3

Creating Screens

This chapter describes how to complete two screens you created earlier, how to copy one screen and adjust it to make a new screen, and how to create a fourth screen.

Overview

In Chapter 2, you created two screens named:

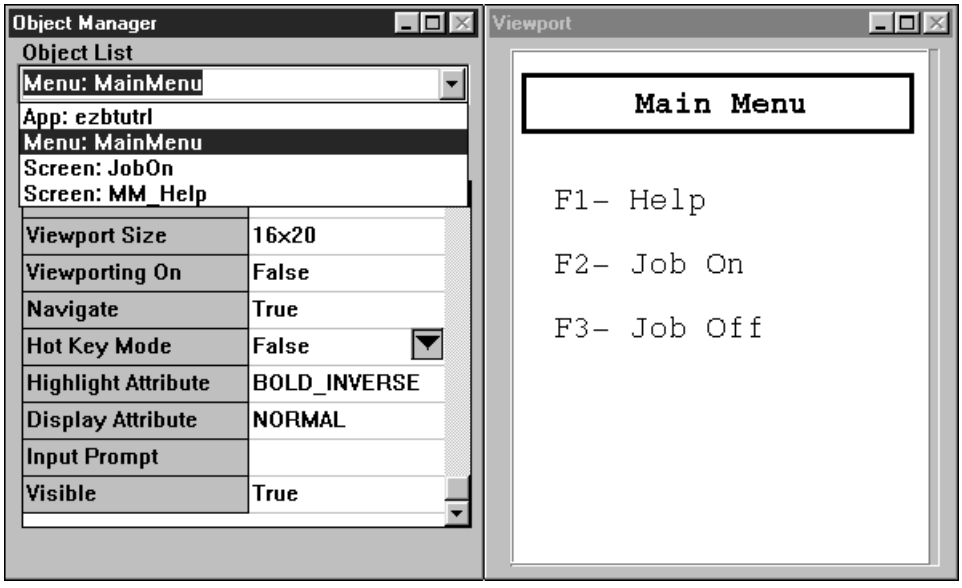
- MM_Help
- JobOn

Do you remember how you did this? In Task 2, you created two menu items (commands) on the Main Menu and indicated the above screen names as Call Screen properties for them. The Call Screen properties indicate the screens the user will go to when one of the menu commands is chosen.

When EZBuilder checked for those screens in your Object List, it found the screens were not yet created and you were prompted as to whether or not you wanted to create the screens. Each screen was created when you responded Yes to the prompt. Although you didn't see the screens at that time, they were added to the Object List.

Whenever you want to see a list of high-level ("parent") components you have created, go to the Object List above the Properties and Actions area. Check the drop-down list for the names of applications, menus, screens, and transactions you have created. To see the "children" objects created for each of these parent components, from the View menu, choose the Resources command.

To see the Object List contents, click on its down arrow. An example Object List is illustrated next. MainMenu is highlighted in the list because the MainMenu menu is currently the active object in the Viewport, and you see its properties.



The screen creation was all done transparently, and you continued your work on the Main Menu at that time. The two screens were created with default settings, but with no data fields or labels on them.

Now, to continue the tutorial exercise, you will design the look of the two screens you already created and you will create the third and fourth screens.

Task 4: Creating the MM_Help Screen

For this part of the tutorial exercise, you will label the MM_Help screen in the same manner that you labeled the Main Menu. You will provide a scrolling section in that menu, and you will enter the help text into that section.

Note: *A scrolling section is a special type of field that allows the user to scroll up and down through text that is too long to be contained in a regular field on the display area.*

Labeling the MM_Help Screen

Note: *Because you did basically the same steps when you created the label for the Main Menu, there are fewer illustrations shown here. If you need to see illustrations of the similar steps, refer back to Task 1 in Chapter 2.*

To make a screen identification label for the MM_Help screen

1. Click the down arrow in the Object List and choose the MM_Help screen. Because the items are listed in alphabetic order, MM_Help is last.
2. From the File menu, choose the Add Label command. The label object is placed in the Viewport.
3. Enter Row = 1 and Column = 4. These coordinates center the label near the top of the Viewport.
4. Click to highlight the default Caption property. Type “Main Menu Help” to replace the default text, and verify the label’s new text in the Viewport.
5. Click the down arrow to access the Display Attribute list of choices, and change the property from NORMAL to BOLD.
6. Click the Box button at the right end of the toolbar. The label that is currently selected (the one you last created) is automatically boxed.

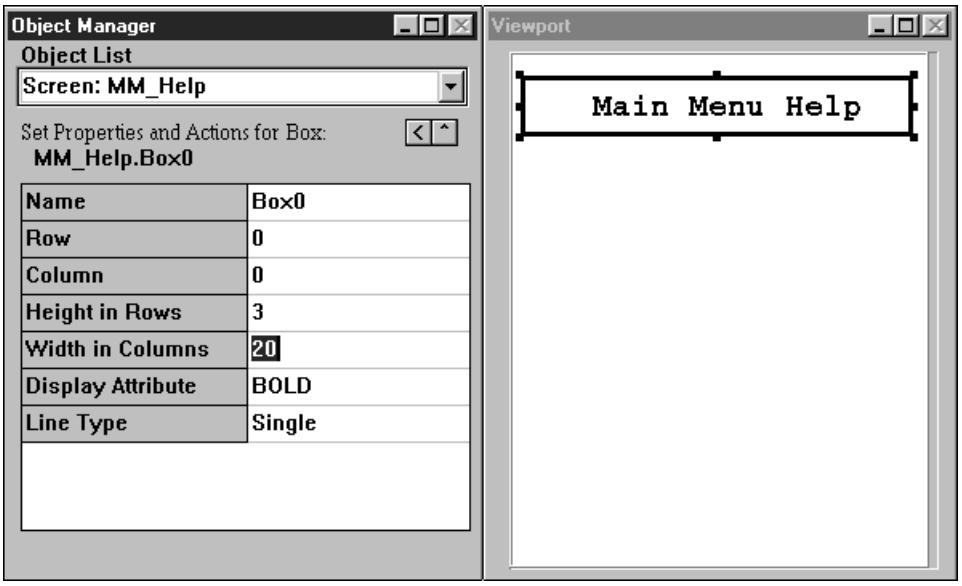
   Note: *Or, from the File menu, you could choose the Add Draw Box command to box a label. Using the toolbar’s Box button is quicker.*

7. Verify results in the Viewport; the text “Main Menu Help” should be boxed.

8. Enter Column = 0 and Width in Columns = 20 to adjust the box size and placement so it matches the label box in the Main Menu (Row and Height in Rows remain as they are). Click any other property to end the entry.

   Note: *You could also go to the Main Menu and select the box object from there. Then from the Edit menu, choose the Copy Object command. Return to the Main Menu Help screen, and from the Edit menu, choose the Paste Object command. But, this method is more time-consuming.*

9. Check your result of the previous steps with the next illustration.

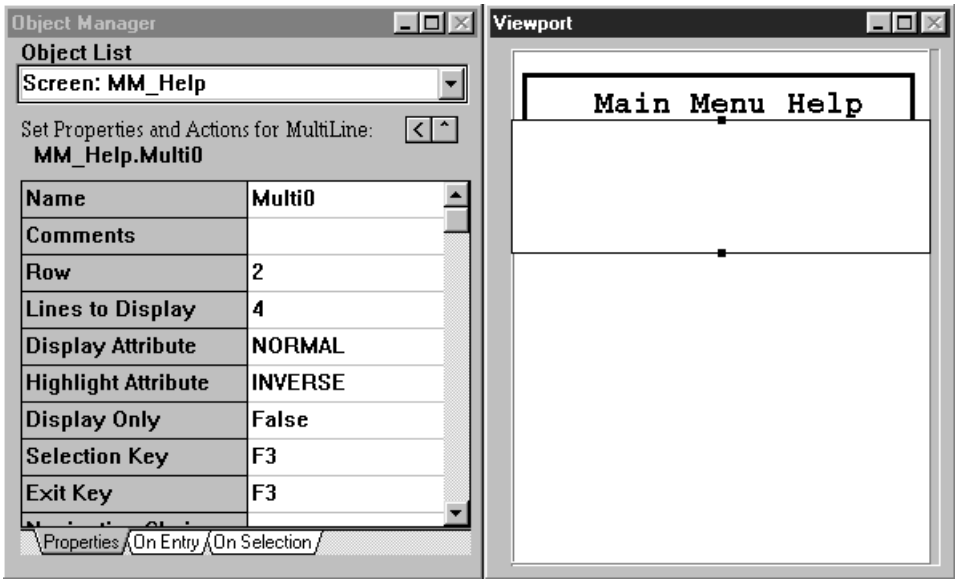


Creating a Scrolling Section to Contain Help Data

When you need to present more text than can be displayed at one time on the screen, you can create a multiple-line object (called a “scrolling section”) which will allow the user of your program to scroll up and down through the text as needed. A scrolling section is often used for help text, as demonstrated in this tutorial exercise.

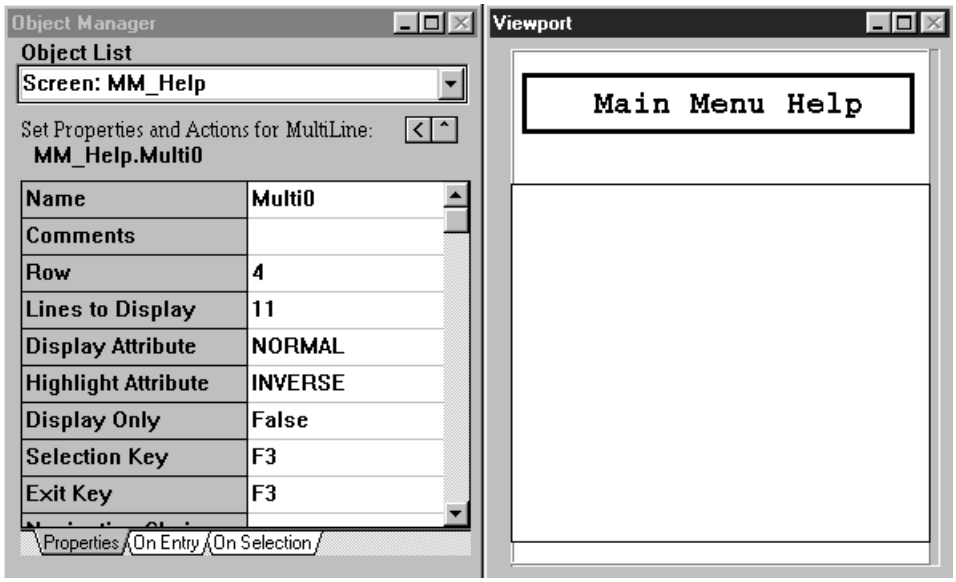
To create a scrolling section and enter its help text

1. Click the Scrolling Section button in the toolbar (fourth button from the right) to quickly create a scrolling section. Or, from the File menu, choose the Add Scrolling Section command.
2. Notice the scrolling section is a multiple-line object with two handles, as illustrated next.



3. Enter Row = 4 to lower the top of this object to Row 4 on your Viewport.
4. Enter Lines to Display = 11 to adjust this multiple-line object to the correct vertical size.
5. Click the Viewport window, somewhere outside the scrolling section, to turn the handles off (deselect the scrolling section). Check your result with the illustration next.

Note: *You cannot enter the help text until the scrolling section is deselected.*



6. Click once inside the scrolling section on the Viewport; this places the cursor inside the scrolling section so you can enter text. Enter the following help text. Do not type the double quote marks.

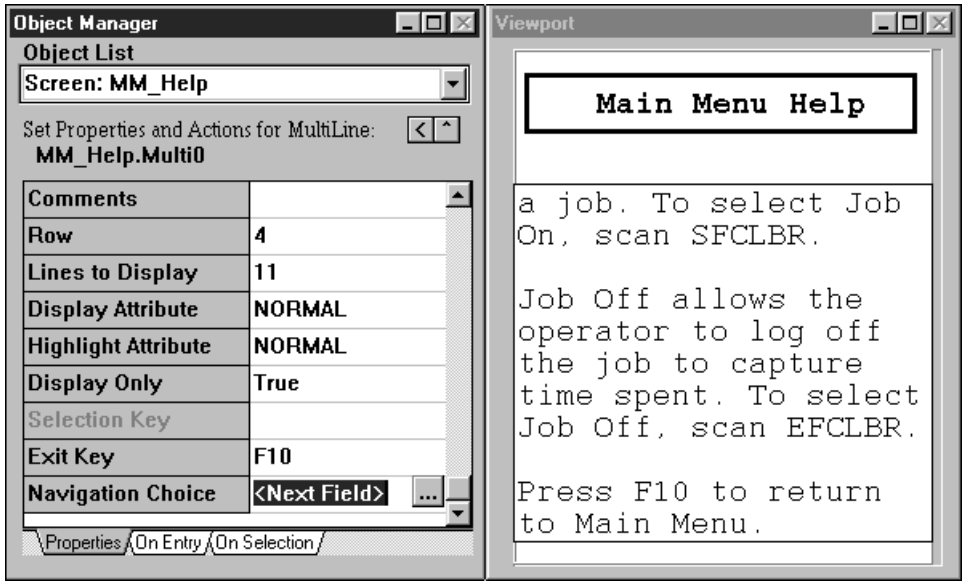
   "Job On allows the operator to log onto a job. To select Job On, scan SFCLBR.

   Job Off allows the operator to log off the job to capture the time spent. To select Job Off, scan EFCLBR.

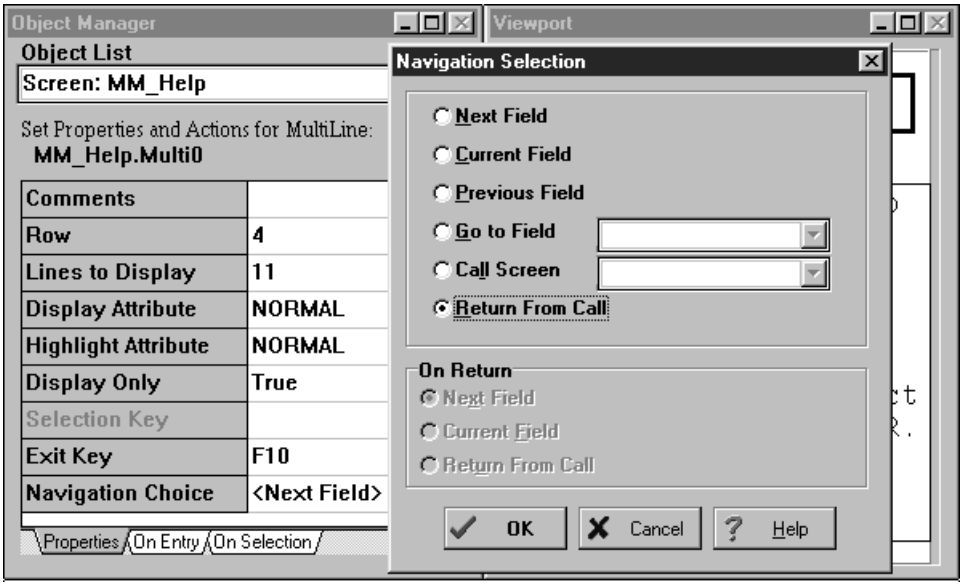
   Press F10 to return to the Main Menu."

   Notice the text wraps around as you continue typing. Eventually, you cannot see the top part of the help text. Use the up and down arrow keys on your keyboard to scroll up and down, and check for typing errors.

7. Change the Highlight Attribute from INVERSE to NORMAL.
8. Double-click the word "False" in the Display Only property to change it to "True."
9. Type "F10" as the Exit Key.
10. Scroll down the Properties and Actions area and click the Navigation Choice property, then click its small button with three dots on it, as illustrated next. This button brings up a Navigation Selection dialog box.

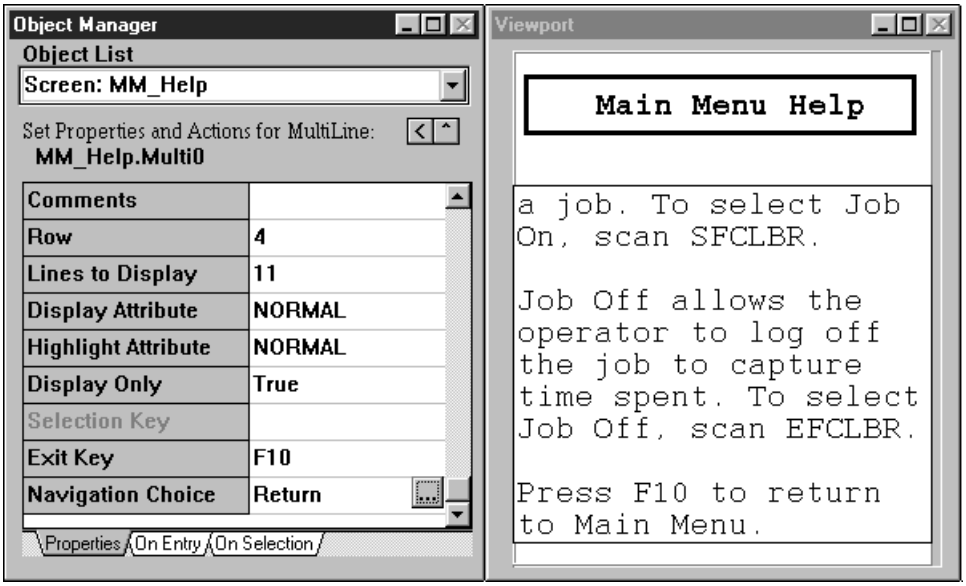


11. The Navigation Selection dialog box shows the default selection (indicated by a small round circle) is the Next Field option. Click to change the selected option to Return From Call, as illustrated next.



12. Click OK in the Navigation Selection dialog box. Notice the word "Return" automatically becomes the Navigation Choice property, as illustrated next. (This allows the return to the previously viewed screen.)

    Note: *Later, during program run time, when the user presses F1 to view the Main Menu Help screen and wants to go back to the previously viewed screen, the user presses F10 and returns to the Main Menu.*



13. You have completed Task 4. Save your file and continue with Task 5.

Task 5: Creating the JobOn Screen

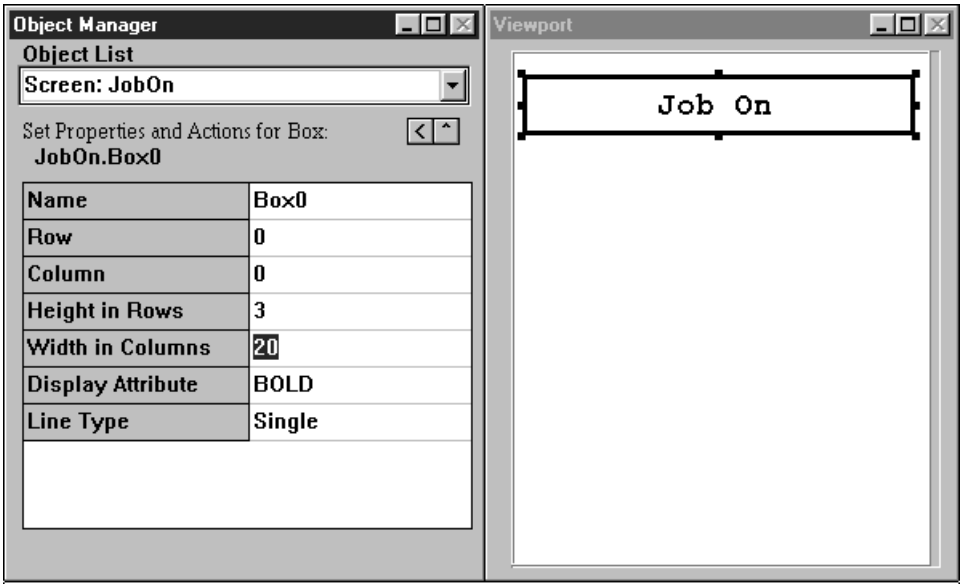
For this part of the tutorial exercise, you will first make a label for the top of the JobOn screen, then you will define three data fields (input areas) for the screen.

Labeling the JobOn Screen

Note: *Because these steps are similar to the steps you did when you created labels for the Main Menu and MM_Help screen, there are fewer illustrations here. If you need to see illustrations of the similar steps, refer back to Task 1 in Chapter 2.*

To make a label to identify the JobOn screen

1. Click the down arrow in the Object List and choose the JobOn screen.
2. From the toolbar, click the Add Label button (the one with the large "A"). The label object is placed in the Viewport.
3. Drag the label in the Viewport to Row = 1, Column = 7 to center the label.
4. Type "Job On" as the Caption property. Verify the label's new text in the Viewport.
5. Change the Display Attribute from NORMAL to BOLD.
6. Click the Box button in the toolbar to box the label "Job On."
7. Enter Column = 0 and Width in Columns = 20 to adjust box. Check your result with the next illustration.



Creating and Labeling the Badge ID Number Data Field

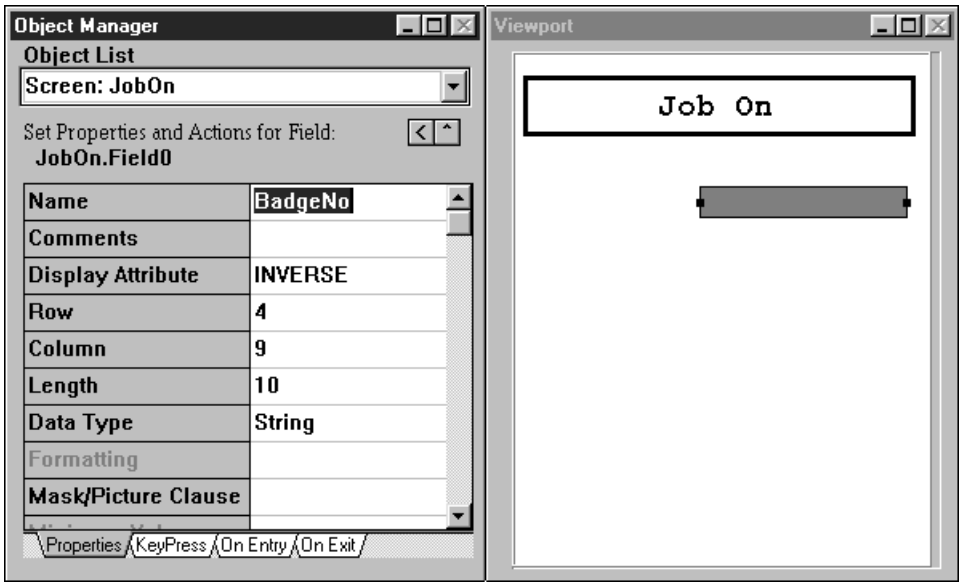
Next, you will define the first of three input data fields needed for the Job On transaction. Each data field must be identified so the user knows what kind of data goes in that area. This identification is done by labeling the data field with appropriate caption text.

Note: *For consistency in this tutorial, for each data field, you will first define the data field and then define its label. The second time you work through this tutorial, you may prefer to define the label first followed by the data field.*

The first data field is a field of ten characters into which Badge Identification Number data will be entered or scanned during your program's run time.

To define the Badge ID Number data field

1. From the File menu, choose the Add Field command.
2. Enter Row = 4 and Column = 9 to position the field on the Viewport.
3. Enter Length = 10.
4. Click once to highlight the default Name property (“Field0”) and type “BadgeNo” as the Name property for this field, as illustrated next.



Note: *The Display Attribute property remains the default INVERSE value. All data fields will be INVERSE to indicate each data field as a shaded area on the screen. This is necessary because, although NORMAL shows an outline of the data*

area on the Viewport, NORMAL does not create an outline of the data area on the user's terminal.

To create the Badge ID Number data field's label

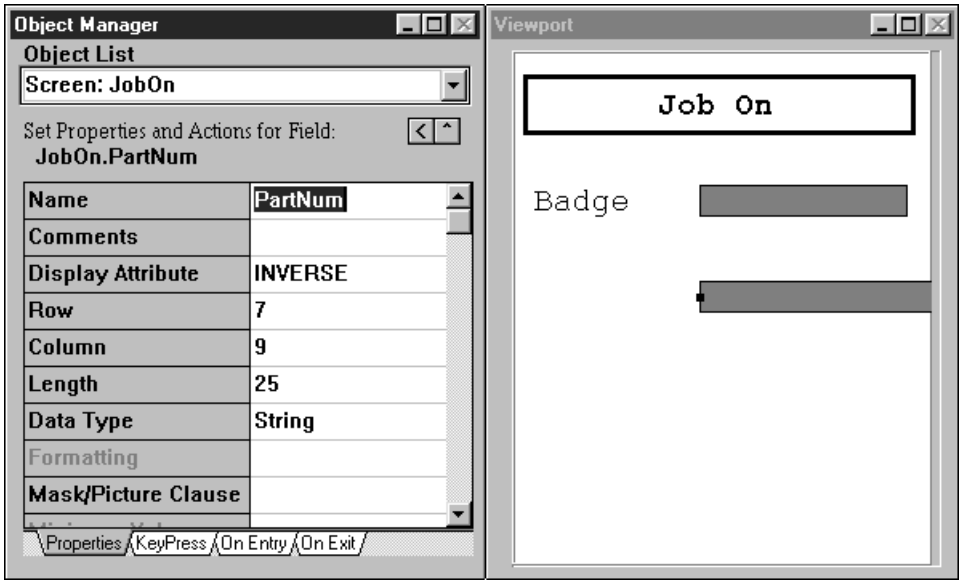
1. From the File menu or the toolbar, choose the Add Label command.
2. Position the label at Row = 4, Column = 1.
3. Type "Badge" as the Caption property. The Caption property identifies the kind of data that goes into the corresponding data field (the shaded area on the Viewport). Check your result with the next illustration.

Creating and Labeling the Part Number Data Field

Next, you will define the second data field. This field has 25 characters and will hold Part Number data which is entered during your program's run time.

To create the Part Number data field

1. From the File menu, choose the Add Field command. Or, from the toolbar, press the button with the letters "ab."
2. Position the data field at Row = 7, Column = 9.
3. Enter Length = 25.
4. Type "PartNum" as the Name property, as illustrated next.



To create the Part Number data field's label

1. From the File menu or toolbar, choose the Add Label command.
2. Enter Row = 7, Column = 1.
3. Type "Part" as the Caption property. Check your result with the next illustration.

Creating and Labeling the Order Number Data Field

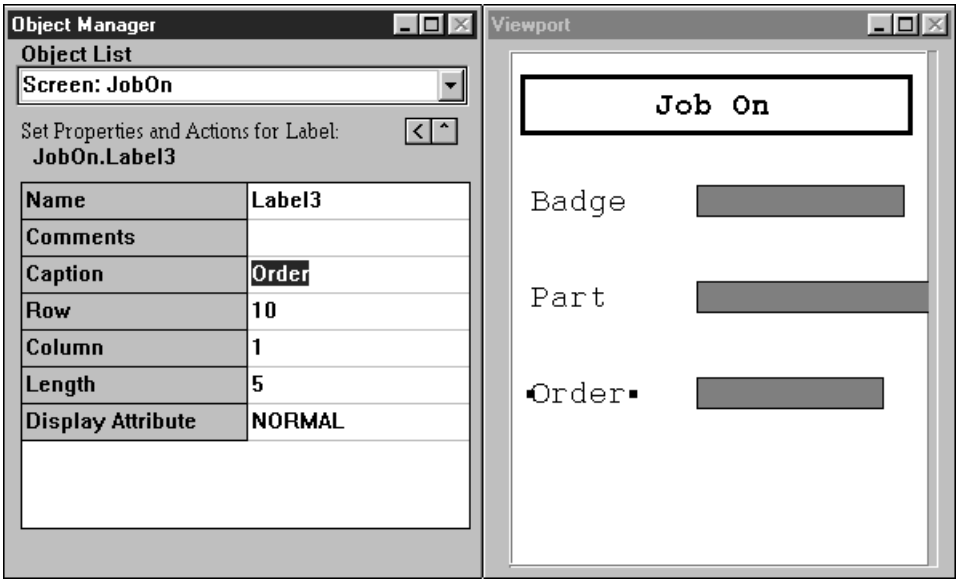
Next, you will define the third data field for this tutorial exercise. This data field of nine characters will contain Order Number data which is entered or scanned during your program's run time.

To create the Order Number data field

1. From the File menu or toolbar, choose the Add Field command.
2. Enter Row = 10, Column = 9.
3. Enter Length = 9.
4. Type “OrderNo” as the Name property. Check your result with the next illustration.

To create the Order Number data field’s label

1. From the File menu or toolbar, choose the Add Label command.
2. Enter Row = 10, Column = 1.
3. Type “Order” as the Caption property. Check your result with the illustration next.



Wrapping Data on the JobOn Screen

Notice in the above illustration that the 25-character Part Number data field is not shown in its entirety on the Viewport. In other words, it does not "wrap" around to another row, but goes off the right edge of the Viewport.

Setting data fields to wrap is not done as attributes for individual fields. It is done on the parent screen level for all fields on the screen.

To set all JobOn data fields to wrap

1. Click the button marked with the ^ symbol, located to the right of the text "Set Properties and Actions for Label" in the Object Manager window. This will make the parent screen, JobOn, active.
2. Double-click on "False" to change the Wrap at Edge property from "False" to "True," as illustrated next.

   Note: *Although the Viewport cannot reflect the above change and show wrapped fields, the wrapped fields will be shown correctly when you build your program and run it on the Simulator or on the terminal.*

   Note: *Up to 240 characters of text can be wrapped.*

Programming the Function Keys

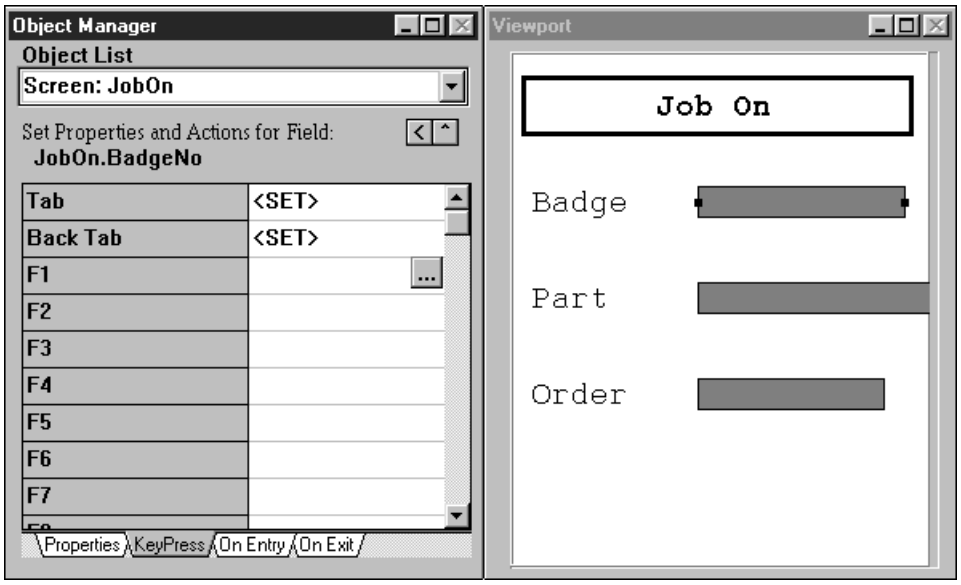
Programming the function keys in EZBuilder is easy and quick to do. This process is described next.

There are at least two reasons why you may want your program's user to be able to press a function key when entering or about to enter data in a field.

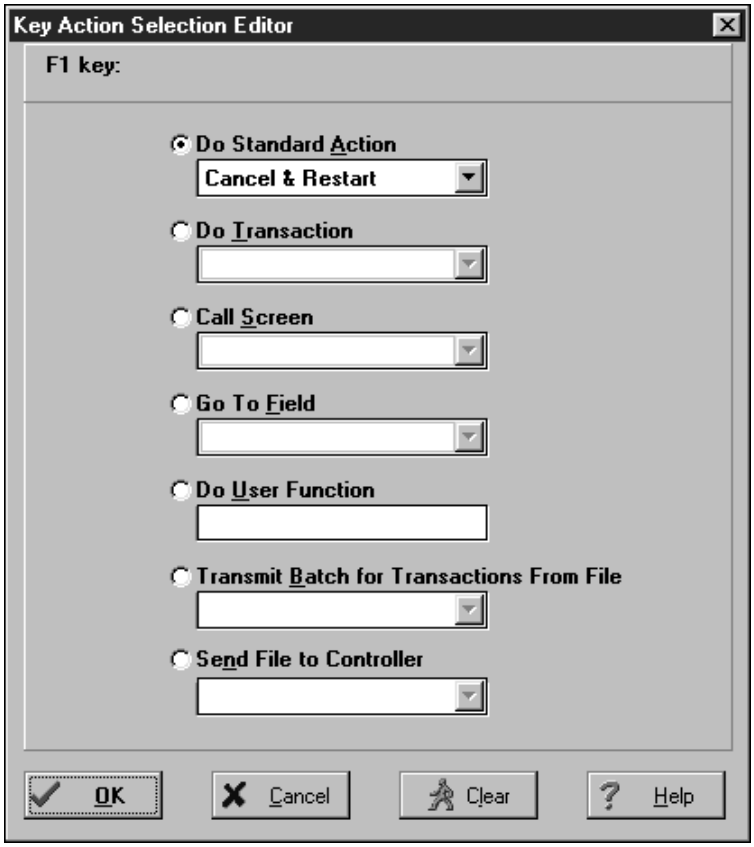
- When the user needs to know what type of data is to be entered in the field and how many characters are allowed for that data, pressing a function key can take the user to a help screen that describes the characteristics of the data field.
- When the user makes a mistake in entering data and wants to re-enter data for that transaction, a function key can take the user back to the previous data field or a different function key can take the user back to the beginning of the same data field.

To associate a help screen with a function key

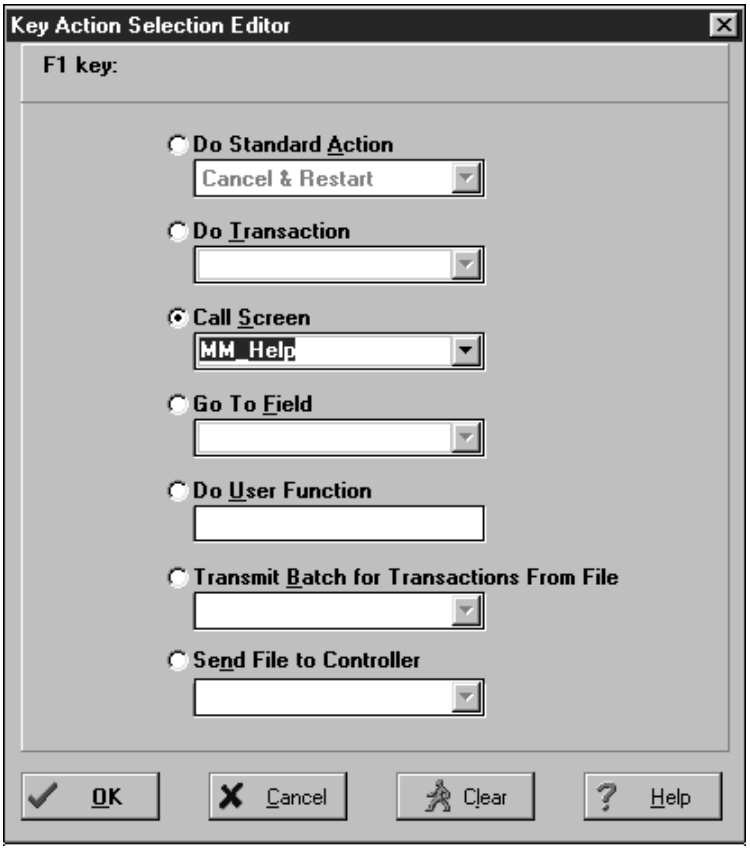
1. Begin by having the JobOn screen in the Viewport (as illustrated above).
2. In the Viewport, double-click the shaded Badge data field to select it. The properties and actions for this field are now available for setting.
3. At the bottom of the Object Manager, click the KeyPress tab. You will see the options illustrated next.



4. Click the (blank) F1 property, then click on its small button with the three dots to bring up the Key Action Selection Editor for the F1 key, as illustrated next.



5. Click to select the small circle button for the Call Screen option, and review its drop-down list. Double-click to choose MM_Help (currently your only list option), as illustrated next.



6. Suppose you now decide to make a new screen that explains the data field characteristics—instead of adding that information to the MM_Help screen. Over the highlighted MM_Help text shown in the above illustration, type "Data_Help" as the screen to be called for the F1 key press action.
7. Click OK on the bottom of the Key Selection Editor dialog box. When prompted if you want to create the Data_Help screen (currently non-existent), click Yes. You will see the F1 key's property is "<SET>."

   Note: *Later, when the program is running and the user presses F1 when the BadgeNo data field is selected (or when Badge ID data has been partially entered), the screen named "Data_Help" will be presented to the user.*

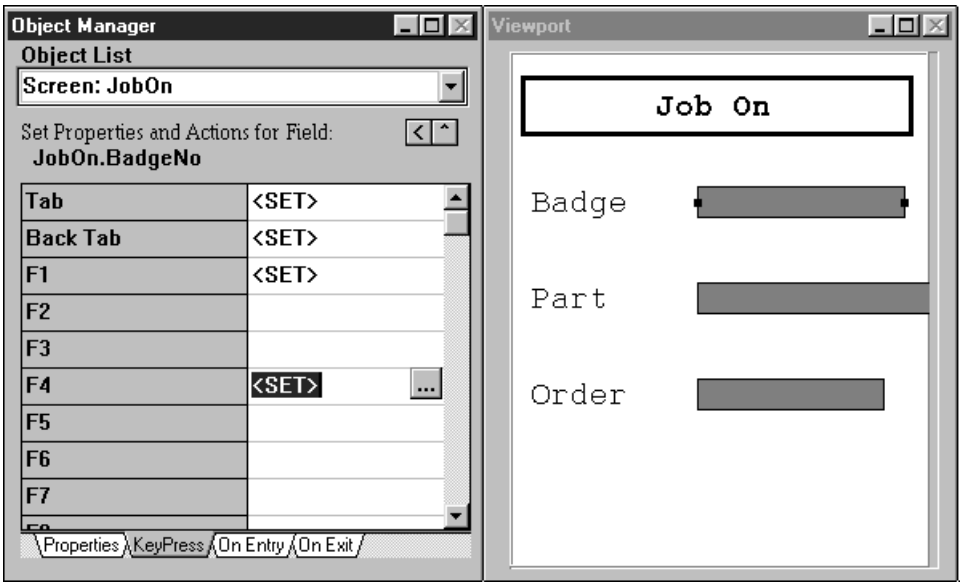
8. Check the Object List to verify that you now have the (empty) screen named Data_Help listed. In Task 7, later in this chapter, you will complete the Data_Help screen in the same manner that you previously created the

MM_Help screen. On the Data_Help screen, you will describe the characteristics of the three data fields in a scrolling section

9. Return to Step 2 and select the PartNum data field instead of the BadgeNo data field. Repeat Steps 3 through 7 to send the user to the Data_Help screen when F1 is pressed in the PartNum field.
10. Return to Step 2 and select the OrderNo data field instead of the BadgeNo data field. Repeat Steps 3 through 7 to send the user to the Data_Help screen when F1 is pressed in the OrderNo field.

To return to a previous data entry field and correct data entry

1. Begin by having the JobOn screen in the Viewport.
2. Double-click in the Viewport to select the BadgeNo data field.
3. At the bottom of the Object Manager, click the KeyPress tab so you can set the properties and actions for field JobOn.BadgeNo.
4. Click the (blank) F4 property, then click on its small button with the three dots to bring up the Key Action Selection Editor for the F4 key, (similar to the previous illustration).
5. Click the circle button for the Go to Field option and review its drop-down list. Select the JobOn.PartNum field from the list. This will allow the user to go down to the next field (PartNum) when done with the BadgeNo field before completely filling the BadgeNo field with data.
6. Click OK on the Key Action Selection Editor dialog box and see the F4 key's property is "<SET>," as illustrated next.



Note: *Later, when the program is running and the user presses F4 when the Badge data field is selected (or when Badge ID data has been partially entered), the cursor is sent to the beginning of the PartNum data field so the user can enter that data.*

7. Set the F4 key also for the PartNum and OrderNo fields (refer back to Steps 2 through 6 on the previous page). Code it so that when the user presses F4 in the PartNum field, the user returns to the OrderNo field. And, when the user presses F4 in the OrderNo field, the user returns to the BadgeNo field.

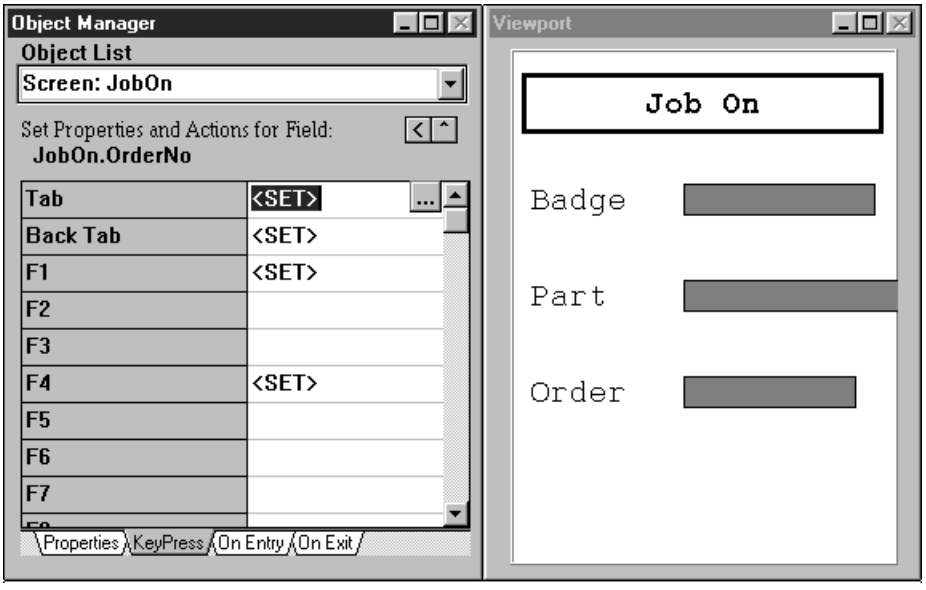
   Note: *This "hot key" approach bypasses all exit validation and field exit action. If a field exit action is desired, you must associate an Exit Field choice from the Do Standard Action drop-down list available from the Key Action Selection Editor. Your introduction to this area comes in Steps 8 through 16.*

Programming the Tab Key

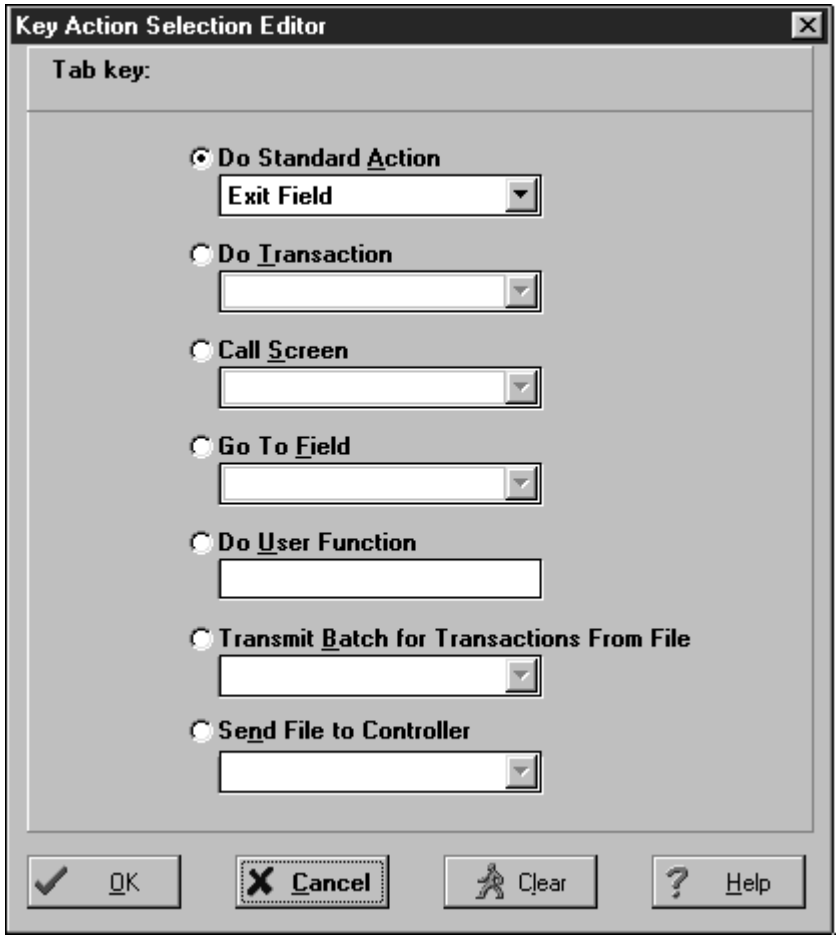
Programming the Tab and Back Tab functions is as easy as programming the function keys (F1 through F10). The Tab key on the user's keyboard exits the field (goes down to the next field) and the Shift-Tab key combination (called the Back Tab function in EZBuilder) goes back up to the previous field. By default, exiting the field has an audible "beep." You can turn the beep on or off and you can set its volume. The Beep property can be set for any key press action (not just tabbing) in the On Val Fail and On Val Succeed pages accessed through the On Exit action page.

To remove the Tab key's audible beep

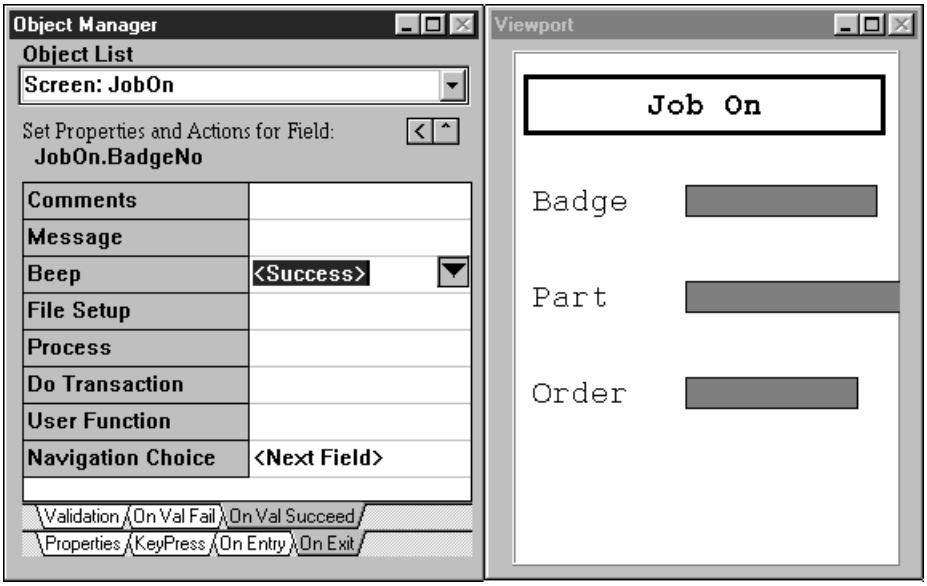
1. Click the default Tab property ("<SET>") to highlight it, as illustrated next.



2. Click the Tab property's small button with the three dots to get the Key Action Selection Editor dialog box, as illustrated next.

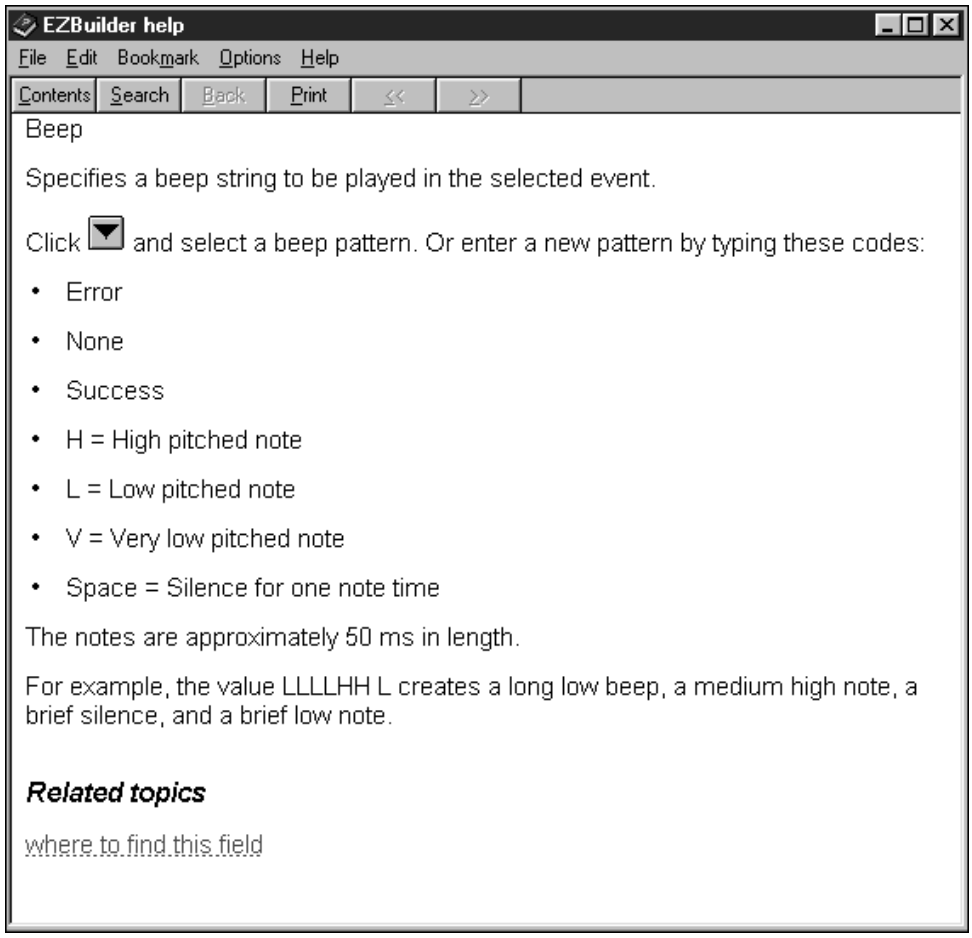


3. Notice the Do Standard Action option is set to Exit Field. Leave that as it is, and click OK to close the Key Action Selection Editor.
4. With your cursor still in the Tab property, click the On Exit tab at the bottom of the Object Manager. This brings up tabs for Validation, On Val Fail and On Val Succeed.
5. Click the On Val Succeed tab to see its properties and actions that can be set for the field, as illustrated next.

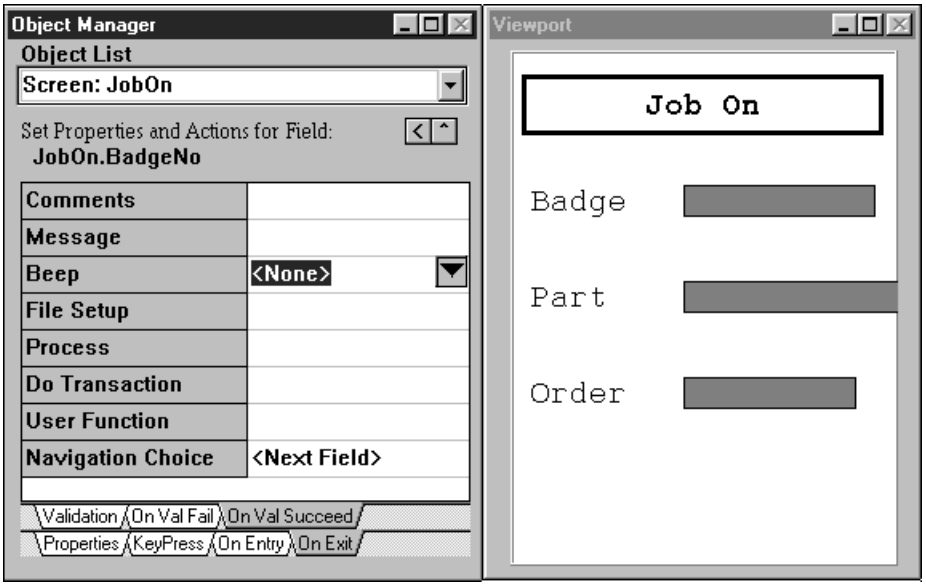


6. Move your cursor to the Beep property, currently set to "<Success>" and view its drop-down list of options, as illustrated next.

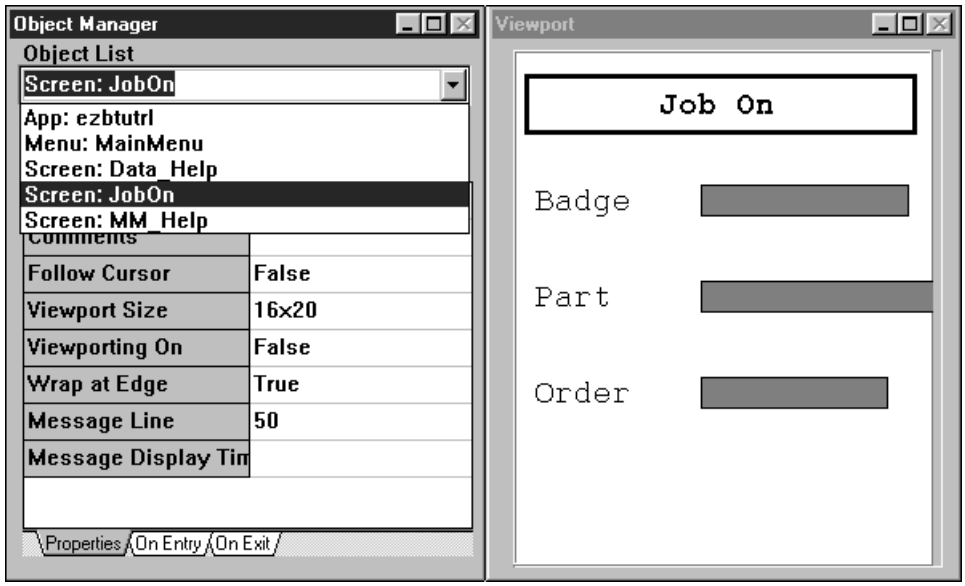
7. To learn the meaning of the last two options in the list, use the EZBuilder online help. Place your cursor anywhere in the beep property list of options and press **F1**. You will see online help relating to the beep property, as illustrated next.



8. Close the online help by clicking the upper right-hand corner “close” box marked X.
9. Choose <None> from the drop-down list of Beep property options, as illustrated next.



10. Click the Properties tab to return to the Properties page, and click the ^ button to make the JobOn parent screen active. Review the Object List to verify that you have three screens, one menu, and one application, as illustrated next.



11. You have completed Task 5. Save your file. Continue with Task 6.

Task 6: Creating the JobOff Screen

In this task, you will create a new screen for Job Off transactions. The JobOff screen will look like the JobOn screen you just created. Both screens will show the same data fields (BadgeNo, PartNum, OrderNo) with their corresponding labels. Both screens will have the property for Wrap at Edge = True. Both screens will have the F1 keys programmed to send the user to the Data_Help screen. The differences between the JobOn and JobOff screens are listed below.

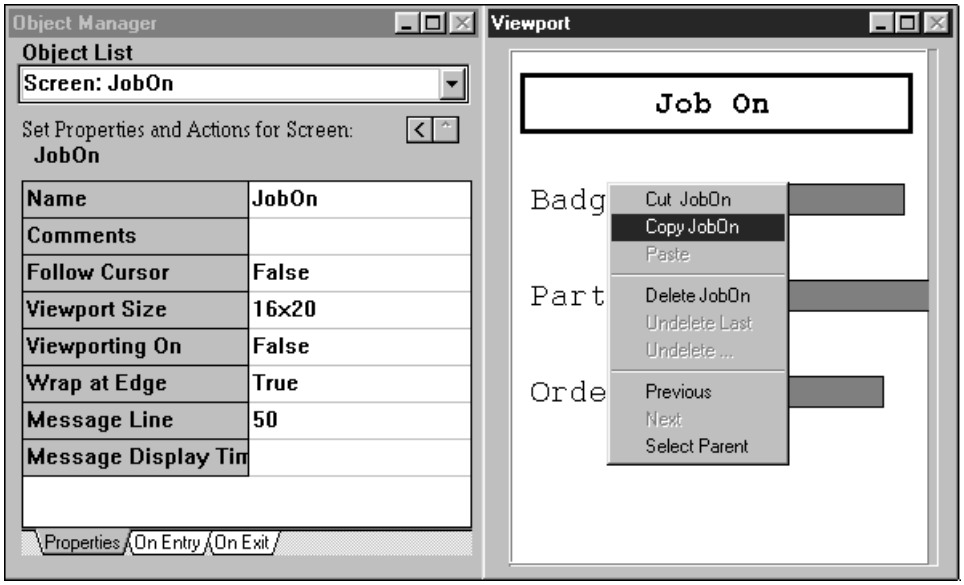
- The boxed identification label will have "Job Off" text (not "Job On" text).
- You will adjust the programmed F4 keys for these fields to send the user to the JobOff screen's fields instead of the JobOn screen's fields.

Copying the JobOn Screen

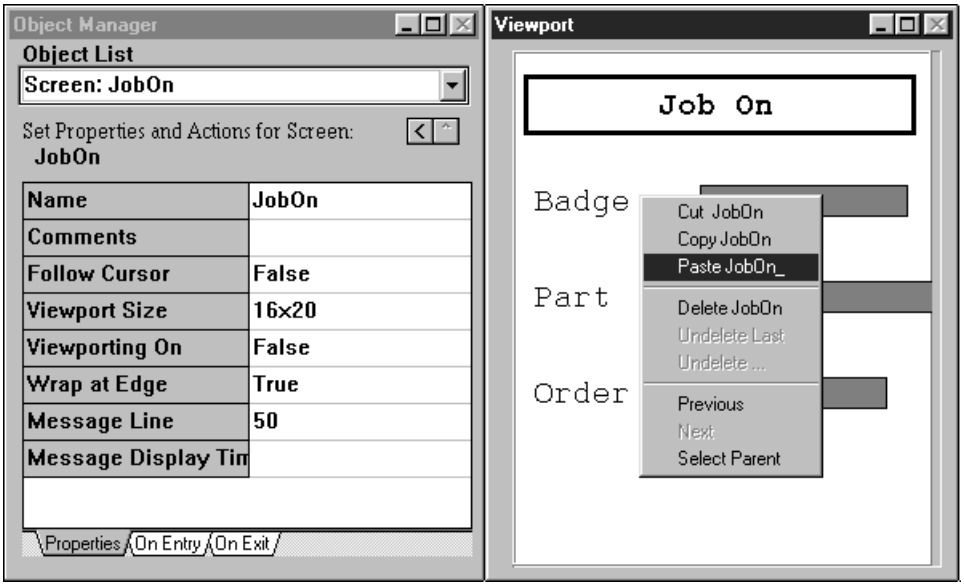
Because of their similarity, you can easily and quickly create the JobOff screen by copying the JobOn screen and simply adjusting the screen's label and the function keys.

To copy the JobOn screen

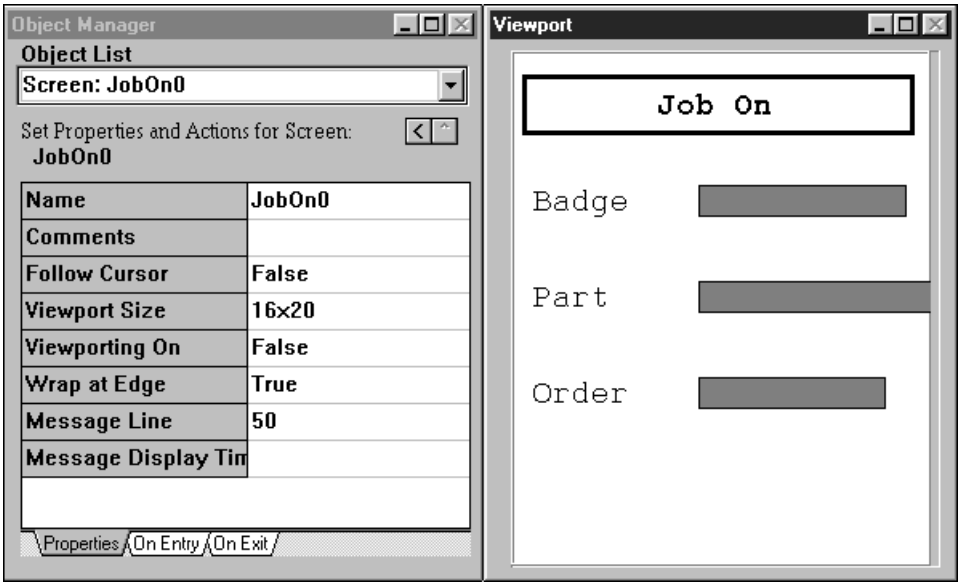
1. From the Object List, choose the JobOn screen, if it is not already available.
2. Place your cursor anywhere inside the blank part of the Viewport (not on a label or data field). Hold down your right mouse button and pull down to Copy JobOn in the drop-down list, as illustrated next. When you release your mouse button, the JobOn screen and its properties are saved in your computer's clipboard.



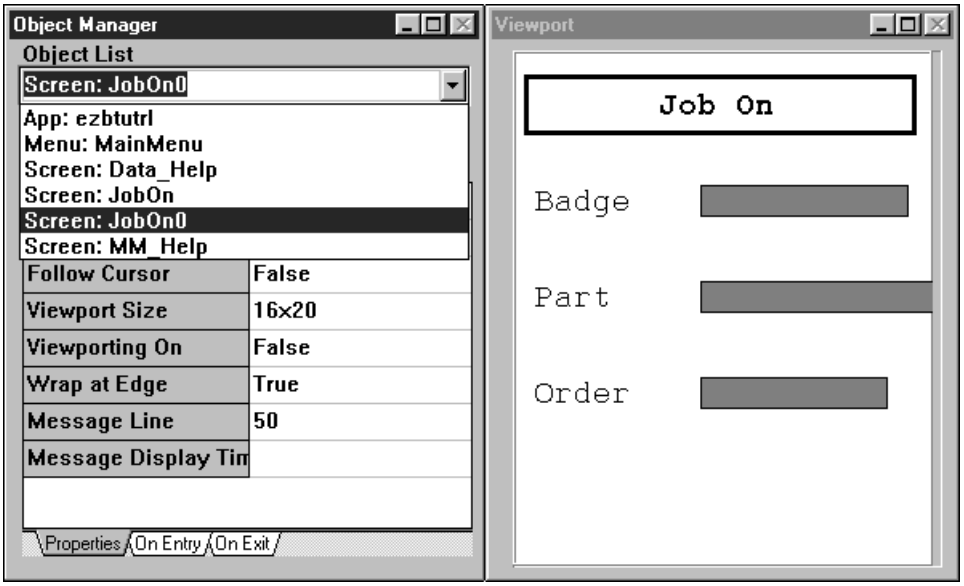
3. Place your cursor inside the Viewport again, and hold down your right mouse button again. Pull down to choose Paste JobOn, as illustrated next. The duplicate JobOn screen that was saved on your computer's clipboard is copied into your EZBuilder program and automatically named JobOn0.



4. The JobOn0 screen is now the active screen, as illustrated next.



5. Click the Object List's down arrow and see that you now have both a JobOn screen and a JobOn0 screen in your application, as illustrated next.

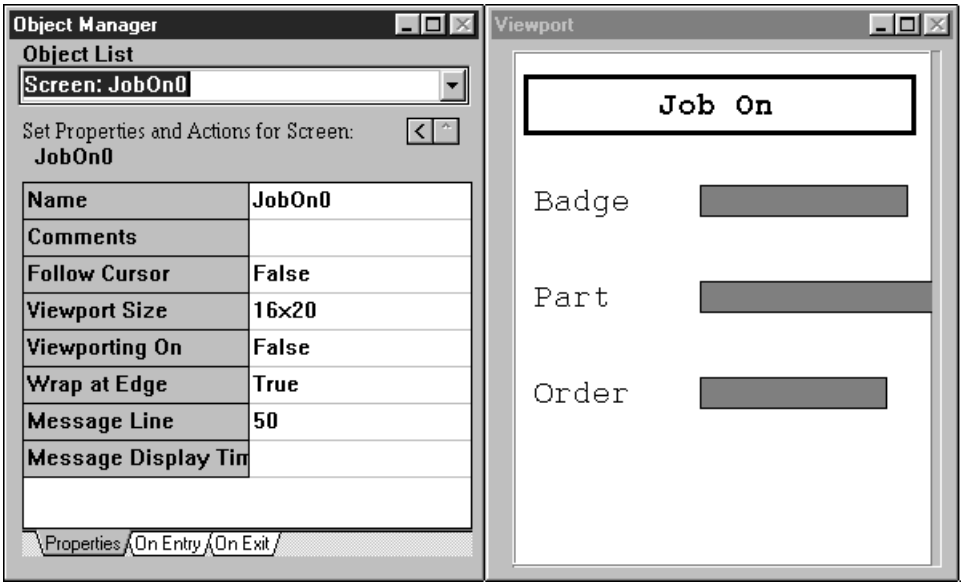


Adjusting the Duplicate JobOn0 Screen

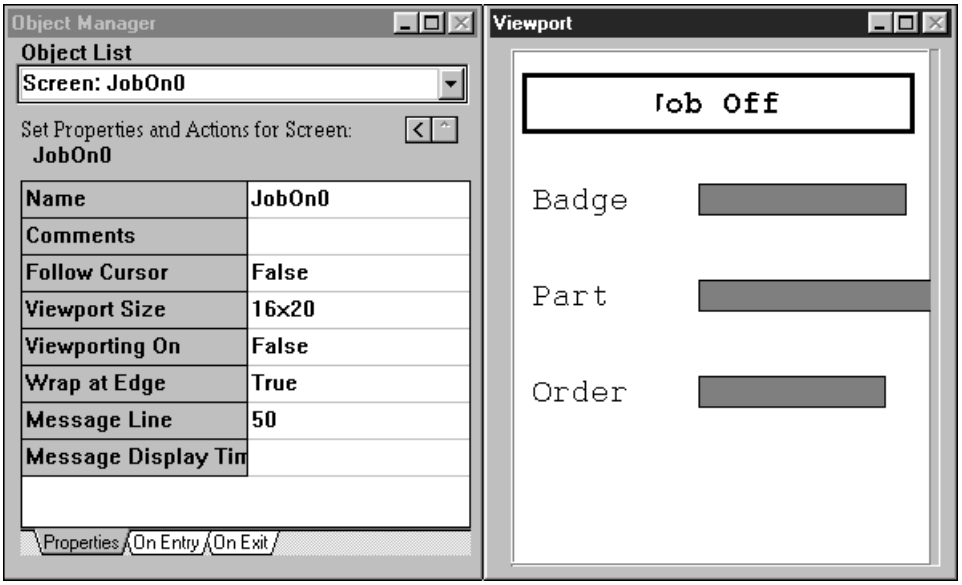
Except for their screen name, the two screens, JobOn and JobOn0, are currently exact duplicates. You will next adjust the JobOn0 screen to become the JobOff screen.

To rename the JobOn0 screen to be JobOff

1. Make sure the JobOn0 screen (not the JobOn screen) is showing on your computer monitor, as illustrated next.

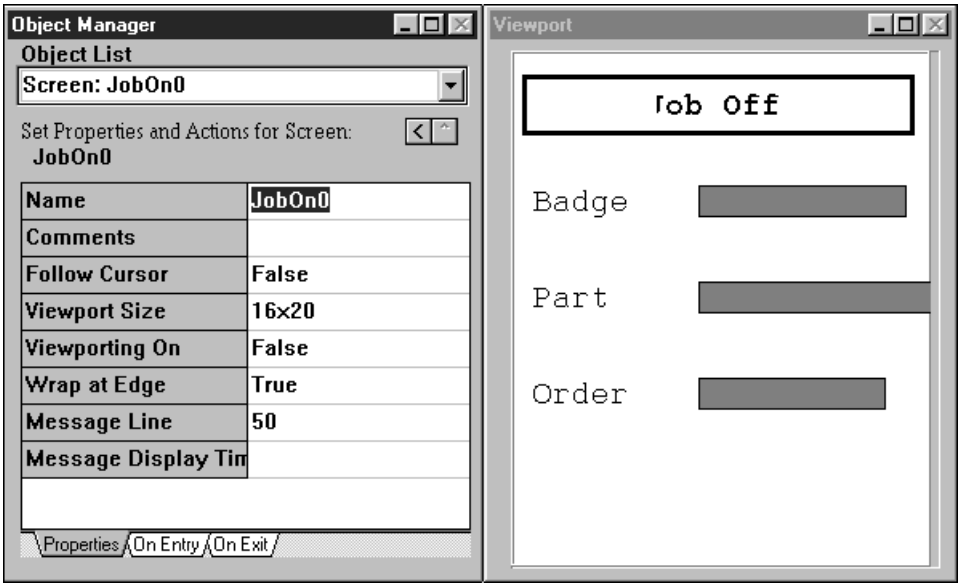


2. In the Viewport, place the mouse cursor on the word “On” in the JobOn screen’s boxed identification label. Highlight the “On” word, and type “Off” to replace “On,” as illustrated next.

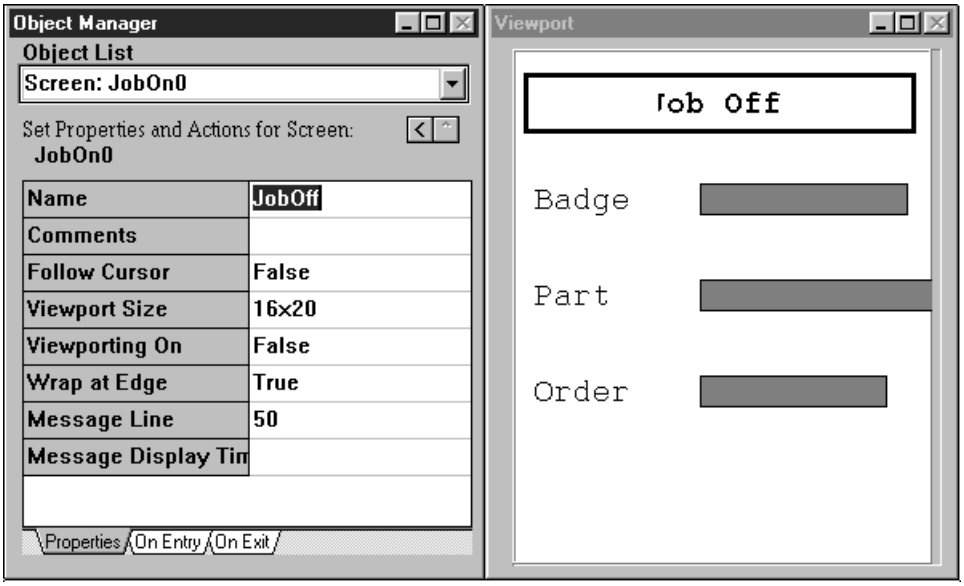


Note: *Notice that part of the “Job Off” text may not show up well. EZBuilder still thinks the Length is one character shorter. Don’t be concerned about the Length property; EZBuilder will fix it automatically later.*

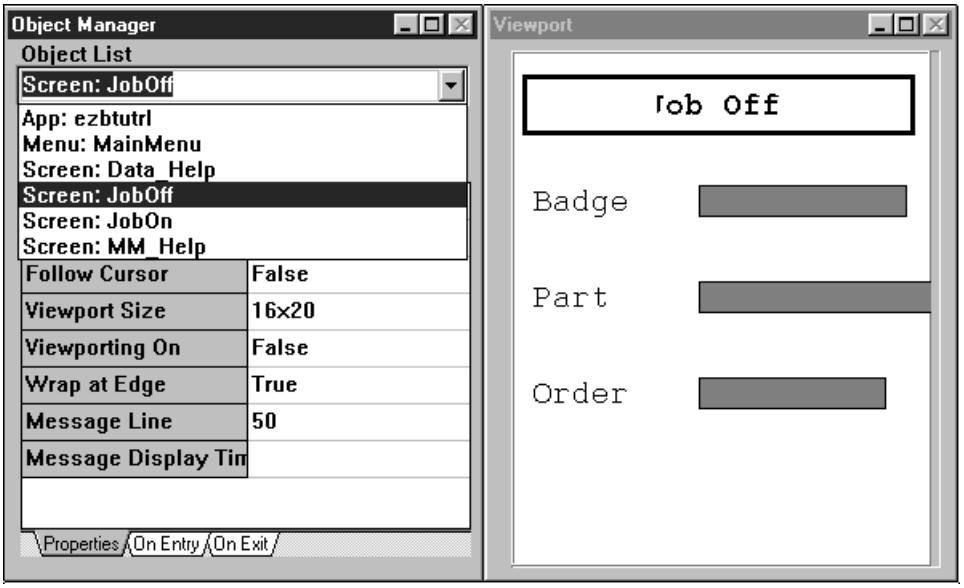
3. Highlight the Name property (JobOn0), as illustrated next.



4. Type over the highlighted Name property to change the property to "JobOff," as illustrated next.

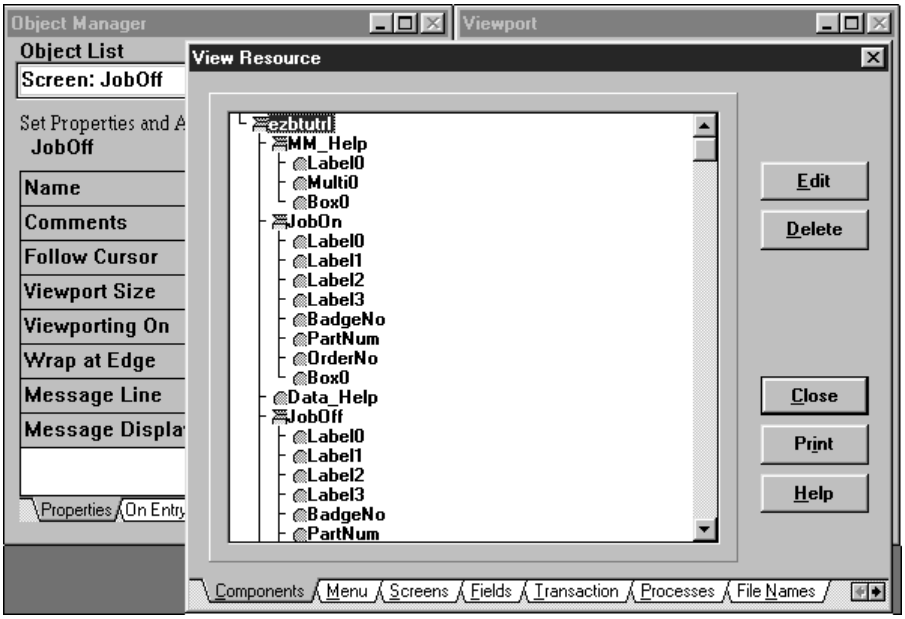


5. Click the Object List's down arrow to verify that you now have both the JobOn and JobOff screen names, as illustrated next.

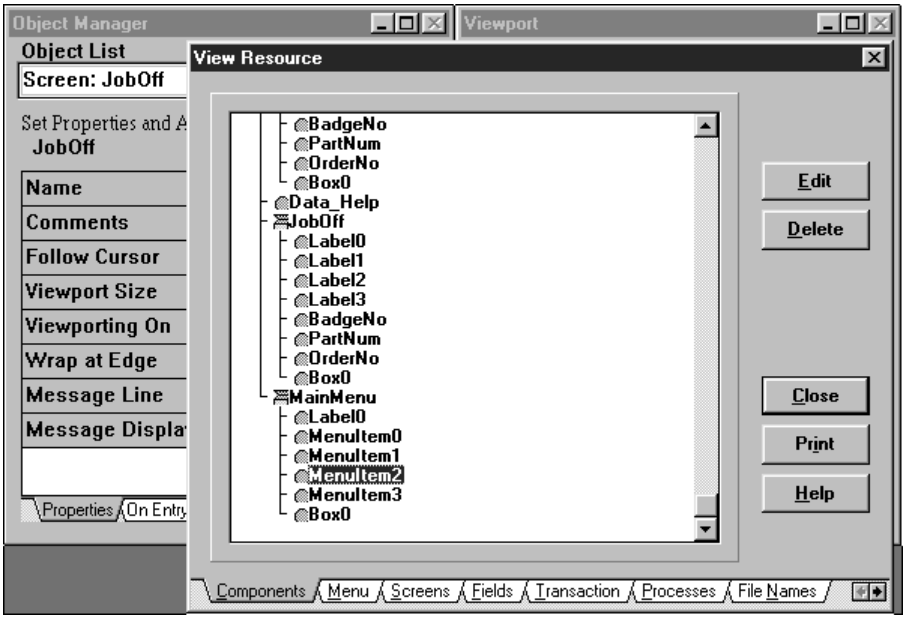


6. From the View menu, choose the Resource command to check your View Resource list. Review your JobOn and JobOff components, as illustrated

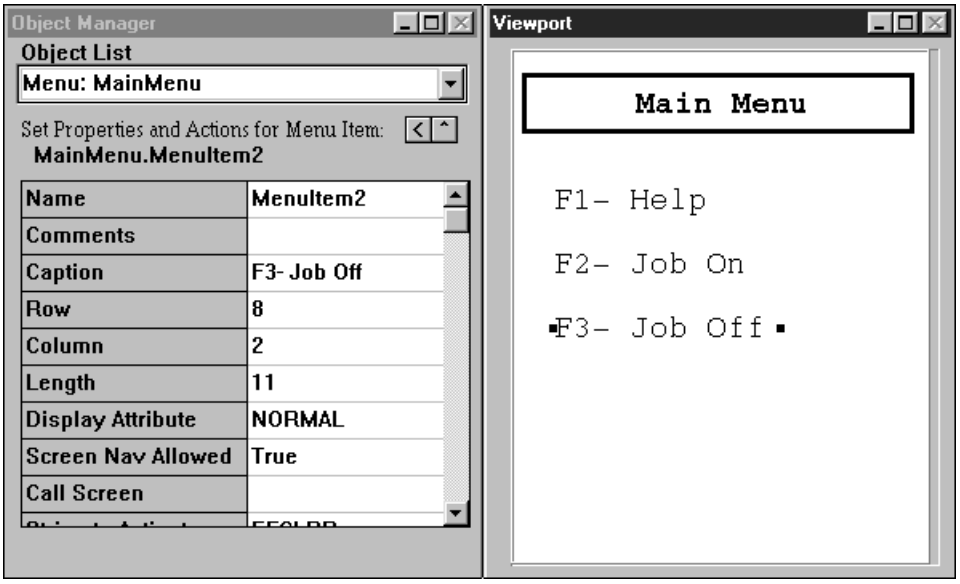
next. While there, notice that the Data_Help screen (listed above JobOff) has no components; these will be added in Task 7.



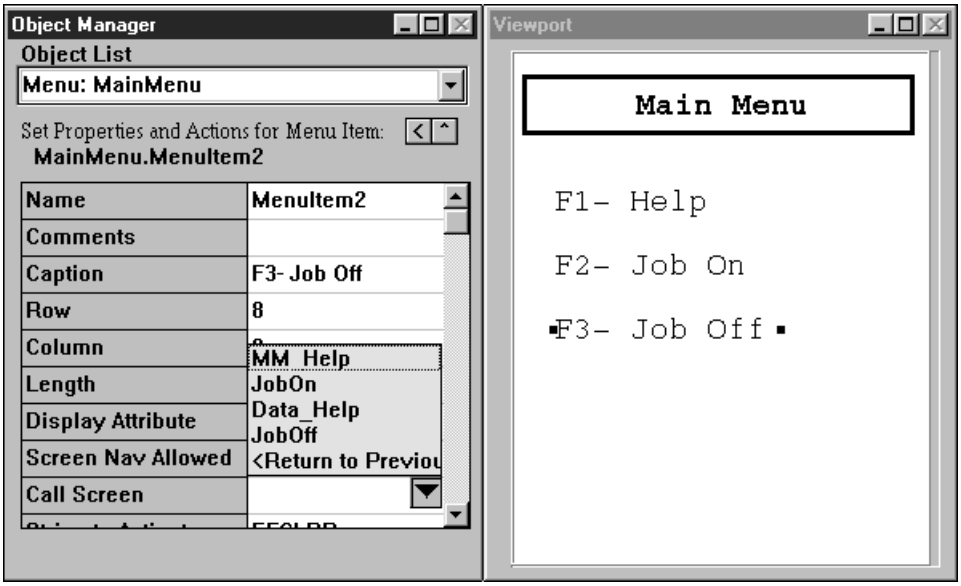
7. Scroll down in the View Resource dialog box to see the Main Menu components you have created. Highlight the MenuItem2, as illustrated next.



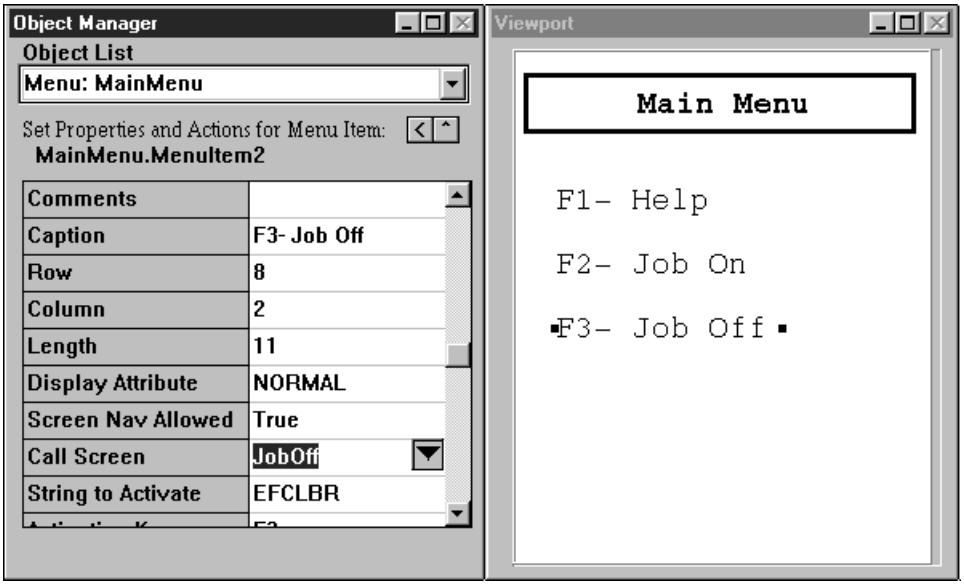
8. Click Edit in the View Resource dialog box. The View Resource dialog box closes and you are automatically taken to the Main Menu. Notice that the MenuItem2 ("F3- Job Off") that you had highlighted in the View Resource dialog box is selected, as illustrated next.



9. Click the (blank) Call Screen property, then click its down arrow. Review the choices that drop down in the list, as illustrated next. Notice that you now have the JobOff screen listed as an option.



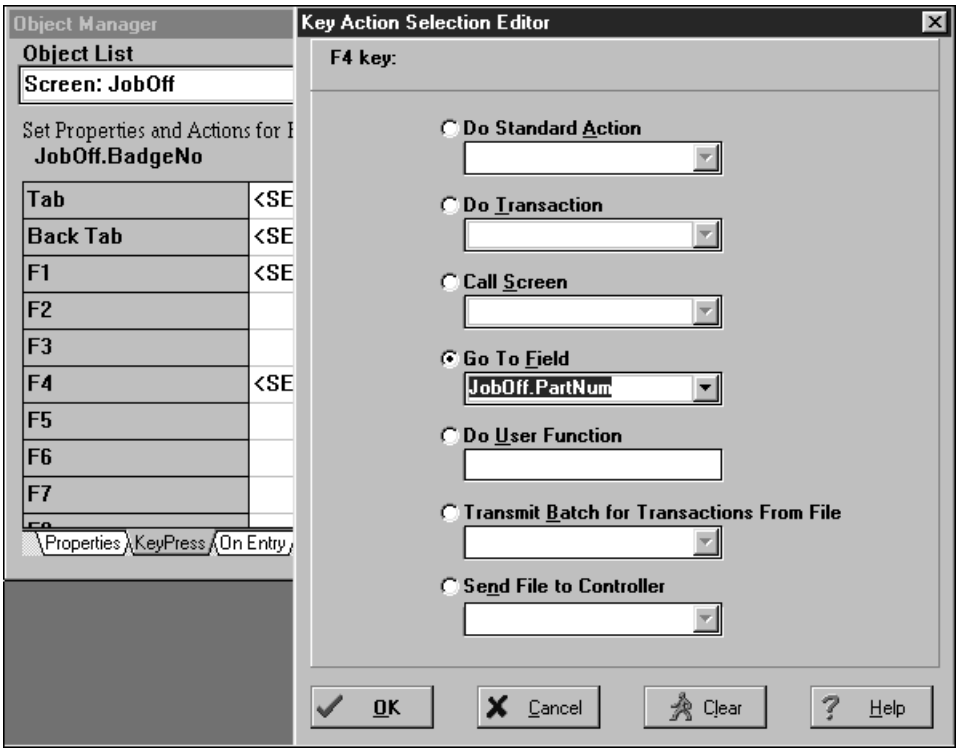
10. Choose the JobOff screen option from the list. This will be the called screen when the user presses F3 when viewing the Main Menu. Check your result with the next illustration.



11. From the Object List, choose the JobOff screen so you can re-program your F4 function keys—which cannot remain the same as the JobOn screen. Or, you can more quickly go back to the JobOff screen by pressing the small button marked with the < (which brings back the previously active object, which in this case was the JobOff screen).

    Note: *The following steps are given in a brief form because you have already done similar steps in Task 5. If you need details or illustrations, see the "Programming the Function Keys" steps in Task 5.*

12. In the Viewport, double-click to select the BadgeNo data field. Click the KeyPress tab in the Object Manager. Click the F4 key's property, then its small button marked with three dots to bring up the Key Action Selection Editor for the F4 key.
13. From the Go To Field's drop-down list, choose JobOff.PartNum to replace JobOn.PartNum, as illustrated next. Click OK.



14. On the JobOff screen, double-click in the Viewport to select the PartNum data field. Click the KeyPress tab in the Object Manager. Click the F4 key's property, then its small button marked with three dots to bring up the Key Action Selection Editor for the F4 key.
15. From the Go To Field's drop-down list, choose JobOff.OrderNo to replace JobOn.OrderNo. Click OK.
16. In the Viewport, double-click to select the OrderNo data field. Click the Key Press tab in the Object Manager. Click the F4 key's property, then its small button marked with three dots to bring up the Key Action Selection Editor for the F4 key.
17. From the Go To Field's drop-down list, choose JobOff.BadgeNo to replace JobOn.BadgeNo. Click OK.
18. You have now completed Task 6. Save your file. Continue with Task 7.

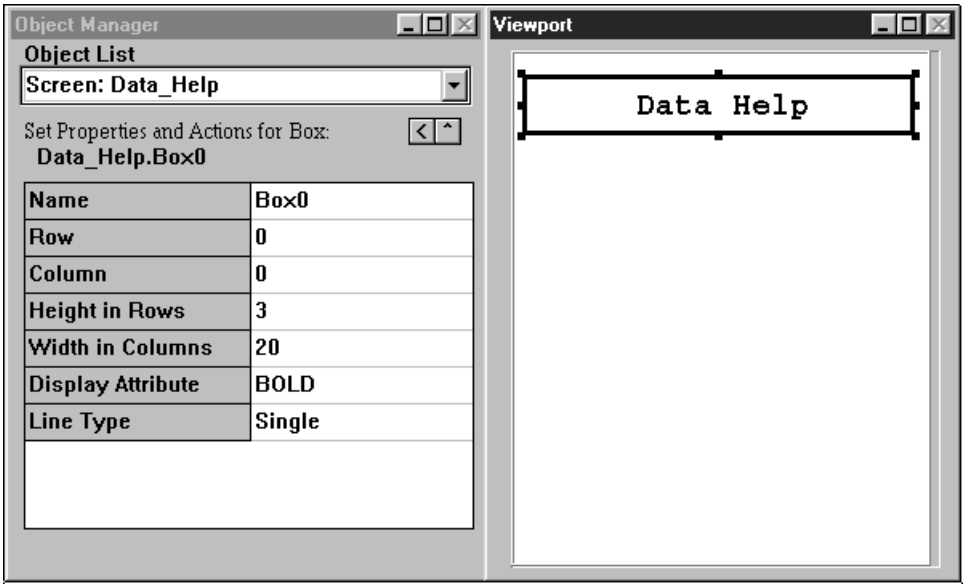
Task 7: Creating the Data_Help Screen

In this task, you will add one boxed screen identification label and one scrolling field with help text to the blank Data_Help screen you created earlier. This is done in the same manner as when you created the MM_Help screen in Task 4.

Note: *Because the following steps are similar to those you did in Task 4 for the MM_Help screen, they are more briefly described here. Refer back to Task 4 if you need more detail or illustrations.*

To complete the Data_Help screen

1. From the Object List, choose the Data_Help screen.
2. From the toolbar, click the button with the big "A" to create a new label.
3. Enter Row = 1, Column = 6 as the label position coordinates.
4. Enter "Data Help" as the Caption property.
5. Change the Display Attribute to BOLD.
6. From the toolbar, click the Box button to box the label.
7. Enter the box properties: Column = 0, Width in Columns = 20. Check the result with the next illustration.



8. From the toolbar, click the Scrolling Section button (fourth from right end).

9. Enter these properties for the Multi0 (multiple-line, scrolling section) field: Row = 4, Lines to Display = 11.
10. Click somewhere in the Viewport window, outside the scrolling section to be sure the scrolling section is not selected (no handles showing).
11. Click once inside the scrolling section to position your cursor inside it.
12. Enter the text and blank lines shown below. Do not type the double quote marks.

    "Each data field is alphanumeric data.

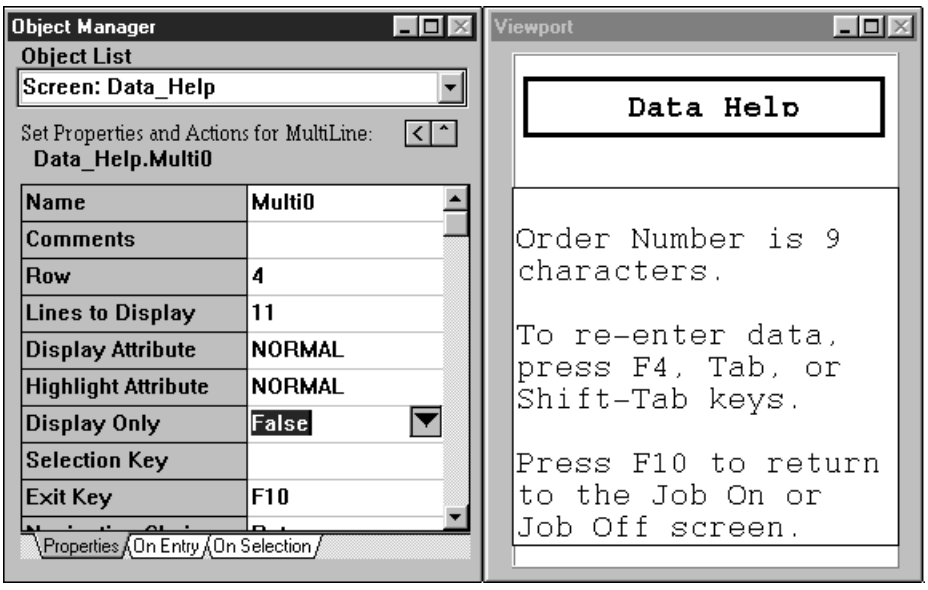
    Badge is employee's ID number, and is 10 characters.

    Part Number is 25 characters maximum.

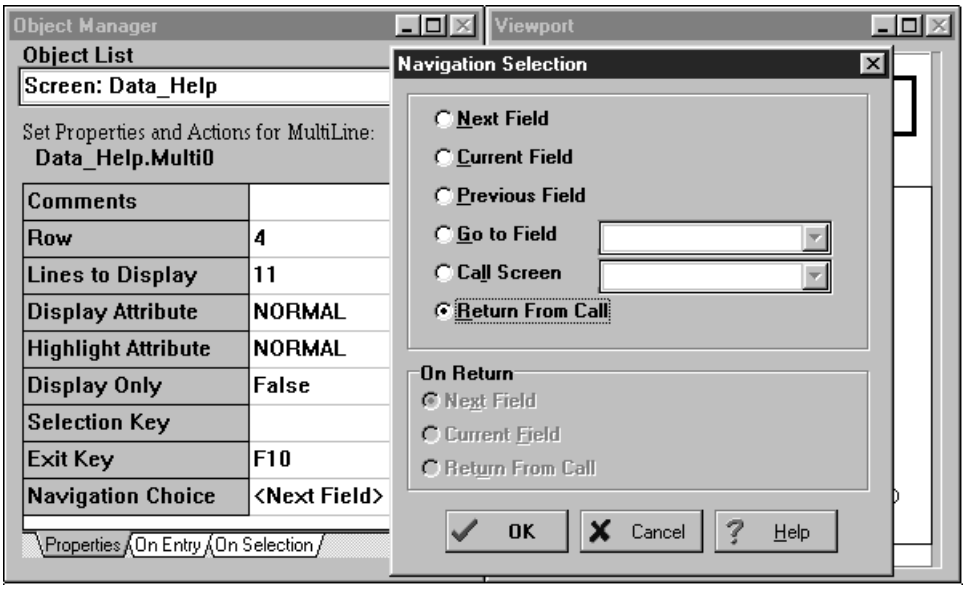
    Order Number is 9 characters.

    To re-enter data, press F4, Tab, or Shift-Tab keys.

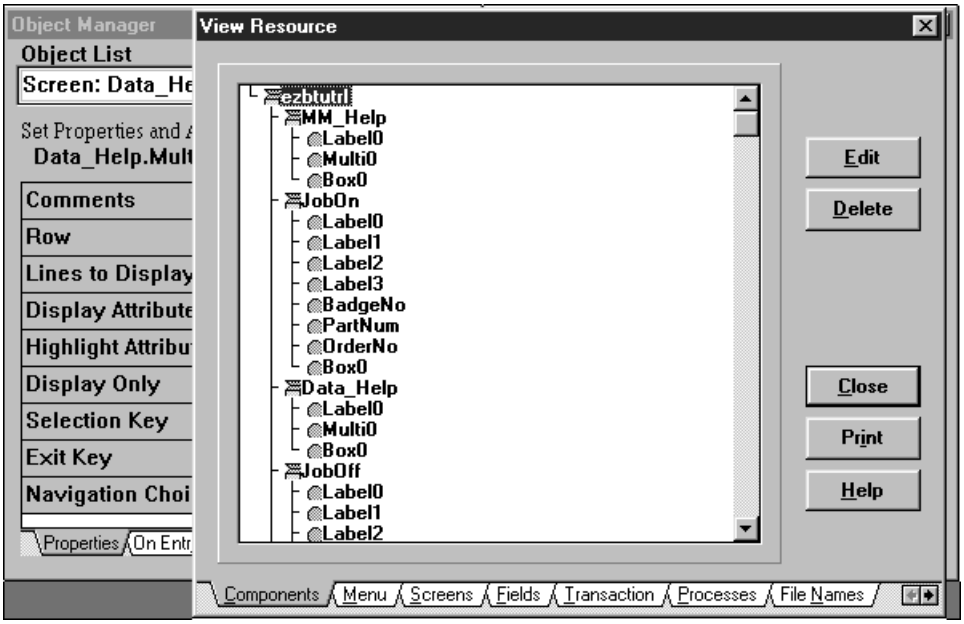
    Press F10 to return to the Job On or Job Off screen."
13. Change the Highlight Attribute from INVERSE to NORMAL.
14. Highlight and delete the Selection Key property.
15. Type "F10" as the Exit Key. Check your result with the next illustration.



16. In the Object Manager window, scroll to the Navigation Choice property and click its small button with three dots to bring up the Navigation Selection dialog box.
17. Select the Return from Call option, as illustrated next.



18. Click OK to close the Navigation Selection dialog box. Notice that “Return” becomes the Navigation Choice property.
19. From the View menu, choose the Resource command. Notice the Data_Help screen now has three components (one label, one multi-line scrolling field, one box), as illustrated next.



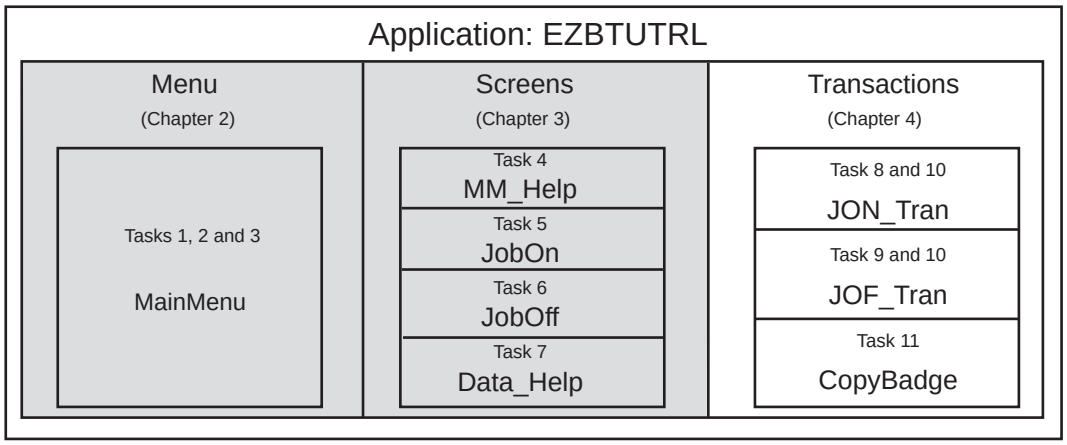
20. Click Close to exit the View Resource dialog box. You have now completed Task 7. From the File menu or toolbar, save your application file.

Chapter Summary

This chapter gave you exercises in creating screens, data entry fields, and their respective labels. You learned how to wrap data and to program function keys for various navigation needs. You learned how to add, remove, and set the volume for audible beeps made upon key press actions. Also, in this chapter, you learned how to duplicate and adjust a screen as a quick way of creating a new screen that is similar to an existing screen.

The EZBuilder components covered up to this point in the tutorial are shaded in the next illustration. To create transactions, continue with Chapter 4.

Tutorial Exercise Summary



EZB.004