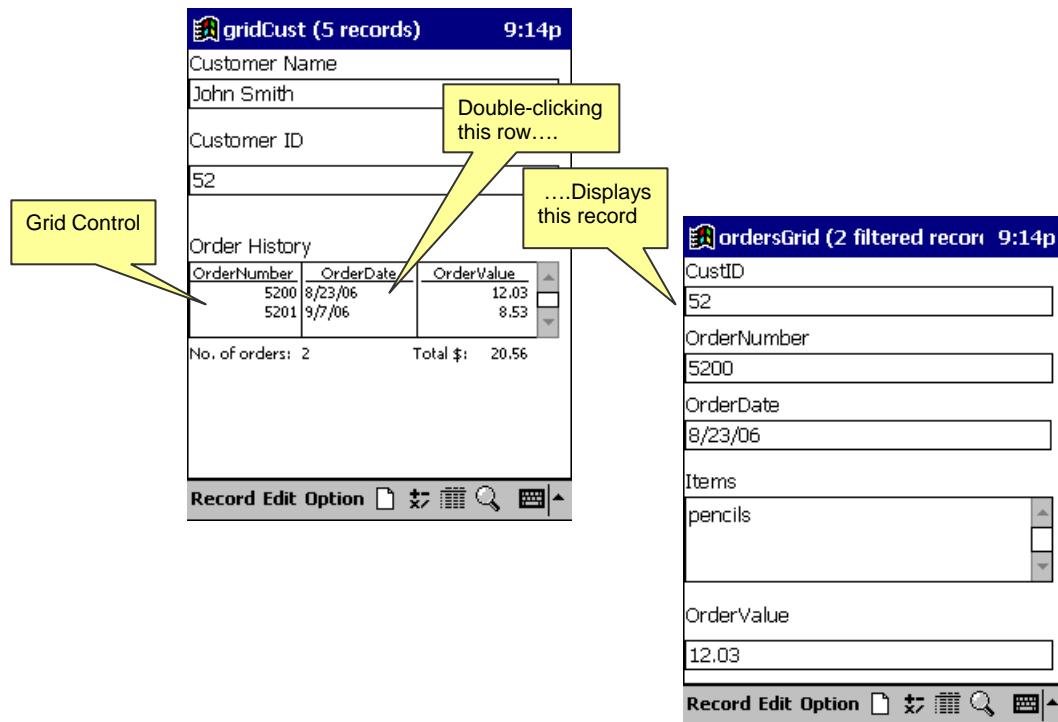
If you intend to run your Visual CE application on a desktop PC, please refer to the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," on page 296.

Creating a Grid Control

A grid control is a scrollable window that displays a list of all of the records in the application table, or related records from a different table. You can choose which record values are displayed, and how the records are sorted. You can allow users to change the values displayed. You can even display summary values for the total number of records in the grid and the sum of the values in a grid column.

How the Grid Works When Displaying Related Records from a Different Table

When used to display related records from a different table, grid controls combine and expand the functions of lookup and jump controls. (See illustration below.) While lookups display just one value from the first related record in another table, grids can display multiple values for all related records in the other table. If the user double-clicks a row in the grid window, Visual CE displays the selected record in another Visual CE application (just like a jump button). You can also turn off the jump, so that the grid is simply a display.



How the Grid Works When Displaying Records from the Current Table

You can also use a Grid Control to display all records from the current application table in the form, giving users a convenient means of viewing a list of records without having to switch to List Mode. If the user double-clicks a row in the grid, Visual CE displays the selected record in the current form. If you want the grid to simply display the records, you have the option of turning off this jump behavior.

Prerequisites if Displaying Related Records from a Different Table

The application table must have one column that matches a column in the other table (which we will call the "jump-to" table). Both columns must have the same column name and store the same data type. This column is referred to as the "key" column, since Visual CE uses it to match records in the two tables. (In the example above, CustID is the key column.)

Alternatively, a global variable may be used to store the key value. For example, you might have a calculated field in the application form that writes a temporary value for the current record to a global variable. Visual CE would compare the computed value to values stored in the jump-to table's key column to identify related records.

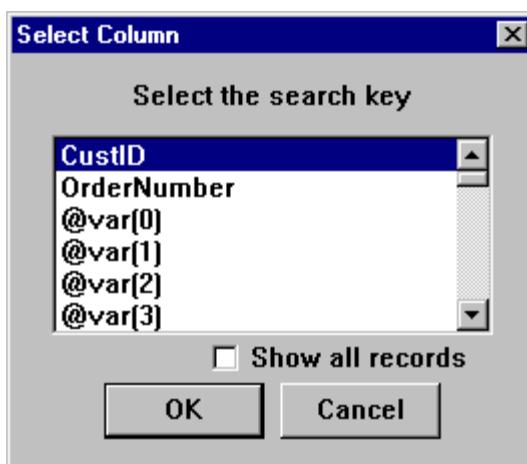
In both cases, the key column in the jump-to table must be indexed. (See the topic, "Indexing Columns," on page 283 for instructions.)

Note

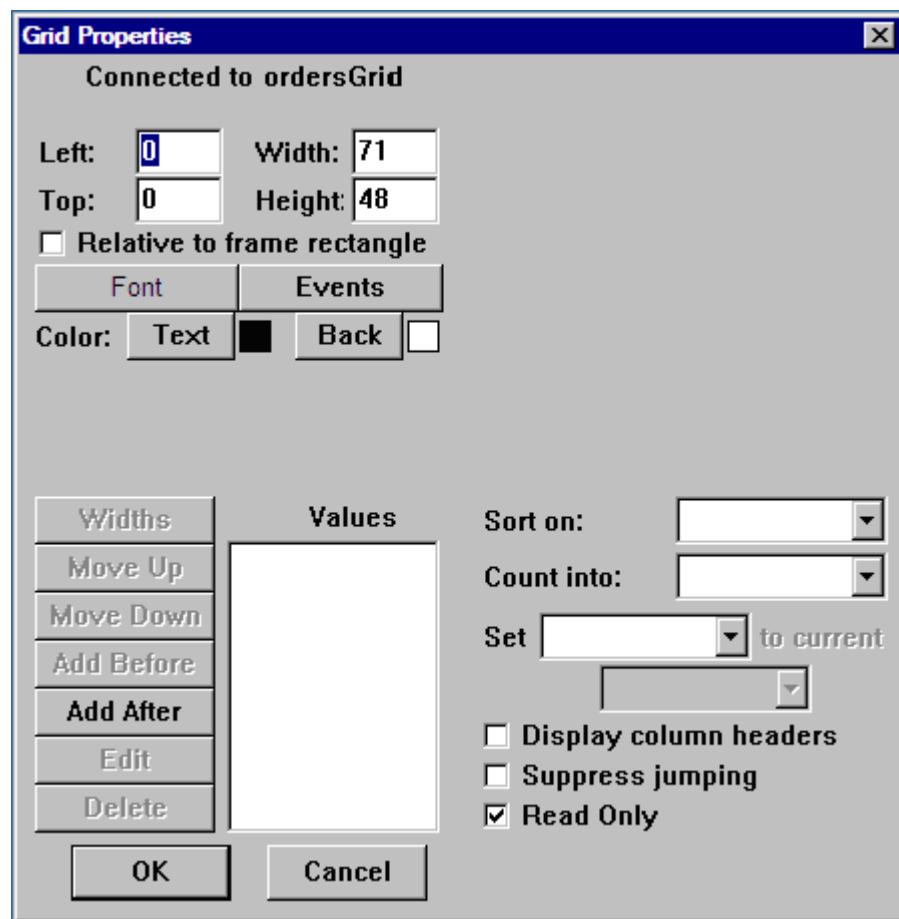
The CE device must be connected to the desktop PC throughout the following procedure.

To create a grid control:

- 1 Select **Control-Grid**, or click the **Grid** button , or right-click where you want the control to appear and select **Grid**.
Visual CE displays the Select Column dialog box.



- 2 Your selection depends upon whether you want the grid to display all records from the current application table, or related records from another table:
 - To display all records in the application table, select **Show all records**, and click **OK**. Go to Step 5.
 - To display records from a related table, select the application "key" column or global variable. Click **OK**. Visual CE displays the "Form to Jump to" box.
- 3 Navigate to the Visual CE application that will be launched when the user double-clicks a row in the grid control. Select it, and click **OK**.
- 4 If you selected a global variable in Step 2, Visual CE asks you to select the key column in the jump-to table. Select it and click **OK**.
Visual CE displays the Grid Properties dialog box.



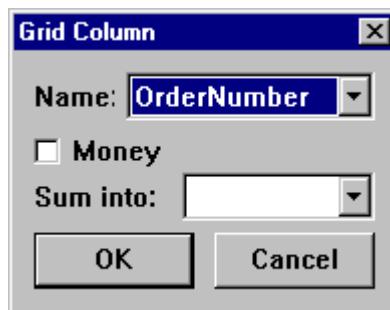
5 Edit the Grid properties as necessary:

Field	Description
Connected to [table]	The application or jump-to table – cannot be changed.
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control remains in the same place on the device screen, even if the user scrolls the display or moves to another region of the form. The control is always visible in the same place on the screen. Refer to the topic, "Fixing a Control's Position on the Screen," on page 266 for more details. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.
Font	The font, size and style of the type displayed inside the grid window. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.
Events	Optional. Select macro to be triggered when the row selected in the control is changed. Refer to "Triggering Macros with Events," on page 320.

Text	The color of the type that appears in the grid window (ignored if your CE device's display is black and white).
Back	The color of the background in the grid window (ignored if your CE device's display is black and white).
Jump To	<i>Shown only if you selected a search key in Step 2.</i> The application that Visual CE opens when the user double-clicks a row in the grid window (cannot be changed).
Keyed on [column or variable]	<i>Shown only if you selected a search key in Step 2.</i> Table column or global variable that jump is keyed on (cannot be changed).
Values	Columns to appear in the grid window. See Step 6 below.
Sort on	Optional. Column containing the values Visual CE will use to sort the rows in the grid (column need not be displayed in the grid window).
Count into	Optional. Global variable to store count of records retrieved and displayed in the grid.
Set [global variable] to current [table column]	Optional. Useful if row selection fires an event. When the user clicks a row in the grid, the value stored in the specified table column for the row will be assigned to the global variable. That way, the processing of the event will know which row was just selected.
Display column headers	If checked, the table column names appear as column headers in the grid window.
Suppress jumping	If checked, the grid is display-only. If unchecked, users can double-click a row in the grid to open that record.
Read Only	If checked, the display is read-only. If unchecked, users can edit data in the grid by tapping and holding on the value to be updated.

6 For each table column you want to appear in the grid window:

- ◊ Click the **Add After** button (or **Add Before** if available and you prefer). Visual CE displays the Grid Column dialog box.



- ◊ In the **Name** field, select a column whose values you want to appear in the grid window.
- ◊ Depending upon the type of data stored in the column, Visual CE displays options for displaying the data in the grid, such as whether or not you wish display integers as money values, or the format for date/time values. For numeric columns, the **Sum into** field is available -- you can select a global variable to store the sum of the values in the grid column. (For example, see the "Total \$" field shown in the illustration at the

- beginning of this topic.) Select the desired options (if any), and click **OK**.
- 7 Visual CE will display the columns in the grid from left to right in the same order they are listed in the Values box from top to bottom. You can change the order by selecting a name in the Values box and clicking **Move Up** or **Move Down**. You can also remove a column from the grid by selecting its name and clicking **Delete**. And, you can change the settings you specified in the Grid Column box by selecting the name and clicking **Edit**.
 - 8 When you have finished specifying the Grid properties, click **OK** to close the Grid Properties dialog box.

Visual CE displays the grid, allotting equal width to each column.

Tip

Visual CE allows you to modify the widths of the columns within the grid window (described in the next step). You may find it easiest if you first make the window itself the desired size, then modify the column widths.

- 9 If you wish to modify the widths of the columns in the grid window:
 - ◊ Double-click the grid control itself.
Visual CE redisplays the Grid Properties dialog box.
 - ◊ Click the **Widths** button in the Grid Properties dialog box.
Visual CE displays the Grid Widths dialog box.



- ◊ Point the cursor at a vertical line dividing two columns and drag to the left or right.
 - ◊ When you have resized the columns as desired, click **OK**. Then click **OK** to close the Grid Properties dialog box.
- 10 Drag the window to the desired location.
 - 11 If you assigned summary values to global variables, create a calculated field control to display each one.

Note

If you intend to run your Visual CE application on a desktop PC, please refer to the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," on page 296.

Creating an mEnable Stoplight Control

An mEnable stoplight control makes it possible for Visual CE/mEnable users to monitor and control the status of an mEnable connection, and to see a warning when the WI-FI signal is weak or lost.

mEnable can be used in either of two modes: *Always Connected* or *Occasionally Connected*. In either mode, the stoplight indicates the strength of the WI-FI signal: blinking yellow means approaching the outskirts of acceptable range (transfer may be slower than normal), while blinking red means out-of-range (lost connection). It also indicates the status of the mEnable connection and synchronization activity.

Additionally, you can make it possible for users working in Occasionally Connected mode to apply the stoplight control to turn the mEnable connection off/on and select synchronization. They can tap each light: red to disconnect from the server database and work locally; yellow to connect and execute full synchronization between all eligible local tables and the server database (users must wait while the light is yellow); and green to connect and continuously synchronize individual records as they are read from/written to the local table and the server database.

Notes

- mEnable is a separate SYWARE product that makes it possible for Visual CE users to access data on remote servers – interactively and in real time—via wired or wireless connections including the Internet. Please refer to the *mEnable User's Guide* for details about the setup and use of mEnable.
- You can use command buttons on your form instead of the stoplight to control synchronization. You would use command buttons instead of the stoplight in three instances: 1) You want to incorporate the mEnable commands in a macro program, 2) You want to allow the user to do one type of synchronization but not the other (for example, full but not continuous), or 3) You want to suppress the log on dialog box that normally appears when the user starts an mEnable session (this dialog is shown on page 418). Refer to the topic, "Creating Command Buttons as mEnable Controls," on page 234 for instructions. Even if you use commands to control synchronization, you can still incorporate a stoplight in your form to display the status of the WI-FI signal.

To create an mEnable stoplight:

- 1 Select **Control – mEnable**, or click the **mEnable** button , or right-click where you want the control to appear and select **mEnable**.

Visual CE displays the mEnable Properties dialog box which contains the following fields:

Field	Description
Left	The initial horizontal position of the left side of the control.
Top	The initial vertical position of the top of the control.
Width	The initial width of the control.
Height	The initial height of the control.
Relative to Frame Rectangle	If checked, the control is positioned relative to the frame; e.g., the device screen. The control remains in the same place on the screen, even if the user scrolls the display or moves to another region of the form. This is useful for making the control a header or footer, for example. If unchecked (the default), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display.

Display Only

Applicable to forms used in Occasionally Connected mode. If checked, the stoplight only displays the strength of the WI-FI signal and the connection/synchronization status. The stoplight's features for initiating an connection and controlling synchronization are turned off. (In this scenario, you would most likely use a macro or command buttons to control synchronization.)

If unchecked, the stoplight enables users to control synchronization by tapping the lights.

In Always Connected mode, the stoplight is only used to display the strength of the WI-FI signal and indicate the connection/synchronization status.

- 2 Make any changes required to the properties, then click **OK**.

If you are not satisfied with the size or location of the control, you can drag it to a different location and/or resize it manually.

Modifying Control Properties

Changing Control Properties

Once you have created a control, you can change some of its properties, but not all.

The only properties you **cannot** change are:

- The type of control. For example, you cannot change radio buttons into a drop-down list.
- The table column or global variable to which the control is associated.

Most of the other properties of a control can be changed.

Note

If you do need to change a control's type, column, or other properties that cannot be changed, delete the existing control and create a new one.

To change a control's properties:

- 1 Double-click the control (or click the control once and select **Edit–Property**, or right-click the control once).
Visual CE displays the control's Property dialog box.
- 2 Make any changes required, then click **OK** on the Property dialog box.
Visual CE changes the control's properties.

Tip

You can change properties for several controls at once (refer to the topic, "Changing Properties for Multiple Controls," on page 267).

Specifying Font Properties for a Control

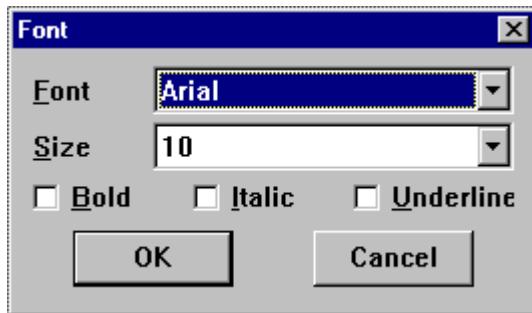
The Properties dialog boxes for most types of controls include a Font button that enables you to assign a font, size, and style (bold, italic, and underline) to the characters in the control. Using this feature, you can give your form a more personalized look, as well as add clarity by emphasizing important elements.

Note

If you do not assign specific font properties to a control using the Font properties dialog box, Visual CE automatically uses the default settings, which are Arial, 10 point, with no styles.

To assign font properties to a control:

- 1 In the Properties dialog box, click the Font button . Visual CE displays the Font dialog box.



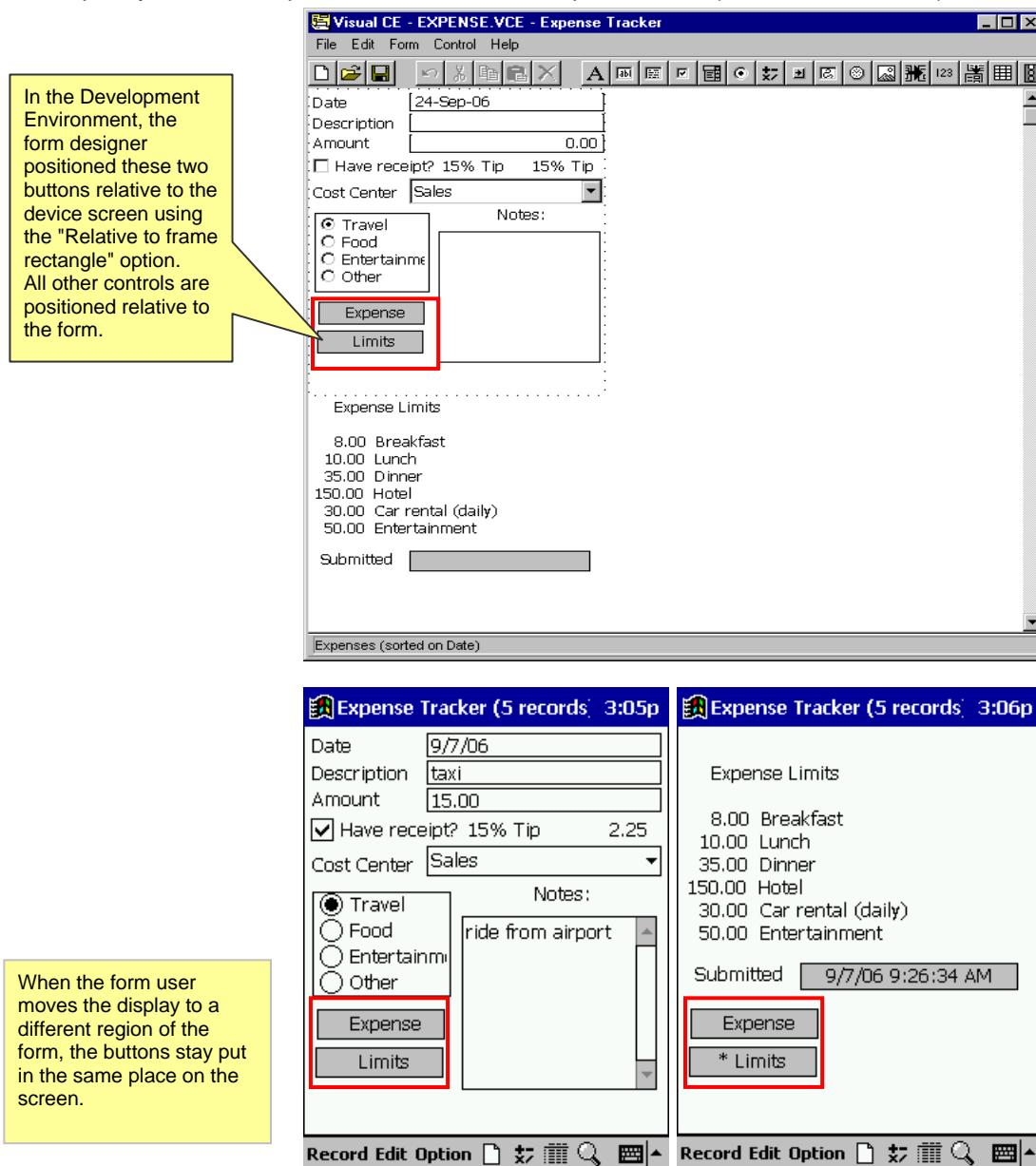
- 2 Assign font properties to the characters displayed by the control:
 - **Font:** Choose between Courier New (a monospaced font), Arial (a proportional sans-serif font), and Times New Roman (a proportional serif font).
 - **Size:** Choose a point size. Visual CE initially displays the most commonly used sizes in the drop-down, but you can use the keyboard to enter any size between 4 and 127 points.
 - **Bold, Italic, Underline:** You can apply any or all of these styles to the control.
- 3 Click **OK** to enter your selections and return to the Properties dialog box.

Fixing a Control's Position on the Screen

The Properties dialog box for most types of controls includes a **Relative to frame rectangle** option.

If unchecked (the default for most control types), the control is positioned relative to the form. It moves along with the other form controls as the user scrolls the display or moves to a new region. It can disappear from view.

If this option is checked, the control is positioned relative to the frame; i.e., the device screen. The control is always visible in the same place on the screen, even if the user scrolls the display or moves to another region of the form. This is useful for making the control a header or footer, for example. It is also useful for creating command buttons that enable users to move back and forth between different regions. The Quick Tour's Expense Tracker sample form provides an example of using frame-relative buttons to enable users to move quickly between expense information and expense limits (as illustrated below).



Changing Properties for Multiple Controls

Using this feature, you can select two or more controls and set the position, size, font, background color, foreground color, "include in list," and "include when printing" properties for all of them at once.

- 1 While pressing the Control key on the keyboard, select each control. Alternatively, you can left-click the mouse and drag the cursor across the controls.
- 2 Select **Edit-Property**.

Visual CE displays the Properties dialog box, which contains the following fields:

Field	Description
Left	The horizontal position of the left side of the controls.
Top	The vertical position of the top of the controls.
Width	The width of the controls.
Height	The height of the controls.
Font	The font, size and style of the type displayed by the controls. Click to modify. Refer to the topic, "Specifying Font Properties for a Control," on page 265 for instructions.
Text	The color of the type that appears in the controls (ignored if your CE device's display is black and white).
Back	The color of the background in the controls (ignored if your CE device's display is black and white).
Include in list	If checked, data from these controls appears in list display mode. If unchecked, it's excluded from the list, but still appears in the single record display.
Include when printing	If checked, data from these controls appears in the printed version of the record.

Note

Visual CE applies any changes you specify in this dialog box to all controls selected in step 1. If you want individual controls to retain their settings for some of the properties, simply leave the initial values in place.

- 3 Make any changes that you want to apply to ALL selected controls, then click **OK**.

Arranging Controls on the Form

Turning Off Snap-to-Grid

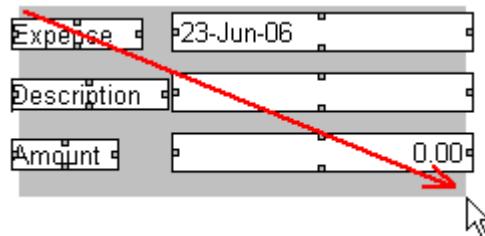
When the "snap-to-grid" feature is on, Visual CE "snaps" controls to an invisible grid as you drag them, helping you align and size controls uniformly. Sometimes, though, you may want to position controls more freely.

- To temporarily turn off the snap-to-grid feature, simply press the Shift key on the keyboard as you drag the control.
- To drag multiple controls this way, hold down the Control key while selecting each one (or left-click and drag across the controls), then hold the Shift key as you drag the controls into position.

Aligning and Sizing Controls on the Form

You can use the Align command in the Edit menu to precisely align and size selected controls relative to one another.

- 1 Select all of the controls you wish to align or make the same size using any of the following methods:
 - ◊ Select **Edit-Select All**, or
 - ◊ While pressing the Control key, click each of the controls, or
 - ◊ Left-click the mouse and drag over the controls. The selection area is indicated by a gray box. Release the mouse button. (This method works the same as a Windows marquee select.)



Note

Visual CE aligns or sizes all of the selected controls to the first control you select. The selection handles around that control have solid boxes, while the others have empty boxes.

- 2 Select **Edit-Align** and the alignment option you want:

Option	Action
Top	Aligns the top edges of the controls.
Center	Aligns the top/bottom centers of the controls in a horizontal line
Bottom	Aligns the bottom edges of the controls.
Left	Aligns the left edges of the controls.
Center	Aligns the left/right centers of the controls in a vertical line
Right	Aligns the right edges of the controls.
Width	Makes all selected controls the same width as the first selected.
Height	Makes all selected controls the same height as the first selected.

Applying Cut, Copy, and Paste to Controls

You can use the Cut, Copy, and Paste commands to duplicate or move one or more controls. The Cut and Copy commands place a copy of the controls on the Clipboard. You can then paste the controls anywhere in the current form or in a different form.

To cut, copy, or paste one or more controls:

- 1 Select the control(s) you want to cut or copy. If you want to perform the operation on multiple controls, press the Control key on the keyboard while selecting each one, or left-click and drag across the controls.
- 2 Choose **Edit–Cut** or **Edit–Copy**. Alternatively, you can click the Cut  or Copy  button
- 3 If you want to paste the control(s) into another form, open the form now. (Otherwise, skip this step.)
- 4 Choose **Edit–Paste** or click the **Paste** button .
- 5 The next step depends upon whether or not you are pasting controls that are connected to table columns:
 - If you are cutting/copying just one control and that control is not connected to a table column or variable, Visual CE just pastes the control into the form.
 - If you are cutting/copying just one control and that control is connected to a table column or variable, Visual CE displays the Select Column dialog box. Click the column to which the control relates, then click **OK**.
 - If you are cutting/copying multiple controls that are connected to table columns and pasting them into the same form, Visual CE automatically pastes the controls and associates them with the same table columns with which they were originally associated.
 - If you are cutting/copying multiple controls that are connected to table columns and pasting them into another form, Visual CE automatically pastes the controls if they have matching table columns (same name and datatype) in the second form's table and ignores any that do not. Visual CE beeps to alert you if it could not paste all of the controls.

Deleting Controls

To delete one or more existing controls:

- 1 Click the control to highlight it. If you want to delete multiple controls, hold down the Control key while clicking each one with the mouse pointer, or left-click and drag across the controls.
- 2 Select **Edit–Delete** or click the **Delete** button .

Customizing the Form

Customizing or Removing the Menu Bar

The standard Visual CE application window contains a menu bar with three menus (Record, Edit, and Option) and toolbar buttons. You can replace this default configuration with up to four custom menus. This feature makes it easy to provide menus precisely tailored to user needs and to create applications for non-English speaking users.

When creating a custom menu, you can include any of the standard Visual CE commands named as desired, commands to launch macros you have created, and horizontal lines to visually separate groups of menu items.

You also have the option of removing the menus and toolbar buttons from the application window altogether. In that case, you can give users access to a customized set of commands by adding command buttons to your form. Refer to the topic, "Creating Command Buttons as Shortcuts for Menu Picks," on page 219.

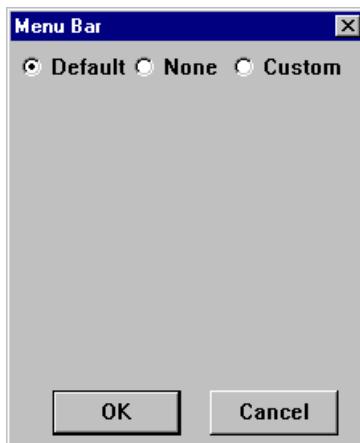
Tips

- If you hide or customize the menu bar, the standard shortcut keys (such as Ctrl+W) are disabled. However, Ctrl+Q is always available to close the form.
- If you hide the menu bar, users will have no obvious way to close the form. You might consider supplying a command button for that purpose (Record | Close). See the topic, "Creating Command Buttons as Shortcuts for Menu Picks," on page 219.

To customize or remove the menu bar:

- 1 Select **Form—Menu Bar**.

Visual CE displays the Menu Bar dialog box.

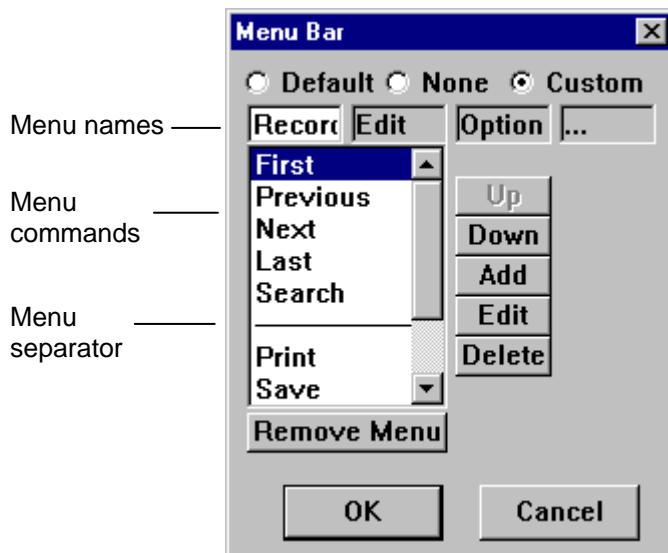


- 2 Choose one of three options:

- **Default:** Display the standard menus (Record, Edit, and Option) and toolbar buttons in the application window.
- **None:** Remove the menu bar and toolbar buttons from the application window. The standard shortcut keys (except Ctrl+Q) will be disabled.
- **Custom:** Create your own menus for the application window. You can simply add a new menu to the default set; modify or delete the default menus; and/or create all new menus. Note: If you select "Custom," the toolbar buttons will no longer be available in the application window and the standard shortcut keys (except Ctrl+Q) will be disabled.

If you select "Custom", the dialog box displays additional options. Proceed with Step 3.

Otherwise, click **OK**.



- 3 You can have one to four menus. You can add new menus, revise or reuse existing menus, or delete menus. Detailed instructions are provided below.

To add a new menu:

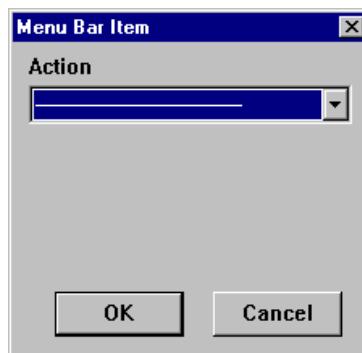
- 1 Click in the box containing the three ellipses (...), and enter a name for the new menu

Note

Each new menu you create is automatically added to the right of existing menus. If you want a new menu to appear to the left of an existing menu (for example, Option), you would have to first delete the Option menu, add the new menu, then recreate the Option menu. If you are creating all new menus, be sure to add them in the order you wish them to appear from left to right.

- 2 You can add any of three types of items to a menu: separators (horizontal lines) to visually group commands, standard Visual CE commands named as desired, and commands to run macros you've previously created. The method to add each type is described below:

- **To add a separator:** Click the **Add** button. Visual CE displays the Menu Bar Item dialog box. The separator (a horizontal line) appears in the Action field. Click **OK**.



- **To add a Visual CE command:** Click the **Add** button. Click the small down arrow to the right of the Action field, and select the desired command from the list. If desired, enter a new name to identify the command in the menu. Click OK.



- **To add a command to run a macro:** *To use this feature, you must have previously created a macro.* Click the **Add** button. Click the small down arrow to the right of the Action field, scroll to the bottom of the list, and select **Run Macro**. In the Macro field, select the macro to run. In the Name field, (optionally) enter a different name to appear in the menu. Click OK.



3 Repeat step 2 for each item you wish to include in the menu.

If desired, you can modify the menu using the instructions below.

To modify a menu (reorder, rename, add, or delete menu commands):

- 1 In the Menu Bar dialog box, click in the box containing the name of the menu to be modified (e.g., Edit).

The commands and any separators that currently make up the menu appear below the menu name.

- 2 Modify the menu as desired:

- To reorder the menu items, click an item to highlight it, then click either the **Up** or **Down** button to change the item's position in the list.
- To rename a command, click the command to highlight it, then click the **Edit** button. Enter the new name in the Name field and click OK.
- To associate a different action with a command, click the command to highlight it, then click **Edit**. Click the small down arrow to the right of the Action field, and select the desired action. Click OK.

- To add a command, macro, or separator, click the **Add** button and proceed as described in Step 2 on page 273.
 - To remove a menu item, click the item to highlight it, then click the **Delete** button.
- 3 When you have finished modifying the menu, click **OK**.

To remove a menu:

- 1 In the Menu Bar dialog box, click in the box containing the name of the menu you wish to remove (for example, Record).
- 2 Click the **Remove Menu** button.
Visual CE asks you to confirm that you want to remove the menu. If you respond "Yes," Visual CE removes the entire menu and slides the remaining menus to the left.

Specifying Control Order

Visual CE applies the order you specify here to five places in the application:

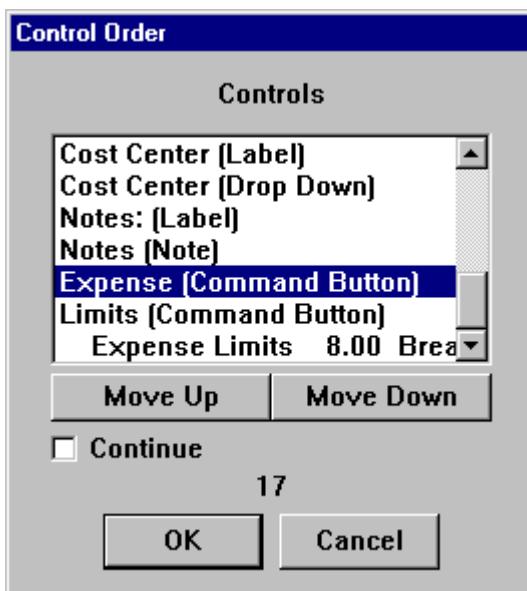
- the order of the columns in the record list display,
- the order of record values in a printed copy,
- the order in which the cursor moves from field to field when the user presses Tab key,
- the order of field values in imported or exported comma delimited text files, if the "All Table Columns" option is not specified
- the order of command buttons in a macro program (discouraged)

There are two methods for specifying control order: reordering a list of the controls in the Control Order dialog box or clicking the controls on the form in the desired order. Both are explained below.

To specify column, print, and tab ordering:

1 Select Form–Control order.

Visual CE displays the Control Order dialog box. Initially, the controls are listed in the order you created them. (All controls are listed, but when the application is used, the cursor tabs only to those fields in which user entries are allowed.)



- 2 In the Control Order dialog box, select one of the controls and click the **Move Up** or **Move Down** button one or more times. Visual CE moves the selected control in the sequence. Repeat this technique with the other controls until the entire list is reordered as desired.
- 3 For controls that are part of a macro sequence (except Skip buttons or the last control in the sequence), select the **Continue** checkbox. Note: To maintain backward compatibility, using a sequence of controls to create a macro is still possible, but discouraged.
- 4 An alternative way to reorder the controls is by clicking them in the desired order on the form layout:
 - Drag the Control Order dialog box out of the way. Notice how the numbering displayed on the form matches the sequence in the Control Order dialog box.
 - On the form, click the first control – the one in which you want the cursor to appear initially.

Visual CE changes that control's number to "1."

- Click the control you want the cursor to move to next.

Visual CE changes that control's number to "2."

- Continue to click each control to indicate the tab order you want.

Visual CE continues to increment each tab order number by "1."

Note

If you make a mistake, click the Cancel button in the Control Order dialog box.

Alternatively, select **Edit–Undo**.

- 5 When you have finished specifying the control order, click **OK** in the Control Order dialog box, or click anywhere in the environment window (except on a control).

Preventing Duplicate Values in a Column

You can specify that only unique values are allowed in a selected table column.

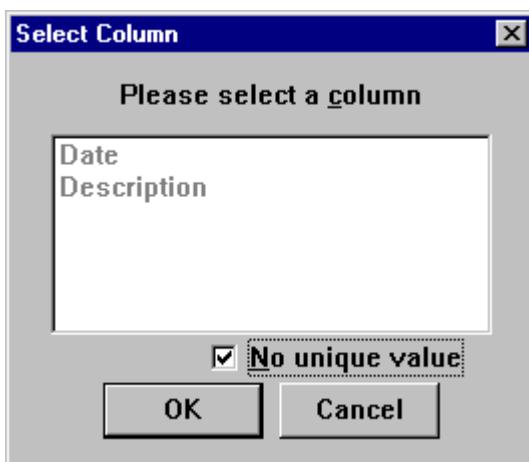
Notes

- The column must be indexed before you can apply this feature. If the column is not already indexed, follow the instructions provided in the topic, "Indexing Columns," on page 283.
- Only one column can be designated to hold unique values.

To disallow duplicate values in a column:

1 Select **Form-Unique**.

Visual CE displays the Select Column dialog box. Initially, the "No unique value" option is selected: duplicates are allowed in all columns.



2 Uncheck **No unique values**.

The list of eligible columns becomes active (no longer gray).

3 Highlight a column name and click **OK**.

Enabling the Auto Recalc Feature

The Auto Recalc feature enables you to specify when Visual CE updates values for the Calculated controls:

- If this option is enabled, Visual CE automatically recalculates the values whenever the user presses a timestamp button or moves the input focus from an edit control that is tied to a text, number, or date/time field.
- If this option is disabled, the user must explicitly select the Option-Recalc command to view the result of the calculation.

Note

If your form includes dependent drop-downs, turn Auto Recalc on. Otherwise, the dependent drop-downs may not work properly.

To specify the Auto Recalc setting:

- 1 Click **Form** in the menu bar.
- 2 The Auto Recalc feature "toggles" on and off. To change the current setting, click the option.

Enabling Bar Code Scanners

A Visual CE form can receive input from most bar code scanners, including those from Symbol Technologies (<http://www.symbol.com>), Socket Communications (<http://www.socketcom.com>), and Intermec (<http://www.intermec.com>). Form users simply click the control that accepts the scanned input, and then press the "scan" button on the scanner.

When designing the form, there is no need to do anything special in order to enable the bar code scanner, but keep the following in mind if you are designing for Symbol, Socket, or Intermec scanners:

Notes for Symbol, Socket, and Intermec Scanners:

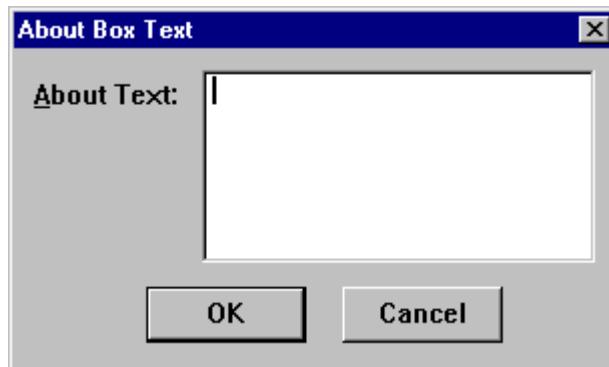
- You must use **text** columns to store barcode values. When your form is in use, the barcode reader is automatically enabled whenever the focus is on an edit control for text input. When the focus is on any control other than an edit control, or is on an edit control for numeric or date input, the barcode reader is automatically disabled.
- If the control that receives the scanned input is followed by a button (command, timestamp, or jump) in the control order, Visual CE automatically activates the button after receiving the scanned input.
- To trigger a Socket In-Hand Scan Card, users click the edit control.

Creating Text for the "About Box"

You can create a brief description of your application that displays when a user selects "About" from the application's Option menu.

- 1 Select **Form->About Text**.

The About Box text dialog box appears.



- 2 Enter the text, as you want it to appear in the About Box, and click **OK**.

Defining Record Sorts and Searches

Specifying an Initial Sort/Search Ordering

You have the option to specify an initial sort/search order for record display – Visual CE's default is unsorted records. The sort/search order can be based on the value in any column, except one containing Pictures.

The sort/search order determines the records that Visual CE scrolls to when the application's user applies the First, Previous, Next, and Last buttons (or their Record menu equivalents). It also enables the Search function, which allows users to locate records based on the column specified.

Note

If you are developing the form on the desktop PC, the CE device must be connected.

To specify a sort/search order:

- 1 Select **Form–Sort on**.

Visual CE displays the Select Column dialog box.

Note

Only "indexed" columns can be used for sorting. If the desired column is not visible in the list, it's probably not indexed. See "Indexing Columns," on page 283 for instructions.

- 2 Click the **Not Sorted** checkbox to remove the checkmark.
- 3 Click the column on which you want sorting based, then click **OK**.

Note

While the form is running, the user can change the sort/search ordering. See "Changing Sort Search Order," on page 394.

Indexing Columns

You can index up to four columns per table at one time.

Note

The following instructions apply ONLY to tables created with Visual CE. You can use the SYWARE utility "Index CDB," or a similar tool to index columns in Pocket Access tables, and "SSCE Utility" for SQL Server CE tables. Refer to the topic, "Visual CE Add-on Tools," on page 17 for instructions on how to obtain a free copy of these utilities.

If you are working in Visual CE's development environment on a desktop PC:

- 1 Close all forms on the CE device.
- 2 Do a soft reset of your CE device and reconnect it to your desktop PC.
- 3 Select the table whose columns you want to index. You can:
 - Open a form (.VCE application file) that is built on the table you want to modify. Select **File–Modify Table**. Click **Yes** in the message box. Visual CE automatically opens the Modify Table dialog box for the table upon which the form is built. Or,
 - Open a form (.VCE application file). While pressing the Shift key, select **File–Modify Table**. Click **Yes** in the message box. Select the table from the list that appears. Or,
 - With no .VCE application file open, select **File–Modify Table**. Click **Yes** in the message box. Then, select the table from the list that appears.

Visual CE displays the Modify Table dialog.

- 4 Visual CE allows up to four indexes at one time. If there are already four column names listed for "Indexes," and the desired column is not among them, you must remove one to make way for the addition. To do so, highlight one of the column names in the Indexes box, and click Remove Index.
- 5 Click the **Add Index** button.
Visual CE displays the Select Column dialog box.
- 6 Highlight the desired column, then click **OK**.
- 7 To prevent users from altering the indexes you've just specified, deselect the **User can change indexes** option. (Users will be unable to add or delete indexes in the application.) This useful if you have a macro that depends upon indexed columns, for example. **Note:** Because this setting is a property of the form, not the table, it is available only if the form associated with the table is currently open in the development environment.

If you are developing your Visual CE application on a CE device:

- 1 Save the form and table in the development environment on the CE device.
- 2 Select **File–Run** within the development environment on the CE device.
Visual CE opens the form in the application window.
- 3 Open the record list display: Select **Options–List**, click the **List** button , or press **Ctrl+L**.
- 4 On the top line, which displays the column names, click the column you want to index. Visual CE places the ^ symbol next to the column heading to denote that it is indexed and the current basis for search/order. Columns that are indexed but not currently used for sort/search order are indicated by the ^ symbol.
- 5 To close the application window and return to the development environment, select **Record–Close**.

Creating Pre-defined Filters for Record Display

A filter is a set of conditions that a record must satisfy in order for it to be included in the record display.

Working in Visual CE's development environment, you can create and save any number of filters as part of the form design. When users run the form, they can select any of the filters you've supplied, or they can create their own.

For each filter, you can specify one to three conditions, and you can specify whether the record must match at least one, or all, of the conditions.

Each condition consists of a column name, a value, and a comparison operator that specifies how a record's value in the column must compare to the value in the filter. The filter value may be a constant or a global variable.

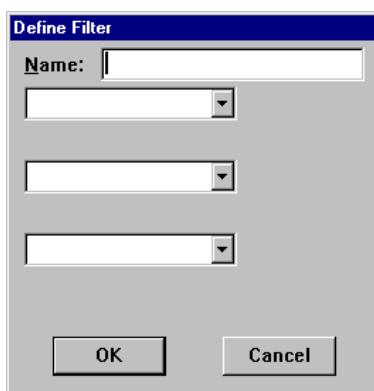
To create and save a filter in the Visual CE development environment:

- 1 Select **Edit-Filters**.

Visual CE displays the Filters dialog box.

- 2 Click the **Add** button.

Visual CE displays the Define Filter dialog box.



- 3 In the **Name** field, assign a short but descriptive name to your filter.

This name will appear as an item in the Visual CE application's menu, under **Option-Filter**.

- 4 Specify the first condition:

- Click the drop-down arrow for the first field, and select the column that contains the data you want to compare.
- In the field that appears to the right of the column name field, select a comparison operator. Not all comparison operators are available for every type of column.

Symbol	Comparison operator
=	Equal to
<>	Not equal to
<	Less than
>	Greater than
<=	Less than or equal to
>=	Greater than or equal to
Contains	Contains a specified text string; e.g., contains cook would find "John Cook," "Cooking," "Chocolate cookies," etc.

- Specify the comparison value – either a constant or a global variable. To specify a

constant, simply type it in the field. To specify a global variable, click the @ sign to the right of the field and select the variable from the list that appears.

Note

Text comparisons are NOT case-sensitive.

Tip

You can test for null values (i.e., empty fields). For the comparison operator, select = (Equal to); and for the comparison value, enter " (two single quote marks with no intervening space).

- 5 If desired, add one or two more conditions in the remaining fields using the same procedure.
- 6 If you supplied two or three conditions, choose one of the following:
 - **Match all** A record will be included in the display only if it satisfies **all** of the conditions
 - **Match at least one** A record will be included in the display if it satisfies **any one** of the conditions
- 7 Click **Close** to close the dialog box.

Security

Password-Protecting an Application

You can control access to an application by requiring users to enter a password.

Tip

If your Visual CE application is built on an SQL Server CE database table (.sdf file) that is password-protected, you can set up the application to either automatically supply the database password or prompt the user for it. Refer to the following topic.

To set up a password for your application:

- 1 Select **Form-Password**.

Visual CE displays the Password dialog box.



- 2 Enter the same password in both fields, and click **OK**. (Your entry will appear as a series of asterisks, to prevent others from seeing it as you type.)

When a user starts the application, they will be required to enter the password as you've specified it. Passwords are case-insensitive.

About Changing a Password for an Existing Application

To change a password for an existing application, you must use Visual CE's development environment.

Supplying Passwords for an SQL Server CE Database Table

If your form is built on an SQL Server CE database table (.sdf file) that requires a password, you can set up the Visual CE application to either automatically supply the password or prompt the user for it. This is in addition to password-protecting the Visual CE application itself.

To set up passwords for applications that use an .sdf file:

- 1 Select **Form-Password**.

Visual CE displays the Password dialog box.



- 2 Complete the Password dialog box as follows:

- **Password:** Optional. Password for the Visual CE application (not the .sdf file). Visual CE passwords are case-insensitive. If you specify a password here, the user will be required to enter the password to open the Visual CE application.
- **Confirm:** Reenter the password for the application just as you entered it in the Password field.

Database:

If the database table (.sdf file) requires a password, either:

- **Password:** Enter the password for the database table. This saves the user from having to supply it when the Visual CE application opens.
- **Prompt for password:** Alternatively, check this option and the user will be prompted for the database table password every time the Visual CE application opens.

If the database table is not password-protected, leave both **Password** and **Prompt for password** blank.

- 3 Click **OK**.

When a user opens the application, they will be required to enter the password(s) as you have specified.

Allowing Users to Insert and/or Delete Records

You can specify whether or not users can insert and/or delete records in the application.

To allow users to insert records:

- 1 Click **Form** in the menu bar.

The **Allow Inserts** option "toggles" on and off:

- If it is checked, users will be allowed to add new records to the table.
- If it is unchecked, users will not be allowed to add new records. (They can still modify existing records.)

- 2 To "toggle" the current setting, click the option.

To allow users to delete records:

- 1 Click **Form** in the menu bar.

The **Allow Deletes** option "toggles" on and off:

- If it is checked, users will be allowed to delete records from the table.
- If it is unchecked, users will not be allowed to delete records. (They can still modify existing records.)

- 2 To "toggle" the current setting, click the option.

Locking the Form Design

You can "lock" a form design to protect it against unauthorized or unwanted modification. This is especially useful if you distribute your Visual CE applications to other users and want to make sure that your design stays unchanged.

Note

Once you lock a form and save it, no one – including you—can open it in design mode. Be sure to make and store an unlocked copy before saving the locked version, in case you decide to modify the form later.

Alternatively – when using the Distribution Files Wizard to create installation files for distributing your Visual CE application (refer to “Creating Distribution Files (Pro and Mobile 360 Editions Only)” on page 298 – you can specify that the forms created in the install distribution folder be locked. In that case, you need not use the locking method described here.

To lock a form:

- Select **Form-Lock**.

Visual CE displays a message warning that the lock will be permanent.

Downloading, Testing, and Managing Applications

Saving an Application

As you create your application, you should periodically save your work to avoid losing it if your computer crashes.

To save the application, select **File—Save** or click the **Save** button .

Note

When you exit Visual CE, your changes to the application are saved automatically.

Saving a Textual Description of the Form

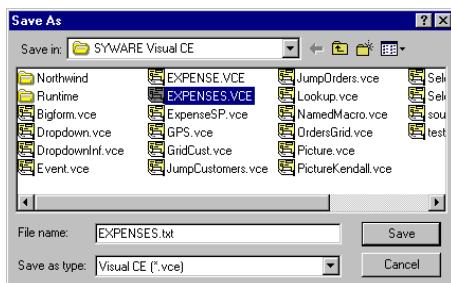
You can use Visual CE to quickly and easily create a text file that contains a comprehensive description of your form – including form features, table column names and datatypes, control dimensions, control order, macros, variables, and more. This description can be an invaluable record and development tool. For example, it can help you keep track of how and where variables are used.

Note

This feature is only available when using Visual CE's development environment on a desktop PC.

To create a textual description of a Visual CE form:

- 1 With the form open in the development environment, select **File–Save As**.
- 2 In the Save As dialog box that appears, navigate to the directory in which you want to save the description file.



- 3 Change the filename extension from **.vce** to **.txt**. Otherwise, you can name the file whatever you like, but it is recommended that you use the same name as the form for easy recognition. For example, if the form application is named **myForm.vce**, you might want to name it **myForm.txt**. (However, if you plan to import or export a comma separated value file into the table, be sure not to confuse this text file with the import or export file.)
- 4 Click **Save**.

Visual CE saves the description file. You can open it with Notepad or other text editor.

```
EXPENSES.txt - Notepad
File Edit Format View Help
Version: 11
Form is not password protected
Database: Windows CE Store
Table: Expenses (user can change indexes)
    Date (Date/Time indexed)
    Description (Text)
    Amount (Integer Number)
    Receipt? (Integer Number)
    Category (Text)
    Cost Center (Integer Number)
    Notes (Text)
    Submitted (Date/Time)
    memo (Memo)
    yesNo (Yes/No)
    picture (Picture)
Default menu bar
Title bar: Expense Tracker (show record count)
Sort on Date
Allow record insertions: On
Allow record deletions: On
Auto Recalc: Off
Background color: white
List View text color: black
Controls
    [1]: Date (Label)
        Left align
        Position: 4,29
        Size: 57x16
        Font: small
        black text with a white background
        Relative to frame rectangle: Off
```

Downloading an Application (Desktop PC Version Only)

When you have completed the application and have saved it to your desktop PC (using the Save command), download it to the CE device. Downloading causes Visual CE to save the application on the CE device, where you can use it.

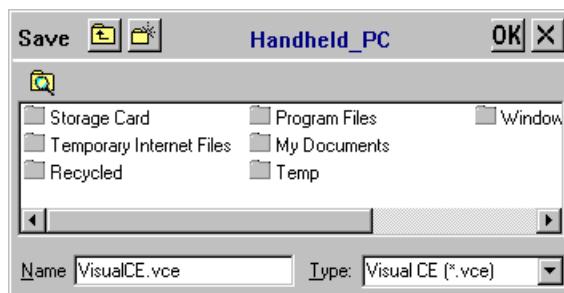
Note

The CE device must be connected to the desktop PC.

If your form uses a table created in Visual CE (or created in Pocket Access and stored in the Windows CE store):

- 1 In Visual CE on your desktop PC, select **File–Download Form**.

Visual CE displays the Save File dialog box.



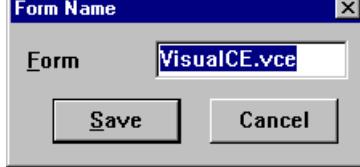
- 2 Navigate to the folder where you want to save the application. It can be anywhere on the CE device's file system, including a flash card.
- 3 In the Name field, replace the default "Visual CE.vce" with an identifying name. Be sure to keep the .vce extension.
- 4 Click **OK**.

Visual CE downloads the application to your CE device.

If your form uses a table created in Pocket Access or SQL Server CE and stored in a database volume (.CDB or .SDF):

- 1 In Visual CE on your desktop PC, select **File–Download Form**.

Visual CE displays the Form dialog box.



- 2 Replace the default "Visual CE.vce" with an identifying name for your form. Be sure to keep the .vce extension.
- 3 Click **Save**.

Visual CE downloads the form to your CE device.

Note

If your form is connected to a .CDB or .SDF file, both the form and the .CDB or .SDF file must be kept in the same folder on the CE device. When you download the form, Visual CE automatically puts the form in the correct folder for you. If you later decide to move the application, be sure that you keep the form and the .CDB or .SDF file together.

Testing the Application

Before using the application in "real life," you should test it thoroughly to be sure it works as you intended. You can change your application easily at this point.

To test an application:

1 Open the Visual CE application:

- **If you have an H/PC or desktop PC:** Double-click the application file icon. (The Visual CE application icon  is displayed above the filename.)
- **If you have a P/PC:** Click the Windows **Start** button, then **Programs**, then the application.

Tip

If you are developing your application on the CE device, you can launch the application from inside the development window by selecting **File-Run**.

Visual CE displays your application in its application window.

2 Create several records to be sure the application works correctly.

3 If your application works correctly, you are ready to use it. If not, you have probably made a simple mistake or two.

To correct any mistakes, close the application and return to the Visual CE development window. If you made a mistake with a control, change the control's properties or delete and recreate the control. If your mistake concerned the title bar, sort/search order, or tab order, respecify the option correctly. After making your changes, click the **Save** button to save your work. If you are working on the desktop PC, download the modified application to the CE device by selecting **File—Download Form**. Open the application and retest.

Note

If you left the old version of the application open while making and downloading your changes, be sure to close and reopen the application before evaluating it. Otherwise, you are still looking at the old version.

Opening an Existing Application

To open an existing application in Visual CE's development environment to view it or make changes:

- 1 Select **File–Open** or click the **Open** button .
Visual CE displays the Open dialog box.
- 2 Navigate to the folder in which you have saved your Visual CE application – change it if another folder is currently displayed.
- 3 Double-click the Visual CE application filename, or click the filename once and then click the **Open** button.
Visual CE displays the selected application in its application development window.

Modifying an Existing Application

All changes to an existing application must be made in Visual CE's development environment.

You can make the following types of changes to an existing Visual CE application:

- Adding new controls
- Changing an existing control's properties
- Deleting and recreating a control
- Respecifying the application title
- Defining a new tab, column, and control order
- Defining a new initial sort/search column
- Changing the password
- Associating a different table with the form

After making your changes, click the **Save** button to save your work, then (if working on the desktop PC) download to the CE device by selecting **File—Download Form**.

Setting Up Visual CE Applications to Run on a Desktop PC

Typically, you will create Visual CE applications on a desktop PC and run them on a CE device. In some instances, though, you might want to run an application on the desktop PC. You can do this, but it requires some additional setup.

There are two key issues: file synchronization and table synchronization.

File Synchronization: The first issue is file pathname recognition. Visual CE keeps track of a Visual CE application and any files it needs to work in the way you've designed it – such scribble box backgrounds or "jump to" applications – using the file system on the CE device. However, when the form runs on the desktop PC, it looks for these files on the desktop PC. Consequently, you must put these files in the correct place in the desktop PC file system. You can accomplish this using Microsoft's ActiveSync File Synchronization, as explained in the instructions below.

Table Synchronization: When the form runs on the desktop PC, it reads and writes data to a table on the desktop PC. (This actually includes the form's application table, and any support tables used in your design such as a table containing lookup values or a word list.) In order to use your application on the desktop PC, you must synchronize all of the tables it uses on the desktop PC with their counterparts on the CE device. You'll use Visual CE's synchronizer to do this.

Following is a step-by-step description of the process for both files and tables.

To set up file synchronization:

- 1 Create a mental checklist for yourself (or put it on paper if it gets too long) of all the files used by your Visual CE application.

At a minimum, you'll have the .vce application itself. But you might also have support files such as:

- scribble control background image (.bmp file)
- files referenced by image controls
- .vce file associated with a jump button or grid control
- target application for a "Run" command button, including any files that program uses
- target Report CE report for **File – Print** (.rce file)
- target file for "Selection on Click"

- 2 When creating your application, all files used must be stored in the Microsoft ActiveSync file synchronization folder or one of its subfolders.

- 3 With the CE device connected to the desktop PC, run Microsoft ActiveSync file synchronization.

The files will be synchronized to the desktop PC.

For example, say that you have designated the following folders for ActiveSync file synchronization: \My Documents on the CE device and C:\SyncFolder\Pocket_PC on the desktop PC. Further, imagine you have a form called MyForm.vce that uses MyBackgrd.bmp for a scribble box.

On the CE device	On the desktop PC
\My Documents\MyForm.vce	C:\SyncFolder\Pocket_PC\MyForm.vce
\My Documents\Graphics\MyBackgrd.bmp	C:\SyncFolder\Pocket_PC\Graphics\MyBackgrd.bmp

- 4 If you are using Windows Vista or Windows 7, please refer to the November 2009 entry on: <http://www.syware.com/totm>.

To set up table synchronization:

- 1 Create a checklist of all of the tables used by your application.

You'll have at least one item--the application table for the form. You might also have "jump-to" tables or tables that contain look-up values, drop-down values, grid rows, or word lists.

Note

The table(s) on the CE device must have been created with Visual CE; Pocket Access tables in a .cdb file or SQL Server CE tables in a .sdf file will not work.

- 2 For each table you listed in Step 1, you must have a desktop PC version. (The Visual CE application on the desktop PC works against data in tables on the desktop PC.) Set up Visual CE synchronization for each table using the instructions in the topic, "Selecting Table Synchronization Options," on page 435.

Note

The desktop database (for example, the .mdb or .accdb file) need not be in the desktop PC file synchronization folder, nor any of its subfolders. In fact, in most cases, you would not want it to be in any of those folders.

- 3 Run the Visual CE synchronization software to synchronize the table(s) on the handheld with those on the desktop PC.

You can now run your Visual CE application on either the desktop PC or CE device in disconnected mode.

Distributing Applications

Creating Distribution Files (Pro and Mobile 360 Editions Only)

If you purchased Visual CE Pro or Mobile 360, SYWARE grants you the right to distribute applications that you have created to third-parties. The Visual CE Personal Edition license does not allow redistribution.

Using the Distribution Files Wizard that is part of the Mobile 360 and Pro Editions, you can quickly and easily create a complete set of installation files for your users – including not only the main form and CE table with indexes, but also any related forms and tables, embedded images such as a logo, a desktop PC database with data, synchronization settings, and a setup.exe. You can distribute these files to your users on a CD or other media including CF and SDIO cards. To install your application on their CE device and desktop PC, all your users have to do is connect the two machines and run setup.exe on the PC.

The later part of this topic explains how to use the wizard to create the distribution files. But before you run the wizard, you must do some preparation, as described below.

Getting Ready

Before creating the installation disks, you must create and thoroughly test the application. If you want to include a desktop PC database with or without data records as part of your design, you must also set up the database and synchronization settings. Here are some tips and guidelines:

- **Test the form**

Get your application working on the CE device. Test it thoroughly.

- **Setup Access (.mdb or .accdb) desktop PC database files**

If your application synchronizes to an Access database (.mdb or .accdb file), you can include it as part of the application installer. If you choose, the installer will put the file on the user's desktop PC and synchronize it with table(s) on the CE device (creating the tables if need be).

In addition, if you want to provide data with your application, you **must** put it in the .mdb or .accdb file on your desktop PC and select the wizard option that includes it in the installation files. During installation, the installer software will automatically create a table on the CE device and populate it with the data from the .mdb or .accdb file.

Before you run the installation wizard, be sure that the OIDs for all records in the .mdb or .accdb file on your desktop PC are set to zero or null (blank).

- **Synchronization settings**

The installation disk you create will install the synchronizer software and the synchronization settings. Specify the synchronization settings using the Visual CE development environment on your desktop PC (see page 435).

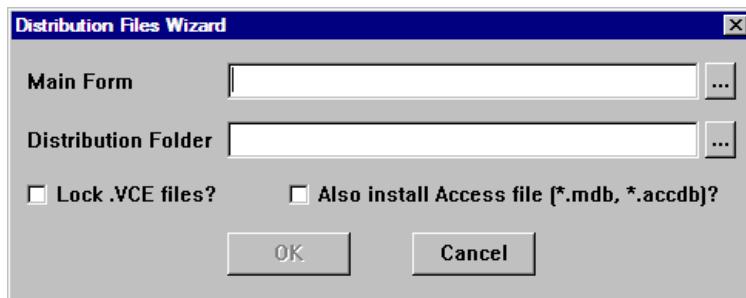
Important: If you want to copy data from the .mdb or .accdb file onto the user's CE device, be sure that the "Synchronize on Install" option is enabled in the synchronization options (see page 437). This option will create a table on the user's CE device and immediately synchronize it with the contents of the .mdb or .accdb file on the desktop PC.

You should now be ready to create the installation files.

To create an installation disk (or other media) for your users using the Distribution Files Wizard:

- 1 Connect your CE device to your desktop PC.
- 2 Open the Visual CE development environment on your desktop PC: Click the Windows Start button, then select **Programs-Visual CE-Visual CE**.
- 3 Select **File – Create Distribution Files**.

Visual CE displays the Distribution Files Wizard.



- 4 Click the Browse button for **Main Form**, and navigate to the folder on the CE device that contains the main form (.vce file). Select the form and click **OK**.
- 5 Click the Browse button for **Distribution Folder**.
Visual CE displays the Save As dialog box. Navigate to – or create – a folder on the desktop PC where Visual CE will put the installation files. To create a new folder now, navigate to the place in your file system where you want the new folder, click the New Folder Icon in the Save As dialog box, name the folder, and finally double-click the folder to open it. Once you have opened the desired folder, click **Save**.
- 6 *Optional.* If you want to include an Access desktop PC database in the installation, select **Also install Access file (*.mdb, *.accdb)?**. If you select this option, Visual CE will automatically include the .mdb or .accdb file that you specified in the application's synchronization settings as part of the distribution files.
- 7 If you want to lock the design of all forms included in the Distribution Folder against modification, select **Lock .VCE files?**.
- 8 Click **OK**.
- 9 After a short time, Visual CE displays the message, "Creation of distribution files complete."
- 10 To create an installation disk for your users, simply copy the contents of the folder that you specified in Step 5 to a CD, disk, or other media.

To install your application, your users need only:

If installing via a desktop PC:

- 1 Connect the CE device and desktop PC.
- 2 Insert the CD in the drive (or copy the installation folder onto the PC hard drive).
- 3 Run setup.exe.

If installing via a CF or SDIO card directly onto the CE device:

- Insert the card in the device slot.
Setup starts automatically.

Part III:

Macros and Events

Macros and Events

Overview of Part III

If you took Quick Tours IV and V, you are familiar with the basics of how macros work, and how to create and debug them. This section contains detailed reference information about the Visual CE Macro Builder, macro commands, events, and other aspects of creating and applying macros, and well as the Visual CE Macro Debugger.

The following topics are discussed:

- Macros and Events: An Overview
 - ◊ Introducing Macros and Events
 - ◊ A Sample Macro Script
 - ◊ Steps in Creating a Macro
- Using the Macro Builder
 - ◊ Starting the Macro Builder
 - ◊ Creating a New Macro
 - ◊ Editing or Deleting a Macro
 - ◊ Converting a Button Macro
- Creating a Way for Users to Run the Macro
 - ◊ Command Buttons
 - ◊ Custom Menu Pick
- Triggering Macros with Events
 - ◊ About Events
 - ◊ Control Events
 - ◊ Form Events (including hardware buttons on the CE device)
- Macro Command Reference
 - ◊ Commands Organized by Category
 - ◊ Commands Organized Alphabetically
- Using the Macro Debugger

Macros and Events: An Overview

Introducing Macros and Events

What is a Macro?

A macro is a script containing a sequence of commands that can be executed with a single command button, menu option, hardware button, or event.

Macros are useful for customizing your forms in any number of ways. For instance, you can save time and effort for your users by defining a series of steps that execute automatically with a single click. You can use a macro to automatically validate user input before it is entered in the database. You can automate complex surveys with many branches using conditional (if-then-else) logic. You can even make your forms dynamic by having macros execute automatically when a given event occurs, such as when the form opens or the user changes a field value or rotates the device.

Examples of Macros

Quick Tour IV at the beginning of this guide showed you how to create a simple macro that dynamically modifies controls on the form according to user input (specifically, the user's selection in a drop-down list determines what checkbox options are displayed on the form).

Here are just a few more examples to illustrate what can be accomplished with macros:

- You could create a "Submit" macro that, when the user clicks a Submit button, automatically saves the current record with a timestamp and the name of the device that created it. The macro would consist of three commands:
 - Step 1: An "Assign" command writes the name of the Windows CE device to a column in the table using the @devicename function.
 - Step 2: An "Assign" command writes the current date and time to a column in the table using the @now function.
 - Step 3: A Record | Save command saves the entire record to the table.
- You could build a questionnaire in which the questions branch in different directions based on the user's responses. For each branch, you could have a macro consisting of two buttons:
 - Step 1: An "Assign" command stores a value based on the user's input.
 - Step 2: A "move to when pressed" command moves to the region of the form that contains the appropriate set of questions based on that response.
- You could apply a validation test to user input (for example, age must be greater than 0). If the response is valid, the macro saves the record; if the response is invalid, it displays an error message. This macro could consist of four commands:
 - Step 1: A "skip" command evaluates the user input: If age is greater than 0, it skips two commands in the sequence; otherwise, it skips zero commands in the sequence.
 - Step 2: A "message" command displays an error message, "Age must be greater than 0."
 - Step 3: A "return from macro" command stops the macro.
 - Step 4: A Record | Save command saves the record to the table.

What are the Components of a Macro?

A macro consists of a sequence of commands that you want performed in a given order.

Visual CE offers more than 50 commands you can use as steps in a macro. Each command type is explained in the command reference section beginning on page 323.

Most commands have parameters you must specify, such as values read into or written out by the command. For these values, you can supply a constant, or the same type of expression used in a calculated field control. Expressions can read values stored in a table column for the current record or in a global variable, as well as apply mathematical operations, if-then-else logic, and functions to return a desired value. Refer to page 201 for details about expressions.

You can also use global variables to pass temporary values between steps in a macro. Refer to page 153 for more information about global variables.

The script commands can execute in a straight linear sequence, or you can incorporate skips, branches and loops based on conditional logic using Skip commands. You can also have submacros (sub-routines) within a macro. Submacros are a sequence of commands that you write once, and can then call from the main script as many times as needed. Simply write the submacro as you would any other, then call it from the main script using a "Run macro" command.

One Visual CE application can have any number of macros. However, each macro is "built into" a single Visual CE application; it is not possible to call an external macro.

How are Macros Run?

Macros can be run in a Visual CE application in either of two ways: they can be run when the user explicitly selects a button or menu item that you have associated with the macro, or they can be run automatically when an event occurs (see below).

To associate a macro with a menu item or command button, refer to page 319.

To associate a macro with an event, refer to page 320.

What is an Event?

Events are generally (but not always) actions initiated by the user. Visual CE recognizes two categories of events: form events and control events. Form events include opening the form, displaying a record, selecting a different tab, leaving a record, pressing a hardware button on the CE device, and rotating the device. Control events include "get focus" (the user clicks or tabs to a control), "Lose focus" (the user clicks or tabs away from the control) and "changed" (the user adds, edits or deletes a value in the control).

As explained above, you can associate a macro with an event, so that when the event occurs, the macro runs automatically.

Events are explained in detail on page 320.

How Do You Create a Macro?

The Visual CE Macro Builder is a structured development environment that has been designed to help guide you through the process of creating a macro. For example, to define each step in your macro, you simply choose from the drop-down list of commands, then complete a few input fields to set any additional properties of the command. You use the Macro Builder to arrange the commands in the order you want them to execute.

The Macro Builder also enables you to assign the macro to a hardware button or other form event.

The Macro Builder is described in detail beginning on page 309.

If you have a named macro created with a previous version of Visual CE you can easily convert it for editing in the Macro Builder. Converting legacy button macros is described on page 317.

Once you have created a macro, be sure to test it. If it doesn't perform as expected, you can use the Visual CE Macro Debugger to identify errors in the macro script.

Putting it All Together

The following page provides a sample macro script that illustrates many of the techniques you can use in your own macros. In addition, be sure to check out the SYWARE website regularly, in particular the Tip of the Month section, <http://www.syware.com/totm>, for more sample macros and ideas.

A Sample Macro Script

The following macro script sets the value in the Amount column to zero for all records in a table. It demonstrates the use of global variables to pass temporary values from one step to the next, as well as the use of Skip commands with If-Then-Else logic that loop through the statements until all records have been processed.

Step	Command	Command parameters	Comment
1	Record First		Go to first record in the table
2	Assign	Value: @count Col: @var(0)	Count the number of records in the table using @count and write the result to the global variable @var(0)
3	Macro step label	Label: StartOfLoop	
4	Skip	Num: If @var(0) = 0 then go to Done	If no more records to do, skip ahead to the step labeled Done; otherwise, continue
5	Assign	Value: 0 Col: Amount	Write the value 0 to the Amount column for the current record
6	Assign	Value: @var(0) - 1 Col: @var(0)	Subtract 1 from the record count and write the result to @var(0)
7	Record Next		Go to the next record in the table
8	Skip	Num: Go to StartOfLoop	Go back to the step labeled StartOfLoop.
9	Macro Step Label	Label: Done	
10	Message box	Msg: 'All amounts set to zero'	Display message for user.
11	Return from macro		Return processing control to the form.

Steps in Creating a Macro

The sequence for creating a macro is:

- Plan the macro
- Create the Visual CE application that will run it, along with any dependencies -- such as filters or indexes -- or external programs that are called in your macro.
- Write the macro using the Macro Builder
- Provide a way for the macro to run
- Test and, if necessary, debug the macro

Plan the Macro

You may find it helpful to plan your macro before writing it. For example:

- What task(s) do you want your macro to perform?
- If the macro processes table records, should the records be sorted or filtered as part of the macro?
- Are there sequences of steps that repeat? If so, can you put them in a submacro?
- How will your macro be launched? Will it be tied to a command button, menu option or hardware button that the user clicks, or should it run automatically when a form or control event happens?

Once you've decided what the macro should do, it's a good idea to write out each of the steps in English or "pseudocode" before actually writing the commands in the Macro Builder.

Create the Visual CE Application

Create the form and table as you would any other application. If any of the macro steps will use a sort or filter, you must index the sort column or define the filter first. Similarly, if the macro calls any external applications, it's a good idea to verify that they exist in the expected location.

Write the Macro Using the Macro Builder

With the application open in the Visual CE development environment on the PC, open the Macro Builder (explained on page 309). The Macro Builder provides a helpful framework for writing and organizing the commands that make up the steps in the macro.

Provide a Way for the Macro to Run

You can have the macro run in any of the following ways:

- Create a command button or menu pick that the user selects (see page 319).
- Assign one of the hardware buttons on the CE device to the macro (see page 320).
- Assign a form event to trigger the macro automatically (see page 320)
- Assign a control event to trigger the macro automatically (see page 321)

Save and Test the Macro

Macros are saved when you save the Visual CE application to which they belong.

Download the application to the CE device and test it thoroughly before using it in production.

Debug the Macro

If the macro doesn't behave as expected, you can use the Visual CE Macro Debugger to isolate and identify the step(s) that are causing errors.

The debugger enables you to “see into” the macro while it is running. You can set breakpoints that tell the debugger to pause execution when it reaches a specific step, so that you can examine the value of a variable or a column in the current record at that point in the execution. You can continue execution one step at a time, or until the next breakpoint.

Using the Macro Builder

Starting the Macro Builder

The Macro Builder is a development environment inside the larger Visual CE development environment that provides a helpful framework for creating, editing, and managing macros and associating them with hardware buttons or form events.

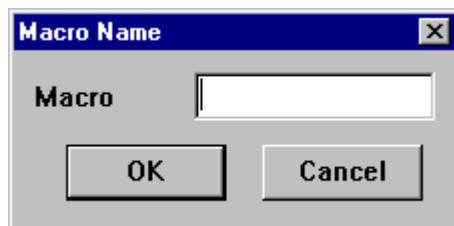
Note

The Macro Builder is available only on the desktop PC.

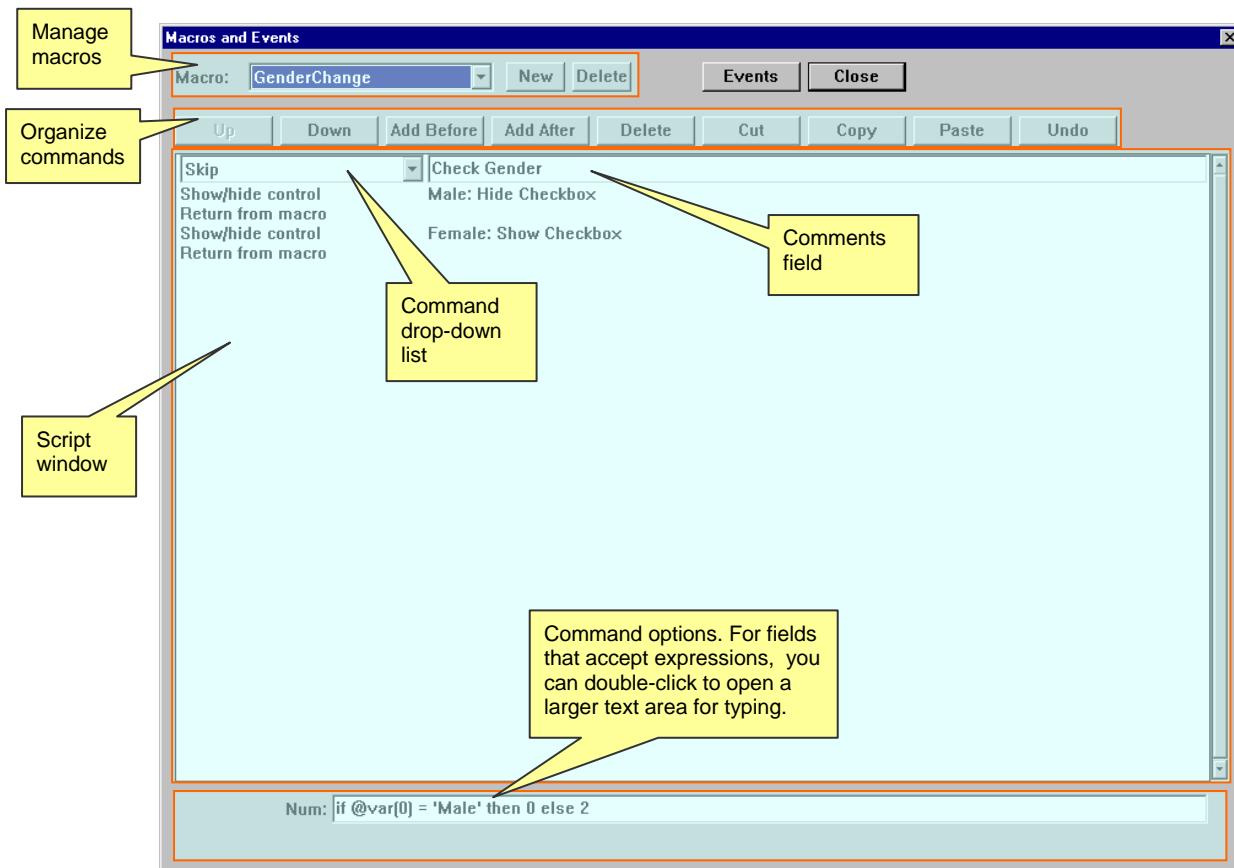
To start the Macro Builder:

- 1 Working in the Visual CE development environment on the desktop PC, open the Visual CE application in which you want to include the macro.
- 2 Select **Edit – Macro/Events**.

If you have previously created a macro as part of this application, Visual CE opens the Macro Builder. Otherwise, Visual CE displays the Macro Name dialog box.



- 3 Enter a meaningful name for your macro that you will be able to recognize later, and click **OK**.
Visual CE opens the Macro Builder. Here you can build your macro scripts and associate them with form events and hardware buttons.



The *Manage macros* area enables you to add a new macro, or select an existing macro to modify or delete:

- The **Macro** drop-down list contains an alphabetical list of all macros that you've created in the current Visual CE application so far.
- The **New** button lets you begin the process of creating a new macro.
- The **Delete** button deletes the macro currently displayed in the Macro drop-down list.

The *Organize commands* area provides buttons you can apply to the currently selected command in the Script window:

- The **Up** button moves the currently selected command up one step in the script.
- The **Down** button moves the currently selected command down one step.
- The **Add Before** button adds a new command to the script, just before the currently selected command.
- The **Add After** button adds a new command to the script, just after the currently selected command.
- The **Delete** button removes the currently selected command from the script.
- The **Cut** button removes the currently selected command(s) from the current location so that you can paste them elsewhere in the script.
- The **Copy** button copies the currently selected command(s) so that you can paste them elsewhere in the script or in another macro.

- The **Paste** button inserts cut or copied command(s) just above the currently selected command.
- The **Undo** button undoes all modifications you've made to a command since you last selected it. If the command is new, Undo effectively deletes it.

The *Script window* lists all of the commands that make up the steps in your macro script.

When you click any command line in the Script window, the Macro Builder displays the following:

- The **Command drop-down list** enables you to choose one of Visual CE's more than 50 pre-defined commands.
- The **Comments field** lets you enter a description of your command.
- The **Command options** area displays fields for entering parameters specific to the command selected in the Command drop-down list. For fields that accept expressions, you can double-click the field to open a larger text area in which to more easily add or edit long expressions.

The **Events** button opens the Events window, in which you can associate any existing macro with a form event. (To associate a macro with a control event, set the Events property of that control.)

The **Close** button closes the Macro Builder window and *temporarily* saves the current version of the macro. You must save the Visual CE application to save the macro.

The following topics explain how to create, edit, and delete macros using the Macro Builder.

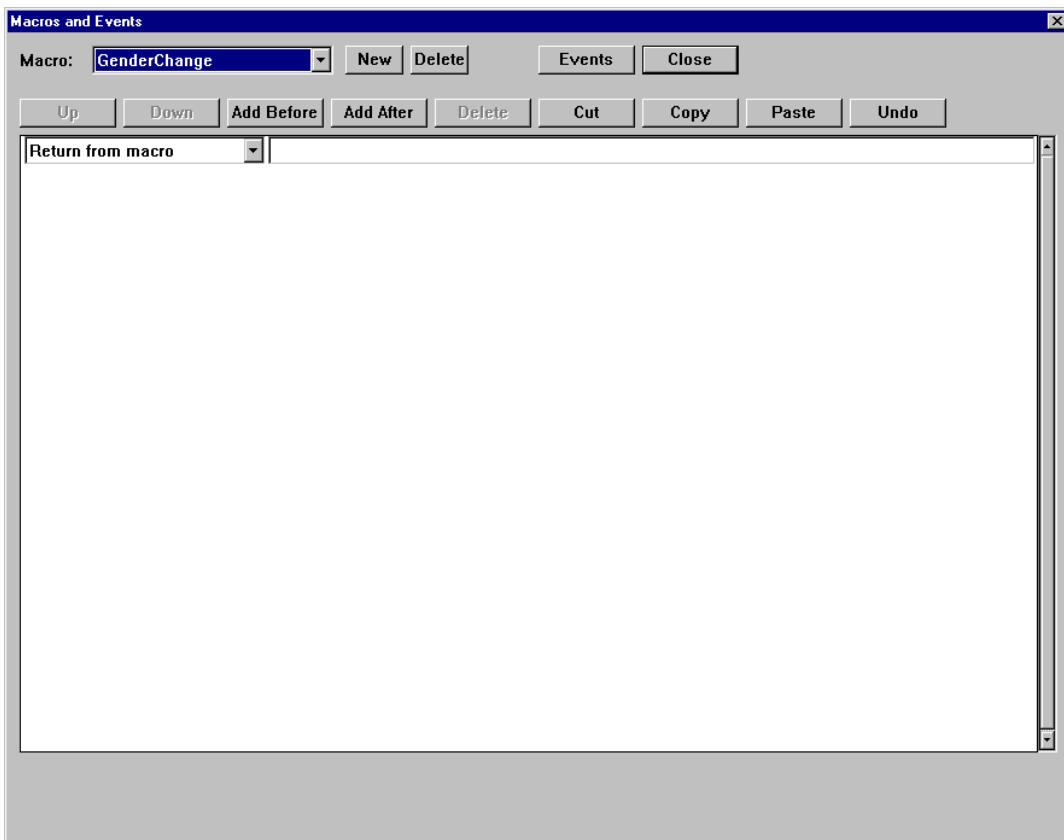
Creating a New Macro

The Macro Builder makes it easy to create a macro.

To create a new macro:

- 1 Open the Macro Builder using the instructions in the previous topic.
- 2 Unless you named your new macro when first opening the Macro Builder, click the **New** button and name your macro now.

The name of the macro you are creating should be visible in the Macro drop-down list at the top-left of the Macro Builder window.



3 Click **Add Before**.

The Macro Builder adds a new command to the top of the Script Window – above the default Return from Macro statement. It is considered good practice that the last statement in your script be a Return from Macro statement, so you should leave it in place.

4 To add a new command to the script:

- ◊ Open the Command drop-down list and select the desired command type.
The list is extensive; you may have to scroll to find the command you want.
- ◊ Once you select a command in the drop-down, Visual CE displays one or more input fields at the bottom of the Macro Builder window. Complete these fields to specify the parameters specific to the selected command type. You can double-click an input field to open a larger area in which to type.

Tip

All of the available command types, and their optional parameters, are described in detail in the section, “Command Reference” beginning on page 323.

- Enter comments in the field to the right of the command. These comments can be very helpful if you need to modify or debug your script in the future. Also, the Script Window displays only the command name, not the specific parameters, so it can be difficult to tell one command from another without either comments or selecting each one to view its parameters.
- 5 Once you have created each one of the commands in your script, put them in the order you want them to execute. To move a command up or down in the sequence, select it and click the **Up** or **Down** button.

Tips

- For more complicated scripts, you may want to change the flow of execution (the sequence in which the commands are executed) to something other than a straight linear sequence. You can incorporate “If-Then-Else” or even “If – Then – Else If – Then – Else” branches and loops using Skip statements. You can also break out statements by putting them in submacros. This is useful if you have a set of statements that are used repeatedly in the macro, for example. Rather than rewriting the same set over and over, you can write them once, save them as a separate macro in the application, and call them when needed using a Run Macro command in the main macro.
- You can also cut, copy, and paste commands within the current script or to and from another macro script. Refer to the topic, “Applying Cut, Copy, and Paste to Macro Commands” on page 314.

- 6 When you have completed your macro, you can close the Macro Builder by clicking the **Close** button.
- 7 To save the macro, you must save the application. Select **File—Save**.

Warning!

If you close the Visual CE development environment without saving the application, your macro will be lost.

Applying Cut, Copy, and Paste to Macro Commands

You can cut and paste one or more commands within the current script, or to and from another macro script.

To cut/copy/paste in the Macro Editor:

- 1 Select the commands you want to cut or copy. If you want to select multiple commands, click the first command to highlight it. Then, while holding down the Shift key, click the last command. Or, you can simply drag the mouse pointer over the desired commands (this works best if you start with the last command and drag upwards). If you change your mind and want to deselect the commands, click the **Undo** button.
- 2 Click the **Cut** or **Copy** button.
- 3 If you want to insert the command(s) in another macro, open that now.
- 4 Select the command that you want to be immediately after the commands pasted in.
- 5 Select the **Paste** button.

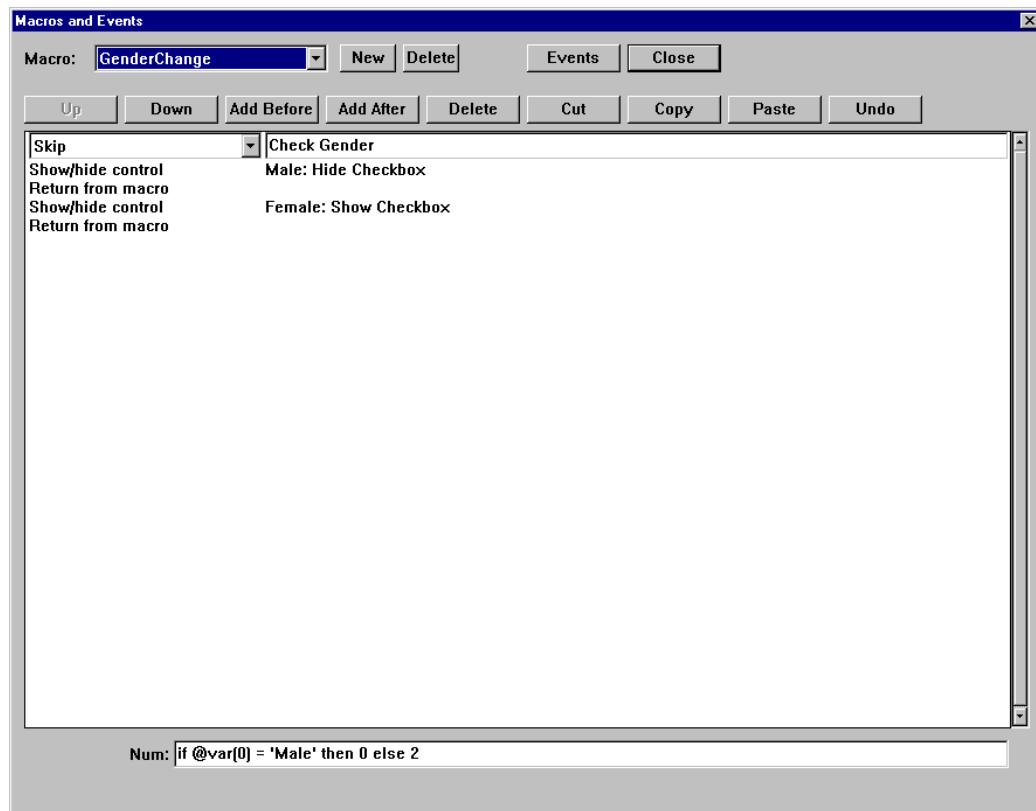
Editing an Existing Macro

The following instructions explain how to edit a macro that was originally created using the Macro Builder. If you wish to edit a legacy button macro, refer to the topic "Converting a Legacy Button Macro to a Buttonless Macro" on page 317.

To edit an existing macro:

- 1 Working in the Visual CE development environment on the desktop PC, open the Visual CE application that includes the macro you wish to edit.
- 2 Select **Edit – Macro/Events**.

The Macro Builder opens.



- 3 From the **Macro** drop-down menu, select the name of the macro you wish to edit.
- 4 Edit the commands and command parameters in the Script Window using the editing features of the Macro Builder (as described in the topic "Starting the Macro Builder" on page 309).
- 5 When you have completed your macro, close the Macro Builder by clicking the **Close** button.
- 6 To save your changes to the macro, you must save the Visual CE application. Select **File–Save**.

Important!

If you close the Visual CE development environment without saving the application, your edits to the macro will be lost.

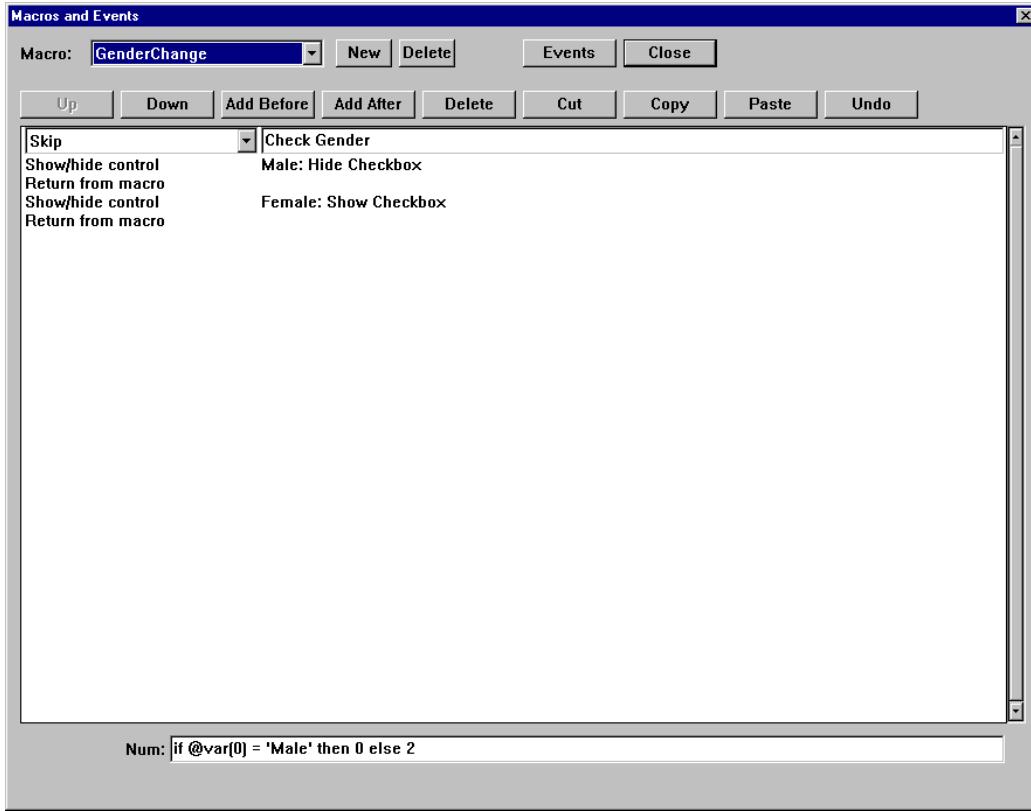
Deleting an Existing Macro

The following instructions explain how to delete a macro that was originally created using the Macro Builder.

To delete an existing macro:

- 1 Working in the Visual CE development environment on the desktop PC, open the Visual CE application that includes the macro you wish to delete.
- 2 Select **Edit – Macro/Events**.

The Macro Builder opens.



- 3 From the **Macro** drop-down menu, select the name of the macro you wish to delete.
- 4 Click the **Delete** button at the top of the Macro Builder window.

Important!

You must save the Visual CE application for any changes to macros to take effect. If you close the Visual CE development environment without saving the application, the macro will not be deleted.

Converting a Legacy Button Macro to a Buttonless Macro

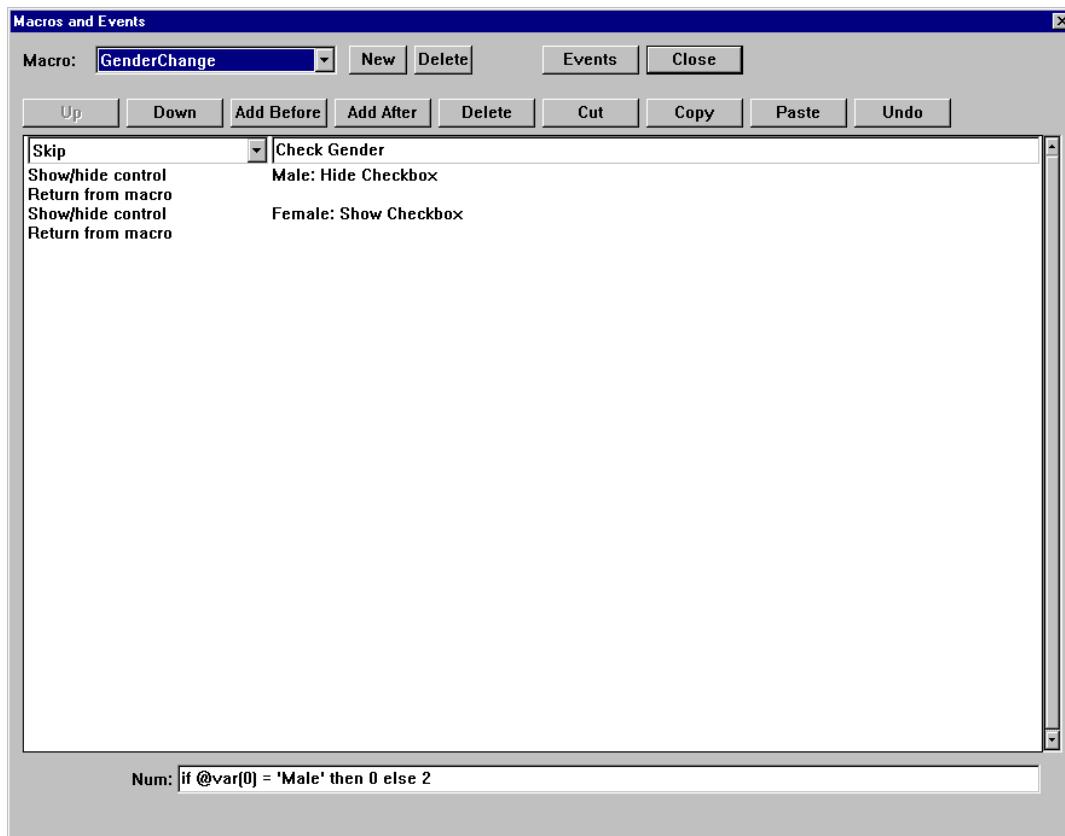
As of Visual CE version 10, the method of creating a macro by assembling a series of Command, Timestamp and Jump buttons is still supported but discouraged in favor of using the Macro Builder.

If you have an old Visual CE application with a button macro, it still works. If you wish to edit that macro using the Macro Builder, you can easily convert it to the buttonless script version using the following instructions. Keep in mind that only named button macros can be opened and converted with the Macro Builder. If the button macro is not named, it cannot be converted.

The Macro Builder converts each button in the macro to a command in the script, and removes the button from the form.

To convert a named macro:

- 1 Working in the Visual CE development environment on the desktop PC, open the Visual CE application that includes the macro you want to convert.
- 2 Select **Edit – Macro/Events**.
The Macro Builder opens.
- 3 In the Macro drop-down list, select the macro you wish to edit. (If your macro is missing from the list, it is not a named macro.)
The script version of the macro is displayed in the script window.



- 4 Carefully review each of the commands in the script, with particular attention to those at the end.

In a button macro, any of the following designates its end:

- Start of Macro command button (denotes the beginning of another macro)
- Any control other than a Command, Jump, or Timestamp button
- The end of the control order

If the Macro Builder has added more commands to the script version than desired, click the **Close** button. Then, insert a dummy Label control on the form, and place it in the control order where it will mark the desired end of the macro sequence.

- 5 If you are satisfied with the commands as they appear in the script window, click **Convert to button-less** macro near the top of the Macro Builder window.
- 6 Click **Yes** when Visual CE asks you to confirm the conversion.
You can now edit the macro script using any of the features of the Macro Builder.
- 7 When you have completed the edits to the macro, click **Close**.
Notice that Visual CE has removed the macro buttons from the form.
- 8 If you added a dummy control in Step 4 to mark the end of the button macro, be sure to remove the control from the form.
- 9 To save the new version of the macro, save the Visual CE application by selecting **File – Save**. If you do not save the application, your changes to the macro will be lost.

Creating a Way for Users to Run the Macro

You can provide a control on the form that users click to run a macro. Two options are available:

- A command button on the form (see page 225)
- A custom menu pick on the form (see page 274)

Alternatively, you may want the macro to execute automatically when an event occurs. Refer to the following topic.

Triggering Macros with Events

About Events

Using events, you can create forms that respond to the user dynamically. You can associate a macro with an event, so that when the event occurs, the macro runs automatically.

For example, you could set up a validation macro to automatically validate a price in the Amount field before the user leaves the record, or a macro that shows additional controls on the form when the user selects a checkbox, or a macro that displays a splash screen when the application first opens.

Tip

You can see events in action by running the SYWARE-supplied sample application, Event.vce (For instructions on installing the samples, please refer to the topic, "Installing Visual CE," on page 15). The samples work only on the CE device, not the desktop PC.

Event.vce illustrates how you can show or hide controls on the form depending upon user input. In this example, the checkbox "Pregnant" is available only if the user chooses "Female" from the drop-down list. The drop-down control's properties contain an event attribute that calls the "GenderChange" macro when the user changes the value.

There are two categories of events: control events and form events. Control events occur when a control on the form (e.g., radio button, checkbox, etc.) gets or loses focus or is used to change a value. Form events include opening the form, selecting a different tab, displaying a record, or leaving a record; when the user presses a hardware button; or when the user rotates the device.

Using Control Events to Trigger a Macro

Visual CE recognizes three types of control events:

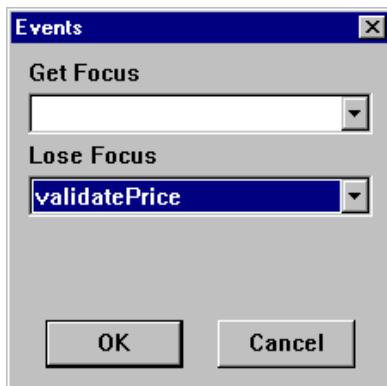
Event	Definition
Get focus	User clicks or tabs to the control, or the Set Focus command puts focus on the control.
Lose focus	User clicks or tabs away from the control, or the control loses focus because of the Set Focus command.
Changed	User adds, edits, or deletes a value in the control. (Assign commands do not trigger a Changed event.)

You define event responses as properties of individual controls. The table below summarizes the events you can use for each control type:

Control	Events		
	Get focus	Lose focus	Changed
Edit box	X	X	
Note box	X	X	
Checkbox	X	X	X
Radio buttons	X	X	X
Drop down list	X	X	X
Scribble box	X	X	
Image control	X	X	
Grid control			X

To set up control events to trigger macros:

- 1 In the main Visual CE development environment, open the Properties dialog box for the control that will trigger the macro. Click the **Events** button. Visual CE displays the Events dialog box.



- 2 From the drop down menus, select the macro you want to associate with each event for the control. You can assign a macro to any or all of the events, depending upon your design.
- 3 Click **OK**.

Using Form Events to Trigger a Macro

Visual CE recognizes several types of form events:

Event	Definition
Hardware button	The user presses a button or rocker switch on the CE device.
On record display	A record is opened.
On leaving record	User goes to another record. This event could be useful for triggering a validation macro to check that input values are within an acceptable range, for example.
On startup	The application opens.
On rotate	The user rotates the device; changing the orientation from landscape to portrait, for example.
On select tab	The user selects a different tab on the form.

To set up form events to trigger macros:

- Unless the Macro Builder is already open, select **Edit – Macro/Events**.

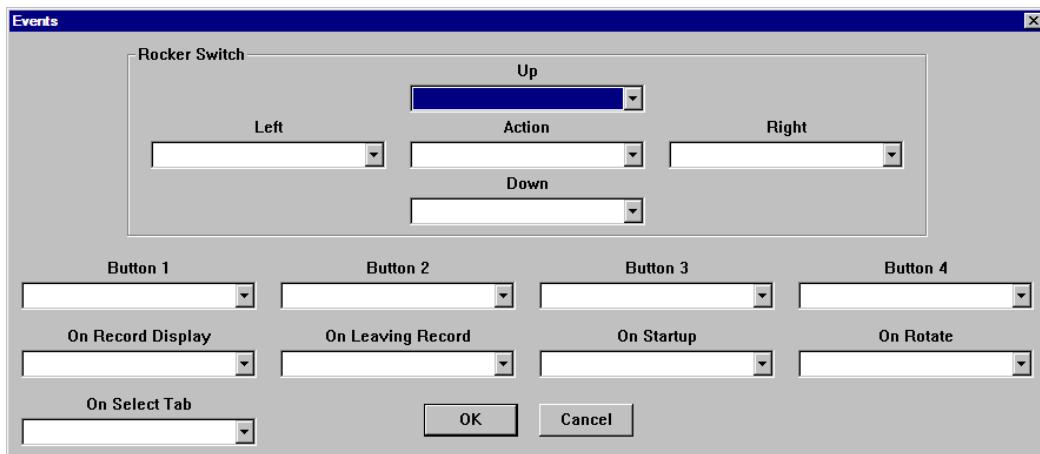
The Macro Builder opens.

Note

If Visual CE displays the Macro Name dialog box when you first open the Macro Builder, you have not yet created a macro for the current Visual CE application. Go to the instructions, "Creating a New Macro" on page 312 for instructions.

- Click the **Events** button near the top of the Macro Builder window.

The Events dialog box opens.



- Select the macro in the drop-down for the desired form event.

Note

The location of the rocker switch and hardware buttons varies with the make and model of CE device running the Visual CE application; refer to the documentation for that device.

- Click **OK**.

Command Reference

This section describes all of the Visual CE commands that you can use as steps in a macro script. The commands are listed two ways – first by category, then in alphabetical order for convenient reference. The alpha listing (page 328) provides a complete description of each command, including parameters you must specify.

Commands by Category

Script Control

Use this command	To
Skip <i>num</i>	Skip forward or back in the macro sequence; can be used to create conditional branches and loops. (page 362).
Macro step label	Identify destination for a Go To clause in a Skip command. (page 362)
Sleep <i>secs</i>	Freeze the application for some number of seconds. (pg 363)
Run macro <i>macro</i>	Start a specified macro. This command makes it possible to incorporate a submacro in the main macro. (page 357)
Timer <i>secs</i> , <i>macro</i>	Start a specified macro after a delay of some number of seconds. (page 368)
Return from macro	Stop the execution of the current macro and return control to the process that called it. If this statement is included in a submacro, processing returns to the main macro. If included in a main macro, processing returns to the form. Recommended as the last step in every macro. (page 355)
Stop macro	Stop the current macro. If included in a submacro, processing does not return to the main macro. Typically used when an error condition is detected. (page 367)

Form Controls

Use this command	To
Show/hide control <i>control</i> , Yes (Show) or No (Hide)	Show or hide any control on the form. (page 361)
Set focus <i>control</i>	Put focus on a form control. Frequently used after validation to put focus on a control with failed input. (page 360)
Select tab <i>num</i>	Select a specific tab.

Grid Controls

Use this command	To
Grid Control Position	Select an item in a grid. (page 336)

Records

Use this command	To
Sort by col	Sort records by values in a table column. (page 366)
Record First	Go to the first record in the table, based on the current sort order, and filter if any.* (page 352)
Record Previous	Go to the previous record in the table, based on the current sort order, and filter if any.* (page 352)
Record Next	Go to the next record in the table, based on the current sort order, and filter if any.* (page 352)
Record Last	Go to the last record in the table, based on the current sort order, and filter if any.* (page 352)
Record Search <i>exact matches only, arg, skip n if not found</i>	Find the first record (based on the current sort order, and filter if any) whose value in a specified table column matches a search value.* (page 352)
Record Print	Print the current record.* (page 352)
Record Save	Save the current record.* (page 352)
Record Insert	Insert a new record.* (page 352)
Record Duplicate	Create a new record by creating a duplicate copy of the current record. (page 352)
Record Delete	Delete the current record.* (page 352)
Edit IR Beam	Beam current record to another CE device.* (page 333)
Edit IR Beam All	Beam all records in the current table or filtered list to another CE device.* (page 333)
Edit IR Beam Receive	Receive records beamed from another CE device.* (page 333)
Option Filter Off	Display all records in the table.* (page 347)
Option Filter Custom	Open the Define Filter dialog box so that the user can filter the record display by desired criteria.* (page 347)
Option Filter Predefined filter, <i>skip n if not found</i>	Run a filter that has been predefined as part of the form design.* Note: At least one filter must already be defined as part of the form design in order for this command to be available in the Macro Builder. (page 347)
Option Clear	Delete all records in the table.* (page 345)

*Performs same action as Visual CE menu option of same name.

Field Values

Use this command	To
Assign Value Col	Assign a value to a table column or global variable. (page 329)
Edit Undo	Undo all changes made to the current record since it was last saved.* (page 334)
Option Recalc	Recalculate values for Calculated Field controls in current record.* (page 350)

*Performs same action as Visual CE menu option of same name.

Form Navigation

Use this command	To
Record Close	Close the current application.* (page 352)
Option List	Toggle the application display between list / single record mode.* (page 349)
Move to when pressed <i>left, top, change background color, color</i>	Move the screen display to a different region of the form. (page 343)
Option About	Displays information in the About box.* (page 344)

*Performs same action as Visual CE menu option of same name.

Serial I/O, including GPS

These commands read data from a serial device into a Visual CE application, and vice versa. For example, you might create a Visual CE application that records latitude, longitude, and altitude from a GPS device, or weight from a digital scale.

Use this command	To
Communicate Open <i>infrared, com</i>	Open the serial port to which the external device is connected. (page 330)
Communicate Send <i>value, port</i>	Send ASCII characters. (page 330)
Communicate Receive <i>port, col</i>	Receive ASCII characters and write them to a table column or global variable. (page 330)
Communicate GPS <i>port, col</i>	Read \$GPGGA sentences and write them to a table column or global variable; specialized version of the Receive command designed for GPS data in NMEA standard format. (page 330)
Communicate Close <i>port</i>	Close the serial port. (page 331)

Read and Write Files

Use this command	To
Option Import file	Import the contents of a comma-delimited ASCII text or XML file into the current table. (page 348)
Option Export file	Export records from the current table or filtered list to a comma-delimited ASCII text or XML file. (page 346)
mEnable put file <i>quiet file</i>	Upload a file of any type from the CE device to the mEnable server via an mEnable connection. (page 340)
mEnable get file <i>quiet file</i>	Download a file of any type from the mEnable server to the CE device via an mEnable connection. (page 340)
File Open <i>file#</i> , <i>file</i> , [read overwrite append]	Open a file in order to perform a specified file operation. (page 333)
File Read <i>file#</i> , <i>col</i>	Read from a file. (page 333)
File Write <i>value</i> , <i>file#</i> , <i>add linefeed</i>	Write to a file. (page 333)
File Close <i>file#</i>	Close a file. (page 333)
File Delete <i>file</i>	Delete a file. (page 333)

Run External Programs

Use this command	To
Jump Search Key, <i>Jump to appl</i> , <i>Only show matches</i> , <i>Cascade Deletes</i> , <i>Record Creation</i>	This command works the same as a Jump button. (page 337)
Run External command, <i>arg</i>	Launch another Visual CE application or other executable program and (optionally) pass a value from the current record to the second program. (page 356)
Report CE Report <i>report</i>	Display the current record in a Report CE layout. (page 354)

Wireless Connection (mEnable)

These commands control the mEnable connection. mEnable is a separate SYWARE product.

Use this command	To
mEnable off	Disconnect from the server database. (page 339)
mEnable synchronize <i>quiet</i>	Execute full synchronization between all eligible local tables and the server database; equivalent of the Yellow Stoplight. (page 339)
mEnable on <i>quiet</i>	Establish an mEnable connection to the server database and synchronize individual records as they are read or written to the local table with the server database. (page 339)

Miscellaneous

Use this command	To
Message box <i>msg</i> , Yes (<i>Skip 1</i>) or No (<i>Skip 2</i>)	Display a message box and, optionally, execute a different set of macro steps depending upon how the user responds. (page 342)
Play sound <i>symbol</i>	Play one of five different sounds from the Windows sound scheme, or a .wav file. (page 351)
Set alarm <i>Col</i>	Set an alarm to go off when the date/time in <i>Col</i> matches the current date/time on the CE device. (page 359)
Dial <i>Col</i>	Dial a phone number stored in the current record or global variable.
SMS send message <i>Msg</i> , <i>Phon</i> , <i>Port</i> , <i>Flow</i> , <i>Baud</i> , <i>Data</i> , <i>Parity</i> , <i>Stop</i>	Send an SMS instant text message to a specified phone number from a PocketPC Phone Edition, Smartphone, or CE device with a mobile phone connection (page Error! Bookmark not defined.).
SMS receive message <i>command</i> , <i>arg</i>	Launch a Visual CE form upon receipt of an SMS instant text message that begins with the string <i>.vce</i> and optionally save the incoming message to a variable. (page 364)

Commands Organized Alphabetically

This section lists in alphabetical order all of the commands you can use as steps in a macro program.

For each command, you will find a detailed explanation of its purpose, as well as any parameters you must specify. When you create a command as a step in a macro, you specify these parameters by completing the Command options at the bottom of the Macro Builder window. (Refer to the section, “Creating a New Macro,” on page 312).

About Expressions

For many command parameters, you can supply the same type of expression that you can specify in a calculated field control. Expressions can read values stored in a table column for the current record or in a global variable. Refer to page 201 for details about creating expressions.

Tip

When entering or editing expressions in the Macro Builder, you can double-click the field to open a larger text area in which to more easily type long expressions.

Assign command

Assign *Value Col*

The Assign command is used in a macro sequence to assign a value to a table column or global variable. It has two parameters:

- **Value:** The value to be assigned. You can supply a constant, or an expression that returns a value. The data type of *Value* can be text, numeric or date/time. You can double-click the field to open a larger area in which to type.
- **Col:** The table column or global variable to which the value is assigned.

Note: If you wish to create a timestamp, use the Assign command with one of the date/time functions such as @now or @today.

Example: Assign @now timestamp

Communicate commands

This group of five commands makes it possible to read data from a serial device into a Visual CE application and vice versa. For example, you might create a Visual CE application that records latitude, longitude, and altitude from a Global Positioning System device, or weight from a digital scale.

Typically, you will use these commands together in a macro to open the serial port, send or receive the data, and close the port.

Tip: To learn more about constructing a macro specifically for GPS data, look at the SYWARE-supplied sample application, GPS.VCE, which was provided with your installation package. (For instructions on installing the samples, please refer to the topic, "Installing Visual CE," on page 15). The samples work only on the CE device, not the desktop PC.

Note: All I/O is done as ASCII characters. For example, the number 123 is read/received as three characters ('1', '2', and '3'), as opposed to a 4-byte binary value.

Communicate | Open Port

Opens the serial port to which the external device is connected. The parameters depend upon the type of port:

- **Port:** Choose the type of port: *infrared* or *com*. Your selection here determines the remaining communications parameters. For proper settings, refer to the documentation for the device.

Communicate | Send Value, Port

Sends a value as ASCII characters. It has two parameters:

- **Value:** The value to be sent; can be a literal string or the result of an expression that may include values from the current record or a global variable. As with any Visual CE expression, you must enclose strings in single quotes ('') and column names that contain a space in double-quotes("). For example:

`"first name" & ' ' & "last name"`

You can double-click the field to open a larger area in which to type.

- **Port:** The serial port to which the external device is connected.

Communicate | Receive Port, Col

Receives a value as ASCII characters and writes it to a table column or global variable. It has two parameters:

- **Port:** The serial port to which the external device is connected.
- **Col:** The table column or global variable to which the value is written.

Communicate | GPS Port, Col

This command reads \$GPGGA sentences and writes them to a table column or global variable. It is a specialized version of the Receive command designed for GPS data in NMEA standard format. It clears characters from the input buffer and scans incoming data until it detects the beginning of a GPS \$GPGGA sentence. It reads the sentence (starting with \$GPGGA and up to but not including <CR><LF>) into the table column or global variable that you specify. It has two parameters:

- **Port:** The serial port to which the external device is connected.
- **Col:** The table column or global variable to receive the GPS sentence.

Tip: You can use the functions @long, @lat, and @alt to get the longitude, latitude, and altitude values from the sentence. Refer to the section, "GPS Functions," near the end of the topic, "Using Advanced Functions in a Calculated Field," on page 208.

Communicate | Close Port

Closes the serial port to which the external device is connected. It has one parameter:

- **Port:** The serial port to which the external device is connected.

Dial command

Dial Col

This command dials a phone number. It has one optional parameter:

- **Col:** The table column or global variable that stores the number to dial. If you do not specify a table column or variable, Visual CE uses the first phone number it finds in the current record, starting its search with the control that has focus.

Edit | IR Beam commands

These commands use an IR connection to copy records from one CE device to another CE device running the same Visual CE application. Each works the same as the menu option of the same name.

Note: Visual CE Visual CE copies records from the sending device to the receiving device, regardless of whether or not the records already exist in the table on the receiving device. That means you could end up with duplicates on the receiving device.

Edit | IR Beam

Beams current record to another CE device.

Edit | IR Beam All

Beams all records in current table or filtered list to another CE device

Edit | Receive IR Beam

Receives records beamed from another CE device

Edit | Undo command

This command undoes all changes made to the current record since it was last saved. It works the same as the Visual CE menu option of the same name.

An Edit | Undo command has no parameters.

File commands

This group of commands makes it possible to read data from and write data to files. You can open up to eight files at once.

File | Open File#, File, Mode

Opens a file for use as a stream. It has three parameters:

- **File#:** Identifier you assign to the file for reference in subsequent file operations.
- **File:** The table column or global variable that contains the full pathname to the file to open.
- **Mode:** The mode in which to open the file: open to read, open to overwrite existing data, open to append new data after existing data.

File | Read File#, Col

Reads the next line of text from an open file. It has two parameters:

- **File#:** The identifier you assigned to the file when opening it.
- **Col:** Table column or global variable to store data read from the file.

Tip

The @endoffile function is very helpful when performing file read operations. You can use it to test whether or not the end of the file has been reached: it returns 0 for No, and 1 for Yes.

File | Write File#, Value, Add linefeed

Writes data to an open file. It has three parameters:

- **File#:** The identifier you assigned to the file when opening it.
- **Value:** The data to be written; can be a literal string or the result of an expression that may include values from the current record or a global variable. As with any Visual CE expression, you must enclose strings in single quotes (') and column names that contain a space in double-quotes("). For example:

```
"first name" & ' ' & "last name"
```

 You can double-click the field to open a larger area in which to type.
- **Add linefeed:** Select this option to add an end of line character to the value.

File | Close File#

Closes a file you are no longer using. It has one parameter.

- **File#:** The identifier of the file to close.

File | Delete File

Deletes a file. It has one parameter.

- **File:** The table column or global variable that contains the full pathname to the file to delete.

Grid command

Grid Control, Position

This command makes it possible to programmatically select (that is, highlight) the first, previous, next, or last item in the grid control. It has two parameters:

- **Control:** The grid control to which the command applies.
- **[Position]:** The item to highlight (First, Prev, Next, Last).

Jump command

Jump *Search Key, Jump to appl, Only show matches, Cascade Deletes, Record Creation*

This command works just like having a Jump button (page 248) embedded in your macro sequence. When the command is executed, it takes the search key value for the current record from a table column or global variable, opens the "Jump to" Visual CE application, and displays the first record in that application's table that stores the same value. It has the following parameters:

- **Search Key and Jump to appl:** Visual CE guides you through selecting these first two parameters.

When you select the command in the Macro Builder, Visual CE displays the Select Columns dialog box. Select the table column or global variable in the current application that stores the value Visual CE will use to match related records in the "jump to" table. If you select a column, keep in mind that the "jump to" application must have an indexed table column with the same name that stores the same datatype.

Then, Visual CE displays the "Form to Jump to" box. Navigate to the application that will be the destination for the jump – the "jump to" application.

If you initially selected a global variable as the "jump from" key value, Visual CE asks you to select the column that stores the values to match in the "jump to" table. The column you select must be indexed.

After you make these selections, the Macro Builder displays the following options at the bottom of the window:

- **Only Show Matches:** If checked, Visual CE filters records in the "jump to" application so that only records that have a matching value in the key column are retrieved for display. If unchecked, all records in the "jump to" application are retrieved, but the form opens to the first record with a matching value.
- **Cascade deletes:** If checked, Visual CE deletes the current record in the "jump from" application table and all related records in the "jump to" table if the user (or macro) selects Record-Delete when the "jump from" from is running.
- **Record Creation:** Action that Visual CE takes when opening the "jump to" application:
 - ◊ **Always** - Always create a new record. If you select the "Only show matches" checkbox, Visual CE automatically fills in the value from the key column or variable; otherwise, the new record is completely empty.
 - ◊ **Only if Needed** - Create a new record only if there are no existing records with a matching value in the key column. If you select the "Only show matches" checkbox, Visual CE automatically fills in the value from the key column or variable; otherwise, the new record is completely empty.
 - ◊ **Never** - Do not create a new record. Display first record in "jumped to" application that has a matching value in the key column. If no matches are found, display messages, "Record not found."

Macro Step Label command

Macro Step Label *Label*

This command provides a destination for a Go To clause in a Skip command. It has one parameter:

- **Label:** A sequence of characters (letters, digits, underscores). A label cannot begin with a digit, nor can it include a space. Labels are not case-sensitive.

Example:

```
StartOfLoop
```

mEnable Connection commands

This group of three commands is used to control mEnable connections via a macro or command buttons.

mEnable is a separate SYWARE product that makes it possible for Visual CE users to access data on remote servers – interactively and in real time—via wired or wireless connections including the Internet. Please refer to the *mEnable User's Guide* for details about the setup and use of mEnable.

Tip: The mEnable Stoplight is an alternative control that performs the same function as the command buttons described here. It is compact and has the added advantage of displaying the status of the mEnable connection. (Refer to the topic, "Creating an mEnable Stoplight Control," on page 262 for details.) You would use the commands described here instead of the Stoplight in three situations: 1) You want to incorporate the mEnable commands in a macro program, 2) You want to allow the user to do one type of synchronization but not the other (for example, full but not continuous), or 3) You want to suppress the log on dialog box that normally appears when the user starts an mEnable session (this dialog is pictured on page 419).

mEnable on quiet

Establish an mEnable connection to the server database and synchronize individual records as they are read or written to the local table with the server database (continuous synchronization -- equivalent of the Green Stoplight). It has one parameter:

- **Quiet:** Suppresses the logon box that otherwise appears when a connection is initiated. (When the user initiates the very first mEnable session on the client device, the logon appears so that the user can specify the connection settings. mEnable uses those settings for subsequent sessions.) If this option is not selected, the logon box appears every time a new connection is opened.

mEnable synchronize quiet

Establish an mEnable connection to the server database (if one does not already exist) and execute full synchronization between all eligible local tables and the server database (equivalent of the Yellow Stoplight). It has one parameter:

- **Quiet:** See Quiet description above.

mEnable off

Disconnect from the server database (equivalent of the Red Stoplight).

mEnable Put and Get commands

These two commands are used to transfer files between the mEnable server and the CE device using an mEnable connection. When one of these commands is executed, it gets the pathname of the file to upload/download from the current record or global variable.

Tip: The function @fileexists (page 208) checks whether or not a file exists on the CE device. You could incorporate this function into your macro to check if a downloaded file would overwrite an existing file, or if a file designated for upload is not available, and alert the user accordingly.

mEnable put file *file*

Establishes an mEnable connection to the server database (if one does not already exist) and uploads a file of any type from the CE device to the mEnable server. It has two parameters:

- **Quiet:** Suppresses the logon box that otherwise appears when a connection is initiated. (When the user initiates the very first mEnable session on the client device, the logon appears so that the user can specify the connection settings. mEnable uses those settings for subsequent sessions.) If this option is not selected, the logon box appears every time a new connection is opened.
- **File:** The table column or global variable that contains the pathname of the file to put to the server.

The file pathname must completely identify the file on the CE device; e.g., \\My Documents\\Photos\\myPhoto.jpg.

When a file is uploaded, mEnable automatically creates the CE device folder structure on the server if the folders do not already exist there. To mirror the CE device's root folder on the server, mEnable creates a subfolder under the folder that contains the mEnable server software and gives it the same name as the CE device.

So for example, say that the mEnable server software is installed at C:\\Program Files\\SYWARE mEnable, and the CE device is named My Handheld. In that case, a file with the pathname \\My Documents\\Photos\\myPhoto.jpg on the handheld would be uploaded to: C:\\Program Files\\SYWARE mEnable\\My Handheld\\My Documents\\Photos\\myPhoto.jpg.

Note: If mEnable Manager is installed, the command utilizes any custom directory structures, rather than the default scenario just described.

mEnable get file *file*

Establishes an mEnable connection to the server database (if one does not already exist) and downloads a file of any type from the CE device to the mEnable server. It has one parameter:

- **Quiet:** See Quiet description above.
- **File:** The table column or global variable that contains the pathname of the file to get from the server.

The pathname must be specified relative to the folder structure of the CE device; e.g., \\My Documents\\Letters\\myFile.txt.

When a file is downloaded, mEnable automatically creates the specified folders on the CE device if they do not already exist.

mEnable assumes that the root directory on the CE device is mirrored on the server by a folder with the same name as the CE device and stored as a subfolder to the folder that contains the mEnable server software.

So for example, say that the mEnable server software is installed at C:\Program Files\SYWARE mEnable, and the CE device is named My Handheld. The file to download is identified as \My Documents\Letters\myFile.txt. In that case, mEnable looks on the server for C:\Program Files\SYWARE mEnable\My Handheld\My Documents\Letters\myFile.txt and downloads the file accordingly, creating \My Documents\Letters on the CE device if necessary.

Note: If mEnable Manager is installed, the command utilizes any custom directory structures, rather than the default scenario just described.

Message Box command

Message box *msg*, Yes (*skip 1*) or No (*skip 0*)

This command displays a message box with custom text that can include values from the current record. Optionally, you can specify that the user must respond Yes or No to the message, and have the macro skip to a different command depending upon how the user responds.

A Message box command has two parameters:

- **msg:** The message to be displayed; can be a literal string or the result of an expression that can include values from a table column for the current record or from a global variable. As with any Visual CE expression, you must enclose literal strings in single quotes ('') and column names that include spaces in double quotes (""). You can double-click the field to open a larger area in which to type.
- **Yes (*skip 1*) or No (*skip 0*):** If you select this option, the message box displays Yes and No buttons from which the user must choose. If the user clicks “Yes,” the macro skips one command in the macro sequence; if the user clicks “No,” the macro skips no commands in the macro sequence. If you leave this option unchecked, the message box simply displays an OK button that the user clicks to acknowledge and close the box.

Move to When Pressed command

Move to when pressed *left, top, change background color, color*

This command moves the screen display to a different region of the form. In order to apply this feature, the form must be larger than the device screen. Refer to the topic, "Specifying the Form Size," on page 158.

It has four parameters:

- **Left:** The horizontal position of the left side of the region to move to.
- **Top:** The vertical position of the top of the region to move to.
- **Change background color:** Check this option if you want to specify a background color for the move-to region. Assigning a different color to the destination region can help orient users to different sections of the form.
- **Color:** Select the background color.

Option | About command

This command opens the About Box to display any information you have provided there. It works the same as the Visual CE menu option of the same name.

An Option | About command has no parameters.

Option | Clear command

This command deletes all records in the current application table. It works the same as the Visual CE menu option of the same name.

An Option | Clear command has no parameters.

Option | Export command

Option | Export file

This command exports records from the current table or filtered list to a comma-delimited ASCII text or XML file.

It has three parameters:

- **All table columns:** *Applicable to text files, not XML.* If this option is selected, the command exports data from all columns in the table. Otherwise, the command exports data from just those columns that are associated with controls on the form. For XML files, export is always from all columns in the table.
- **No headers:** *Applicable to text files, not XML.* If you select this option, the table column names will not be included in the export file. If you don't select this option, the first line of the export file will consist of the table column names.
- **File:** The table column or global variable that stores the file pathname to export to.

To export an XML file, the full file pathname – including the .xml extension – must be supplied. If you do not supply the .xml extension, the command assumes that you are exporting to a text file. If you supply any extension other than .xml, a comma delimited text file is created.

If you do not provide any value for File, Visual CE automatically assigns the name of the table to export file and puts it in the same folder as the application. For example if the current application, MyForm.vce, uses MyTable, Visual CE will export the contents of the table to a text file called MyTable.txt and puts it in the same folder as MyForm.vce.

If the file to export to already exists, it will be overwritten. You can avoid this by checking with the @fileexists function and notifying the user accordingly.

Notes:

- Field values are exported in the same order as the control order on the form. Refer to “Specifying Control Order,” on page 276.
- All date/time values are automatically exported in the following format:
 - ◊ *Text:* YYYY-MM-DD HH:MM:SS
 - ◊ *XML:* YYYY-MM-DDTHH:MM:SS
- Pictures can be exported to an XML file; they cannot be exported to a text file.
- After the table is exported, you may want to delete all records in the table. (Be sure to keep the archived records in a safe place.) If you don't delete the records, and you later import the file back in, you will have two of every record.

Option | Filter commands

These commands filter the records by desired criteria. They work the same as the Visual CE menu options with the same name.

Option | Filter | Custom

Opens the Define Filter dialog box so that the user can filter the record display by desired criteria.

It has no parameters.

Option | Filter | Predefined *filter*, *skip n if not found*

Runs a filter that has been predefined as part of the form design. At least one filter must already be defined as part of the form design in order for this command to be available in the Macro Builder. It has two parameters:

- **Filter:** The name of the filter to be applied.
- **Skip *n* if not found:** (Optional.) Use this setting to supply a custom message/behavior if the record search finds no matches.

If you leave the 0 value in place, Visual CE applies the default behavior when a search yields no results: that is, it displays the message "Record not found" and quits the macro.

Alternatively, you can suppress the "Record not found" error that normally occurs when no record is found and have your macro skip forward or backward *n* commands in the macro sequence. For example, you could create a custom message box command, and supply a value for *n* that skips to that command when no match is found.

Option | Filter | Off

Displays all records in the table. It has no parameters.

Option | Import command

Option | Import file

This command imports the contents of a comma delimited ASCII text or XML file into the current table.

It has three parameters:

- **All table columns:** *Applicable to text files, not XML.* If this option is selected, the command imports data to all columns in the table. Otherwise, the command imports data to just those columns that are associated with controls on the form. For XML files, export is always to all columns in the table.
- **No headers:** *Applicable to text files, not XML.* Select this option if the file contains no headers; that is, if the first line of the file does not contain the table column names.
- **File:** The table column or global variable that stores the file pathname to import.

To import an XML file, the full file pathname – including the .xml extension – must be supplied. If the extension is not .xml, the file is assumed to be a comma delimited text file.

If you do not provide any value for File, Visual CE automatically looks in the folder that contains the Visual CE application (.vce file) for a text file with the same name as the table associated with the application. For example, if MyForm.vce uses MyTable, Visual CE will look for MyTable.txt in the same folder as MyForm.vce.

Notes

- To determine the proper format for an import file of either text or XML, including field order, export the current contents of the table to the desired file type and examine the results. (To export a file, open the form on the CE device and select **Options-Export**.)
- Visual CE considers any of the following date/time formats valid for either text or XML files, and imports values that conform to them as is:
 - ◊ YYYY-MM-DD
 - ◊ YYYY-MM-DD HH:MM
 - ◊ YYYY-MM-DD HH:MM:SS
 - ◊ YYYY-MM-DDTHH:MM:SS
 - ◊ Formats specified via the Windows Regional Settings
- Pictures can be imported from an XML file; they cannot be imported from a text file.

Option | List command

This command toggles the application display between list / single record mode. It works the same as the Visual CE menu option of the same name.

An Option | List command has no parameters.

Option | Recalc command

This command recalculate the values for all Calculated Fields in the current record. It works the same as the Visual CE menu option of the same name.

An Option | Recalc command has no parameters.

Play Sound command

Play sound *sound*

This command, typically used as a step in a macro sequence, causes the CE device to play a sound. You can select from five different sounds in the Windows sound scheme or you can supply your own .wav file. It has one parameter:

- Select from the symbols **OK**, *****, **!**, **X** and **?** -- each is associated with an event that generates a sound in the standard Windows sound scheme. Or, click [...] and select the desired .wav on the CE device.

In the Macro Builder, click **Play Sound** to hear the sounds on the PC. On the CE device, go to **Settings – Sounds & Reminders** to preview them. (The sounds may not match on different systems.)

Record commands

These commands manipulate records or the record display in a wide variety of ways. Each command works the same as the Visual CE menu option of the same name. Unless otherwise indicated, they have no parameters.

Record | Close

Closes the Visual CE application.

Record | Delete

Deletes the current record.

Record | Duplicate

Duplicates the current record.

Record | First

Moves to the first record in the table or filtered list.

Record | Insert

Creates a new record.

Record | Last

Moves to the last record in the table or filtered list.

Record | Next

Moves to the next record in the table or filtered list.

Record | Previous

Moves to the previous record in the table or filtered list.

Record | Print

Prints the current record.

Record | Save

Saves the current record.

Record | Search *exact matches only, arg, skip n if not found*

Finds the first record (based on the current sort order, and filter if any) whose value in a specified table column matches a search value. It has three parameters:

- **Exact matches only:** Select this option to find exact matches only (case is ignored). For example, if **Fred** were the search value, the result would be the first record in the table that stored the value **Fred** or **fred** in the sort column. **Fred Smith** would not match. If the table does not contain a record with the exact match in the sort column, the result would be the message, "Record not found.", or custom behavior you specify; see Skip if not found below. If you do not select this option, the command finds the first record that begins with the search value, or the next greatest value if no exact matches exist.
- **Arg:** (Optional.) The global variable that holds the search value. If you do not select a variable, the form will prompt the user for a search value.

- **Skip *n* if not found:** (Optional.) Use this setting to supply a behavior if the record search finds no matches.

If you leave the 0 value in place, Visual CE applies the default behavior when a search yields no results: that is, it displays the message “Record not found” and quits the macro.

Alternatively, you can supply your own message and/or behavior. For example, you could create a custom message box command, and supply a value for *n* that skips to that command when a match is not found.

Report CE Report command

Report CE Report *report*

This command opens Report CE and displays the current record in a Report CE report template (.rce file).

Create the report template (.rce file) according to the instructions in the Report CE documentation. The application table for the Visual CE form and the top level table for the Report CE layout must be the same table. **Tip:** If the tables are not the same, you can alternatively set up a Run External command to start another program (page 356), and specify the Report CE template (.rce file) as the command.

A Report CE Report command has one parameter:

- **Report:** The report (.rce file) to run. Use the Browse button to locate it in the CE device's file system.

Tip about Report CE switches

If you want to supply one or more switches to a Report CE layout, you can simply type them before the pathname; for example:

```
/f13345 \Program Files\Report CE\MyReport.rce
```

Refer to the topic, "Running a Report from a Visual Basic or C/C++ Application" in the *Report CE User's Guide* for information about switches.

The report can be run on either the CE device or the desktop PC. In either case, Report CE reports data from tables on the CE device. Therefore, if Report CE is run from the desktop PC, the CE device must be connected.

Return from Macro command

This command stops the execution of the current macro and returns control to the process that called it. If this statement is included in a submacro, processing returns to the main macro. If included in a main macro, processing returns to the form. It is recommended as the last step in every macro.

Run External command

Run External command, arg

This command launches another Visual CE application or other executable program and (optionally) passes a value from the current record to the second program.

It has two parameters:

- **Command:** The program to run. You can either type its full file pathname, or use the Browse button to locate it in the CE device's file system.

Tip about Report CE switches

If you want to supply one or more switches to a Report CE layout, you can simply type them before the pathname; for example:

`/f13345 \Program Files\Report CE\MyReport.rce`

Refer to the topic, “Running a Report from a Visual Basic or C/C++ Application” in the *Report CE User’s Guide* for information about switches.

- **Arg:** (Optional) The table column or global variable that stores the value to be passed to the second program as a command line argument.

Run Macro command

Run macro *macro*

This command starts a specified macro as a submacro.

It has one parameter:

- **Macro:** The name of the macro. (The macro must have already been defined in the current application.)

Select Tab command

Select tab num

This command selects a specified tab on the form.

It has one parameter:

- **Num:** The number that identifies the tab to select. The first tab is identified by 0, the second by 1, the third by 2, and so on.

Set Alarm command

Set alarm Col

This command sets an alarm to go off when the date/time in Col matches the current date/time on the CE device. When the alarm goes off, Visual CE beeps and displays a message. As long as any Visual CE application is running, the alarm goes off at the specified time. The alarm window contains a View button that the user clicks to open the record that triggered the alarm.

Note:

You must define the alarm message as part of the form design by selecting **Edit-Alarm** in the Visual CE development environment. Refer to page 232 for additional details.

The Set alarm command has one parameter:

- **Col:** The date/time column or a global variable that stores the date and time that the alarm sounds. If you do not supply a value, the user will be asked to provide a date/time when setting the alarm.

Set Focus command

Set focus control

This command puts focus on a form control. You might use it in a validation macro to put focus on a control with failed input so that the user can more easily correct the value, for example.

Note

This command can trigger a Get Focus or Lose Focus event.

It has one parameter:

- **Control:** The form control to get focus.

Show/Hide Control command

Show/hide control *control, Yes (Show) or No (Hide)*

This command, typically used as a step in a macro sequence that is triggered by an event, shows or hides any control on the form. (This application is illustrated in Quick Tour IV).

A Show/hide control has two parameters:

- **Control:** The form control to show or hide.
- **Yes (Show) or no (Hide):** If you select this option, the control is displayed when the command is executed. If you deselect this option, the control is hidden when the command is executed.

Skip command

Skip num

This command is used to skip forward or back in the macro sequence. Skip commands make it possible to have conditional branches and loops in a macro.

A Skip command has one parameter:

- **Num:** May be a number of steps to skip, or a GO TO clause with a label.

Number: The number of commands to skip; positive values skip forward, negative values skip backward. You can supply a number or an expression that evaluates to a number.

GO TO clause: You can type “Go To” and a label that matches a macro step label elsewhere in the script. When the skip command is executed, processing jumps to the line identified by the label. (See example on page 306.) “Go To” and macro step labels are not case-sensitive.

Tip

- You can double-click the Num field to open a larger area in which to type the expression or Go To clause.

To create a conditional branch, you can use an “if-then-else” expression for *Num* that controls how far to skip depending on values in the current record. For example, the following command would execute the next step in the macro if the value for age in the current record were less than or equal to 0, otherwise it would skip the next step.

```
if age <= 0 then 0 else 1
```

You can even nest one “if” inside another to create “if...else if ... else” expressions, as illustrated below. This sample macro applies a 25% discount if the total is more than \$1000, or a 10% discount if the total is between \$100 and \$1000.

Step	Command	Command parameters
1	Assign	Value: ITEM_1 + ITEM_2 + ITEM_3 Col: TOTAL
2	Skip	Num: If TOTAL > 1000.00 then 0 else if TOTAL >= 100.00 then 2 else 3
3	Assign	Value: TOTAL * 0.75 Col: TOTAL
4	Skip	Value: 1
5	Assign	Value: TOTAL * 0.9 Col: TOTAL
6	Return from macro	

Alternatively, you can create a conditional branch with a GO TO clause; i.e.,

```
if condition then go to label
```

For example, if age <= 0 then go to ErrorMessage

Note that there is no 'Else' when you use a GO TO clause; if the condition is not true, no skipping is done and the next step is executed.

Quick Tour IV (page 106) provides another example of conditional branching. The sample script at the beginning of this section (page 306) demonstrates how to create a loop with Skip commands and Go Tos.

Sleep command

Sleep secs

This command delays execution of the next step in the macro for a specified number of seconds. In fact, it freezes the entire application for the specified period. You might use this command in a GPS macro, for example, to freeze the application between the step that opens the GPS comm port and the step that reads the GPS signal.

A Sleep command has one parameter:

- **Secs:** The number of seconds to sleep; may be a constant or the result of an expression. You can double-click the field to open a larger area in which to type the expression.

See also **Timer** command (page 368).

SMS Receive Message command

Send Receive Message Command, Arg

This command automatically launches a Visual CE application upon receipt of an SMS instant text message that begins with the string `.vce.` and optionally saves the incoming message to a global variable.

To receive the message, the Visual CE application need not be running.

It has two parameters:

- **Command:** The Visual CE application to launch.
- **Arg:** Global variable in which to store the message. If no argument is supplied here, the Visual CE form is launched but the message is not saved.

SMS Send Message command

Send SMS Message *Msg, Phon, Port, Flow, Baud, Data, Parity, Stop*

This command sends an SMS instant text message from the Visual CE application to a phone number contained in the current record or global variable. To send the message, the Visual CE application must be running on a PocketPC Phone Edition or a CE device that is connected to a mobile phone via infrared, Bluetooth, or serial cable.

It has the following parameters:

- **Msg:** The message to be sent; can be a literal string or the result of an expression. As with any expression in Visual CE, you must enclose literal strings in single quotes ('') and column names that included spaces in double quotes ("").

For example, to create a message template for notifying customers of package shipments, you could specify the following expression (assuming that firstName is a table column):

```
'Dear ' & firstName & ', ' & 'Your package was mailed on ' & @now & '.'
```

You can double-click the field to open a larger area in which to type the message.

- **Phon:** The table column or global variable containing the phone number of the intended recipient.
- **Port:** Leave blank if the Visual CE application will run on a PocketPC Phone Edition. Otherwise, select the COM port on the CE device that will be used to connect with the phone's modem.
- **Flow, Baud, Data, Parity, Stop:** (If you selected a COM Port) Communications settings for the phone modem. Refer to the documentation for that device.

Sort by command

Sort by *col*

This command sorts records in the table or filtered list by values in a table column. You would typically apply this command prior to another macro step that depends on the records being in a certain order.

A Sort by command has one parameter:

- **Col:** The table column to sort by. This column must be indexed (see page 283).

Stop Macro command

This command stops the current macro. If included in a submacro, processing does not return to the main macro. It is typically used when an error condition is detected.

Timer command

Timer *secs, macro*

This command starts another macro after a delay of some number of seconds. Unlike the Sleep command which freezes the application, the timer allows the user to work with the form while it counts down. It has two parameters:

- **Secs:** The number of seconds between the time the command is executed and the designated macro is started; may be a constant or the result of an expression. You can double-click the field to open a larger area in which to type the expression.
- **Macro:** The name of the macro to start; must be a macro already defined in the current application.

See also **Sleep** command (page 363).

Timestamp (discouraged)

With the introduction of the Macro Builder in Visual CE version 10, use of Timestamp buttons as a step in a macro is discouraged. Instead, use an Assign command with one of the date/time functions such as @now or @today.

Example: Assign @now timestamp

Using the Macro Debugger

Overview

The Visual CE Macro Debugger is a visual tool that enables you to more quickly and easily identify errors in your macros. Working with the Macro Debugger on the desktop PC, you can “look into” your macro as it runs on the CE device. That is, you can pause execution at any step in the macro, so that you can examine how variable or table column values are effected when a particular step is executed. You can set “breakpoints” that stop the run at specific points, and you can resume the execution step-by-step and/or breakpoint-to-breakpoint.

Quick Summary: Steps for Using the Macro Debugger

Briefly, here are the steps for using the Macro Debugger. These steps are covered fully in the remaining topics in this chapter.

- 1 Connect the CE device to the desktop PC.
- 2 Open Visual CE on the desktop PC.
- 3 In Visual CE on the desktop PC, select **File–Open** and open the .vce file that contains the macro you wish to debug.*
- 4 Select **File – Debug**.
Visual CE asks you to save the application on CE device. Once you have done that, the Macro Debugger opens on the desktop PC.
- 5 In the Macro Debugger, enter one or more breakpoints where you want to pause execution of the macro script. To enter a breakpoint, highlight the step just *after* the point you want the pause to occur, then click the **Breakpoint** button.
- 6 In the Macro Debugger, click **Go**.
Visual CE starts the application on the CE device.
- 7 Working in the form on the CE device, launch the macro as you normally would.
The macro runs until it reaches the first breakpoint you inserted in the script.
- 8 Use the **Value** drop-down menu to examine the values contained by global variables and/or table columns at this point in the macro run.
- 9 Resume the script execution by clicking either **Step** or **Go** in the Macro Debugger. **Step** executes the next step, while **Go** runs the script up to the next breakpoint.
- 10 When you have identified the errors, close the Macro Debugger (so that you can open the Macro Builder where you can correct the problems). You can either click **Close** in the Macro Debugger, or select **Record – Close** in the .vce application. Either action closes the Macro Debugger *and* the .vce application, and saves your breakpoints.
- 11 Once you have successfully debugged your macro, delete from the CE device the .dbg file that has the same name and folder as the .vce file containing the macro. (The .dbg file saves the breakpoints.)

*If your application has multiple forms:

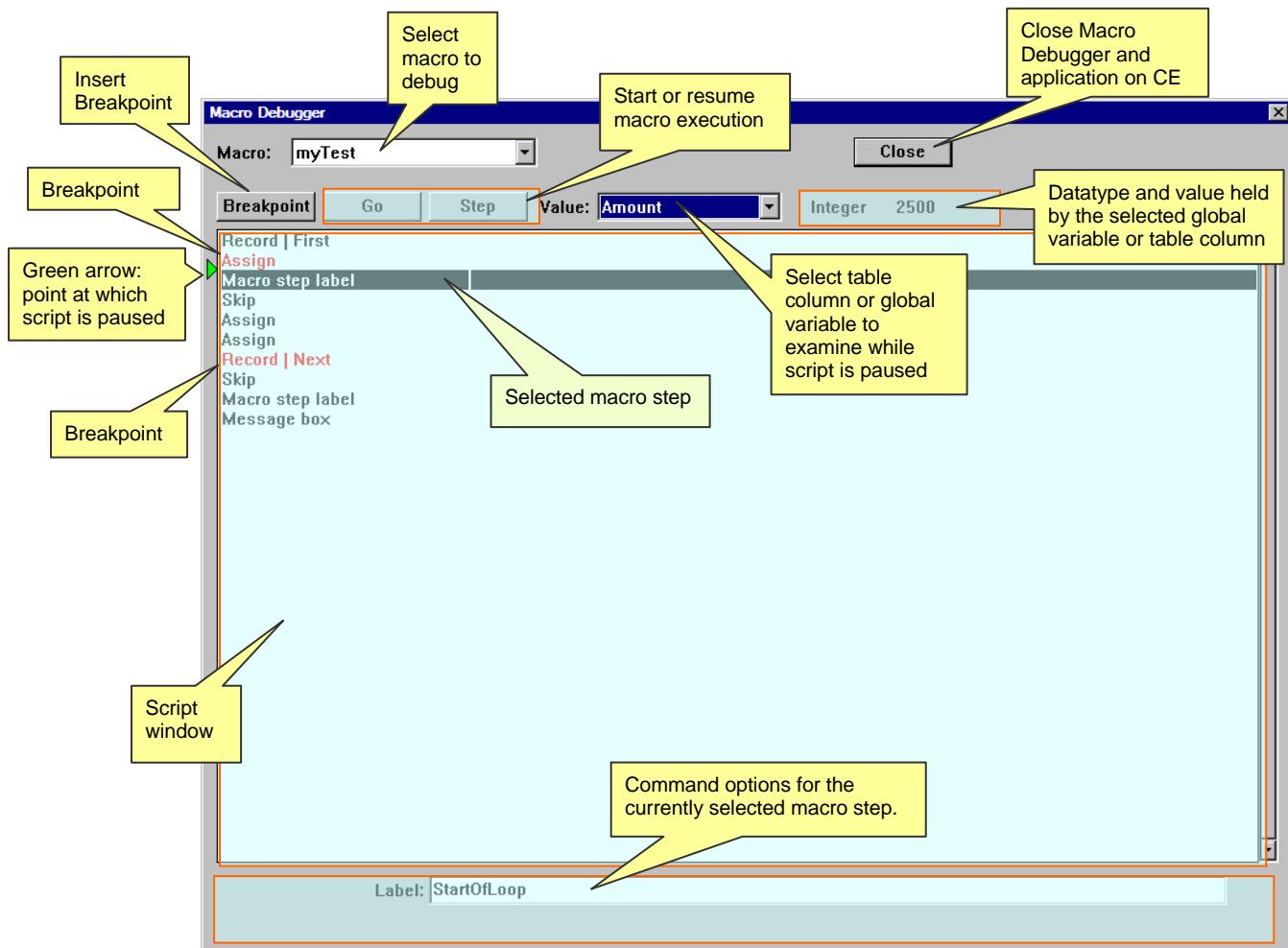
If your application has more than one form, open the first form (FILE–OPEN), run the debugger (FILE–DEBUG), set your breakpoints, and then (without clicking the GO button) click CLOSE to close the Debugger. Repeat this (FILE–OPEN, FILE–DEBUG, set breakpoints, CLOSE) for each form you want breakpoints in. Then, open your main form (FILE–OPEN), start the debugger (FILE–DEBUG), and press the GO button.

Starting the Macro Debugger

The Macro Debugger is available only on the desktop PC. The CE device must be connected to the desktop PC throughout the debugging process.

To start the Macro Debugger:

- 1 Connect the CE device to the desktop PC.
- 2 Working in the Visual CE development environment on the desktop PC, open the Visual CE application that includes the macro(s) to debug (**File—Open**).
- 3 Select **File—Debug**.
Visual CE displays the Save dialog box. (If you have modified the form on desktop, Visual CE asks you to save it. In addition, Visual CE always asks you to save the current form to the CE device, ensuring that the form whose macro you are debugging on the CE device is consistent with the version with which you are working on the desktop PC.)
- 4 Navigate to the location on the CE device where you want to save the form, and click **OK**.
Visual CE opens the Macro Debugger on the desktop PC. Here, you can step through the execution of your macro to pinpoint any problems in it.



- The **Macro** drop-down menu contains an alphabetical list of all macros that are included in the current form.

- The **Script window** lists all of the commands that make up the steps in the currently selected macro.
- The **Command options** area at the bottom of the Debugger window displays the parameters entered for the command (step) currently highlighted in the script window. These are non-editable. (If you find an error here, you must close the Debugger and open the macro in the Macro Builder.)
- The **Breakpoint** button either inserts a new breakpoint or removes an existing breakpoint in the macro script. When you are running a macro in the Debugger, the execution automatically stops at each breakpoint you've inserted, so you can examine the values contained by table columns or global variables at each point.
- The **Go** button either starts the Visual CE application running on the CE device, or, if the macro has been started and paused, resumes execution up to the next breakpoint.
- The **Step** button advances the execution one step.
- The **Value** drop-down menu lets you examine the value and datatype held by a selected table column or global variable at the point the macro is paused.
- The *green arrow* in the script window indicates the point at which the macro execution is currently paused.
- The **Close** button closes both the Debugger window and the CE application being debugged on the CE device. If you inserted any breakpoints into the macro, they are saved.

The following topics explain in more detail how to use each feature of the Macro Debugger.

Setting and Removing Breakpoints

You can insert breakpoints in the macro which tell the Debugger to pause execution before executing a given step. While the execution is paused, you can see the value contained in each table column or global variable at that point.

You can insert any number of breakpoints in your macro, but you must insert at least one.

When you close the Debugger, your breakpoints are saved.

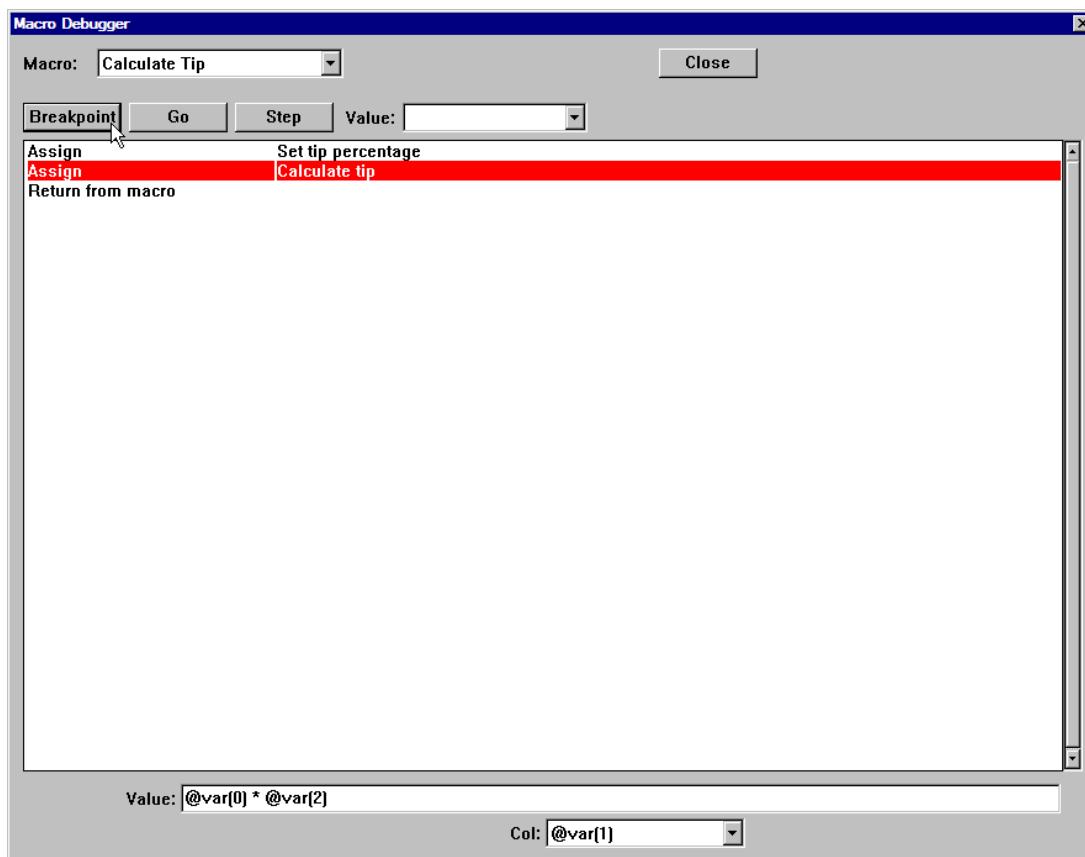
To insert a breakpoint:

- 1 In the Macro Debugger, select the macro you want to debug from the **Macro** drop-down menu.

The steps (commands) that make up the macro are displayed in the script window.

- 2 In the script window, highlight the line where you want to insert the breakpoint.

Note: When you run the Debugger, execution will stop just *before* the highlighted step.



- 3 Click the **Breakpoint** button.

The line turns red to indicate the breakpoint.

To remove a breakpoint:

- 1 In the script window, highlight the line containing the breakpoint.

- 2 Click the **Breakpoint** button.

The line returns to its normal state (black) to indicate that it is no longer contains a breakpoint.

Note about .dbg file on CE device

When you insert breakpoints in the macro, Visual CE notes their location in a .dbg file. This file is automatically saved on the CE device with the same name and folder as the form that contains the macro. When you have finished debugging the form, be sure to delete this file from the CE device. You will not be able to run the .vce application on the CE device until you do.

If your application has multiple forms

If your application has multiple forms each containing a macro, you can insert breakpoints into each one. That way, when you run the application in the Debugger (as explained in the next topic), you can actually debug any and all of the macros contained in it.

To insert breakpoints in multiple forms, simply repeat these steps for each form:

- 1 Select **File–Open** to open the form.
- 2 Select **File–Debug**.
- 3 Set your breakpoints as described above.
- 4 Click **Close** to close the Debugger.

After you have added the breakpoints to each form, *open the main form (**File–Open**)*, start the Debugger (**File–Debug**) and proceed with the instructions in the next topic.

Running a Macro in the Debugger

Once you have inserted at least one breakpoint into your macro, you are ready to debug it.

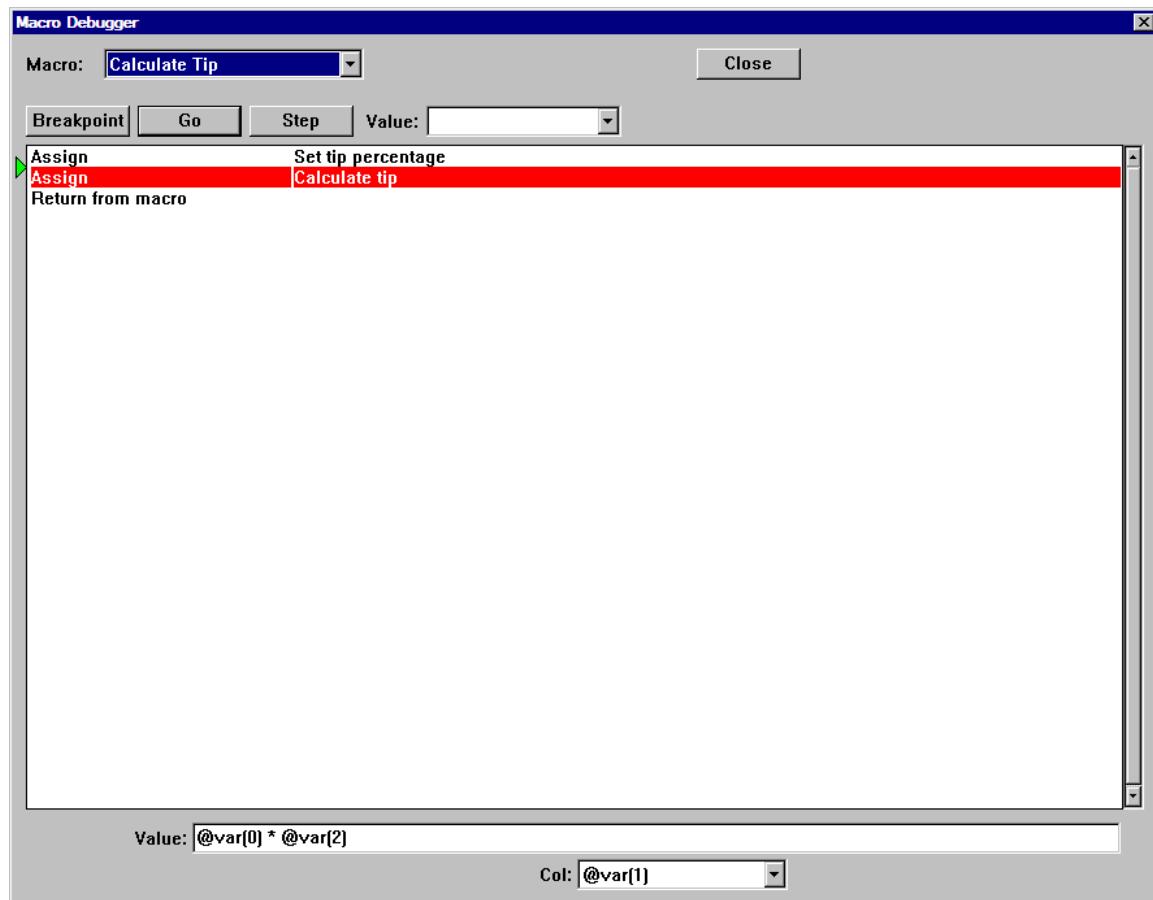
To execute a macro in the Debugger:

- 1 Click the **Go** button.

Visual CE opens the application on the CE device.

- 2 In the Visual CE application on the CE device, launch the macro as you normally would (click a command button, initiate an event, etc.).

The Debugger pauses the execution of the macro at the first breakpoint. In the script window on the desktop PC, it displays a green arrow to indicate where in the script it has paused.



- 3 With the script paused, use the **Value** drop-down to select a variable or table column whose value you wish to examine. To the right of the Value drop-down menu, the Debugger displays the value and datatype contained in the selected variable or table column.
- 4 Click **Step** to execute the next step in the macro, or click **Go** to continue the execution up to the next breakpoint.

Closing the Debugger

Once you have identified any errors in the macro, close the Macro Debugger on the desktop PC and the application on the CE device. Then, you can open the macro in the Macro Builder where you can fix the problems.

To close the Macro Debugger:

Either of the following actions will close the Macro Debugger and the application:

- In the Macro Debugger, click **Close**, or,
- In the Visual CE application on the CE device, elect **Record–Close** or click the **Exit** button .

Reminder

Once you have finished debugging the form, remove the .dbg file from the CE device. (This file contains the breakpoints. It is automatically saved with the same name and in the same folder as the .vce file.) If you do not remove the .dbg file and you attempt to run the form on the CE device without a connection to the Debugger on the desktop PC, Visual CE will display an error message.

Part IV:

Using Visual CE Applications

Using Visual CE Applications

Overview of Part IV

If you used the sample application in Quick Tour I or Ia, you already know the basics of using Visual CE applications on CE devices with or without a touchscreen. This section contains detailed reference information for using Visual CE applications on both types of devices, as well as on desktop PCs, including features not covered in the quick tour.

The following topics are discussed:

- Visual CE Application Window
 - ◊ Starting Visual CE applications
 - ◊ Exiting Visual CE applications
- Viewing Records
 - ◊ Scrolling through records
 - ◊ Displaying a list of the records
 - ◊ Switching from list display to single record display
- Sorting and Finding Records
 - ◊ Searching for specific records in the table
 - ◊ Changing sort/search order
 - ◊ Filtering records by specific criteria
- Adding, Editing, and Deleting Records
 - ◊ Creating new records
 - ◊ Creating a new record by duplicating an existing record
 - ◊ Editing records
 - ◊ Saving records
 - ◊ Deleting records
- Entering Data into Records
 - ◊ Using scribble boxes
 - ◊ Using image controls
 - ◊ Entering a null value for a date/time or numeric field
 - ◊ Entering negative numbers and decimal points on non-touchscreen devices
 - ◊ Using word lists in edit or note boxes
 - ◊ Copying or moving data between records
 - ◊ Using a bar code scanner with Visual CE
- Using Alarms
 - ◊ Setting and Responding to Alarms
- Printing Records
 - ◊ Printing records
 - ◊ Using Report CE to Print a Visual CE Record
- Exporting and Importing Data
 - ◊ Exporting a table
 - ◊ Importing an ASCII text file
 - ◊ Clearing a table
 - ◊ Beaming records between CE devices
- Using Visual CE applications with mEnable
- Running a Visual CE Application from a Visual Basic or C/C++ Application

Visual CE Application Window

Starting Visual CE Applications

Visual CE applications can be run on CE devices with or without a touchscreen, on and desktop PCs.

To start a Visual CE application on a CE device with a touchscreen:

- Click the Windows **Start** button, then select **Programs**, then the name of the Visual CE application. Or, use Windows Explorer to navigate to the Visual CE application icon,  and double-click it. (The Visual CE application icon  is displayed above the filename.)

Visual CE displays your application in its application environment window.

To start a Visual CE application on a CE device without a touchscreen (e.g., a Smartphone):

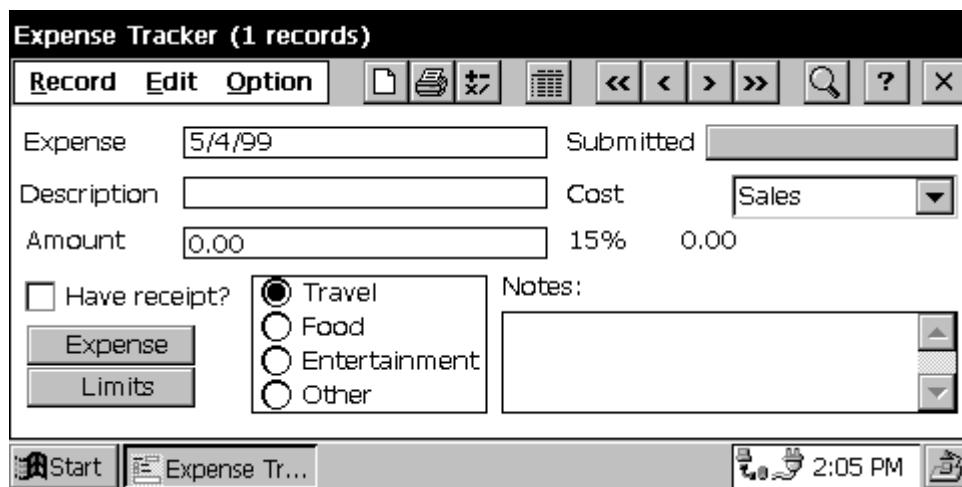
- On the Home screen, click **Start**, then the name of the Visual CE application. Visual CE displays your application in its application environment window.

To start a Visual CE application on a desktop PC:

- Verify that the underlying table has been set up for synchronization (refer to the topic, "Setting Up Visual CE Applications to Run on a Desktop PC," on page 296).
- Navigate to the folder that contains the icon for the Visual CE application.
- Double-click the icon.

Note

Visual CE applications have the .VCE filename extension.



On touchscreen devices and desktop PCs, the default Visual CE application window provides three menus – Record, Edit, and Option. You can select menu options by clicking them with the stylus or mouse pointer, or by pressing the shortcut keys on your keyboard. In addition, Visual CE toolbar buttons offer a faster way to activate the most frequently-used commands listed on the menus.

The screenshot shows the 'Expense Tracker' application window. At the top, it displays 'Expense Tracker (1 record)' and a close button (X). Below the title, there are several input fields and controls:

- Date:** 2 / 5 /2008
- Description:** (empty)
- Amount:** 0.00
- Tip:** 0.00 Receipt?
- Cost Center:** Sales (with a dropdown arrow)
- Category Selection:** Radio buttons for Travel (selected), Food, and Other.
- Notes:** A text area containing placeholder text: 'Notes:' followed by three empty lines.
- Buttons at the bottom:** Expense, Limits, Dial, and Record.

On non-touchscreen devices (e.g., Smartphones), the default Visual CE application window provides the Record and Dial commands activated by pressing the softkeys immediately below them. 'Record' opens a menu of the most frequently-used Visual CE commands. 'Dial' dials a phone number in the current record (if one is available). It dials the first phone number it finds in the current record, starting its search with the control that has focus.

Record Menu Commands

On Touchscreen Devices and Desktop PCs:

On touchscreen devices and desktop PCs, the default Record menu offers the following commands. (Refer to the following page for a description of the Record Menu commands on non-touchscreen devices.)

Note

Not all buttons are available on the P/PC.

Command	Button	Shortcut	Function
First		(none)	Displays the first record.
Previous		Ctrl+Down Arrow On P/PC: move the Up/Down switch up.	Displays the previous record.
Next		Ctrl+Up Arrow On P/PC: move the Up/Down switch down.	Displays the next record.
Last		(none)	Displays the last record.
Search		Ctrl+F	Locates a record based on the current sort/search order.
Print		Ctrl+P	Prints the current record.
Save	(none)	Ctrl+S	Saves the current record.
Insert*		Ctrl+N	Creates a new record.
Delete	(none)	Ctrl+DEL	Deletes the current record.
Close		Ctrl+W Ctrl+Q	Closes the application.

* If you hold down the Shift key while clicking **Insert** (or the Insert button), Visual CE creates a new record by duplicating the current record. Refer to the topic, "Creating a New Record by Duplicating an Existing Record," on page 398 for information about using the Shift key on a P/PC.

On Non-Touchscreen Devices:

On non-touchscreen devices (e.g., Smartphones), the default Record menu offers the following commands. (Refer to the previous page for a description of the Record Menu commands on touchscreen devices and desktop PCs.)

Command	Function
List	Toggles the display between list and single record mode.
Search	Locates a record based on the current sort/search order.
First	Displays the first record.
Previous	Displays the previous record.
Next	Displays the next record.
Last	Displays the last record.
About	Displays information in the About box.
Close	Closes the application.
Delete	Deletes the current record.
Insert	Creates a new record.

Edit Menu Commands

On Touchscreen Devices and Desktop PCs:

On touchscreen devices and desktop PCs, the default Edit menu offers the following commands.

Command	Button	Shortcut	Function
Undo	(none)	Ctrl+Z	Undoes last edit.
Cut	(none)	Ctrl+X	Deletes the currently-selected information from the edit box.
Copy	(none)	Ctrl+C	Copies the currently-selected edit box information into the Windows clipboard.
Paste	(none)	Ctrl+V	Inserts information from the Windows clipboard into the current edit box.
IR Beam*	(none)	(none)	Beam current record to another CE device.
IR Beam All*	(none)	(none)	Beam all records in current table or filtered list to another CE device.
Receive IR Beam*	(none)	(none)	Receive records beamed from another CE device.

*Available only when running on the handheld

On Non-Touchscreen Devices:

The Edit Menu is not available in the default version of the Visual CE application window for non-touchscreen devices (e.g., Smartphones).

Option Menu Commands

On Touchscreen Devices and Desktop PCs:

On touchscreen devices and desktop PCs, the default Option menu offers the following commands.

Command	Button	Shortcut	Function
List		Ctrl+L	Toggles the display between list and single record mode.
Zoom	(none)	(none)	List View Only. Enlarges text in the display.
Filter	(none)	(none)	Off: Displays all records in table. Custom: Allows user to specify conditions that records must meet in order to appear in display; other records are hidden until filter is turned Off. [filter_name]: Applies a pre-defined filter (set of conditions); other records are hidden until filter is turned Off.
Import	(none)	(none)	Copies data from an ASCII file to the table on your CE device. The ASCII file must have been created using the Export command.
Export	(none)	(none)	Copies the data from the table to an ASCII file on your CE device. Note: Pictures are not exported or imported.
Clear	(none)	(none)	Deletes all of the records in the table.
Recalculate		(none)	Recalculates the values in the current record's calculated field(s).
About	(none)	(none)	Displays information in the About box.

On Non-Touchscreen Devices:

The Option Menu is not available in the default version of the Visual CE application window for non-touchscreen devices (e.g., Smartphones).

Application Window Toolbar Buttons

On Touchscreen Devices and Desktop PCs:

If the form designer used the default menu bar for touchscreen devices and desktop PCs, the following buttons are available.

Applications on H/PCs have these buttons:

Button	Name	Function
	Insert	Creates a new record in the table.
	Print	Prints the current record.
	Recalc	Recalculates the values in the current record's calculated field(s).
	List	Toggles the display between list and single record mode.
	First	Displays the first record.
	Previous	Displays the previous record.
	Next	Displays the next record.
	Last	Displays the last record.
	Search	Locates a record based on the current sort/search order.
	Help	Displays online help for Visual CE.
	Exit	Exits the application.

Applications on P/PC have these buttons:

Button	Name	Function
	Insert	Creates a new record in the table.
	Recalc	Recalculates the values in the current record's calculated field(s).
	List	Toggles the display between list and single record mode.
	Search	Locates a record based on the current sort/search order.

On Non-Touchscreen Devices:

The toolbar buttons are not available in the default version of the Visual CE application window for non-touchscreen devices (e.g., Smartphones).

Application Window Shortcut Keys

On Touchscreen Devices and Desktop PCs:

Note

If the form designer customized or hid the menu bar, the Shortcut Keys are unavailable, except Ctrl+Q which is always available to close the application.

Record Functions

To do...	Press
Display the previous record	Ctrl+Up arrow
Display the next record	Ctrl+Down arrow
Locate a record based on the current sort/search order.	Ctrl+F
Print the current record.	Ctrl+P
Save the current record.	Ctrl+S
Create a new record in the table.	Ctrl+N
Create a new record by duplicating values from the currently displayed record	Shift+Ctrl+N
Delete the current record.	Ctrl+DEL
Close the application.	Ctrl+W or Ctrl+Q
Go from list display to single record display.	Enter key or Action button

Edit Functions

To do...	Press
Undo last edit.	Ctrl+Z
Delete the currently selected information from the edit box.	Ctrl+X
Copy the currently selected edit box information into the Windows clipboard.	Ctrl+C
Insert information from the Windows clipboard into the current edit box.	Ctrl+V

Option Functions

To do...	Press
Toggle between list and single record display modes.	Ctrl+L

P/PC Only

To ...	Do this
Go from list display to single record display.	Press action button.
Display the previous record.	Move Up/Down switch up.
Display the next record.	Move Up/Down switch down.

On Non-Touchscreen Devices:

On non-touchscreen devices (e.g., Smartphones), the following shortcut keys are available:

To ...	Do this
Display the previous record.	Move Up/Down rocker up.
Display the next record.	Move Up/Down rocker down.

Exiting Visual CE Applications

To exit a Visual CE application on a CE device with a touchscreen or a desktop PC:

- Select **Record–Close** or click the **Exit** button 

To exit a Visual CE application on a CE device without a touchscreen:

- Click **Record–Close**.

Visual CE saves any changes or new records created since your last save, then closes the Visual CE application window.

Viewing Records

Scrolling Through Records

Once you have created two or more records (or if records have been downloaded from an application on your desktop PC), you can scroll through the records in the table.

The default application window's Record menu lists four scrolling commands (First, Previous, Next, and Last). The menu commands have equivalent toolbar buttons or shortcuts (depending on the computer or device) for faster command execution.

Command	Button*	Function	Shortcut
First		Displays the first table record.	(none)
Previous		Displays the previous table record.	Keyboard: Ctrl+Up arrow P/PC or non-touchscreen device: Move Up/Down switch up.
Next		Displays the next table record.	Keyboard: Ctrl+Down arrow P/PC or non-touchscreen device: Move Up/Down switch down.
Last		Displays the last table record.	(none)

*Available only on touchscreen devices and desktop PCs.

Tip

On touchscreen devices and desktop PCs: Visual CE's filter option enables you to temporarily limit the display to just those records of current interest; e.g., that match a set of conditions you specify. In that way, you can scroll through a subset of the records, rather than entire table. Refer to the topic, "Filtering the Record Display," on page 395.

Displaying a Record List

On Touchscreen Devices and Desktop PCs:

To view a list of records in the table, select **Options–List**, click the **List** button , or press Ctrl+L.

Visual CE displays one line of data for each record. If the window is full, use the scroll bar to display additional records.

Tips

- You can adjust the column widths by dragging the squares between the column headings.
- You can sort the record list by any column in the table. Refer to the topic, "Changing Sort/Search Order," on page 394.
- If the sort/search order is based on a text column, you can scroll directly to the first record in the table whose entry in the column begins with a specific letter or sequence of letters (this is useful if the list is very long). Simply select the sort/search order, then press or tap the letters on the keyboard or soft keypad. Note: If you enter the letters too slowly, Visual CE interprets them as separate instructions. For example, if you enter **b**
r the list would scroll to the first entry that begins with br (say, "Brandon"). But if you enter **b** then enter **r** after a pause, the list would scroll to Barbara, then Raymond.
- You can view a list of all records in the table, or just records that match specific criteria (a "filtered" list). Refer to the topic, "Filtering the Record Display," on page 395.
- Depending upon the form design and the field, you may be able to edit data in the list display. Tap and hold on the value you wish to change.

On Non-Touchscreen Devices:

To view a list of records in the table, click **Record–List**.

Visual CE displays one line of data for each record. Because of the limited screen size, only one column -- the table column used for the current sort/search order -- is displayed.

If the list is long, use the up/down rocker to scroll through the records.

Tips

- Depending upon how the form designer set up the list display, you may be able to sort and list records by other table columns. If that is the case, the form designer would have provided buttons on the Record display that you could use to select the desired table column.
- *If your device has a keyboard:* If the current sort/search order is based on a text column, you can scroll directly to the first record in the table whose entry in the column begins with a specific letter or sequence of letters (this is useful if the list is very long). Simply press the letters on the keyboard, and Visual CE scrolls the list to the first record whose entry begins with those letters. Note: If you enter the letters too slowly, Visual CE interprets them as separate instructions. For example, if you enter **b**
r the list would scroll to the first entry that begins with br (say, "Brandon"). But if you enter **b** then enter **r** after a pause, the list would scroll to Barbara, then Raymond.

Switching from List Display to Single Record Display

On Touchscreen Devices and Desktop PCs:

There are several ways to "turn off" the list display and return to single record mode:

- Select **Options–List**.

- Click the **List** button .

- Press **Ctrl+L**.

- On an H/PC, press the **Enter key**.

- On a P/PC, press the **Action button**.

Visual CE returns to the display of a single record at a time in its application window.

On Non-Touchscreen Devices:

To "turn off" the list display and return to single record mode, you can either:

- Press the **Action** button.

- Select **Record–List**.

Visual CE returns to the display of a single record at a time in its application window.

Sorting and Finding Records

Searching for Specific Records in the Table

Visual CE provides three ways to quickly locate specific records:

- **Search:** The Search feature scrolls the display to the first record whose value in a specified column matches a value you provide. Using the Quick Tour's Expense Tracker application as an example, you could scroll to the first record created in May, or to the first record in the Travel category, or to the first record whose description begins with the word Dallas, and so on.
- **Filter:** You can use a list "filter" to find and display all records that match up to three conditions. Your search criteria are not limited to a single column. In fact, you can even ask Visual CE to find all records that have a specified text string in ANY text or memo column. For example, you could find all records in the Expense Tracker that were created after January 1, 2001, and were assigned to the Travel category, and had the word Dallas anywhere in any text column.
- **Scroll to record by prefix:** (List display mode only) If the sort/search order is based on a text column, you can scroll directly to the first record in the table whose entry in the column begins with a specific letter or set of letters. Simply select the sort/search order (see page 394), then press or tap the letters on the keyboard or soft keypad. Note: If you enter the letters too slowly, Visual CE interprets them as separate instructions. For example, if you enter **br** the list would scroll to the first entry that begins with br (say, "Brandon"). But if you enter **b** then enter **r** after a pause, the list would scroll to Barbara, then Raymond.

This topic describes the Search command in more detail. For a complete description of list filters, refer to the topic, "Filtering the Record Display," on page 395.

To apply the search command on a touchscreen device or desktop PC:

- 1 If you are not already in list display mode, select **Options-List**, click the List  button, or press **Ctrl+L**.
- 2 On the top line, which displays the column names, click the column you want to search on.
- 3 If the column is not already indexed, Visual CE asks if you want to index it now. Click **Yes**.

Note

If the table was created in Pocket Access or SQL Server CE, you must create the column indexes in advance. SYWARE offers free utilities for that purpose. Refer to the topic, "Visual CE Add-on Tools," on page 17 for instructions on obtaining a copy.

Visual CE places two symbols (^) next to the column name, to indicate that the current sort/search order is based on that column.

- 4 Select **Record-Search** or click the **Search** button .
- Visual CE displays an input box that is appropriate to the column's data type.
- 5 Enter the value you want to find.

Note

If you are searching on a text column, it is not necessary to enter a complete word. For example, if you entered just **day**, Visual CE scrolls to the first record whose value in that column begins with those three letters – e.g., Day, days, Dayton, etc.

To apply the search command on a non-touchscreen device (e.g., a Smartphone):

- 1 In order to search for values in a particular table column, the form designer must have made it possible to sort the table records by that column. Set focus on the button provided by the form designer for that purpose and press the Action key.
- 2 Click **Record-Search**.
Visual CE displays an input box that is appropriate to the column's data type.
- 3 Enter the value you want to find.

Note

If you are searching on a text column, it is not necessary to enter a complete word. For example, if you entered just **day**, Visual CE scrolls to the first record whose value in that column begins with those three letters – e.g., Day, days, Dayton, etc.

Changing Sort/Search Order

You can change the record sort/search order in the record list display. That means you can sort the table records by values in any column, at any time.

Note

This feature is standard in Visual CE applications on touchscreen devices and desktop PC. It is not standard on non-touchscreen devices; however, the form designer may have provided a special command button or menu option for changing the sort/search order (as illustrated in "Quick Lesson 5a: Creating a Command Button to Sort Records").

To change the sort/search order on a touchscreen device or desktop PC:

- 1 Select **Options-List**, click the **List** button , or press Ctrl+L.
- 2 On the top line, which displays the column names, click the column you want to use to sort the records.
- 3 If the selected column is not already indexed, Visual CE asks if you want to index it now. Click **Yes**.

Note

If the table was created in Pocket Access or SQL Server CE, you must create the column indexes in advance. SYWARE offers free utilities for that purpose. Refer to the topic, "Visual CE Add-on Tools," on page 17 for instructions on obtaining a copy.

Visual CE displays the records in the newly selected order, and places a [^] symbol next to the selected column heading to indicate that it is the basis for the current sort/search order. You can scroll the new order in either list or single record display mode.

Notes

- Creating a sort/search order can take a long time.
- The Search command applies only to the current sort/search order. For example, if Date is the current sort/search order, you can only search for records by date. If you want to search for records by description, you must make Description the current sort/search order. This restriction applies only to the Search command, not the Filter option.
- Once you select a column for sorting/searching, that column automatically becomes indexed (indexed columns are denoted by the [^] symbol in the column heading). Up to four columns in a table can be indexed at any given time.

Filtering the Record Display

Note

This feature is available in Visual CE applications on touchscreen devices and desktop PC. It is not available on non-touchscreen devices.

If there are numerous records in a table, working with the entire list can be cumbersome. You can narrow your view to just those records that match specific criteria.

Using the Quick Tour's Expense Tracker as an example, you might ask Visual CE to list just those expense records that contain a client's name in any text field, or whose amount exceeds \$10.00, or that were created after a particular date. Those records that do not match are temporarily filtered out (hidden).

The set of conditions is called a "filter." A filter can have up to three conditions. Each condition consists of a column name, a value, and a comparison operator that specifies how the record's value in the column must compare to the specified value in order for the record to be included in the list. You can specify that a record must match at least one of the conditions, or all of them.

You can select a filter provided by the form's designer, or you can create your own. You can quickly turn a filter on/off to switch the display between the filtered list and the entire list.

To use filters on a touchscreen device or desktop PC:

Visual CE's **Option-Filter** submenu offers three items:

- **Off:** Switches filters off, so that all records in the table are displayed
- **Custom:** Brings up a dialog box in which you can apply your own filter criteria (see description below)
- **[filter name]:** Applies a filter provided by the form designer. (If no options other than "Off" and "Custom" appear in the sub-menu, the form designer has not provided any pre-defined filters with this form. See page 284 for information about pre-defined filters.)

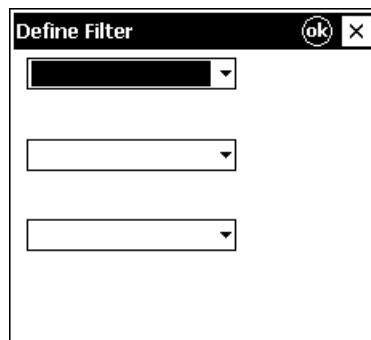
To apply your own filter criteria on a touchscreen device or desktop PC:

- 1 Select **Option-Filter-Custom**.

Visual CE displays the Define Filter dialog box. If you had applied a filter previously, or if the form designer supplied a filter, Visual CE displays those conditions. You can use them as is, or modify them.

Tip

To clear a value, click the drop-down arrow, scroll to the top of the display, and click the blank line.



- 2 Specify the first condition:

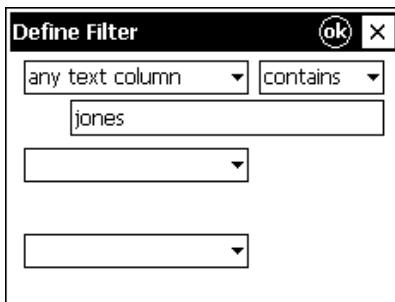
- Click the drop-down arrow for the first field in the dialog box, and select the column that contains the data you want to compare.

- In the field that appears to the right of the column name field, select a comparison operator. Not all comparison operators are available for every type of column.

Symbol	Comparison operator
=	Equal to
<>	Not equal to
<	Less than
>	Greater than
<=	Less than or equal to
>=	Greater than or equal to
Contains	Contains a specified text string; e.g., contains "cook" would find "John Cook," "Cooking," "Chocolate cookies," etc.

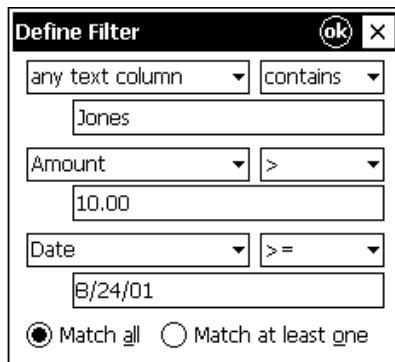
- Specify the comparison value.

Note: Text comparisons are NOT case-sensitive.



Tip: You can test for null values (e.g., empty fields). For the comparison operator, select = (Equal to); and for the comparison value, enter "" (two single quote marks with no intervening space).

- If desired, add one or two more conditions in the remaining fields using the same procedure.



- If you supplied two or three conditions, choose one of the following:

- Match all:** A record will be included in the filtered list only if it satisfies all of the conditions
- Match at least one:** A record will be included in the filtered list if it satisfies any one of the conditions

- Click **OK** to close the dialog box. Visual CE applies your filter.

When you view the record display (either list or single record mode), only records that match the filter criteria are visible.

Adding, Editing, and Deleting Records

Creating New Records

To create a new table record on a touchscreen device or desktop PC:

Select **Record–Insert** or click the **Insert** button .

Visual CE displays a new record (with default values) in which you can make your entries.

See also: "Creating a New Record by Duplicating an Existing Record" (page 398)

To create a new table record on a non-touchscreen device:

Click **Record–Insert**.

Visual CE displays a new record (with default values) in which you can make your entries.

Creating a New Record by Duplicating an Existing Record

Note

This feature is available in Visual CE applications on touchscreen devices and desktop PC. It is not available on non-touchscreen devices.

If a new record that you want to create has many of the same entries as a previous record, you may want to start by making a copy of the existing record.

1 Display the record that you want to duplicate.

2 While pressing the Shift key*, select **Record-Insert** or click the **Insert** button .

Visual CE displays the duplicate – you can tell that a new record has been created because Visual CE increments the number of records displayed in the application title bar by one.

3 Edit the entries as desired.

4 Select **Record-Save** to save your new record.

See also: "Creating New Records" (page 397)

*Shift Key on the P/PC

You apply the Shift key on the P/PC's soft keypad just as you would on a regular keyboard:

1 Activate the P/PC's soft keypad by clicking the keyboard image in the P/PC's taskbar.

2 Tap the image of the **Shift** key so that it is highlighted.

Note

Unless you want to continue to create duplicate records, be sure to "de-select" the Shift key after you have applied the Insert command.



Editing Records

Visual CE allows you to easily edit existing records. You can edit records in single record mode and list mode.

To change an existing field value:

- **In single record mode:** Simply click or set focus on the field, and enter or select a new value. If the form contains grid controls, they may be editable as well (contact the form designer). If so, use the tap-and-hold technique described below.
- **In list mode:** Tap and hold on the field. If the field is editable, a text entry box opens for you to complete. (This feature is applicable to touchscreen devices only.)

On touchscreen devices or desktop PCs only: To recalculate the values shown in the current record's calculated field(s), select **Option–Recalculate** or click the **Recalc** button .

Saving Records

Visual CE automatically saves new records or changes to records every time you:

- Scroll to another record
- Insert a new record
- Delete a record
- List all records
- Exit the application

On touchscreen devices or desktop PCs only: To save your work periodically, you can select **Record–Save**.

Deleting Records

Note

If you delete a record that has dependent records (as designated by the form designer using the Jump button's "Cascade deletes" feature), then Visual CE automatically deletes the dependent records as well.

To delete a single record from the table (all device types):

- 1 Display the record in the application window (scroll to it, if necessary).
- 2 Select **Record–Delete**.
Visual CE deletes the record.

To delete all the records from a table (touchscreen and desktop PCs only),
use Clear Table (see page 416).

To delete multiple records from the table (touchscreen and desktop PCs only):

- 1 Switch to the list display. (Select **Options-List** or click the List button )
- 2 Highlight the records to be deleted.
You can select multiple, contiguous records by dragging the stylus or mouse pointer over them, or by holding down the Shift key while you select each one.
- 3 Select **Record-Delete**.
Visual CE asks you to confirm that you want to delete the selected records.

Entering Data into Records

Using Scribble Boxes

Note

This feature is available in Visual CE applications on touchscreen devices and desktop PC. It is not available on non-touchscreen devices.

To make an entry in a Scribble box, simply draw or write inside the box with the stylus or mouse pointer, just as you would with a pencil on paper.



To clear the entire scribble and start over, click the X that appears on the left side of the box.

To erase a portion of a scribble:

- **On an H/PC or desktop PC:** While you press and hold the Shift key, use the stylus or mouse pointer like an eraser. When you release the Shift key, you return to pencil mode. Alternatively, you can press and release the Enter key to toggle between pencil and eraser mode.
- **On a P/PC:** Click the Action button to toggle between pencil and eraser mode.

Scribble boxes that have no border are view-only.

Using Image Controls

Image controls let you select a picture (a .jpg, .gif, or .bmp file) to save with a record.

The screenshot shows a Visual CE application window titled "MLS (2 records)" with a timestamp of "7:42p". The main area contains a form with the following fields:

- Listing Number:** 798562342
- Image Control:** A thumbnail image of a two-story house with a garage.
- New listing:** Single family
Detached
- Comments:** Stunning home with all the amenities.
- Address:** 23 Elm Street

At the bottom of the screen, there is a toolbar with icons for Record, Edit, Option, and other functions.

To clear an image, click the X that appears on the left side of the box.

To select an image:

- 1 Click inside the image control box or press the Space key. Or, if you have a non-touchscreen device, put focus on the Image control box and press the Space key. (If you are creating a new record, click inside the empty box. If you are editing an existing record, click the image to replace it.)

Visual CE displays the file selection window.

Note

If nothing happens when you click an image, the control is "read only." That means the form designer has specified that the images can be viewed, but not changed.

- 2 Select the desired image from the list in the file selection window.

Visual CE closes the file selection box and displays the selected picture in the image control box.

Note

Visual CE stores the image according to the form designer's specifications – Visual CE either copies the image into the table, or stores just the path and filename of the original picture file. If the table stores just the text reference (path and filename), you must be sure that the original picture is maintained in the same location on the device's file system and with the same name. Otherwise, Visual CE will be unable to locate and display the image when the record is opened. Consult your form designer for details.

Tip

If your CE device has a digital camera integrated or connected to it, you can use an Image control to take a picture and save it in a Visual CE record. Refer to "Using Image Controls to Take Pictures," on page 404.

Using Image Controls to Take Pictures with a Digital Camera

If your CE device has a digital camera integrated or connected to it, you can use an Image control to take a picture and save it in a record. All you do is click the control, aim the camera using the Image control as the viewfinder, and click the control to take the picture.

Prerequisites

- The Image control must have been set up by the form designer for this purpose
- The camera must be one of the supported models. As of this writing, the following digital cameras are supported:
 - ◊ HP SDIO 1.3MP digital camera
 - ◊ LifeView FlyCAM 1.3MP digital camera
 - ◊ HP built-in cameras
 - ◊ Cameras that support the Windows CE Camera API
- Refer to the SYWARE website (www.syware.com) for the most up-to-date list.
- The camera manufacturer's software must be installed on the CE device.
- The HP SDIO 1.3MP digital camera requires additional software from SYWARE. For this model in particular, download <http://www.syware.com/download/wce/sdiocam.zip> and, using Active Sync, put it into the \Windows folder on the CE device.
- HP built-in cameras DO NOT require additional software from SYWARE.

Once your system is set up, taking pictures is easy.

To use an Image control to take and save pictures:

- 1 Connect the camera to the CE device.
- 2 Open the form. (In other words, the camera must be connected to the CE device before you open the form.)
- 3 Click inside the Image control box or press the Space key. Or, if you have a non-touchscreen device, put focus on the Image control box and press the Space key. (If you are taking a new picture, click inside the empty box. If you are replacing an existing picture, click the image to replace it.)
- 4 Using the Image control box as the viewfinder, aim the camera.
- 5 To take the picture, click inside the Image control box.

Note

Visual CE stores the picture according to the form designer's specifications – it either puts the picture file into the table, or stores just the pathname of the picture file in the table. In the later case, Visual CE puts the picture file in the same folder as the application (.vce file) and assigns the filename using the convention:
deviceName_dateTimeTaken.

If the table stores just the text reference (file pathname), you must be sure that the original picture is maintained in the same location on the device's file system and with the same name. Otherwise, Visual CE will be unable to locate and display the picture when the record is opened. Consult your form designer for details.

If the picture is stored in the table, it will be an OLE Object (bitmap) in the desktop database when the record is synchronized to the desktop.

Tip

On touchscreen devices and desktop PCs: To clear a picture from the display, click the X that appears on the left side of the box.

Entering Null Values in a Date/Time or Numeric Field

If you don't want to specify a date in a Date/Time field or a value in a numeric field, you can enter a null value.

To enter a null value on a touchscreen device or desktop PC:

- 1 Make the field the focus by pressing the **Tab** key on your keyboard until the field is highlighted. (You enter null values directly into the field, not via the Calendar dialog box or number pad.)
- 2 Press **Delete** on the keyboard.

To enter a null value on a non-touchscreen device:

- Set focus on the field and check the checkbox.

Entering Negative Numbers and Decimal Points (Non-Touchscreen Devices)

When entering numbers in a number field on a non-touchscreen device, you can use the * key to enter a minus sign for a negative value and the # key for a decimal point.

Using Word Lists in an Edit or Note Box

Note

This feature is available in Visual CE applications on touchscreen devices and desktop PC. It is not available on non-touchscreen devices.

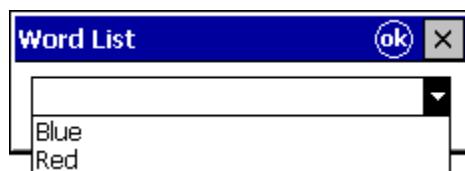
A word list is a set of words or phrases that you can insert into a edit or note box by simply picking entries from a pop-up display, rather than having to type them from scratch.

Word lists are intended to eliminate the need to type commonly-used text. They are not restrictive—you can type your own text into the edit or note box and simply chose from the pop-up list as needed. You can select any number of items from the word list, and modify your selections as desired. You can even add your own items to the word list for use in the future.

To insert text from a word list:

- 1 Double-click inside an edit or text box:

Visual CE displays the pop-up word list. Note: If nothing happens when you double-click, then the form designer has not associated a word list with the box.



- 2 Select an item from the list.
- 3 Visual CE inserts your selection into the edit box in the Word List window. You can either:
 - Click OK to insert your selection into the form's edit or note box, as is.
 - Modify the text and click OK. Visual CE asks if you wish to save the modified text in the word list. If you click Yes, Visual CE inserts the modified text into the form's edit or note box, and adds it to the word list for future use. If you click No, Visual CE just inserts the text into the form's edit or note box.

Copying or Moving Data Between Records

Note

This feature is available in Visual CE applications on touchscreen devices and desktop PC. It is not available on non-touchscreen devices.

You can copy or move data from one table record to another using the Visual CE Edit menu's cut, copy, and paste commands.

Tip

You can use the following instructions to copy data on a field-by-field basis. Depending on your needs, you may find it easier to duplicate an existing record. See "Creating a New Record by Duplicating an Existing Record," on page 398.

Command	Function
Cut	Deletes the currently selected information from the edit box.
Copy	Copies the currently selected edit box information into the Windows clipboard.
Paste	Inserts information from the Windows clipboard into the current edit box.

To copy data from one record to another:

- 1 Display the record that contains the data you want to copy (scroll to the record, if necessary).
- 2 Drag the data to highlight it, and select **Edit–Copy**. Or, tap and hold in the desired field and select **Copy** from the menu that appears.
- 3 Scroll to the record to which you want to copy the data.
- 4 Click the field in which you want the data, then select **Edit–Paste**. Or, tap and hold in the desired field and select **Paste** from the menu that appears.
Visual CE pastes the copied data into the field.

To move data from one record to another:

- 1 Display the record that contains the data you want to move (scroll to the record, if necessary).
- 2 Drag the data to highlight it, and select **Edit–Cut**. Or, tap and hold in the desired field and select **Cut** from the menu that appears.
- 3 Scroll to the record to which you want to insert the data.
- 4 Click the field in which you want the data, then select **Edit–Paste**. Or, tap and hold in the desired field and select **Paste** from the menu that appears.
Visual CE pastes the moved data into the field.

Using a Bar Code Scanner with Visual CE

A Visual CE form can receive input from most bar code scanners, including those from Symbol Technologies (<http://www.symbol.com>), Socket Communications (<http://www.socketcom.com>), and Intermec (<http://www.intermec.com>).

When using the form, simply click on the control that accepts the scanned input, and then press the "scan" button on the scanner.

Notes for Symbol, Socket, Intermec Scanners:

The following notes pertain if you are using a Symbol, Socket, or Intermec scanner:

- While running your form, the barcode reader is automatically enabled whenever the focus is on an edit control for text input. If focus is on an edit control for numeric or date input, or any control other than an edit control, the barcode reader is automatically disabled.
- If the form designer placed a timestamp, jump, or command button in the control order after the control that receives the scanned input, Visual CE automatically activates that button after receiving the scanned input.
- To trigger a Socket In-Hand Scan Card, click the edit control.

Using Alarms

Setting and Responding to Alarms

Your form designer may have provided an Alarm button that you can use to set an alarm. For each record, you can specify a different date and time that the alarm is triggered.

When the alarm goes off, Visual CE beeps and displays a message. The message is set by the form designer, as is the defer time (similar to a snooze button). See your form designer for specifics.

The form need not be running for the alarm to go off. As long as any Visual CE application is running, the alarm will display at the specified time.

SalesContacts (1 records) 6:05p

Name
Ferguson

Phone
555 555-1212

NextContact
11/18/03 10:30:00 AM

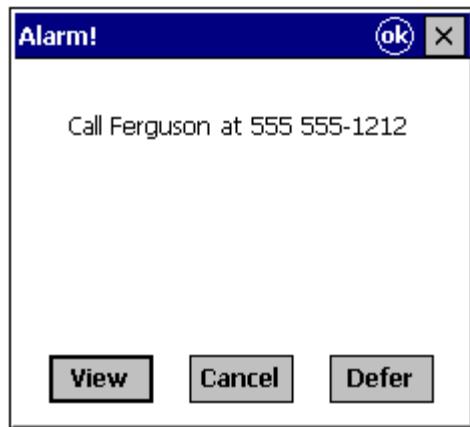
Remind me

Notes
Invite to open house.

Record Edit Option

To set an alarm:

- 1 Using the control provided by the form designer, specify the date and time you want the alarm to go off. If the designer did not provide a separate control, Visual CE will prompt you in the next step.
 - 2 Click the alarm button to activate the alarm.
 - 3 If you did not specify a date/time in Step 1, Visual CE displays the calendar box. Enter the date and time you want the alarm to go off, and click **OK**.
- When the time you specified matches the current time according to the device's clock, Visual CE beeps and displays the Alarm window.

**To respond to an alarm:**

The Alarm window displays a message supplied by the form designer, and three buttons:

- **View:** Click to display the record that triggered the alarm.
- **Cancel:** Click to deactivate the alarm and close the window.
- **Defer:** Click to close the window. The alarm will go off again, after a time period specified by the form designer.

Printing Records

Printing a Record

Note

This feature is available in Visual CE applications on touchscreen devices and desktop PC. It is not available on non-touchscreen devices.

You can print a paper copy of a selected record. The print includes only record fields whose control property is set to "Include when printing." (This property must be set in Visual CE's development environment. Refer to "Changing a Control's Properties," on page 264 and the description of the control type in the section on creating controls, which begins on page 166.)

Tip

If Report CE is installed on your CE device, you can print the currently displayed record in a Report CE layout by selecting Visual CE's Record-Print command. Refer to the topic, "Using Report CE to Print a Visual CE Record," on page 413 for instructions on setting this up.

To print a record:

- 1 Display the record that you want to print. Alternatively, you can highlight its entry in the record list.
- 2 Select **Record-Print** or click the **Print** button .

If the record does not print, verify that your printer is properly connected. Try printing a test sample using Word, Pocket Word or a similar program. If this test fails, contact Microsoft.

Using Report CE to Print a Visual CE Record

Note

This feature is available in Visual CE applications on touchscreen devices and desktop PC. It is not available on non-touchscreen devices.

SYWARE's Report CE software enables you to design, view, and print customized reports of data that you have collected with Visual CE. If you have Report CE, you can organize your system so that Visual CE's Record-Print command automatically prints the currently displayed record in a Report CE layout.

To set up:

Create the report template (.rce file) according to the instructions in the Report CE documentation. Be sure to:

- Give the report template the same name as the corresponding Visual CE form, but with the .rce extension (e.g., MyStuff.rce and MyStuff.vce)
- Save the .rce and .vce files in the same folder
- The Visual CE application table and the top level table for the Report CE template (.rce file) must be the same table.

If you are running Visual CE on the desktop PC:

Report CE reports data from tables on the CE device, regardless of whether you are running the Visual CE application on the desktop PC or the CE device. That means the CE device must be connected throughout the following procedure. In addition, keep in mind that the data displayed in the report will come from the table on the CE device--not from the table on the desktop PC.

To print a record in the layout:

1 Display the record that you want to print. Alternatively, you can highlight its entry in the record list.

2 Select **Record-Print** or click the **Print** button .

Visual CE launches the .rce file that shares the same name as the form. The report displays just data for the currently selected record.

Exporting and Importing

Export Table

On the CE device, you can export the data the form is using to an ASCII comma delimited text file. You can then email your data and/or import it into other applications. This feature is also useful for archiving records.

You can export a filtered list or the entire contents of the table.

Notes

- The Export feature is available in the default version of Visual CE applications on touchscreen devices and desktop PC. It is not available on non-touchscreen devices.
- The form designer may have set up the Export command to create an XML file (.xml) rather than an ASCII text file.

To export records to an ASCII file:

- 1 Open the Visual CE application on the CE device.
- 2 Optional: If you want to export a subset of the table records, apply a filter (either custom or pre-defined).
- 3 Select **Options-Export**.
Visual CE exports the file.

After the table is exported, you may want to delete all records in the table (see page 416). (Be sure to keep the archived records in a safe place.) If you don't delete the records, and you later import the file back in, you will have two of every record.

Notes

- If a form based on a table is running, such as Myform.vce based on MyTable, Visual CE automatically names the exported file *table_name.txt*. (e.g., MyTable.txt) and places it in the same folder as the form. However if the form designer provided a command button to export records, any file pathname specified as part of that command overrides these defaults.
- The form designer may have set up the Export command to export values from all table columns, or just the values from table columns that are associated with controls on the form.
- All date/time values are automatically exported in the following format:
YYYY-MM-DD HH:MM:SS.
- Pictures are not exported to text files.
- If the export is for all table columns, values are exported in the order of the table columns. If the export is for just table columns associated with form controls, values are exported in the same order as the control order on the form. Refer to "Specifying Control Order," on page 276.

Import an ASCII File

You can import an ASCII comma delimited text file into a table on your CE device. Typically, this file will have been created using the Export Table command.

Notes

- The Import feature is available in the default version of Visual CE applications on touchscreen devices and desktop PC. It is not available on non-touchscreen devices.
- The form designer may have set up the Import command to import an XML file (.xml) rather than an ASCII text file.
- If a form based on a table is running, such as Myform.vce based on MyTable, Visual CE assumes that the text file to import is named *table_name.txt* (e.g., MyTable.txt), and looks for it in the same folder as the form. However if the form designer provided a command button to import records, any file pathname specified as part of that command overrides these defaults.
- Visual CE considers any of the following date/time formats valid and imports values that conform to them, as is:
 - ◊ YYYY-MM-DD
 - ◊ YYYY-MM-DD HH:MM
 - ◊ YYYY-MM-DD HH:MM:SS
 - ◊ YYYY-MM-DDTHH:MM:SS
 - ◊ Formats specified via the Windows Regional Settings.
- To determine the proper format for an import file of either text or XML, including field order, export the current contents of the table to the desired file type and examine the results. (To export a file, open the form on the CE device and select **Options-Export**.)
- Pictures cannot be imported from a text file.

To import the file:

- 1 Make sure that the filename matches the name of the target table. Also, both the import file and the application must be located in the same folder.
- 2 Open the Visual CE application on the CE device.
- 3 Select **Option-Import**.
The records are now available to the application.

Clear Table

Note

This feature is available in Visual CE applications on touchscreen devices and desktop PC. It is not available on non-touchscreen devices.

To delete all records in the table, select **Options-Clear**.

Beaming Records from One CE Device to Another

Note

This feature is available in Visual CE applications on touchscreen devices and desktop PC. It is not available on non-touchscreen devices.

Using an IR connection, you can copy records from one CE device to another CE device running the same application. The application must be open on both devices while the process is taking place. You can beam a single record, the records in a filtered list, or the entire contents of the application's table.

Note

Visual CE copies records from the sending device to the receiving device, regardless of whether or not the records already exist in the table on the receiving device. That means you could end up with duplicates on the receiving device.

To beam records:

- 1 Open the same Visual CE application on both devices.
- 2 Optional: You can beam a subset of the table's records:
 - To beam a single record: Display the record on the sending device
 - To beam a filtered list of records: Apply the filter (either custom or pre-defined) to the records on the sending device
- 3 In Visual CE on the sending device, select either:
 - **Edit-IR Beam:** Copies the current record.
 - **Edit-IR Beam All:** Copies all records in a filtered list (if applied in Step 2) or all records in the current table to the receiving device.
- 4 In Visual CE on the receiving device, select **Edit-Receive IR Beam**.

Visual CE and mEnable

Using Visual CE Applications with mEnable

mEnable is a separate SYWARE product that makes it possible for Visual CE users to access and update data on remote servers – interactively and in real time—via wired or wireless connections including the Internet.

Depending upon how the Visual CE developer set up your application, it may work with mEnable in **Occasionally Connected** or **Always Connected** mode.

When you use a Visual CE application with mEnable in Occasionally Connected mode, the application works with data in the local table, while mEnable works in the background to synchronize each record that the application reads or writes with the database on the server. If the connection with the server is broken, either intentionally or accidentally, you can continue to work with data on the client (CE device). Later, you can run a full synchronization to bring the local table and/or server database up to date.

In Always Connected mode, you are always working with the most-up-to-date records on the server. No data is stored on the CE device. If the connection with the server is broken, you must wait until it is restored.

Note

mEnable must be installed on both the client device and the server computer. While you are using Visual CE with a live mEnable connection, the mEnable Server software must be running on the server computer. Please refer to the *mEnable User's Guide* for complete set up instructions.

The form designer may have supplied customized command buttons that you can use to control the mEnable connection, or they may have incorporated a macro that starts and controls the connection "behind the scenes."



Alternatively, the form designer may have supplied an mEnable stoplight for monitoring the mEnable connection and possibly controlling it.



- | | |
|---------------|--|
| Red | Disconnect from the server database (work on local data only). |
| Yellow | Execute full synchronization between all eligible local tables (not just the current table) and the server database. You must wait while the light is yellow. |
| Green | Establish an mEnable connection to the server database (every time the client reads or writes a record from/to the local table, that record is synchronized with the server database). |

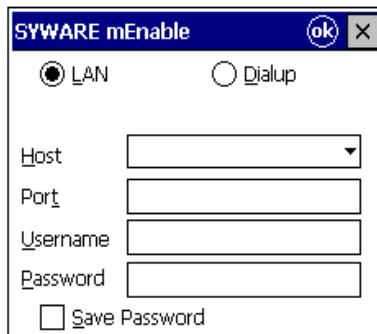
Depending upon how the form designer set up the stoplight, you may be able to tap the lights to control the connection and synchronization, or it may be there to simply inform you of the status.

Additionally, the stoplight indicates the strength of the WI-FI signal. A **blinking yellow light** means you are nearly out of range and should move to an area with a stronger connection if possible. You can continue to work but the transfer may be slower than normal. A **blinking red light** means you are out of range and have lost connection.

To start an mEnable session (either mode):

- 1 If your form works in Occasionally Connected mode, click the yellow or green stoplight (or select an mEnable command button provided by the form designer). If your form works in Always Connected mode, an mEnable session starts automatically when you open the form.

Depending upon how the form designer set up the application, mEnable may display the log on dialog box.



Note

Depending upon how the form designer set up the application, the log on box may appear just the first time you start an mEnable session on the CE device, so that you can specify the connection settings. From then on, mEnable will use those settings and the log on box will no longer appear so that you need not re-enter the settings each time.

It is also possible that the form designer supplied all values for you, so that you never see the logon box.

- 2 Specify the connection to the server:

- **LAN or Dialup:** Select your network connection type -- LAN for local area network, or Dialup for an RAS dialup connection.
- **Host:** The IP address of the server (for example, 208.77.50.186).
- **Port:** Leave blank (Visual CE provides this value automatically).
- **Username and Password:** Enter the values required for your database connection.
- Click **OK**.

- 3 *For users with Occasionally Connected mode:* If you had selected the yellow stoplight or an "mEnable synchronize" command button (full synchronization) in Step 1, mEnable synchronizes the records on the CE device with those on the server, and vice versa. This can take some time; you must wait while the stoplight is yellow. If you had selected the green stoplight or an "mEnable on" command button, mEnable asks if you want to perform full synchronization. In most cases, it is recommended that you respond **Yes**. If you know that the databases are already synchronized, or if you want to synchronize only the records you modify in this session, select **No**.
- 4 When you have completed your mEnable session and want to disconnect from the server, click the red stoplight.

To close an mEnable session (either mode):

If your form works in Occasionally Connected mode, click the red stoplight or the "mEnable off" command button. If your form works in Always Connected mode, the mEnable session closes automatically when you close the Visual CE application.

Running Visual CE from Another Application

Running a Visual CE Application from a Visual Basic or C/C++ Application

If you launch a Visual CE application from a Visual Basic or C/C++ application (using ShellExecuteEx or CreateProcess), you can use any of the following optional switches on the command line: /F, /f, or /a. /F opens the display to the first record that has a specified value in the sort column; /f filters the records so that only those records that have a specified value in the sort column are displayed; /a supplies a value for a given global variable.

The commands use the following syntax:

/F*value application_name.vce*

Opens the Visual CE application to the first record that has *value* in the "sort on" column in the application table. No filter is applied. (For information about designating the sort column, refer to the topic, "Specifying an Initial Sort/Search Ordering," on page 282.) If the value contains spaces, enclose it in double quotes. The value may be either a constant, an expression that returns a constant, or a global variable.

For example:

/F "John Smith" MyContacts.vce

If MyContacts.vce has "sort on" set to the "name" column, then the first record displayed will be the one for John Smith. All other records in the table will be available for viewing and modification.

/f*filtervalue application_name.vce*

Restricts the display to just those records that have *filtervalue* stored in the "sort on" column in the application table. (For information about designating the sort column, refer to the topic, "Specifying an Initial Sort/Search Ordering," on page 282.) If the filter value contains spaces, enclose it in double quotes. The filter value may be either a constant, an expression that returns a constant, or a global variable.

Example 1:

/f13345 MyOrders.vce

If MyOrders.vce has "sort on" set to the "partNumber" column, then the application will show only information about part number 13345.

Example 2:

/f "John Smith" MyContacts.vce

If MyContacts.vce has "sort on" set to the "name" column, then the application will show only information about John Smith.

Example 3:

/f@var(0) MyContacts.vce

If MyContacts.vce has "sort on" set to the "name" column, and @var(0) returns John Smith, then the application will show only information about John Smith.

/a*variable value application_name.vce*

Supplies a value for a given global variable.

Examples:

/a20 13345 MyOrders.vce

/a21 "Fred Smith" MyContacts.vce

Part V:

Desktop Connectivity

Desktop Connectivity

Overview of Part V

Visual CE offers a wide range of options for moving data between CE devices and the desktop PC; between Visual CE applications that share the same table structure; and even between Visual CE applications and other database and spreadsheet programs such as dBase, Microsoft Access, and Microsoft Excel.

The following options are available:

- Downloading a Desktop PC Table to a Windows CE Table
- Downloading Data from an Excel Spreadsheet to a Windows CE Table
- Importing a Windows CE Table into Microsoft Excel
- Importing a Windows CE Table Into Microsoft Access
- Linking a Windows CE Table to Microsoft Access
- Exporting a Windows CE Table to an ASCII Text or XML File
- Importing an ASCII Text or XML File into a Windows CE Table

Downloading a Desktop PC Table to a Windows CE Table

You can quickly download a table (and any data that it contains) from the desktop PC into a new Windows CE table. You can then create a Visual CE form for the table as you normally would.

This process works for tabular files created with any ODBC-enabled desktop PC application, including Microsoft Access, Excel, or FoxPro.

Refer to the topic, "Downloading a Table from an ODBC-Enabled Application," on page 146 for instructions.

Downloading Data from an Excel Spreadsheet to a Windows CE Table

If you simply want to copy data (one time) from an Excel spreadsheet to a Windows CE table on the CE device, follow these steps:

- 1 Open the spreadsheet in Excel, select the desired data, and give it a name (**Insert – Name – Define**). Close Excel.
- 2 In the Visual CE development environment, select **File–Download Table**.
- 3 When asked if you want to keep the data synchronized, click **No**.
Visual CE displays the Download Table dialog box.
- 4 For "Desktop Data Source", select **Excel Files**. For "Desktop Database", select the .xls file you created in Step 1. Click **OK**.
Visual CE displays the Select Table box.
- 5 Select the name you defined in step 1 and click **OK**.
Visual CE downloads the table to the CE device.

Importing a Windows CE Table into Microsoft Excel

You can import data collected on your CE device to a Microsoft Excel spreadsheet using ODBC.

While in Microsoft Excel, follow these steps to import data from a Windows CE table:

- 1 Select **Data-Get External Data-Create New Query**. If you do not see this menu item, Microsoft Query is not installed. Install Microsoft Query from your Excel setup disks.
- 2 Select the **Databases** tab.
- 3 Select **Visual CE Data Source** listbox item and click **OK**.
- 4 Under "Available tables and columns" select the Windows CE table to import, click **>**, then click **Next**.
- 5 Click **Next** through three consecutive dialog boxes.
- 6 Click **Finish**.

Importing a Windows CE Table Into Microsoft Access

You can import data collected on your CE device into a Microsoft Access table using ODBC.

Note

This process copies data from the Windows CE table into a table in the Access database. Subsequent changes made to the data in the Microsoft Access table will not appear in the Windows CE table, and vice versa. If you want to keep the two tables up-to-date with one another on an on-going basis, see the topic, "Selecting Table Synchronization Options," on page 435 instead.

While in Microsoft Access, follow these steps to import data from a Windows CE table into a Microsoft Access table:

- 1 From **File-Get External Data** select **Import**.
- 2 From **Files of Type** (the drop down list in the lower left-hand corner), select **ODBC Databases**. "ODBC Databases" is the last item in the list; you may have to scroll the list to bring it into view.
- 3 Select **Visual CE Data Source** listbox item and click **OK**.
- 4 Select the Windows CE table to import and click **OK**.

Linking a Windows CE Table with Microsoft Access

You can establish a link between a Microsoft Access table on the desktop PC and a Visual CE table on the CE device. Any changes made to the Windows CE table are immediately visible on the desktop PC. Keep in mind that the CE device must be connected to the desktop PC whenever you use the table.

While in Microsoft Access:

- 1 From **File-Get External Data** select **LinkTables**.
- 2 From **Files of Type** (the drop down list in the lower left-hand corner), select **ODBC Databases**. "ODBC Databases" is the last item in the list; you may have to scroll the list to bring it into view.
- 3 Select **Visual CE Data Source** listbox item and click **OK**.
- 4 Select the Windows CE table to link and click **OK**.

Exporting a Windows CE Table to an ASCII Text or XML File

While running a Visual CE form on a CE device, you can export the data that the form is using to an ASCII comma delimited text or XML file. See "Export Table" on page 414.

Importing an ASCII Text or XML File into a Windows CE Table

You can import an ASCII comma delimited text or XML file into a Windows CE table for use with Visual CE. See "Import File" on page 415.

Part VI:

Synchronizing Tables

Table Synchronization

Overview of Part VI

When you synchronize tables, changes made to the data in one are applied to the other. This section explains how to set up and take advantage of this feature.

Topics covered in this section:

- Selecting Table Synchronization Options
- Using Timestamp-Based Synchronization
- Synchronizing Multiple Handhelds
- Synchronizing Tables On Command Using Visual CE
- Synchronizing Tables Using Vicesync
- Synchronizing Tables Using mEnable
- Synchronizing Tables with Excel Spreadsheets

Selecting Table Synchronization Options

Synchronization enables you to maintain and use the same data in tables on both the desktop PC and CE device – since the changes made to the data in one table are automatically applied to the other.

Visual CE gives you complete control over the synchronization process. For example, you can specify whether modifications are copied just from the desktop PC to the CE device; just from the CE device to the desktop PC; or to and from both machines simultaneously. You can move records from the CE device to the desktop PC. You can even distinguish between records that have been deleted, changed, and added, and how such records should be handled in both directions.

You can set up synchronization for any number of pairs of desktop/handheld tables and apply different synchronization options to each pair. You can also set up synchronization between one desktop table and multiple handhelds. Additionally, you specify whether synchronization occurs automatically whenever MobileDevices/ActiveSync starts, or just when you explicitly apply the Synchronize command.

This topic explains how to set up and fine-tune the synchronization options to fit your needs.

What Tables Can be Synchronized?

Any table created using Visual CE's "File–Create Table" or "File–Download Table" options can be synchronized. (Windows CE tables created in other ways cannot.) The desktop PC table can be any tabular file created with an ODBC-enabled desktop application such as Microsoft Access.

Typically, you create a table on the desktop PC and copy it to the CE device (or vice versa) using one of the techniques described in the Desktop Connectivity section.

Synchronizing Multiple Handhelds (Pro and Mobile 360 Editions Only)

If you have the Pro or Mobile 360 Edition of Visual CE, you can set up synchronization between a table on the desktop PC and multiple CE devices. Refer to the topic, "Synchronizing Multiple Handhelds," on page 440 for instructions.

Prerequisites

Before you can set up and apply synchronization:

- The first column of the desktop PC table must be an "OID" column. (Visual CE automatically adds a corresponding column to the Windows CE table on the CE device and uses the columns to keep records in the two tables in sync.) If you have not already done so, open the desktop table in the application that was used to create it, e.g., Microsoft Access. Add a new first column to the table with the following specifications (use default values for any other settings):

- Field name: OID
- Data Type: Number
- Field Size: Long Integer (for Access tables)
- Indexed: No

When creating new records in the desktop PC table, always set the OID column to null (blank) or zero.

- You must have a Level 2 read/write ODBC Driver for the desktop PC database and an ODBC Data Source built on that ODBC Driver. See the documentation for your ODBC Driver (provided by your database vendor) for installation and setup instructions.
- SQL Server tables must have a primary key defined.

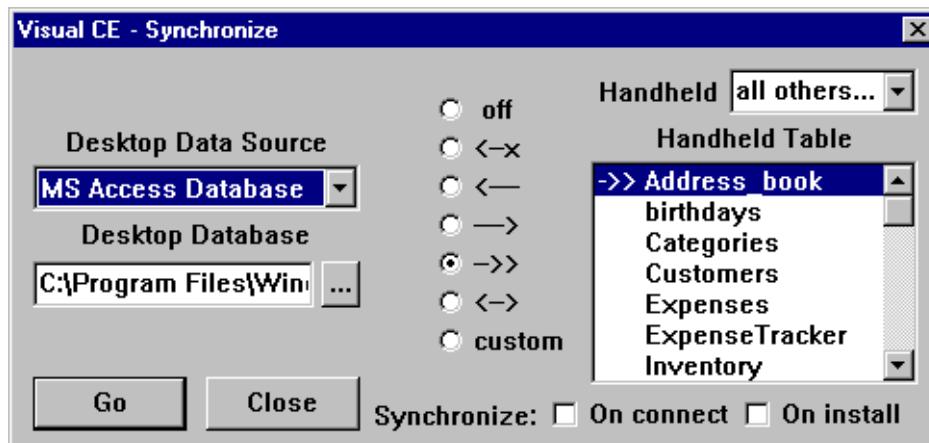
Note

Your CE device must be connected to your desktop PC throughout the following procedure.

To specify synchronization options:

- 1 Open Visual CE on your desktop PC. Then, select **File—Synchronize**.

The Visual CE Synchronize dialog box appears.



- 2 **Handheld:** If you have just one CE device, ignore this field. If you want to use the same synchronization settings for all of your devices, leave this set to “all others...”. Otherwise, select the name of the CE device to which the synchronization settings will apply.
- 3 **Desktop Data Source:** Select the ODBC data source type.
- 4 **Desktop Database:** Select the desktop database that contains the tables you want to synchronize.

- Click the Browse button ..., and navigate to the folder that contains the database
- Select the database, then click **Open**

Notes

- If the data source that you selected in Step 3 intrinsically identifies this database, leave the Desktop Database field blank—see your database documentation.

Tip

- It is assumed that most of the tables on the handheld will be synchronized with tables contained in a single desktop database (the one you pick here). However, you can override this selection for individual handheld tables using the Custom Synchronization options described in the following topic (on page 438).

- 5 **Handheld Table:** Select the Windows CE table that you want to specify synchronization settings for.
- 6 Click a radio button to specify how Visual CE will synchronize data between the Windows CE table you just highlighted and its corresponding table on the desktop PC:

Use this radio button**To do this**

off

Prevent synchronization of the Windows CE table with the desktop PC table. Any changes made to records in one table are NOT applied to the other. This is the default.

←x

Copy new or modified records from the Windows CE table to the desktop PC table, and delete them from the

	Windows CE table (e.g., move records from the handheld to the desktop PC).
←	Apply all changes to records in the Windows CE table to the desktop PC table. Changes in the desktop PC table are NOT applied to the Windows CE table.
→	Apply all changes to records in the desktop PC table to the Windows CE table. Changes in the Windows CE table are NOT applied to the desktop PC table.
⇒	Publish. Replace all records in the Windows CE table with all records from the desktop PC table.
↔	Apply all changes to records in the Windows CE table to the desktop PC table and vice-versa (e.g., all changes appear in both tables).
custom	<p>Opens the [table] Synchronization dialog box, which enables you to further fine-turn the synchronization options. Refer to the topic, "Custom Synchronization Option," on page 438.</p> <p>Tip: Even if you don't modify the settings, this dialog box is useful for viewing the specific actions associated with each of the standard options listed above (off, <-, etc.)</p>

Notice that Visual CE puts the synchronization symbol next to the table name. (A straight bar, —, indicates that you have specified a custom option.) Whenever synchronization occurs, Visual CE applies the option indicated by that symbol.

- 7 Repeat Steps 3 through 6 for every Windows CE table that you want to synchronize with a table on the desktop PC.
- 8 If you want synchronization to happen automatically every time MobileDevices/ActiveSync starts, click the **Synchronize on connect** checkbox. Otherwise, you must specifically tell Visual CE to synchronize the tables whenever you want the process to occur. See "Synchronizing Tables On Command," on page 441.
- 9 **(Pro and Mobile 360 Users Only)** If you select **Synchronize on install**, the runtime installer will automatically synchronize the newly installed table on the CE device with the desktop PC table as part of the installation process. Refer to the topic, "Distributing an Application," on page 298 for more information.
- 10 Click **Close** to save your settings and close the dialog box; or click **Go** to save your settings and run synchronization now.

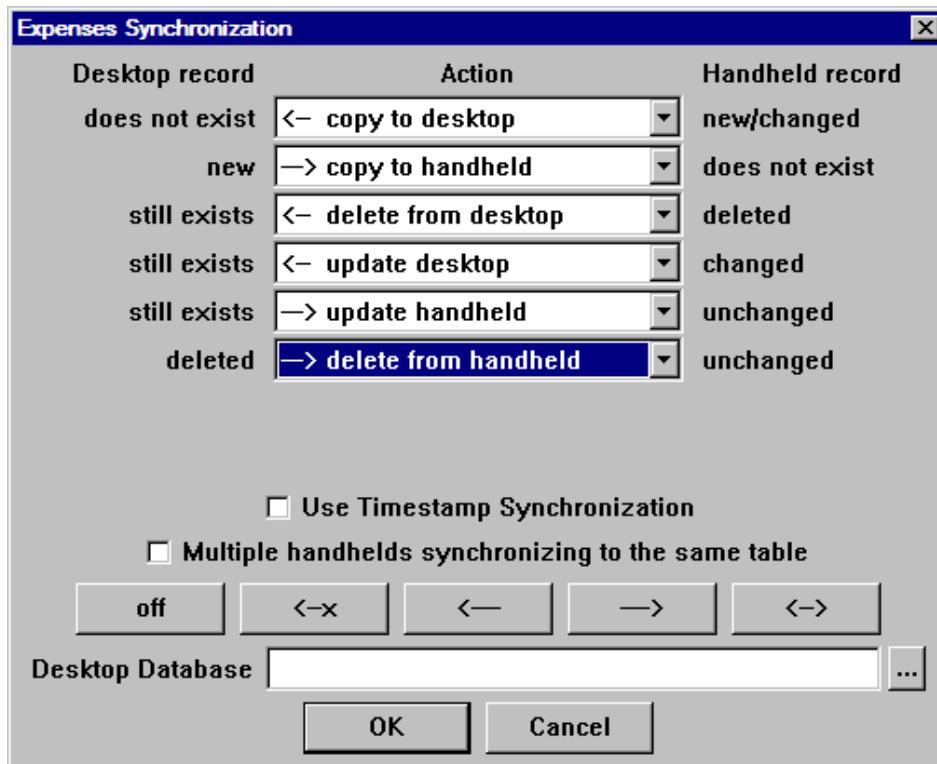
From now on (or until you change the settings), Visual CE synchronizes the tables according to the settings you just specified.

Note

When creating new records on the desktop PC, always set the value in the OID column to null (blank) or zero.

Custom Synchronization Option

Visual CE's Synchronize dialog box offers a "custom" option. When you select that option in the Synchronize dialog box (page 436), Visual CE displays a second dialog box that enables you to fine-tune how updates are copied from one table to another.



You can fine-tune the settings for specific conditions (e.g., desktop record does not exist) using the drop-down arrows at the right side of the Action column. To re-apply the set of default actions associated with one of the standard synchronization options across all conditions, click the corresponding button near the bottom of the dialog box (**off**, **<-x**, **<-**, **→** or **<→**). The standard synchronization options associated with each of these buttons are described near the end of the previous topic, "Selecting Table Synchronization Options."

Timestamp Synchronization

If you want to use timestamp-based synchronization, select the **Use TIMESTAMP Synchronization** checkbox. Timestamp-based synchronization is faster and provides greater control than regular synchronization but requires additional effort to maintain the timestamps. (These timestamps are unrelated to Timestamp controls.) See "Using Timestamp-Based Synchronization," on page 439 for details.

Multiple Handhelds Synchronizing to the Same Table (Pro and Mobile 360s Only)

If you have the Pro or Mobile 360 Edition of Visual CE, you can synchronize a table on the desktop PC with any number of CE devices. For instructions on applying this feature, refer to the topic, "Synchronizing Multiple Handhelds," on page 440.

Desktop Database

You can use this optional setting to override the value for the Desktop Database field in the Visual CE Synchronize dialog box for the table selected in the Handheld Table window (page 436).

Using Timestamp-Based Synchronization

When synchronizing, Visual CE normally compares every record in the desktop PC tables to every record in the CE device tables to determine what has changed. Even if only a few records have changed, this process can take a long time because the synchronizer must check all records, even those that did not change.

If you know that only a few records have changed, you can minimize synchronization time by using a technique called Timestamp-Based Synchronization.

Note

Timestamp-Based Synchronization is completely unrelated to Timestamp controls on forms.

Timestamp-Based Synchronization relies on you, the designer, to designate which records on the desktop PC have changed. You do this by adding an additional column at the end of the table in the desktop PC database. The name of this column must be **TIMESTAMP** and it must be a date/time column.

Whenever you create or modify a record on the desktop PC, you must set the value for that record in the **TIMESTAMP** column to the current date and time. This is your signal to the synchronizer that the desktop record is new or changed.

Do NOT make any modifications to the table or records on the CE device to accommodate Timestamp-Based Synchronization. Visual CE knows what records it has changed there. In particular, do NOT add a Timestamp column to the table on the CE device.

Other than setting the **TIMESTAMP** column on the desktop PC when you create or modify a record, you do nothing differently. When you synchronize, Visual CE reads these timestamps and resets them as need be.

Using timestamps is optional and involves certain tradeoffs:

- **If you use timestamps, note the following:**

- ◊ You must set the timestamps on the desktop PC database every time you update a record. Failure to set the timestamp causes Visual CE to ignore your desktop PC updates.
- ◊ You must set the timestamps on the desktop PC database every time you insert a record. Failure to set the timestamp causes Visual CE to delete inserted records at synchronization time.

- **If you do not use timestamps, note the following:**

- ◊ Synchronization is slower.
- ◊ If you update a record on both the CE device and the desktop PC, the CE device record overwrites the desktop PC record. (If you use a timestamp column, Visual CE asks which record to use.)
- ◊ If you update a record on the desktop PC and delete it on the CE device, Visual CE deletes the record from the desktop PC. (If you use a Timestamp column, Visual CE copies the record from the CE device to the desktop PC.)

About timestamp synchronization and existing desktop PC tables:

If you enable timestamp synchronization for a desktop PC table that already exists, you must manually add a Timestamp column to the table before synchronizing it. The column must meet the following requirements:

- It must be the last column in the table
- It must be named **TIMESTAMP**
- Its data type must be defined as Date/Time

If you disable timestamp synchronization for a desktop PC table, you must manually remove the Timestamp column from the table before synchronizing it again.

Synchronizing Multiple Handhelds (Pro and Mobile 360 Editions Only)

If you have the Pro or Mobile 360 Edition of Visual CE, you can synchronize a table on the desktop PC with the corresponding table used on any number of CE devices.

To set up synchronization for multiple handhelds:

- 1 Assign a unique name to each CE device:
 - **On a P/PC:** Click the Windows **Start** button, then select **Settings – System – About - Device ID**. Then, enter a unique name in the **Device name** field.
 - **On an H/PC:** Click the Windows **Start** button, then select **Settings - Control Panel – Communications – Device Name**. Then, enter a unique name in the **Device Name** field.
- 2 Set up synchronization between the desktop PC table and the table on one CE device as you normally would, following the instructions in the earlier topic, "Selecting Table Synchronization Options," on page 435.
- 3 In the Custom Synchronization dialog box (page 438) on the desktop PC, check the **Multiple handhelds synchronizing to the same table** option.

You can now synchronize the desktop PC table with any of the CE devices.

Synchronizing Tables On Command Using Visual CE

Once you've set up the synchronization settings for a table on the desktop PC and a table on the CE device, keeping the two sets of information up-to-date with one another is a simple process. (See "Selecting Table Synchronization Options," on page 435 if you have not already performed an initial synchronization.)

Notes

- If you checked the **Synchronize on connect** option when you specified the settings for the two tables, you do not need to do the following since synchronization will occur automatically.
- Two methods are available for synchronizing on command: using the Visual CE development environment (described here), and using the Vicesync.exe program (described in the following topic).
- You must be working on the desktop PC in connected mode.

To synchronize tables on command using the Visual CE development environment:

- 1 Open Visual CE on the desktop PC.
- 2 Open the application for the table you want to synchronize.
- 3 Select **File-Synchronize**.
- 4 Click **Go**.

Visual CE synchronizes the two tables according to the options specified.

Note

- When creating new records on the desktop, always set the OID column to null (blank) or zero.

Synchronizing Tables Using Vicesync

Vicesync.exe makes it possible to synchronize tables without opening the Visual CE development environment. When you launch this program, it automatically synchronizes all eligible tables on both the desktop PC and the CE device. If a table is missing, it offers to create it.

Pro and Mobile 360 Edition Users: Vicesync is included as part of the distribution files. Refer to the topic, "Creating Distribution Files," on page 298 for details.

Notes

- Before running Vicesync.exe, you must have already set up the synchronization options using the Visual CE development environment, as described in the topic, "Selecting Table Synchronization Options," on page 435.
- You must be working on the desktop PC in connected mode.

To synchronize tables using Vicesync.exe:

- 1 Locate **Vicesync.exe** on the desktop PC (the default location is \Program Files\Windows CE Services\SYWARE Visual CE).
- 2 Launch Vicesync.exe.
Vicesync automatically synchronizes all eligible tables on the desktop PC and CE device. It first displays a message on the CE device alerting you that it has begun the synchronization process. As it synchronizes the tables, it notes each table with a message displayed on the desktop PC. If it encounters any problems (e.g., a table is missing), it will stop the process and display a dialog box on the desktop PC for your response. When the process has completed successfully, Vicesync displays the message "Synchronization complete" on the CE device.
- 3 Click **OK** to clear the messages on the CE device.

Note

- When creating new records on the desktop, always set the OID column to null (blank) or zero.

Synchronizing Tables Using mEnable

With mEnable, a separate SYWARE product, you can synchronize tables on the CE device with tables on a remote server in "continuous" or "full" mode. In "continuous" mode, each change made to one table is immediately reflected in the other, so that you are always working with up-to-the-second information. "Full" mode is the same as the standard Visual CE synchronization described in earlier topics.

You can use mEnable with a wired or wireless connection, including the Internet.

For more information about mEnable, refer to the topic, "Using Visual CE Applications with mEnable" on page 418 or the *mEnable User's Guide*.

Synchronizing Tables on the CE Device with Excel Spreadsheets on the Desktop PC

It is possible, although not recommended, to use Excel files as the desktop database and synchronize them with tables on the CE device using mEnable or Visual CE synchronization.

This requires some set up:

- Visual CE does not work with Excel files by default, but you can change this by editing the WCEODBC.INI file that is located in the Windows folder on the desktop PC. (You may wish to make a backup copy of this file before changing it.) Using a text editor such as Notepad, open the file and locate the line that reads [VICESYNC]. Just below that line, insert a new line that reads: Excel=1
- When creating an Excel ODBC Data source, be sure to click on “Options” and turn off the “Read Only” option. (“Read Only” is on by default.)
- You must define the table in the spreadsheet. To do this:
 1. Open the spreadsheet in Excel.
 2. Select the region containing the data.
 3. Select **Insert – Name – Define** from the menu bar.(For more details, consult your Microsoft Excel documentation.)

When using Excel as your desktop PC database, keep in mind the following limitations:

- If you create a new record in the Excel file, it must not duplicate a record that already exists in the table in the Excel file.
- Record deletions on the CE device cannot be propagated to the desktop.
- Synchronization of multiple CE devices to one desktop table is not supported in this case.

Index

&	
& operator	201
.	
.dbg files	374
@	
@ functions	
Advanced	208
Basic.....	207
@count	207
@days.....	207
@devicename.....	207
@now.....	207
@prev	207
@sum(columnName).....	207
@today	207
@var function	211
See also Global Variables.....	153
A	
About Box	
Specifying text for.....	281
Access	See Microsoft Access
Adding columns to a table	149
Add-on tools	
Backup table	17
Clean out table definitions.....	17
Expunge form.....	17
Index CDB.....	17
Rebuild table from form.....	17
SSCE utility	17
Advanced functions	208
Date/Time.....	211
Global variables	211
GPS.....	212
Numeric.....	209
System	213
Text	208
Alarm	
Creating command button to set.....	232
Creating macro command to set.....	359
Responding to	410
Setting	410
Aligning controls	269
Application title	
Removing	163
Specifying.....	163
Application window	
Opening.....	379
Applications (Visual CE)	
Overview	379
Shortcut keys	386
Toolbar buttons.....	385
Changing	295
Creating	124
Creation steps.....	136
Distributing.....	298
Downloading from desktop PC to CE device	292
Exiting	388
Modifying	295
Opening in development environment	294
Planning	137
Saving	290
Saving a textual description.....	291
Setting up to run on desktop PC.....	296
Starting	379
Testing	293
Using.....	378
Archiving	
Records	414
Tables	17
ASCII text	
Exporting.....	414
Importing.....	415
Assign command.....	329
Auto Recalc	
Enabling	279
AutoNumber controls	
Creating	252
B	
Backup table (Add-on tool).....	17
Bar code scanners	
Designing Visual CE applications to use	280
Using with Visual CE	409
Basic functions	207
Beaming records between CE devices with IR	417
Bitmap files	
Creating image controls.....	241
Creating scribble boxes	237
Boolean expressions	
Using in a calculated field.....	206
Breakpoints	
Adding and removing in macros	373
Button bar	
Removing from application window	272

C	
C/C++ applications	
Launching a Visual CE application from	421
Calculated fields	
Creating	201
Enabling Auto Recalc	279
Functions	201
Recalculating existing value in application	399
Using dates and times in expressions	205
Using functions in expressions	207
Using If-Then-Else expressions	206
Using to concatenate text	201
Using to perform arithmetic operations	
.....	201
Camera	See Digital cameras
Cascade deletes	
Enabling.....	249, 251
Checkboxes	
Creating	180
Clean table definitions (Add-on tool).....	17
Clear	
Scribble box.....	402
Table.....	416
Cloning tables	142
Color	
Adding to form design	160
Defining custom colors	162
Text in list mode	161
Column order	
Specifying in record list display	276
Column width (in record list display)	
Changing	390
Columns	
Adding	149
Deleting	149
Find sum of values in a	207
Indexing	283
Modifying	149
Selecting to store unique values in	278
Specifying order in record list display..	64, 95, 276
Command buttons	
Command shortcuts	219
Control mEnable connection	234
Dial	231
Display current record in Report CE ..	228
Export text or XML file	221
General instructions for creating	214
Import text or XML file	221
List of types that can be created	216
Move between form regions	223
Put and get files via mEnable.....	235
Search records	220
Set alarm	232
SMS Send message	230
Sort records	229
Start another program.....	226
Using	28, 36
Using to create macros	214
Command line switches	421
Commands (macro) alphabetically	
Assign	329
Communicate	330
Dial	332
Edit Undo	334
File	335
IR Beam	333
Jump	336, 337
Macro Step Label command	338
mEnable connection	339
mEnable Put and Get commands.....	340
Message box.....	342
Move to when pressed command.....	343
Option About	344
Option Clear	345
Option Export	346
Option Filter.....	347
Option Import	348
Option List	349
Option Recalc.....	350
Play sound	351
Record commands.....	352
Report CE Report	354
Return from macro	355
Run external.....	356
Run macro.....	357
Select tab	358
Set alarm	359
Set focus	360
Show/hide control	361
Skip	362
Sleep	363
SMS Receive message.....	364
SMS send message	365
Sort by	366
Stop macro.....	367
Timer	368
Commands (macro) by category	
Field values	325
Form controls	323
Form navigation	325
Grid controls.....	323
Miscellaneous	327
Read/write files	326
Records	324
Run external programs	326
Script control	323
Serial I/O	325
Wireless connections	326

Commands (Visual CE menu)	
Control menu	132
Edit menu (Application window)	383
Edit menu (Development environment)	130
File menu	129
Form menu	131
Option menu	384
Record menu	381, 382, 387
Communicate commands	330
Continue flag	
Setting	276
Control events	321
Control menu commands	132
Control order	
Specifying	276
Controls	
Aligning	269
Changing properties	264
Changing properties for multiple controls at once	267
Copying	270
Creating	166
Creating AutoNumbers	252
Creating Calculated Fields	201
Creating Checkboxes	180
Creating Command Buttons	214
Creating Drop-Down Lists	184, 187, 189, 192, 199
Creating Edit Boxes	167
Creating Grid Controls	257
Creating Image Controls	241
Creating Jump Buttons	248
Creating Labels	179
Creating Lookups	254
Creating mEnable Stoplight	262
Creating Note Boxes	177
Creating Radio Buttons	182
Creating Scribble Boxes	237
Creating Timestamp Buttons	239
Cutting	270
Deleting	271
Dragging without snap-to-grid	268
List of types available	11
Order	276
Pasting	270
Relative to frame rectangle	266
Selecting multiple	269
Sizing relative to other controls	269
Specifying font properties	265
Copying data between records	408
Creating controls	166
Creating records in an application	
By duplicating an existing	398
From scratch	397
Creating tables	138
By cloning	142
By copying an Access Database to the CE device	145
By downloading from an ODBC-Enabled application	146
In Visual CE	139
Using Pocket Access	143
Using SQL Server CE	144
Creating Visual CE applications	124
Custom colors	
Adding or Modifying	162
Custom filters	
Creating and applying	395
Custom synchronization option	438
D	
Data types	
Definitions of	140
Date/Time functions	211
Date/time values	
Creating controls to enter	171
Entering null values	405
Specifying formats for	174
Using in calculated field expressions	205
Debugging	
Macros in applications with multiple forms	370, 374
Macros with Macro Debugger	370
Decimal point	
Entering on a non-touchscreen device	406
Default values	
Creating for edit boxes	171
Delete records	
Allowing/disallowing user to	288
Enabling automatic deletion of dependent records	249
How to	401
Deleting	
Columns	149
Controls	271
Records	401
Tables	151
Dependent drop-down lists	
Creating	189
Multi-tiered	192
Description of form	
Saving	291
Desktop connectivity	424, 425, 427, 428, 429, 430, 435
Desktop PC	296
Setting up Visual CE applications to run on	296
Development environment window	
Hiding scroll bars in	159

Shortcut keys.....	134
Starting on CE device.....	126
Starting on desktop PC	126
Toolbar buttons	133
Device screen size.....	128
Dial command	
Macro.....	332
On Smartphones	380
Dial command button	
Creating.....	231
Digital cameras	
Creating Image control for using with	
Visual CE	244
Using with Visual CE applications.....	404
Distributing applications	
Creating installation disk	298
Downloading	
Files from server via mEnable.....	235
Downloading from desktop PC to CE device	
Microsoft Access table to a Windows CE	
table	425
Microsoft Excel spreadsheet to Windows	
CE table	426
Table from an ODBC-enabled application	
for use with Visual CE	146
Visual CE applications.....	292
Dragging controls	
Without snap-to-grid	268
Drop-Down lists	
Creating	184
Creating dependent drop-down lists ..	189
Creating multi-tiered dependent.....	192
Creating navigational drop-down lists	199
Using a column from another table as	187
Dumping the table.....	416
Duplicate values	
Preventing in a selected column	278
Duplicating records	
Creating command button to	219
How to	398
E	
Edit Undo command	334
Edit Boxes	
Creating	167
Creating default values for	171
Creating input masks for	170
Creating pop-up word lists for	176
Date/time formats for	174
Making input required by setting Min/Max	
values.....	175
Money values	173
Using word lists	407
Edit menu commands	
Application window.....	383
Customizing.....	272
Development Environment.....	130
Editing records.....	399
Empty record fields	
Testing for using Boolean expression	206
Erasing scribbles	402
Events	
Overview	320
Quick Tour.....	106
Setting up control events to trigger	
macros	321
Setting up form events to trigger macros	
.....	322
Excel.....	See Microsoft Excel
Exiting	
Applications.....	388
Visual CE's development environment	
.....	135
Export	
Create command button to	221
Records.....	346, 414
Table to text file.....	414
Table to xml file.....	414
Windows CE table into Microsoft Access	
.....	428
Windows CE table to an ASCII text file	
.....	430
Windows CE table to Microsoft Excel	427
Expressions	
Creating.....	201
Using in a calculated field	201
Expunge form (Add-on tool)	17
F	
File I/O commands	335
File menu commands	129
File synchronization	
Applying to use applications on desktop	
PC.....	296
Filtering the record display	395
Filters (Dependent drop-down)	
Creating.....	189, 192
Purpose of.....	189
Filters for record display	
Applying	395
Creating custom in application window	
.....	395
Creating pre-defined as part of form	
design	284
Specifying with a command line switch	
.....	421
Turning off/on	395
Font properties	
Specifying for a control	265

Form events.....	322
Form menu commands.....	131
Form Wizard	156
Forms	
Adding color	160
Beginning a new.....	155
Changing size	158
Changing tables for	152
Creating.....	126
Creating tabs.....	164
Expunging from CE device	17
Locking to prevent modification .	289, 299
Planning	137
Saving description of.....	291
Setting up to run on a desktop PC	296
Using	379
Without tables	157
Functions	
@count.....	207
@days.....	207
@devicename.....	207
@now	207
@prev.....	207
@sum.....	207
@today.....	207
Advanced	208
Basic.....	207
Date/Time.....	211
Global variable	211
GPS.....	212
Numeric.....	209
System	213
Text	208
Using in a calculated field expression	201
G	
Get files from mEnable server	
Create command button to	235
GIF files	
Creating image controls	241
Creating scribble boxes.....	237
Global variables	
function.....	211
Overview, how to use.....	153
Sample use in macro	306
Using in expressions	202
Writing values to. See also various control types and macro commands	
GPS	
Commands.....	330
Functions.....	212
Graphics	
Creating image controls	241
Creating scribble boxes.....	237
Grid controls	
Creating.....	257
H	
Hardware buttons	
Setting up to trigger macros	322
Help	
Contacting SYWARE	16
I	
If-Then-Else expressions	
Using in a calculated field.....	206
Image controls	
Adding to form	241
Creating	241
Creating to display web pages	246
Creating to take pictures with digital cameras.....	244
Using.....	403
Using to take pictures with digital cameras.....	404
Import	
Create command button to	221
Text files	415
Windows CE table into Microsoft Access	428
Windows CE table into Microsoft Excel	427
XML files	415
Index CDB - Add-on tool	17
Indexing columns	283
Allowing/disallowing users to change	150, 283
In list mode	392
Input masks	
Creating for edit boxes.....	170
Insert records	
Allowing/disallowing users to	288
Installation disk	
Creating to distribute applications (Pro and Mobile 360 Editions only)	298
Installing Visual CE	15
IR	
Beaming records between CE devices	417
IR Beam commands.....	333
J	
JPEG files	
Creating image controls.....	241
Creating scribble boxes	237
Jump buttons	
Creating	248
Jump command.....	336, 337
Jumping to another form	248
L	
Labels	
Creating	179

Launching Visual CE from another application	421	Editing with Macro Builder	315																																																																																																																																																		
Linking to Microsoft Access	429	Overview	303																																																																																																																																																		
List display		Quick Tour.....	106																																																																																																																																																		
Counting records in	207	Running in the debugger.....	375																																																																																																																																																		
Editing.....	25	Sample.....	117, 306, 362																																																																																																																																																		
Filter using command line switches ...	421	Triggering with control events	321																																																																																																																																																		
Filtering.....	395	Triggering with events (Overview)	320																																																																																																																																																		
Finding sum of values in column.....	207	Triggering with form events.....	322																																																																																																																																																		
Scrolling to specific records by initial letters	390	Triggering with hardware buttons	322																																																																																																																																																		
Sorting	394	Maintenance Utilities See Add-on tools.....	17																																																																																																																																																		
Specifying color for text.....	161	mEnable																																																																																																																																																			
Specifying column order for.....	276	Adding an mEnable stoplight control to a form.....	262	Switching to single record display	391	Adding mEnable command buttons to a form.....	234	Viewing	25, 34, 390	Creating command buttons to put and get files	235	List filters		Using to synchronize tables	443	Applying.....	395	Using with Visual CE.....	418	Command line switches	421	mEnable connection commands	339	Creating and applying custom.....	395	mEnable Put and Get	340	Creating pre-defined as part of form design.....	284	Menu bar		Purpose of	395	Customizing	272	Turning off/on	395	Removing from application window ...	272	Listing all records	390	Message box command	342	Lock form	289	Microsoft Access	425	Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316
Adding an mEnable stoplight control to a form.....	262																																																																																																																																																				
Switching to single record display	391	Adding mEnable command buttons to a form.....	234	Viewing	25, 34, 390	Creating command buttons to put and get files	235	List filters		Using to synchronize tables	443	Applying.....	395	Using with Visual CE.....	418	Command line switches	421	mEnable connection commands	339	Creating and applying custom.....	395	mEnable Put and Get	340	Creating pre-defined as part of form design.....	284	Menu bar		Purpose of	395	Customizing	272	Turning off/on	395	Removing from application window ...	272	Listing all records	390	Message box command	342	Lock form	289	Microsoft Access	425	Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316				
Adding mEnable command buttons to a form.....	234																																																																																																																																																				
Viewing	25, 34, 390	Creating command buttons to put and get files	235	List filters		Using to synchronize tables	443	Applying.....	395	Using with Visual CE.....	418	Command line switches	421	mEnable connection commands	339	Creating and applying custom.....	395	mEnable Put and Get	340	Creating pre-defined as part of form design.....	284	Menu bar		Purpose of	395	Customizing	272	Turning off/on	395	Removing from application window ...	272	Listing all records	390	Message box command	342	Lock form	289	Microsoft Access	425	Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316								
Creating command buttons to put and get files	235																																																																																																																																																				
List filters		Using to synchronize tables	443	Applying.....	395	Using with Visual CE.....	418	Command line switches	421	mEnable connection commands	339	Creating and applying custom.....	395	mEnable Put and Get	340	Creating pre-defined as part of form design.....	284	Menu bar		Purpose of	395	Customizing	272	Turning off/on	395	Removing from application window ...	272	Listing all records	390	Message box command	342	Lock form	289	Microsoft Access	425	Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316												
Using to synchronize tables	443																																																																																																																																																				
Applying.....	395	Using with Visual CE.....	418	Command line switches	421	mEnable connection commands	339	Creating and applying custom.....	395	mEnable Put and Get	340	Creating pre-defined as part of form design.....	284	Menu bar		Purpose of	395	Customizing	272	Turning off/on	395	Removing from application window ...	272	Listing all records	390	Message box command	342	Lock form	289	Microsoft Access	425	Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																
Using with Visual CE.....	418																																																																																																																																																				
Command line switches	421	mEnable connection commands	339	Creating and applying custom.....	395	mEnable Put and Get	340	Creating pre-defined as part of form design.....	284	Menu bar		Purpose of	395	Customizing	272	Turning off/on	395	Removing from application window ...	272	Listing all records	390	Message box command	342	Lock form	289	Microsoft Access	425	Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																				
mEnable connection commands	339																																																																																																																																																				
Creating and applying custom.....	395	mEnable Put and Get	340	Creating pre-defined as part of form design.....	284	Menu bar		Purpose of	395	Customizing	272	Turning off/on	395	Removing from application window ...	272	Listing all records	390	Message box command	342	Lock form	289	Microsoft Access	425	Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																								
mEnable Put and Get	340																																																																																																																																																				
Creating pre-defined as part of form design.....	284	Menu bar		Purpose of	395	Customizing	272	Turning off/on	395	Removing from application window ...	272	Listing all records	390	Message box command	342	Lock form	289	Microsoft Access	425	Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																												
Menu bar																																																																																																																																																					
Purpose of	395	Customizing	272	Turning off/on	395	Removing from application window ...	272	Listing all records	390	Message box command	342	Lock form	289	Microsoft Access	425	Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																
Customizing	272																																																																																																																																																				
Turning off/on	395	Removing from application window ...	272	Listing all records	390	Message box command	342	Lock form	289	Microsoft Access	425	Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																				
Removing from application window ...	272																																																																																																																																																				
Listing all records	390	Message box command	342	Lock form	289	Microsoft Access	425	Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																								
Message box command	342																																																																																																																																																				
Lock form	289	Microsoft Access	425	Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																												
Microsoft Access	425																																																																																																																																																				
Lookup control		Copying a table to the CE device	145	Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																
Copying a table to the CE device	145																																																																																																																																																				
Creating	254	Downloading table definitions and data from.....	146	M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																				
Downloading table definitions and data from.....	146																																																																																																																																																				
M		Importing data	428	Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																								
Importing data	428																																																																																																																																																				
Macro Builder		Including .mdb file with installation files	298	Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																												
Including .mdb file with installation files	298																																																																																																																																																				
Overview.....	309	Linking to a Windows CE table	429	Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																
Linking to a Windows CE table	429																																																																																																																																																				
Starting	309	Synchronizing tables.....	146, 435	Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																				
Synchronizing tables.....	146, 435																																																																																																																																																				
Macro Debugger		Microsoft ActiveSync		Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																								
Microsoft ActiveSync																																																																																																																																																					
Breakpoints.....	373	Applying file synchronization.....	296	Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																												
Applying file synchronization.....	296																																																																																																																																																				
Closing.....	376	Microsoft Excel		Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																
Microsoft Excel																																																																																																																																																					
Overview.....	370	Downloading data from an Excel spreadsheet to a Windows CE table	426	Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																				
Downloading data from an Excel spreadsheet to a Windows CE table	426																																																																																																																																																				
Quick Tour	116	Importing data from a Visual CE table to a spreadsheet	427	Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																								
Importing data from a Visual CE table to a spreadsheet	427																																																																																																																																																				
Running a macro in	375	Synchronize spreadsheets with tables	444	Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																												
Synchronize spreadsheets with tables	444																																																																																																																																																				
Starting	371	Minus sign		Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																																
Minus sign																																																																																																																																																					
Steps for using	370	Entering on a non-touchscreen device	406	Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																																				
Entering on a non-touchscreen device	406																																																																																																																																																				
Macro Step Label command	338	Mobile 360 (Visual CE)		Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																																								
Mobile 360 (Visual CE)																																																																																																																																																					
Macros		Description of	10	Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																																												
Description of	10																																																																																																																																																				
Adding and removing breakpoints for debugging	373	Modifying		Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																																																
Modifying																																																																																																																																																					
Command reference.....	323	Applications.....	295	Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																																																				
Applications.....	295																																																																																																																																																				
Commands See Commands (macro)		Columns	149, 150	Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																																																								
Columns	149, 150																																																																																																																																																				
Converting a button macro	317	Control properties	264	Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																																																												
Control properties	264																																																																																																																																																				
Creating (Overview)	307	Synchronization settings	435	Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																																																																
Synchronization settings	435																																																																																																																																																				
Creating a command button to launch.....	225	Money values		Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																																																																				
Money values																																																																																																																																																					
Creating custom menu pick to launch	272	Edit Box.....	173	Creating with Command buttons.....	214	Creating with Macro Builder	309, 312	Cut, copy, paste commands.....	314	Debugging with Macro Debugger.....	370	Deleting with Macro Builder.....	316																																																																																																																																								
Edit Box.....	173																																																																																																																																																				
Creating with Command buttons.....	214																																																																																																																																																				
Creating with Macro Builder	309, 312																																																																																																																																																				
Cut, copy, paste commands.....	314																																																																																																																																																				
Debugging with Macro Debugger.....	370																																																																																																																																																				
Deleting with Macro Builder.....	316																																																																																																																																																				

Move to when pressed	343
Moving controls	
Aligning	269
In precise increments	49, 80
Without snap-to-grid.....	268
Moving data between records	408
Multi-tiered dependent drop-down lists	
Creating.....	192
N	
Navigational drop-down lists	
Creating.....	199
Negative numbers	
Entering on non-touchscreen devices	406
No headers	222, 346, 348
No underlying database.....	157
Non-touchscreen devices	
Entering negative number or decimal point	406
Tutorial for creating Visual CE applications for use on.....	68
Tutorial for using Visual CE application on	30
Note Boxes	
Creating.....	177
Creating pop-up word lists for	176
Using pop-up word lists.....	407
Null values	
Entering in a date/time field	405
Entering in a numeric field	405
Testing for in a calculated field expression	206
Numeric Functions.....	209
Numeric values	
Entering negative numbers and decimal points on non-touchscreen devices	406
Entering null values.....	405
O	
ODBC	425, 427, 428, 429, 435, 436
ODBC-enabled applications	
Downloading tables created with	146
OID columns	
Adding to desktop PC tables prior to synchronization.....	435
Opening applications	294
Option About command	344
Option Clear command.....	345
Option Export command	346
Option Filter commands.....	347
Option Import command	348
Option List command.....	349
Option Recalc command	350
Option menu commands	384
Customizing	272
P	
Password box	
Creating	167, 168
Passwords	
Changing	286
Creating	286
Supplying for SQL Server CE database tables	287
PCs	
Setting up Visual CE applications to run on.....	296
Personal Edition (Visual CE)	
Description of.....	10
Pictures	
Creating image controls.....	241
Creating scribble boxes	237
Taking with Image controls and digital cameras.....	404
Planning Forms	137
Play sound command.....	351
Pocket Access	
Creating a Pocket Access table by copying an Access database	145
Creating tables with	143
SYWARE indexing utility for	17
Pre-defined filters	
Applying to record display	395
Creating	284
Printing Records	
From Visual CE.....	412
Specifying order of fields in printouts..	276
Using Report CE to.....	413
Pro and Mobile 360 Editions only	
Distributing an application	298
Synchronizing multiple handhelds	440
Pro Edition (Visual CE)	
Description of.....	10
Programs	
Creating command buttons to start from Visual CE application	226
Put files to mEnable server	
Create command button to	235
R	
Radio Buttons	
Creating	182
Rebuild table (Add-on tool)	17
Record commands	352
Record count	
Removing from application title bar	163
Record lists	
Changing sort order	394
Changing width of columns	25, 34, 390
Counting records in (function)	207
Creating pre-defined filter as part of form design	284

Filtering list in application window	395
Finding sum of values in column (function)	207
Sorting	394
Specifying initial order in design.....	282
Specifying order of columns in	276
Switching to single record display	391
Viewing.....	25, 34, 390
Record menu commands	
Customizing.....	272
On non-touchscreen devices.....	382, 387
On touchscreen devices and desktop PCs	381
Record order	
Changing in application	394
Specifying initial (in application design)	282
Records	
Allowing users to delete	288
Allowing users to insert	288
Archiving.....	414
Autonumbering	252
Beaming between CE devices with IR417	
Copying or moving data between.....	408
Creating new	397, 398
Deleting	401
Displaying a list of	25, 34
Duplicating.....	398
Editing.....	399
Enabling automatic deletion of dependent records (cascade deletes).....	249
Exporting to text file.....	414
Filtering list of	395
Finding by desired criteria	395
Listing all.....	390
Listing selected.....	395
Locating.....	389
Printing	412
Saving.....	400
Scrolling.....	389
Searching for	27, 35, 389, 392, 394
Specifying an initial sort/search order	282
Specifying order of fields in printouts .	276
Report CE	
Creating a command button to open..	228
Using to print Visual CE records	413
Report CE Report command	354
Required	
Making input required for edit box (Lite Edition)	168
Making input required for edit box (Personal, Pro, and Mobile 360 Editions)	175
Return from macro command.....	355
Run external command	356
Run macro command	357
S	
Sample Visual CE applications	
Installing.....	15
Saving applications.....	290
Saving records.....	400
Screen size	
Changing.....	128
Scribble boxes	
Adding background images to	237
Adding scribble box control to form ...	237
Drawing in	402
Erasing	402
Scroll bars	
Hiding in development window	159
Scrolling through records	389
Search command button	
Creating.....	220
Searching for records	392
By column value.....	392
By multiple criteria.....	395
By text strings in any text field	395
Creating command button to.....	220
Overview of methods	392
Security	
Allow/disallow record insert/delete....	288
Passwords.....	286
Protecting form against modification..	289
Select tab command.....	358
Selecting controls	269
Serial communication	325, 330
Set alarm command	359
Set alarm command button	
Creating.....	232
Set focus command	360
Shift key	
Using on P/PC	398
Shortcut Keys	
Application window	386
Development environment window....	134
Show/hide control command	361
Sizing Controls	269
Skip command.....	362
Sleep command.....	363
Smartphones .. See Non-touchscreen devices	
SMS message	
Creating command button to send....	230
Creating macro command to receive .	364
Creating macro command to send....	365
SMS receive message command.....	364
SMS send message command	365
SMS send message command button	
Creating.....	230

Snap-to-Grid	
Turning on/off	268
Socket scanners	280, 409
Sort by command	366
Sort command button	
Creating.....	229
Sort Order	
Changing.....	394
Creating a command button to change	229
Specifying.....	282
Using a command line switch	421
SQL Server CE	144
Creating passwords for	287
Creating tables with.....	144
Requirements to use with Visual CE....	14
SYWARE indexing utility for.....	17
Start flag	
Hiding on CE device.....	163
Starting Visual CE	
Applications.....	379
Development environment	126
Stop macro command	367
Sum of values in column	
Finding	207
Switches	
For launching a Visual CE application from the command line	421
Symbol scanners	280, 409
SynchronizationSee Table synchronization	
System functions	213
System requirements.....	14
SYWARE technical support.....	16
T	
Tab order	
Specifying.....	276
Table synchronization	
Custom options	438
Establishing during table download ...	100
Methods available	434
Multiple handhelds	440
On command.....	441
On install	435
Quick Tour.....	99
Setting up	435
Timestamp-based	438, 439
Using mEnable.....	443
Using Vicesyn.exe.....	442
With Excel spreadsheets.....	444
Tables	
Adding columns to.....	149
Archiving	17
Changing for a form	152
Clearing.....	416
Cloning	142
Creating (Overview).....	138
Creating a Pocket Access table by copying an Access database	145
Creating form without.....	157
Creating jumps between.....	248
Creating with Pocket Access	143
Creating with SQL Server CE.....	144
Creating with Visual CE.....	139
Deleting.....	151
Deleting columns from	149
Downloading from desktop PC ...	100, 146
Downloading from Microsoft Access or other ODBC-enabled application ...	146, 425
Exporting to an ASCII text file.....	430
Importing into Microsoft Access.....	428
Importing into Microsoft Excel	427
Importing text files into.....	415
Indexing columns in	283
Modifying columns in	149
Rebuilding from a form	17
Removing definitions from computer....	17
Selecting one column to store unique values in	278
Specifying for a form.....	155
Synchronizing	434
Tabs	
Adding to form	164
Testing applications	293
Text files	
Exporting table to.....	414
Importing into table	415
Text functions	208
Text search	
Finding records by specifying text strings	395
Text strings	
Concatenating using calculated field..	201
Textual description of form	
Creating	291
Time	
Entering null values for	405
Formats, specifying.....	174
Using in calculated field expressions..	205
Timer command	368
Timestamp buttons	
Creating	239
Timestamp-based synchronization	439
Title bar	
Removing from application window	163
Removing record count from	163
Specifying	163
Toolbar buttons	
Application window	385
Development environment window....	133
Removing from application window	272

Tutorials	
Creating Visual CE Applications	
On non-touchscreen device.....	68
On touchscreen device	38
Macro Debugger.....	116
Macros and Events.....	106
Table Synchronization.....	99
Using Visual CE Applications	
On non-touchscreen device.....	30
On touchscreen device	20
Type style of controls	265

U

Unique	
Selecting a column to store unique values.....	278
Uploading	
Files to server via mEnable	235

V

Variables	
See global variables	153
Vicesyn.exe	
Using to synchronize tables	442

Visual Basic applications

Launching a Visual CE application from	
.....	421

Visual CE

Add-on tools.....	17
Editions	10
Installing.....	15
Overview	9
System Requirements.....	14
Technical Support.....	16
Using with mEnable	418

W

Web pages	
Creating Image control to display	246
Windows Start flag	
Hiding on CE device	163
Word lists	
Creating.....	176
Using	407

X

XML files	
Exporting table to	346, 414
Importing into table	348, 415
Importing or exporting	221