LAB 3.2 (4.0 HOURS)

**Assessment Preparation Checklist:**

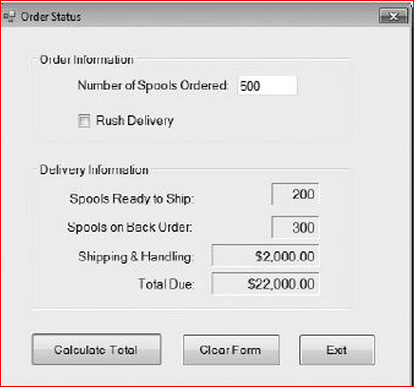
To prepare for the assessment, read Chapter 6 in the textbook, *Starting Out with Visual Basic® 2012*. This chapter examines how to write general purpose procedures and functions. In addition, go through the lesson for this module that explores how to create, call, and pass arguments to the procedures as well as various techniques for debugging applications that use them.

**Title: Build an Order Status Application**

The Middletown Wire Company sells spools of copper wiring for $100 each. The normal delivery charge is $10 per spool. Rush delivery costs $15 per spool. Create an application that displays the status of an order. The status should include the following:

* The number of spools ready to ship
* The number of spools on back order
* The shipping and handling charges
* The total amount due

The application form should resemble the one shown in the figure below:



The user should note the number of spools ordered and enter it into the text box and check the Rush Delivery check box if rush delivery is desired. When the Calculate Total button is clicked, an input box should appear asking the user to enter the number of spools currently in stock. If the user has ordered more spools than are in stock, a portion of the order is back-ordered. For example, if the user orders 200 spools and there are only 150 spools in stock, then 150 spools are ready to ship and 50 spools are back-ordered.

The application should have the following functions, called from the Click event handler of the Calculate Total button:

|  |  |
| --- | --- |
| GetInStock | Displays an input box asking the user to enter the number of spools in stock. The function should return the value entered by the user. |
| ReadyToShip | Accepts the following arguments: the number of spools in stock and the number of spools ordered. The function returns the number of spools ready to ship. |
| BackOrdered | Accepts the following arguments: the number of spools in stock and the number of spools ordered. The function returns the number of spools on back order. If no spools are on back order, it returns 0. |
| ShippingCharges | Accepts the following arguments: the number of spools ready to ship and the per-spool shipping charges. The function returns the total shipping and handling charges. |

The application should have the following procedures, called from the Click event handler of the Clear Form button:

|  |  |
| --- | --- |
| ResetSpools | Clears the text box and the check box. |
| ResetDelivery | Clears the labels that display the delivery information. |

Input Validation: Do not accept orders for less than one spool.

**Required Setup and Resources:**

* VMware Player 5.x
* ITT-Lab VM

**Recommended Procedure:**

**Note:** For the steps that require you to paste the screenshot or answer a question, document your response in a Microsoft Word document titled “SD3320\_Module3\_Lab2.docx.” Make sure to assign the corresponding task number against each response or screenshot. In addition, document the source code wherever applicable.

**Task 1: Design the Application**

1. Design the program as described.
2. Make a list of the controls you will access, using code. Include the Name property.
3. Make a list of the event procedures you will code.

**Task 2: Implement the Order Status Application**

1. Launch **Visual Studio 2012**.
2. Create a new Visual Basic Windows Application project name **Module3\_Lab2\_OrderStatus**.
3. Add the necessary controls to the form.
4. Set the Name property of each control.
5. Set any other form and control properties, as required by the design.
6. Write the code to implement data validation. Make sure to document your code using comments.
7. Test your code with valid and invalid values.
8. Write the code to implement the Exit button.
9. Test your code.
10. Write the code to implement each of the sub procedures called by the event procedure of the Clear Form button.
11. Write the code to implement the Clear Form button.
12. Test the Clear Form button. Use debugging tools to locate and correct any problems in your code.
13. Write the code to implement the GetInStock function.
14. Add the code to the Click event procedure of the Calculate Total button to call the GetInStock function.

***Question 1:*** *What data type did you use for the return value?*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Test your code.
2. Write the code to implement the ReadyToShip function.

***Question 2:*** *Did you use* ByVal or ByRef *arguments?*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Modify the Click event procedure of the Calculate Total button so that it displays the number of spools that are ready to ship.
2. Test your code.
3. Write the code to implement the BackOrdered function.

***Question 3:*** *How many parameters does the function require?*

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Modify the Click event procedure of the Calculate Total button so that it displays the number of spools back ordered.
2. Test your code.
3. Write the code to implement the ShippingCharges function.
4. Finish implementing the Click event procedure of the Calculate Total button.
5. Test your code with the values shown in the table:

|  |  |  |  |
| --- | --- | --- | --- |
| **Number in Stock** | **Number Ordered** | **Rush Delivery** | **Expected Results** |
|  | 0 |  | MessageBox is displayed. |
|  | “1 spool” |  | MessageBox is displayed. |
| 6 | 4 | No | Spools ready: 4  Spools on back order: 0  Shipping: 40  Total: 440 |
| 6 | 4 | Yes | Spools ready: 4  Spools on back order: 0  Shipping: 60  Total: 460 |
| 2 | 7 | No | Spools ready: 2  Spools on back order: 5  Shipping: 20  Total: 220 |
| 2 | 7 | Yes | Spools ready: 2  Spools on back order: 5  Shipping: 30  Total: 230 |

1. Turn in your written answers to the lab questions.

**Submission Requirements:**

Submit the Word document titled “SD3320\_Module3\_Lab2.docx” to your instructor for grading. Make sure to include a screenshot that shows that the application developed is working correctly. In addition, make sure to submit the project titled “Module3\_Lab2\_OrderStatus.” The completed project file package should include all the sources files used to develop the application. The Word document should have the following specifications:

* Font: Arial; font size: 12; double-spaced
* Length: 1 page

**Evaluation Criteria:**

Your submission will be evaluated against the following points:

* Did you test the code against given values?
* Did the test data generate the correct results?
* Does Clear Form work correctly?
* Does Exit close the application?
* Did you modify the Click event procedure of the Calculate Total button?