

Stack Simulator

Problem Description: The stacking of coins is a common activity for people as they count money. You are required to write a program to simulate the counting of money by using an array and a stack data structure. Your program should allow the user to remove (pop) values off the top of the stack or add (push) values on to the stack.

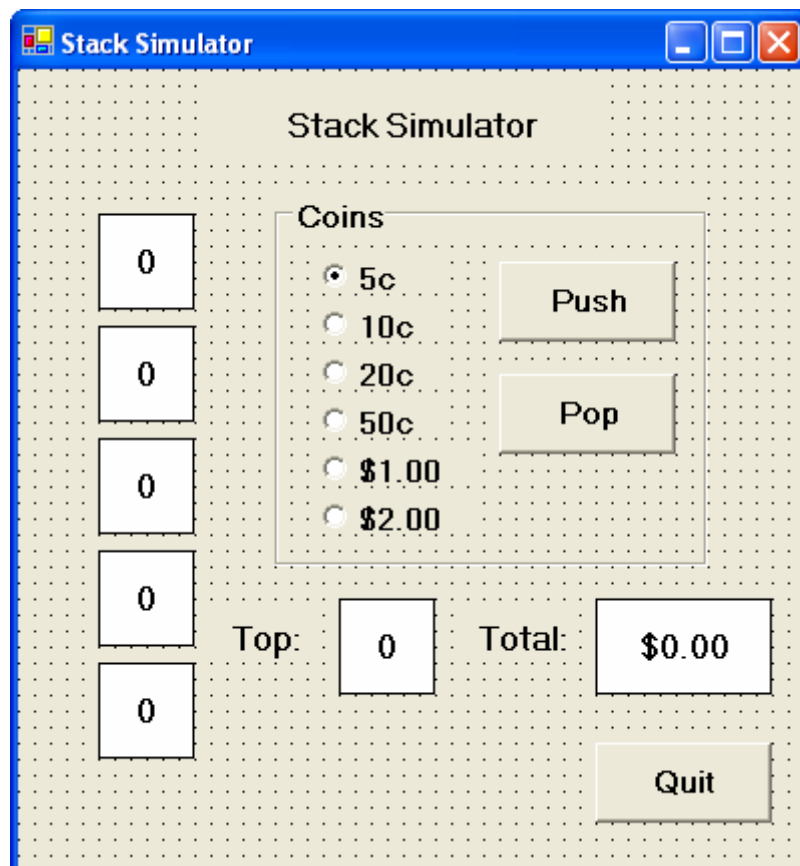
Notes:

1. The program must provide for the following coins: 5c, 10c, 20c, 50c, \$1, \$2.
2. The stack must be limited to 5 items (coins).
3. The processing of the coins must use a stack process where the 'last on' is the 'first out' (LIFO).
4. A total of the coins as well as the value of the 'top' pointer for the stack must be displayed after each transaction.

Programming Level: Intermediate

Skills Covered: Arrays, radio buttons, group boxes.

A screen shot of the stack simulator is shown below:



Setting up the objects for our program

Follow these steps to create the program:

1. Start a new Visual Basic.NET program called 'Stack Simulator'.
2. Select the Label tool from the toolbox. Use it to draw ten rectangular boxes on the form as shown. Five labels will be used to form the stack elements on the left hand side of the form. Name these labels from the top to bottom as 'lblStack5' down to 'lblStack1'. Three labels will be used as headings. Name the label at the top of the form 'lblTitle'. The two other heading labels will be used to display the words 'Top' and 'Total'. Name these 'lblHeading1' and 'lblHeading2'. Lastly the two labels next to these headings should be named 'lblTop' and 'lblTotal'.
3. Use the GroupBox tool to a single group box on the form. Name it 'grpCoinSelect'.
4. Use the RadioButton tool to draw six radio buttons inside the group box to represent each of the valid coins that can be added to the stack. Name each from top to bottom like this: 'rad5c', 'rad10c', 'rad20c', 'rad50c', 'rad100c' and 'rad200c'.
5. Use the Button tool to create three buttons on the form. Two of these buttons should be inside the group box. The last button should be in the bottom right hand corner of the form. Name the buttons inside the group box 'btnPush' and 'btnPop'. Name the button in the bottom right hand corner 'btnQuit'.

Changing the properties of the objects

Click on each object in turn and set the following properties. Note that properties such as font and colour have not been listed here. Customise these properties as you wish to make the form look well presented.

Form1

StartPosition: CenterScreen

Text: Stack Simulator

lblTitle

Text: Stack Simulator

lblHeading1

Text: Top:

lblHeading2

Text: Total:

The properties for the labels that will display the stack elements are listed below. As the topmost one is 'lblStack5', the labels have been listed in descending order.

lblStack5**Text:** 0**BorderStyle:** FixedSingle**BackColor:** White**lblStack4****Text:** 0**BorderStyle:** FixedSingle**BackColor:** White**lblStack3****Text:** 0**BorderStyle:** FixedSingle**BackColor:** White**lblStack2****Text:** 0**BorderStyle:** FixedSingle**BackColor:** White**lblStack1****Text:** 0**BorderStyle:** FixedSingle**BackColor:** White**lblTop****Text:** 0**BorderStyle:** FixedSingle**BackColor:** White**lblTotal****Text:** \$0.00**BorderStyle:** FixedSingle**BackColor:** White

Adding code to our program

1. Double click on the form and click in the declarations section (underneath the Inherits command). Type in the following variable declarations:

```
Dim Coin_Stack(5) As Decimal
Dim Stack_Top As Integer = 0
Dim Stack_Total As Decimal
Dim Stack_Loop As Integer
Dim Current_Value As Decimal = 0.05
```

The first line sets up the stack array that will be used to store all of the coin values. The 'Stack_Top' variable will keep track of the top of the stack and is set to an initial value of '0' to indicate that the stack is empty. The 'Stack_Total' variable will be used to store the total of all of the coin values in the stack. The 'Stack_Loop' variable will be used to loop through the stack and the 'Current_Value' variable will be used to store the currently selected coin before it is placed on the stack.

2. Double click on the form and type the following code into the form1 load event:

```
Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As _
System.EventArgs) Handles MyBase.Load

    For Stack_Loop = 1 To 5
        Coin_Stack(Stack_Loop) = 0
    Next

End Sub
```

3. When each of the radio buttons are clicked, we will place their value into the 'Current_Value' variable. Type in the following code into the click event of each of the radio buttons. Note that when double clicking on the radio buttons from design mode, the default event that is opened is 'CheckChanged' and not 'Click'. You will need to select the click event from the event name drop down box.

```
Private Sub rad5c_Click(ByVal sender As Object, ByVal e As _
System.EventArgs) Handles rad5c.Click

    Current_Value = 0.05

End Sub
```

```
Private Sub rad10c_Click(ByVal sender As Object, ByVal e As _
System.EventArgs) Handles rad10c.Click

    Current_Value = 0.1

End Sub
```

```
Private Sub rad20c_Click(ByVal sender As Object, ByVal e As _
System.EventArgs) Handles rad20c.Click

    Current_Value = 0.2

End Sub
```

```
Private Sub rad50c_Click(ByVal sender As Object, ByVal e As _
System.EventArgs) Handles rad50c.Click
```

```
    Current_Value = 0.5
```

```
End Sub
```

```
Private Sub rad100c_Click(ByVal sender As Object, ByVal e As _
System.EventArgs) Handles rad100c.Click
```

```
    Current_Value = 1
```

```
End Sub
```

```
Private Sub rad200c_Click(ByVal sender As Object, ByVal e As _
System.EventArgs) Handles rad200c.Click
```

```
    Current_Value = 2
```

```
End Sub
```

4. Type in the following general procedure that will be called to copy the stack values into the labels, display the top pointer and the total of the coins.

```
Private Sub CopyStack()
```

```
    lblStack1.Text = Coin_Stack(1)
```

```
    lblStack2.Text = Coin_Stack(2)
```

```
    lblStack3.Text = Coin_Stack(3)
```

```
    lblStack4.Text = Coin_Stack(4)
```

```
    lblStack5.Text = Coin_Stack(5)
```

```
    lblTop.Text = Stack_Top
```

```
    Stack_Total = 0
```

```
    For Stack_Loop = 1 To 5
```

```
        Stack_Total = Stack_Total + Coin_Stack(Stack_Loop)
```

```
    Next
```

```
    lblTotal.Text = Format(Stack_Total, "currency")
```

```
End Sub
```

5. Double click on each of the 'Push' and 'Pop' buttons and enter the following code:

```
Private Sub btnPush_Click(ByVal sender As System.Object, ByVal e As _
System.EventArgs) Handles btnPush.Click
```

```
    If Stack_Top < 5 Then
```

```
        Stack_Top = Stack_Top + 1
```

```
        Coin_Stack(Stack_Top) = Current_Value
```

```
        CopyStack()
```

```
    Else
```

```
        MsgBox("The stack is full. Pop an item before pushing _
the next item.", MsgBoxStyle.Critical, "Stack Error")
```

```
    End If
```

```
End Sub
```

```

Private Sub btnPop_Click(ByVal sender As System.Object, ByVal e As _
System.EventArgs) Handles btnPop.Click

    If Stack_Top > 0 Then
        Coin_Stack(Stack_Top) = 0
        Stack_Top = Stack_Top - 1
        CopyStack()
    Else
        MsgBox("The stack is empty. Push an item before popping _
the next item.", MsgBoxStyle.Critical, "Stack Error")
    End If

End Sub

```

6. Double click on the 'Quit' button and enter the following code to end the program:

```

Private Sub btnQuit_Click(ByVal sender As System.Object, ByVal e As _
System.EventArgs) Handles btnQuit.Click

    End

End Sub

```

Testing the program

Run and test the program and ensure that it behaves as expected. Test that you can add each of the coins and that you cannot add more than 5. Test to see that if the stack is empty, a coin cannot be removed. Try different combinations of the coins and ensure that the total and the top pointer displays work correctly.

Further ideas to develop

- Make the coin stack larger.
- Investigate the use of the 'Stack' data structure. Some of the lines of code that you will need are shown below:

```
Dim Coins As New Stack
```

This line can be used to declare the stack.

```
Coins.Push("0.05")
Value = Coins.Pop
```

These lines can be used to 'push' values on to the stack and 'pop' them off the stack, respectively.