

Sample Macros

The sample macros provided with ReportSmith show how you can use the macro language to perform sophisticated tasks such as conditional formatting, creating derived fields, customizing menus, printing and loading a series of reports, and creating temporary tables for complex manipulation of report data.

You can create macros to produce more detailed tasks, such as calculating a percent of total. Or, you can use macros as “wizards” to perform commonly requested functions like filling list boxes with *.RPT* files in a given folder and allowing users to run and print them from the dialog box.

Chapter 10 documents the actual code for several of the available sample macros, found in the *MACROS* subfolder. These samples show you how to:

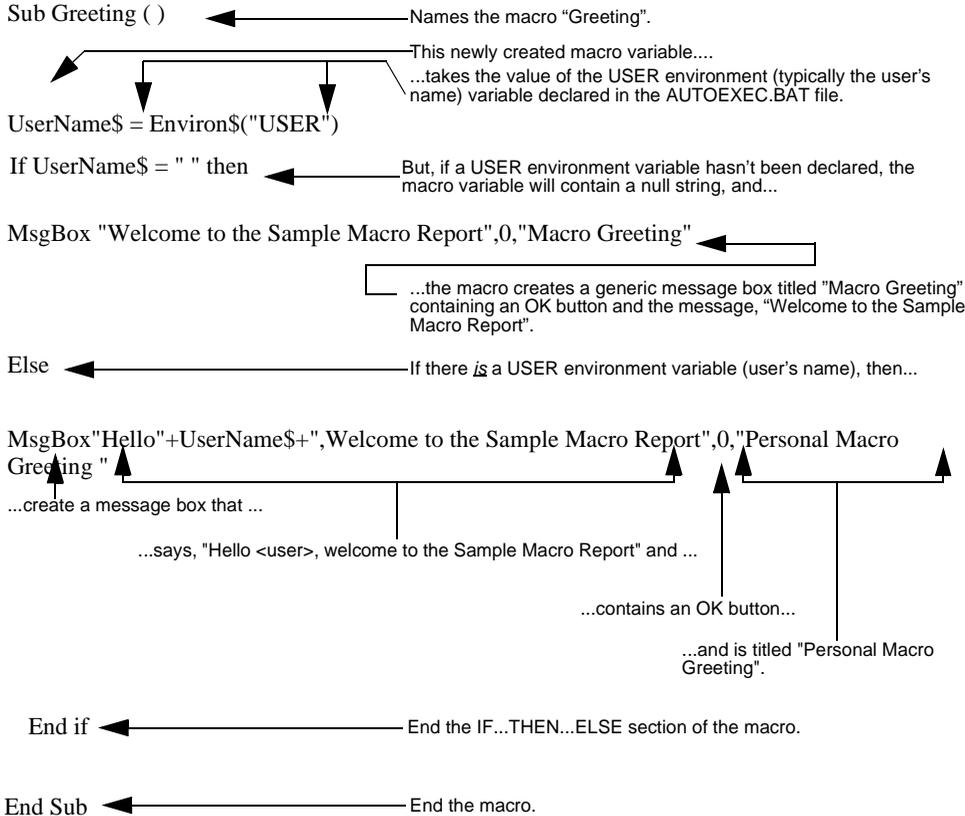
- Display a greeting.
- Connect to a database.
- Place a line counter field on a report.
- Determine a percent of total.
- Sum the value of another macro-derived field.
- Generate a dialog box using all available functions.
- Refresh a temporary table with the *DataSet* object.

Each sample includes comments to help familiarize you with the macro process, but the chapter assumes that you are already familiar with the concepts discussed in Chapter 9, “Macros: An Overview”. For a detailed explanation of each macro involved, please see “Command reference” on page 261.

Example 1: Displaying a greeting (GREETING.MAC)

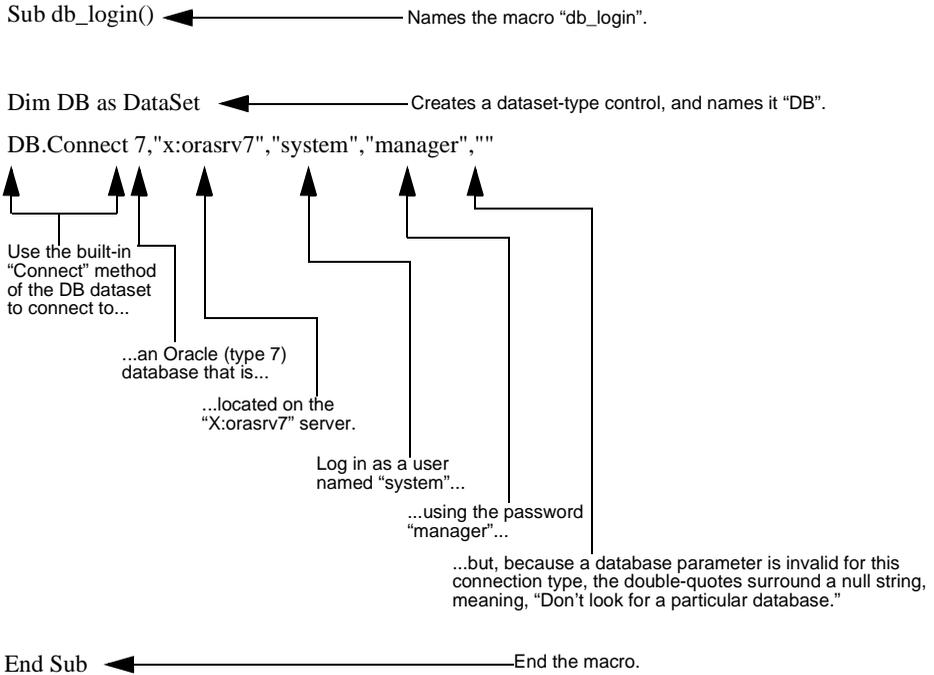
This macro displays a greeting using the name stored in the USER environment variable (typically found in the AUTOEXEC.BAT file of the computer system). You can link this macro to the opening of ReportSmith, the opening of a specific report, or both.

Note: The line spacing in the following examples is not necessary when creating your macros. It is for illustration purposes only.



Example 2: Connecting to a database

This macro shows you how to use the macro language to create a database connection. The example below connects to an Oracle database, but there are thirteen other types to which you can connect. (See the macro command reference on page 261 for details.).



Example 3: Creating a counter field

This macro defines a derived field that numbers the records in a report.

```
Sub RecordNumber() ← Name this macro "RecordNumber".  
  
DerivedField Str$(Current)  
    ↑ Using the record number of the current record...  
    ↑ ...convert that number (an integer value) to its string-type  
    ↑ representation, and...  
    ↑ ...use that string as the value of a new derived field.  
  
End Sub ← End the macro.
```

After creating this derived field, choose Tools|Derived Field, name the derived field, and then choose the macro that defines the field. Then you may use Insert|Field to insert it into your report.

End If ←———— End the If...Then section. (Remember that there is no "Else" provision so macro execution is halted if an exception occurs.)

End Sub ←———— End the macro.

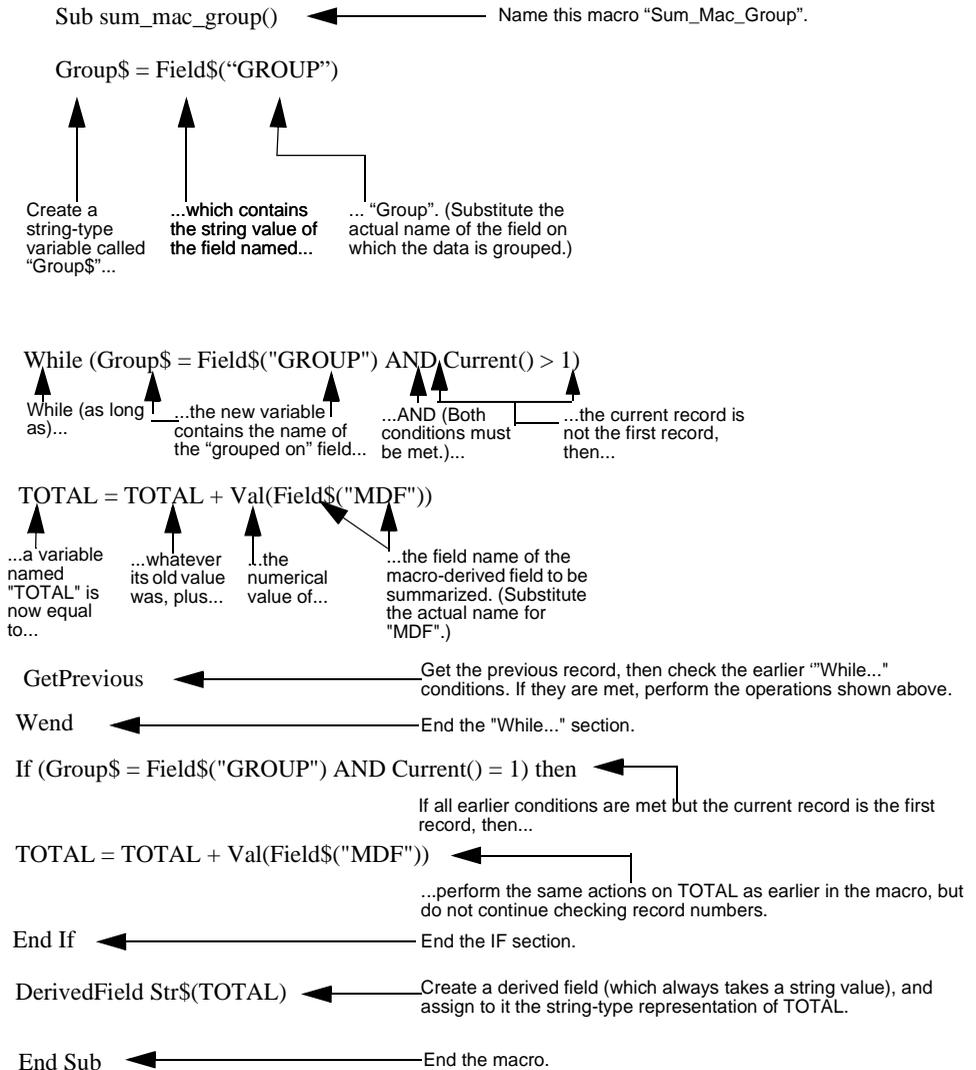
In the case of other sample macros, the task was complete once the macro had been created. However, this macro creates a derived field that must still be inserted into the appropriate location within the report.

- To insert the derived field,
 - 1 In the Edit Macro dialog box, choose OK to return to the Create a Macro dialog box.
 - 2 Choose OK to exit the Create a Macro dialog box, and then choose OK to exit the Derived Fields dialog box.
 - 3 Choose Insert|Field to drop the derived field as a column of data in the report.
The Insert Field dialog box appears.
 - 4 Select the Derived Field type, select the name you assigned to the derived field, and then choose Insert.
The Insert cursor appears.
 - 5 Insert the derived into the appropriate header or footer.
 - 6 Use the Percent button on the ribbon to format the derived field.

Example 5: Creating a summary

The following is an example of macro-derived fields that sum the values of another macro-derived field. In this example, substitute the field name of the macro-derived field to be summarized in place of "MDF." Also, if you are working with grouped data, in place of "GROUP" substitute the name of the field on which the data is grouped. If the report is grouped on more than one field, and you want a summary for each group, create a macro-derived field for each group.

This macro summarizes the value of another macro-derived field in a report with a group break.



Examples 6 and 7: Creating and refreshing a temporary table

You can create temporary tables to perform more complex functions within ReportSmith. A temporary table contains static data that doesn't change when the originating table changes (unless you instruct it to). Temporary tables are useful for reporting on a selection of groups or for grouping and sorting on macro-derived fields. To create a temporary table, you must first create a set of data to populate the table. Types of data used are "included" columns (such as columns listed in the SELECT statement, including SQL-derived fields), summary fields, and macro-derived fields. After creating a report with the desired data, you can use a macro to export the table through an ODBC driver.

For example, you can use a temporary table to create a view of a table which contains only a subreport of its data. Also, you can use temporary tables to perform summaries on macro-derived fields or to sort data by a summary field.

Note: The temporary table itself must be stored in an ODBC connection. However, the original data can originate anywhere.

When you save a report as a temporary table,

- Each aspect of the report is saved as a separate column.
- Data columns and derived fields are named numerically in order.
- Summary fields appear as columns and use the following naming convention:

SumFunction|GroupLevel_Column#

Element	Description
SumFunction	The summary function for the summary field (Min, Max, Sum and so on).
Group Level	The group level to which the summary is set.
Column#	The position or order number of the summary column in your report.

For example, the name of a summary field that counted the number of values in the 3rd column at the Entire Report Group level in your report is COUNT1_3, and the name of a summary field that summarized the first column in your report at your primary group (the first group you have created) is SUM2_1.

- To create and use a temporary table,

- 1 Open a report.
- 2 Create a report macro like the one below to export your report data to a temporary table:

```

Sub temptable()
Dim DS as Dataset
DS.SetFromActive
DS.ExportTable "mytable", 55, "RS_dBASE", "", " ", " "
End Sub

```

Name this macro "temptable".
 Create a dataset variable and name it "DS".
 Use a data set from the current report.
 Use the ExportTable method of the dataset object to...
 ...create a temporary table named "mytable".
 You can use another driver type here, but you *must* use an ODBC type.
 End the macro.

- 3 Link the macro you created to a report-level event, such as the closing of your report.
- 4 After making sure this macro has run, create a new report, and choose the table with the name that matches the name you indicated in the ExportTable command in your macro. In the above example, it is "mytable".

You now have a new report containing a separate column for each data field, derived field, and summary field from your original report. You can now manipulate these fields in the same manner as regular data fields (sort, group, and perform summaries).

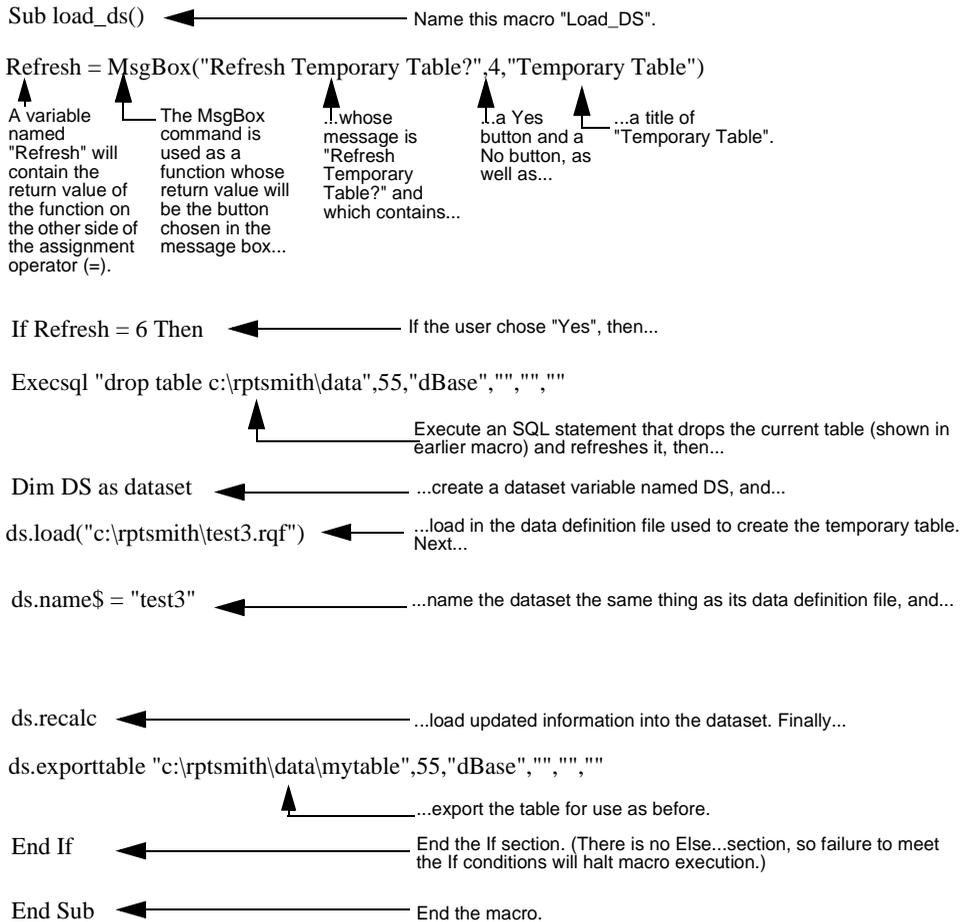
You can save your report containing the temporary table as an .RQF (ReportSmith Query File) file. This type of file can be used to update a temporary table each time you run a report. It contains a description of how to get the data you just exported from the source. Creating an .RQF file can be done through File|Save As or via a macro. If you choose to save the file via a macro, add the following line to the macro above that is used to export the table (substituting an actual file name for "Filename.RQF" in the following code fragment):

```
DS.Save "Filename.RQF"
```

- To create a ReportSmith query file (.RQF),

- 1 Create a new report.
- 2 Choose File|Save As. The Save Report dialog box appears.
- 3 Under File Type, choose ReportSmith query (*.RQF) files.

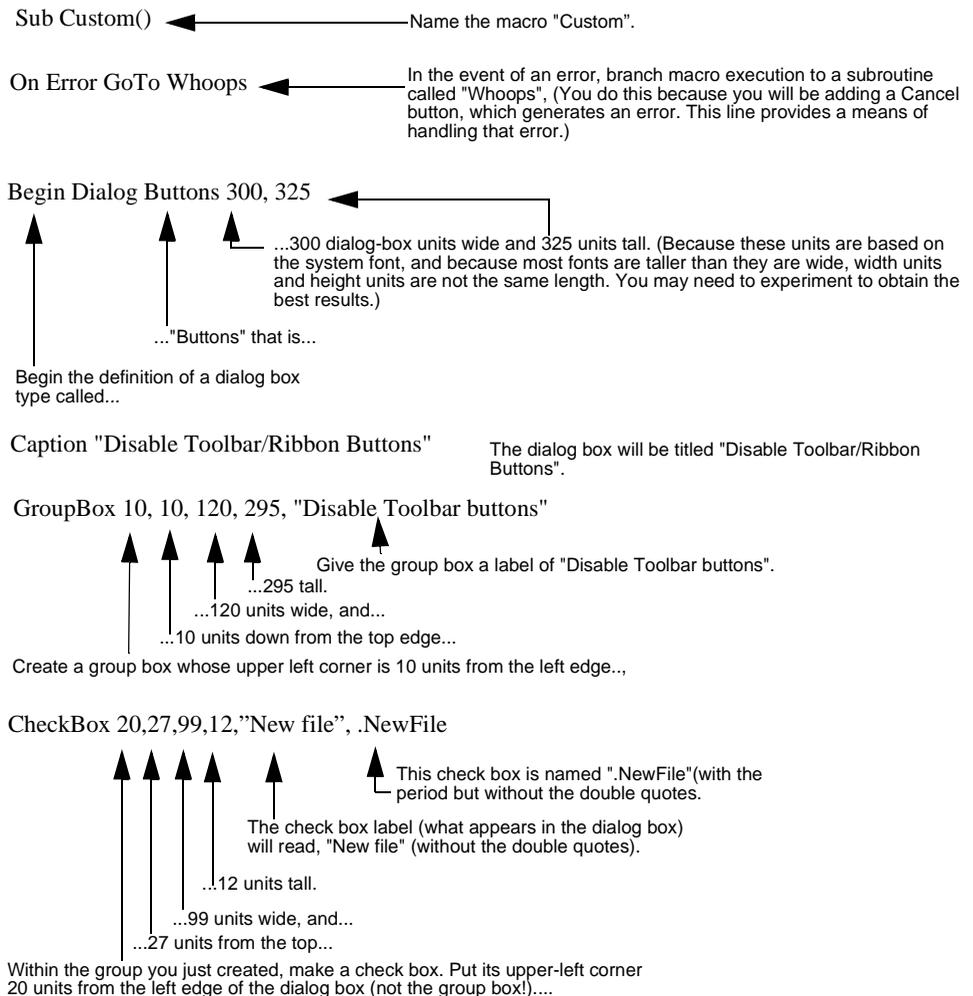
You will probably want to name an .RQF file the same thing that you named your temporary table. The following macro prompts a user to refresh the temporary table from which a report has been built. Link this report-level macro to the **Before Report Load** event.



Advanced example (CUSTOM.MAC)

This example takes you on a “behind-the-scenes tour” of CUSTOM.MAC, the macro used in Appendix A of this manual, to enable or disable various toolbar/ribbon buttons and their equivalent menu commands. As you examine the ReportBasic code used to create CUSTOM.MAC, you see how to create dialog boxes and their constituent elements, how to determine whether a menu command is enabled or disabled, and how to set the enable/disable state of buttons and menu commands.

As described in Appendix A, if you intend to use CUSTOM.MAC or the code contained in it to customize ReportSmith for use with one of your applications, we recommend that you ensure that the macro file is not distributed with the report to which it pertains to prevent users from changing your settings.



CheckBox 20,39,99,12, "Open file", .OpenFile
 CheckBox 20,51,99,12,"Save file", .SaveFile
 CheckBox 20,66,99,12,"Print", .Print
 CheckBox 20,81,99,12,"View Page", .ViewPage
 CheckBox 20,93,99,12,"Page Width",.PageWidth
 CheckBox 20,105,99,12,"100 Percent",.HundredPercent
 CheckBox 20, 120, 99, 12, "Column Mode", .ColumnMode
 CheckBox 20, 132, 99, 12, "Form Mode", .FormMode
 CheckBox 20, 147, 99, 12, "Header", .Header
 CheckBox 20, 159, 99, 12, "Footer", .Footer
 CheckBox 20, 174, 99, 12, "Ascending sort", .AscendSort
 CheckBox 20, 186, 99, 12, "Descending sort", .DescendSort
 CheckBox 20, 201, 99, 12, "Sum", .Sum
 CheckBox 20, 213, 99, 12, "Average", .AbsValue
 CheckBox 20, 225, 99, 12, "Minimum", .Minimum
 CheckBox 20, 237, 99, 12, "Maximum", .Maximum
 CheckBox 20, 249, 99, 12, "Item count", .Count
 CheckBox 20, 264, 99, 12, "Merge reports", .Merge
 CheckBox 20, 276, 99, 12, "SQL", .SQL
 CheckBox 20, 288, 99, 12, "Best fit for columns", .BestFit

Repeat the check box-creation steps, using the coordinates, labels, and names shown here.

GroupBox 155, 10, 120, 250, "Disable Ribbon buttons"

Create a second group using these parameters...

CheckBox 165, 27, 99, 12, "Font", .Font
 CheckBox 165, 42, 99, 12, "Point size", .Points
 CheckBox 165, 57, 99, 12, "Bold", .Bold
 CheckBox 165, 69, 99, 12, "Italic", .Italic
 CheckBox 165, 81, 99, 12, "Underline", .Underline
 CheckBox 165, 96, 99, 12, "Align flush left", .Left
 CheckBox 165, 108, 99, 12, "Align centered", .Center
 CheckBox 165, 120, 99, 12, "Align flush right", .Right
 CheckBox 165, 135, 99, 12, "Currency format", .Currency
 CheckBox 165, 147, 99, 12, "Numerical format", .Number
 CheckBox 165, 159, 99, 12, "Percentage format", .Percentage
 CheckBox 165, 174, 99, 12, "Insert text", .Text
 CheckBox 165, 186, 99, 12, "Insert drawing", .Drawing
 CheckBox 165, 198, 99, 12, "Insert graph", .Graph
 CheckBox 165, 210, 99, 12, "Insert crosstab", .Crosstab
 CheckBox 165, 225, 99, 12, "Style Extractor", .Extract
 CheckBox 165, 237, 99, 12, "Edit/Apply styles", .Styles

...and populate it using these parameters.

OKButton 155,275,60,18

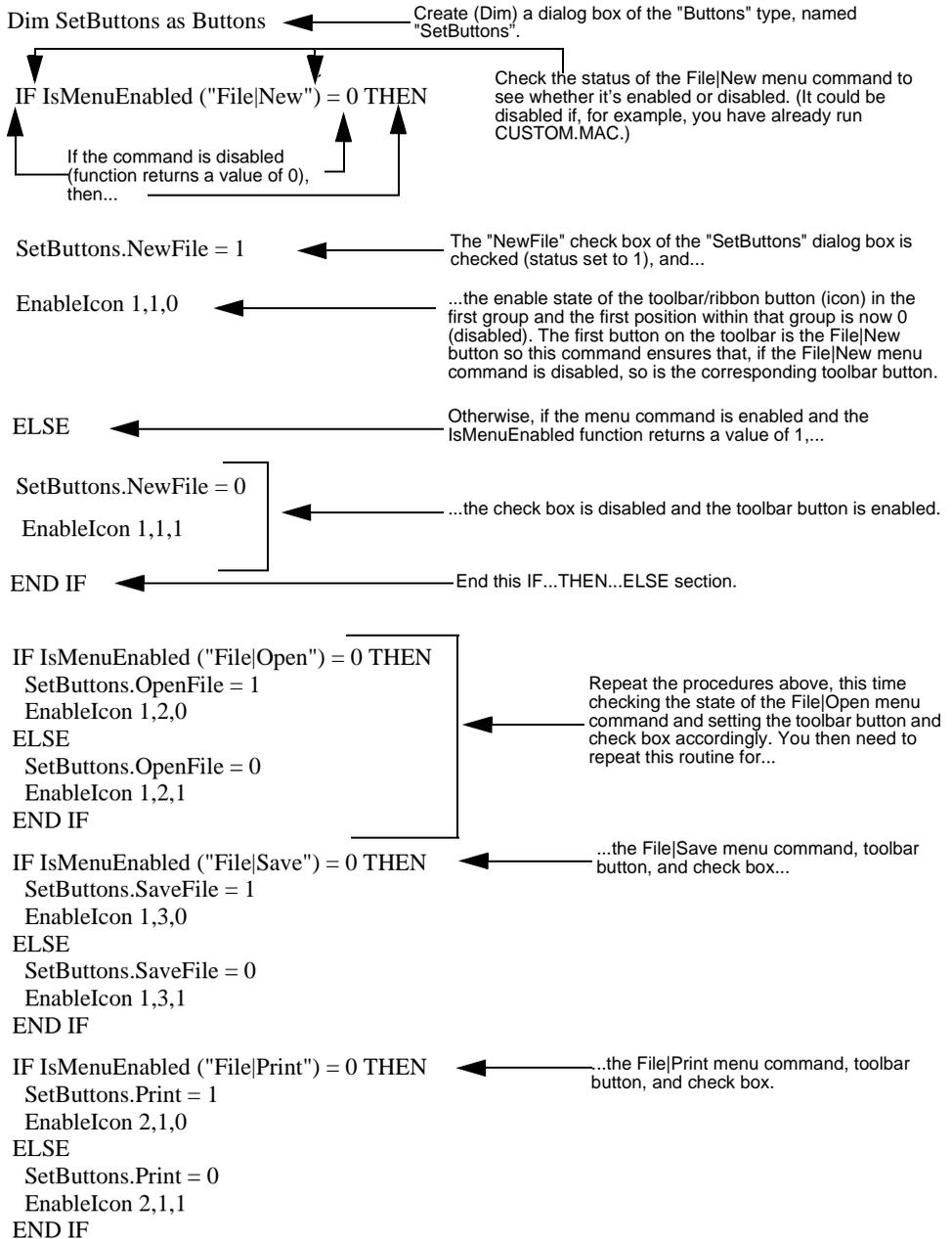
Create an OK button and place it 155 units from the left edge of the dialog box and 275 units from the top. Make it 60 units wide and 18 units tall (This is a standard button size.). Next to this button...

CancelButton 215, 275, 60, 18

...create and place a Cancel button. (This is why you created an "OnError" routine. Choosing this button will generate an error which your routine will then handle properly.)

End Dialog

End the definition of this dialog-box type.



```

IF IsMenuEnabled ("View|Zoom") = 0 THEN
  SetButtons.ViewPage = 1
  SetButtons.PageWidth = 1
  SetButtons.HundredPercent = 1
  EnableIcon 3,1,0
  EnableIcon 3,2,0
  EnableIcon 3,3,0
ELSE
  SetButtons.ViewPage = 0
  SetButtons.PageWidth = 0
  SetButtons.HundredPercent = 0
  EnableIcon 3,1,1
  EnableIcon 3,2,1
  EnableIcon 3,3,1
END IF

```

This routine is somewhat different because:
 TheView|Zoom menu command has no single toolbar button equivalent, but rather a group of zoom options (and check boxes).
 Disabling this menu command should also disable the entire group of zoom option buttons (and check boxes).

```

IF IsMenuEnabled ("Insert|Headers/Footers") = 0 THEN
  SetButtons.Header = 1
  SetButtons.Footer = 1
  EnableIcon 5,1,0
  EnableIcon 5,2,0
ELSE
  SetButtons.Header = 0
  SetButtons.Footer = 0
  EnableIcon 5,1,1
  EnableIcon 5,2,1
END IF

```

This routine likewise requires disabling a group of buttons and a group of check boxes.

```

IF IsMenuEnabled ("Tools|Sorting") = 0 THEN
  SetButtons.AscendSort = 1
  SetButtons.DescendSort = 1
  EnableIcon 6,1,0
  EnableIcon 6,2,0
ELSE
  SetButtons.AscendSort = 0
  SetButtons.DescendSort = 0
  EnableIcon 6,1,1
  EnableIcon 6,2,1
END IF

```

This routine likewise requires disabling a group of buttons and a group of check boxes.

```

IF IsMenuEnabled ("Tools|Summary Fields") = 0 THEN
  SetButtons.Sum = 1
  SetButtons.AbsValue = 1
  SetButtons.Minimum = 1
  SetButtons.Maximum = 1
  SetButtons.Count = 1
  EnableIcon 7,1,0
  EnableIcon 7,2,0
  EnableIcon 7,3,0
  EnableIcon 7,4,0
  EnableIcon 7,5,0

```

This routine likewise requires disabling a group of buttons and a group of check boxes.

```
ELSE
  SetButtons.Sum = 0
  SetButtons.AbsValue = 0
  SetButtons.Minimum = 0
  SetButtons.Maximum = 0
  SetButtons.Count = 0
  EnableIcon 7,1,1
  EnableIcon 7,2,1
  EnableIcon 7,3,1
  EnableIcon 7,4,1
  EnableIcon 7,5,1
END IF
```

```
IF IsMenuEnabled ("Tools|Merge reports") = 0 THEN
  SetButtons.Merge = 1
  EnableIcon 8,1,0
ELSE
  SetButtons.Merge = 0
  EnableIcon 8,1,1
END IF
```

```
IF IsMenuEnabled ("Tools|SQL Text") = 0 THEN
  SetButtons.SQL = 1
  EnableIcon 8,2,0
ELSE
  SetButtons.SQL = 0
  EnableIcon 8,2,1
END IF
```

```
IF IsMenuEnabled ("Format|Character") = 0 THEN
  SetButtons.Font = 1
  SetButtons.Points = 1
  SetButtons.Bold = 1
  SetButtons.Italic = 1
  SetButtons.Underline = 1
  EnableIcon 12,1,0
  EnableIcon 13,1,0
  EnableIcon 14,2,0
  EnableIcon 14,3,0
ELSE
  SetButtons.Font = 0
  SetButtons.Points = 0
  SetButtons.Bold = 0
  SetButtons.Italic = 0
  SetButtons.Underline = 0
  EnableIcon 12,1,1
  EnableIcon 13,1,1
  EnableIcon 14,1,1
  EnableIcon 14,2,1
  EnableIcon 14,3,1
END IF
```

← In this routine, there is once again a 1-to-1 correspondence between the menu command (Tools|Merge reports) and the button or check box.

← This routine actually disables several groups of ribbon options: the Fonts drop-down box (which is its own group), and the group containing the Bold, Italic, and Underline options.

```
IF IsMenuEnabled ("Format|Text Alignment") = 0 THEN
  SetButtons.Left = 1
  SetButtons.Center = 1
  SetButtons.Right = 1
  EnableIcon 15,1,0
  EnableIcon 15,2,0
  EnableIcon 15,3,0
ELSE
  SetButtons.Left = 0
  SetButtons.Center = 0
  SetButtons.Right = 0
  EnableIcon 15,1,1
  EnableIcon 15,2,1
  EnableIcon 15,3,1
END IF
```

```
IF IsMenuEnabled ("Format|Field Formats") = 0 THEN
  SetButtons.Currency = 1
  SetButtons.Number = 1
  SetButtons.Percentage = 1
  EnableIcon 16,1,0
  EnableIcon 16,2,0
  EnableIcon 16,3,0
ELSE
  SetButtons.Currency = 0
  SetButtons.Number = 0
  SetButtons.Percentage = 0
  EnableIcon 16,1,1
  EnableIcon 16,2,1
  EnableIcon 16,3,1
END IF
```

```
IF IsMenuEnabled ("Insert|Picture") = 0 THEN
  SetButtons.Drawing = 1
  EnableIcon 17,2,0
ELSE
  SetButtons.Drawing = 0
  EnableIcon 17,2,1
END IF
```

```
IF IsMenuEnabled ("Insert|Object") = 0 THEN
  SetButtons.Graph = 1
  EnableIcon 17,3,0
ELSE
  SetButtons.Graph = 0
  EnableIcon 17,3,1
END IF
```

```

IF IsMenuEnabled ("Insert|Crosstab") = 0 THEN
  SetButtons.Crosstab = 1
  EnableIcon 17,4,0
ELSE
  SetButtons.Crosstab = 0
  EnableIcon 17,4,1
END IF

```

```

IF IsMenuEnabled ("Format|Report Styles") = 0 THEN
  SetButtons.Styles = 1
  EnableIcon 18,2,0
ELSE
  SetButtons.Styles = 0
  EnableIcon 18,2,1
END IF

```

DIALOG SetButtons ←

Now that all check box and toolbar/ribbon button settings have been determined, display the SetButtons dialog box.

```

IF SetButtons.NewFile = 1 THEN
  EnableIcon 1,1,0
  EnableMenu "File|New", 0
ELSE
  EnableIcon 1,1,1
  EnableMenu "File|New", 1
END IF

```

← In earlier routines, you assessed the state of a menu command, and then set the toolbar/ribbon buttons and dialog box check boxes accordingly. Now, you implement this concept "in the other direction." That is, this routine and those that follow it ask the state of a check box, and then set the corresponding toolbar/ribbon buttons and menu command (if any). This routine specifies that if the NewFile check box of the SetButtons dialog box is checked (if its status is 1), then both the button and the menu command should be disabled. Otherwise (if its status is 0, meaning it's unchecked), both the button and the menu command should be enabled.

```

IF SetButtons.OpenFile = 1 THEN
  EnableIcon 1,2,0
  EnableMenu "File|Open", 0
ELSE
  EnableIcon 1,2,1
  EnableMenu "File|Open", 1
END IF

```

```

IF SetButtons.SaveFile = 1 THEN
  EnableIcon 1,3,0
  EnableMenu "File|Save", 0
ELSE
  EnableIcon 1,3,1
  EnableMenu "File|Save", 1
END IF

```

```

IF SetButtons.Print = 1 THEN
  EnableIcon 2,1,0
  EnableMenu ("File|Print"), 0
ELSE
  EnableIcon 2,1,1
  EnableMenu ("File|Print"), 1
END IF

```

```

IF SetButtons.ViewPage = 1 THEN
  SetButtons.PageWidth = 1
  SetButtons.HundredPercent = 1
  EnableIcon 3,1,0
  EnableIcon 3,2,0
  EnableIcon 3,3,0
  EnableMenu "View|Zoom", 0
ELSE
  SetButtons.ViewPage = 0
  SetButtons.PageWidth = 0
  SetButtons.HundredPercent = 0
  EnableIcon 3,1,1
  EnableIcon 3,2,1
  EnableIcon 3,3,1
  EnableMenu "View|Zoom", 1
END IF

```

This routine is slightly different. If this check box is checked, then the other check boxes in its group (corresponding to a group of toolbar buttons) should also be checked. However, all of the buttons (and check boxes) in this group pertain to only one menu command.

The next two routines handle the other two check boxes and buttons in this group, so that if any one check box of the group is checked, the entire group is checked in the dialog box, and the entire group of toolbar buttons becomes disabled.

You will not actually see the other check boxes become checked until the next time the dialog box appears because the code that performs this operation will not be executed until the dialog box closes.

```

IF SetButtons.PageWidth = 1 THEN
  SetButtons.ViewPage = 1
  SetButtons.HundredPercent = 1
  EnableIcon 3,1,0
  EnableIcon 3,2,0
  EnableIcon 3,3,0
  EnableMenu "View|Zoom", 0
ELSE
  SetButtons.ViewPage = 0
  SetButtons.PageWidth = 0
  SetButtons.HundredPercent = 0
  EnableIcon 3,1,1
  EnableIcon 3,2,1
  EnableIcon 3,3,1
  EnableMenu "View|Zoom", 1
END IF

```

```

IF SetButtons.HundredPercent = 1
THEN
  SetButtons.ViewPage = 1
  SetButtons.PageWidth = 1
  EnableIcon 3,1,0
  EnableIcon 3,2,0
  EnableIcon 3,3,0
  EnableMenu "View|Zoom", 0
ELSE
  SetButtons.ViewPage = 0
  SetButtons.PageWidth = 0
  SetButtons.HundredPercent = 0
  EnableIcon 3,1,1
  EnableIcon 3,2,1
  EnableIcon 3,3,1
  EnableMenu "View|Zoom", 1
END IF

```

```

IF SetButtons.ColumnMode = 1 THEN
    EnableIcon 4,1,0
ELSE
    EnableIcon 4,1,1
END IF

IF SetButtons.FormMode = 1 THEN
    EnableIcon 4,2,0
ELSE
    EnableIcon 4,2,1
END IF

IF SetButtons.Header = 1 THEN
    EnableIcon 5,1,0
    EnableIcon 5,2,0
    EnableMenu "Insert|Headers\Footers", 0
ELSE
    EnableIcon 5,1,1
    EnableIcon 5,2,1
    EnableMenu "Insert|Headers\Footers", 1
END IF

IF SetButtons.Footer = 1 THEN
    EnableIcon 5,1,0
    EnableIcon 5,2,0
    EnableMenu "Insert|Header\Footer", 0
ELSE
    EnableIcon 5,1,1
    EnableIcon 5,2,1
    EnableMenu "Insert|Header\Footer", 1
END IF

IF SetButtons.AscendSort = 1 THEN
    EnableIcon 6,1,0
    EnableIcon 6,2,0
    EnableMenu "Tools|Sorting", 0
ELSE
    EnableIcon 6,1,1
    EnableIcon 6,2,1
    EnableMenu "Tools|Sorting", 1
END IF

IF SetButtons.DescendSort = 1 THEN
    EnableIcon 6,1,0
    EnableIcon 6,2,0
    EnableMenu "Tools|Sorting", 0
ELSE
    EnableIcon 6,1,1
    EnableIcon 6,2,1
    EnableMenu "Tools|Sorting", 1
END IF

```

```

IF SetButtons.Sum = 1 THEN
  EnableIcon 7,1,0
  EnableIcon 7,2,0
  EnableIcon 7,3,0
  EnableIcon 7,4,0
  EnableIcon 7,5,0
  EnableMenu "Tools|Summary Fields", 0
  SetButtons.AbsValue = 1
  SetButtons.Minimum = 1
  SetButtons.Maximum = 1
  SetButtons.Count = 1
ELSE
  EnableIcon 7,1,1
  EnableIcon 7,2,1
  EnableIcon 7,3,1
  EnableIcon 7,4,1
  EnableIcon 7,5,1
  EnableMenu "Tools|Summary Fields", 1
  SetButtons.AbsValue = 0
  SetButtons.Minimum = 0
  SetButtons.Maximum = 0
  SetButtons.Count = 0
END IF

IF SetButtons.AbsValue = 1 THEN
  EnableIcon 7,1,0
  EnableIcon 7,2,0
  EnableIcon 7,3,0
  EnableIcon 7,4,0
  EnableIcon 7,5,0
  EnableMenu "Tools|Summary Fields", 0
  SetButtons.Sum = 1
  SetButtons.Minimum = 1
  SetButtons.Maximum = 1
  SetButtons.Count = 1
ELSE
  EnableIcon 7,1,1
  EnableIcon 7,2,1
  EnableIcon 7,3,1
  EnableIcon 7,4,1
  EnableIcon 7,5,1
  EnableMenu "Tools|Summary Fields", 1
  SetButtons.AbsValue = 0
  SetButtons.Minimum = 0
  SetButtons.Maximum = 0
  SetButtons.Count = 0
END IF

```

```

IF SetButtons.Minimum = 1 THEN
  EnableIcon 7,1,0
  EnableIcon 7,2,0
  EnableIcon 7,3,0
  EnableIcon 7,4,0
  EnableIcon 7,5,0
  EnableMenu "Tools|Summary Fields", 0
  SetButtons.AbsValue = 1
  SetButtons.Sum = 1
  SetButtons.Maximum = 1
  SetButtons.Count = 1
ELSE
  EnableIcon 7,1,1
  EnableIcon 7,2,1
  EnableIcon 7,3,1
  EnableIcon 7,4,1
  EnableIcon 7,5,1
  EnableMenu "Tools|Summary Fields", 1
  SetButtons.AbsValue = 0
  SetButtons.Sum = 0
  SetButtons.Maximum = 0
  SetButtons.Count = 0
END IF

IF SetButtons.Maximum = 1 THEN
  EnableIcon 7,1,0
  EnableIcon 7,2,0
  EnableIcon 7,3,0
  EnableIcon 7,4,0
  EnableIcon 7,5,0
  EnableMenu "Tools|Summary Fields", 0
  SetButtons.AbsValue = 1
  SetButtons.Minimum = 1
  SetButtons.Sum = 1
  SetButtons.Count = 1
ELSE
  EnableIcon 7,1,1
  EnableIcon 7,2,1
  EnableIcon 7,3,1
  EnableIcon 7,4,1
  EnableIcon 7,5,1
  EnableMenu "Tools|Summary Fields", 1
  SetButtons.AbsValue = 0
  SetButtons.Minimum = 0
  SetButtons.Sum = 0
  SetButtons.Count = 0
END IF

```

```

If SetButtons.Count = 1 THEN
  EnableIcon 7,1,0
  EnableIcon 7,2,0
  EnableIcon 7,3,0
  EnableIcon 7,4,0
  EnableIcon 7,5,0
  EnableMenu "Tools|Summary Fields", 0
  SetButtons.AbsValue = 1
  SetButtons.Minimum = 1
  SetButtons.Maximum = 1
  SetButtons.Sum = 1
ELSE
  EnableIcon 7,1,1
  EnableIcon 7,2,1
  EnableIcon 7,3,1
  EnableIcon 7,4,1
  EnableIcon 7,5,1
  EnableMenu "Tools|Summary Fields", 1
  SetButtons.AbsValue = 0
  SetButtons.Minimum = 0
  SetButtons.Maximum = 0
  SetButtons.Sum = 0
END IF

IF SetButtons.Merge = 1 THEN
  EnableIcon 8,1,0
  EnableMenu "Tools|Merge reports", 0
ELSE
  EnableIcon 8,1,1
  EnableMenu "Tools|Merge reports", 1
END IF

IF SetButtons.SQL = 1 THEN
  EnableIcon 8,2,0
  EnableMenu "Tools|SQL Text", 0
ELSE
  EnableIcon 8,2,1
  EnableMenu "Tools|SQL Text", 1
END IF

IF SetButtons.BestFit = 1 THEN
  EnableIcon 8,3,0
ELSE
  EnableIcon 8,3,1
END IF

IF SetButtons.Font = 1 THEN
  EnableIcon 12,1,0
  EnableIcon 13,1,0
  EnableIcon 14,1,0
  EnableIcon 14,2,0
  EnableIcon 14,3,0
  EnableMenu "Format|Character", 0

```

```

SetButtons.Points = 1
SetButtons.Bold = 1
SetButtons.Italic = 1
  SetButtons.Underline = 1
ELSE
  EnableIcon 12,1,1
  EnableIcon 13,1,1
  EnableIcon 14,1,1
  EnableIcon 14,2,1
  EnableIcon 14,3,1
  EnableMenu "Format|Character", 1
  SetButtons.Points = 0
  SetButtons.Bold = 0
  SetButtons.Italic = 0
  SetButtons.Underline = 0
END IF

IF SetButtons.Points = 1 THEN
  EnableIcon 12,1,0
  EnableIcon 13,1,0
  EnableIcon 14,1,0
  EnableIcon 14,2,0
  EnableIcon 14,3,0
  EnableMenu "Format|Character", 0
  SetButtons.Font = 1
  SetButtons.Bold = 1
  SetButtons.Italic = 1
  SetButtons.Underline = 1
ELSE
  EnableIcon 12,1,1
  EnableIcon 13,1,1
  EnableIcon 14,1,1
  EnableIcon 14,2,1
  EnableIcon 14,3,1
  EnableMenu "Format|Character", 1
  SetButtons.Font = 0
  SetButtons.Bold = 0
  SetButtons.Italic = 0
  SetButtons.Underline = 0
END IF

IF SetButtons.Bold = 1 THEN
  EnableIcon 12,1,0
  EnableIcon 13,1,0
  EnableIcon 14,1,0
  EnableIcon 14,2,0
  EnableIcon 14,3,0
  EnableMenu "Format|Character", 0
  SetButtons.Points = 1
  SetButtons.Font = 1
  SetButtons.Italic = 1
  SetButtons.Underline = 1

```

```

ELSE
  EnableIcon 12,1,1
  EnableIcon 13,1,1
  EnableIcon 14,1,1
  EnableIcon 14,2,1
  EnableIcon 14,3,1
  EnableMenu "Format|Character", 1
  SetButtons.Points = 0
  SetButtons.Font = 0
  SetButtons.Italic = 0
  SetButtons.Underline = 0
END IF

IF SetButtons.Italic = 1 THEN
  EnableIcon 12,1,0
  EnableIcon 13,1,0
  EnableIcon 14,1,0
  EnableIcon 14,2,0
  EnableIcon 14,3,0
  EnableMenu "Format|Character", 0
  SetButtons.Points = 1
  SetButtons.Bold = 1
  SetButtons.Font = 1
  SetButtons.Underline = 1
ELSE
  EnableIcon 12,1,1
  EnableIcon 13,1,1
  EnableIcon 14,1,1
  EnableIcon 14,2,1
  EnableIcon 14,3,1
  EnableMenu "Format|Character", 1
  SetButtons.Points = 0
  SetButtons.Bold = 0
  SetButtons.Italic = 0
  SetButtons.Underline = 0
END IF

IF SetButtons.Underline = 1 THEN
  EnableIcon 12,1,0
  EnableIcon 13,1,0
  EnableIcon 14,1,0
  EnableIcon 14,2,0
  EnableIcon 14,3,0
  EnableMenu "Format|Character", 0
  SetButtons.Points = 1
  SetButtons.Bold = 1
  SetButtons.Italic = 1
  SetButtons.Underline = 1

```

```

ELSE
  EnableIcon 12,1,1
  EnableIcon 13,1,1
  EnableIcon 14,1,1
  EnableIcon 14,2,1
  EnableIcon 14,3,1
  EnableMenu "Format|Character", 1
  SetButtons.Points = 0
  SetButtons.Bold = 0
  SetButtons.Italic = 0
  SetButtons.Underline = 0
END IF

IF SetButtons.Left = 1 THEN
  EnableIcon 15,1,0
  EnableIcon 15,2,0
  EnableIcon 15,3,0
  EnableMenu "Format|Text Alignment", 0
  SetButtons.Center = 1
  SetButtons.Right = 1
ELSE
  EnableIcon 15,1,1
  EnableIcon 15,2,1
  EnableIcon 15,3,1
  EnableMenu "Format|Text Alignment", 1
  SetButtons.Center = 0
  SetButtons.Right = 0
END IF

IF SetButtons.Center = 1 THEN
  EnableIcon 15,1,0
  EnableIcon 15,2,0
  EnableIcon 15,3,0
  EnableMenu "Format|Text Alignment", 0
  SetButtons.Left = 1
  SetButtons.Right = 1
ELSE
  EnableIcon 15,1,1
  EnableIcon 15,2,1
  EnableIcon 15,3,1
  EnableMenu "Format|Text Alignment", 1
  SetButtons.Left = 0
  SetButtons.Right = 0
END IF

IF SetButtons.Right = 1 THEN
  EnableIcon 15,1,0
  EnableIcon 15,2,0
  EnableIcon 15,3,0
  EnableMenu "Format|Text Alignment", 0
  SetButtons.Left = 1
  SetButtons.Center = 1

```

```

ELSE
  EnableIcon 15,1,1
  EnableIcon 15,2,1
  EnableIcon 15,3,1
  EnableMenu "Format|Text Alignment", 1
  SetButtons.Left = 0
  SetButtons.Center = 0
END IF

IF SetButtons.Currency = 1 THEN
  EnableIcon 16,1,0
  EnableIcon 16,2,0
  EnableIcon 16,3,0
  EnableMenu "Format|Field Formats", 0
  SetButtons.Number = 1
  SetButtons.Percentage = 1
ELSE
  EnableIcon 16,1,1
  EnableIcon 16,2,1
  EnableIcon 16,3,1
  EnableMenu "Format|Field Formats", 1
  SetButtons.Number = 0
  SetButtons.Percentage = 0
END IF

IF SetButtons.Number = 1 THEN
  EnableIcon 16,1,0
  EnableIcon 16,2,0
  EnableIcon 16,3,0
  EnableMenu "Format|Field Formats", 0
  SetButtons.Currency = 1
  SetButtons.Percentage = 1
ELSE
  EnableIcon 16,1,1
  EnableIcon 16,2,1
  EnableIcon 16,3,1
  EnableMenu "Format|Field Formats", 1
  SetButtons.Currency = 0
  SetButtons.Percentage = 0
END IF

IF SetButtons.Percentage = 1 THEN
  EnableIcon 16,1,0
  EnableIcon 16,2,0
  EnableIcon 16,3,0
  EnableMenu "Format|Field Formats", 0
  SetButtons.Number = 1
  SetButtons.Currency = 1

```

```

ELSE
  EnableIcon 16,1,1
  EnableIcon 16,2,1
  EnableIcon 16,3,1
  EnableMenu "Format|Field Formats", 1
  SetButtons.Number = 0
  SetButtons.Currency = 0
END IF

IF SetButtons.Text = 1 THEN
  EnableIcon 17,1,0
ELSE
  EnableIcon 17,1,1
END IF

IF SetButtons.Drawing = 1 THEN
  EnableIcon 17,2,0
  EnableMenu "Insert|Picture", 0
ELSE
  EnableIcon 17,2,1
  EnableMenu "Insert|Picture", 1
END IF

IF SetButtons.Graph = 1 THEN
  EnableIcon 17,3,0
  EnableMenu "Insert|Object", 0
ELSE
  EnableIcon 17,3,1
  EnableMenu "Insert|Object", 1
END IF

IF SetButtons.Crosstab = 1 THEN
  EnableIcon 17,4,0
  EnableMenu "Insert|Crosstab", 0
ELSE
  EnableIcon 17,4,1
  EnableMenu "Insert|Crosstab", 1
END IF

IF SetButtons.Extract = 1 THEN
  EnableIcon 18,1,0
ELSE
  EnableIcon 18,1,1
END IF

IF SetButtons.Styles = 1 THEN
  EnableIcon 18,2,0
  EnableMenu "Format|Report Styles", 0
ELSE
  EnableIcon 18,2,1
  EnableMenu "Format|Report Styles", 1
END IF

```



← There are some cases in which a toolbar or ribbon button does not have an equivalent or corresponding menu command.

