LAB 2.2 (4.0 HOURS)

**Assessment Preparation Checklist:**

To prepare for the assessment, read Chapter 3 in the textbook, *Starting Out with Visual Basic® 2012*. This chapter covers the concept of variables, data types, calculation, and exception handling. In addition, go through the lesson for this module that explores the use of text boxes to gather input from users.

**Title: Create a Payroll Calculator**

In this lab, you will create a use case for payroll calculator application and then design and create a program to calculate an employee’s pay for a week, using various operators.

You need to design and create a program that accepts input of employee name, employee type, and the number of hours worked each week for two weeks.

You must be able to select an employee type. An employee of type Trainee makes $10 per hour. An employee of type Regular makes $15 per hour. An employee of type Manager makes $20 per hour.

The program should calculate the employee’s pay for the week. The employee should receive 1.5 times the regular pay for any hours worked over the prescribed 40 hours in a week. The program should output the following information:

* The employee’s name, without trailing or leading spaces
* Regular hours worked
* Regular pay
* Overtime hours worked
* Overtime pay
* Total pay

A message should be displayed with an Error icon if the value entered is not numeric. If an input field contains a value greater than 60 or less than 0, display a message with an Error icon.

The application should use a class-level variable to store the employee type.

**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\_Module2\_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. Use Microsoft PowerPoint to sketch the user interface for the application.
2. Copy the sketch and paste it into a Word document.

Payroll Calculator

Robbie Rob

43

Regular

Trainee

Calculate Pay

Employee Name

Week 1

Total Pay for 2 Weeks -

**$1,425.00**

Manager

47

Week 2

**Hours**

1. Create a table that lists the type of control and the name. Make sure to use standard naming conventions and use names that adhere to Visual Basic naming rules.

***Question 1:*** *Which control will you use to allow users to select the employee type?*

***Answer 1:*** Radio Buttons

1. Add a column to the table and list the default Text property for each control.
2. Implement the event procedures shown in the table.

|  |  |
| --- | --- |
| **Event Procedure** | **Description** |
| CheckChanged event for the Trainee RadioButton | If the button is checked, sets strEmployeeType to “Trainee.” |
| CheckChanged event for the Regular RadioButton | If the button is checked, sets strEmployeeType to “Regular.” |
| CheckChanged event for the Manager RadioButton | If the button is checked, sets strEmployeeType to “Manager.” |
| Click event for the Calculate Pay button | Calculates the required values and displays the output in a Label control. |

1. List each variable that the event procedure will use. Define the name, data type, scope, and initial value.
2. List any constants you will use in the event procedure. Define the name, scope, data type, and value.
3. Use Microsoft Visio to draw a flowchart for each event procedure. Save the flowchart as a graphic file and import it into your Word document.
4. Save your Word document.

**Task 2: Create and test the application you designed**

1. Start **Visual Studio 2012**.
2. Create a new Windows Application project named **Module2\_Lab2.2\_Calculator**.
3. Set the title of the form to “**Payroll Calculator**.”
4. Add the controls you identified in Task 1 to the form. Set their positions to match your sketch. Lock the controls when you are satisfied with your design.
5. Set the Name and Text properties for each control to match your design.
6. Set the tab order.
7. Add code to set Option **Explicit On**.
8. Declare any constants as class-level constants.

***Question 2:*** *Why is it safe to declare the constants as class-level constants?*

***Answer 2:*** As a constant it will not be changed and if declared as class-level will be able to be accessed anywhere in the class.

1. Declare a variable to store the employee type as a class-level variable.

***Question 3:*** *Why do you need to declare this variable as a class-level variable?*

***Answer 3:*** The value needs to be changed and/or accessed in more than one function.

1. Write an event procedure to calculate and display the output. Make sure to add the necessary input validation. Use comments to document your code.

***Question 4:*** *Did you use nested decision statements?*

***Answer 4:*** No.

***Question 5:*** *Did you use If...Then...Else statements?*

***Answer 5:*** Only If…Then… statements, no …Else statements. Used a Select Case statement.

1. Test the application and correct any values.
2. Save the project when you are finished.
3. Turn in your written answers to the lab questions.

**Submission Requirements:**

Submit the Word document titled “SD3320\_Module2\_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 “Module2\_Lab2\_Calculator.” 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 performance will be evaluated against the following points:

* Does your application display a dialog box with an error icon for non-numeric input?
* Does your application display a dialog box with an error icon for input values less than 0 or greater than 60?
* Does the program display the correct result for each type of employee?

EXERCISE 2.1 (2.0 HOURS)

**Assessment Preparation Checklist:**

To prepare for the assessment, read Chapters 4 and 5 in the textbook, *Starting Out with Visual Basic® 2012*. These chapters cover the concept of decision-making and loops. In addition, go through the lesson of this module that explores the use of different statements for decision-making and repetition.

**Title: Decisions and Repetition**

**Part 1:** Answer the following questions:

1. For each of the following Visual Basic code snippets, identify the syntax error.
2. Dim str As String = "Hello"

Dim intLength As Integer

intLength = Length(str)

**Syntax Error: Length(str) will throw a syntax error.**

1. If intZ < 10 Then

lblResult.Text = "Invalid Data"

**Syntax Error: No “End If”**

1. Dim str As String = "123"

If str.IsNumeric Then

lblResult.Text = "It is a number."

End If

**Syntax Error: str.IsNumeric will throw a syntax error.**

1. Describe the difference between the If...Then...ElseIf statement and a series of If...Then statements.

If If…Then…ElseIf Then

1. In an If...Then...ElseIf statement, what is the purpose of a trailing Else?
2. What is a flag and how does it work?
3. Briefly describe how the AND operator works.
4. Briefly describe how the OR operator works.
5. How is the XOR operator different from the OR operator?

**Part 2:** Answer the following questions:

1. Why is it important to assign an initial value to a loop counter variable?
2. Why should you be careful not to place a statement in the body of a For... Next loop that changes the value of the counter variable of the loop?
3. You need to write a loop that iterates until the user enters a specific value into an input box. Which type of loop should you choose? Why?
4. You need to write a loop that will repeat 224 times. Which type of loop will you choose? Why?
5. You need to write a loop that iterates as long as a variable has a specific value stored in it. Which type of loop will you choose? Why?
6. Why is a computer system time a good source of random seed values?
7. You use the statement lstNames.Items.RemoveAt(6) to remove an item from a list box. Does the statement remove the sixth or seventh item in the list? Why?
8. What kind of control(s) do you use when you want to provide the user a list of items to select from, but do not want to limit the user input to the items on the list?
9. What kind of control(s) do you use when you want to limit the user selection to a list of items?

**Submission Requirements:**

Submit your answers in a Microsoft Word document.

**Evaluation Criteria:**

The [exercise rubric](#h.30j0zll) will be used to evaluate the assessment.