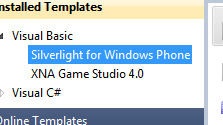
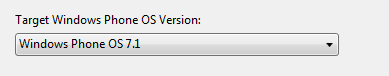
**Mr. Moon’s Programming Symposium!**

Phase I: Design the Form

* **Create a Windows Phone 7 SDK *Silverlight for Windows* Phone Application** Open Visual Studio for Windows Phone using the Start button on the Windows taskbar and the All Programs submenu. Create a new Silverlight for Windows Phone Application project by completing the following: Click the New Project button on the Standard toolbar; select and expand Visual Basic in the left pane under Project types; select Silverlight for Windows Phone; select Windows Phone Application Project in the right (Templates) pane; name the project MediaPlayer in the Name text box; then click the OK button in the New Project dialog box *(ref: Figure 5-6).*

Hint

Hint



* **Choose Target Platform** Select Windows Phone OS 7.1, and then click the OK button.
* **Change the Title on the Title Bar** To change the title on the title bar, double click MainPage.xaml, go to the form, click on my application on the phone screen, Go to the properties, and under **Title** type *MediaPlayer* then press the ENTER key.

Hint

* **Change the name of the text block (Text block)** Drag the first text block onto the MainPage.xaml Form object and name the text block lblLinearFeet.. Set the Text property for the Text Block object to *Linear Feet?*. Set the font to Segoe WP, Size 25. Position the text block to resemble Figure 5-93 on the next page.
* **Add TextBox Object** Drag a TextBox object onto the form . Using snap lines, align the top of the TextBox object with the top of the Text block object. Name the TextBox object txtLinearFeet. Change the font to Segoe WP, Size 25. Reduce the width of the TextBox object to closely resemble Figure 5-93. Center the Text Block object and the TextBox object horizontally in the frmWoodCabinetEstimate Form object. Don't forget to remove any text inside the box.

*The Text block object and TextBox object occupy the first line of the frmWoodCabinetEstimate Form object (Figure* 5-93) . *They are centered horizontally in the form.*

**

Figure 5-93

**Add a Second Text Block** Drag a Text Block object onto the form below the lblLinearFeet Text Block object. Name the Text Block lblWoodType. Change the text in the Text Block to Wood Type:. Change the font to Segoe WP, Regular, Size 25. Center the text block horizontally in the frmWoodCabinetEstimate Form object.

**Add a Rectagle** Drag a Rectangle object onto the frmWoodCabinetEstimate Form object. Name the Rectangle pnlWoodType. Set the Fill to a dark gray so the rectangle will stand out on the form. Set the width of the Rectangle to 296. Set the Height of the Rectangle to 268. Center the Rectangle object horizontally in the frmWoodCabinetEstimate

Form object *(ref: Figure 5-12).*

**Add Radio Buttons** Place three RadioButton objects on the Rectangle object. Name the first RadioButton radPine and change its Content property to Pine. Name the second RadioButton radOak and change its Content property to Oak. Name the third RadioButton radCherry and change its Content property to Cherry. Select the three RadioButtons and change the font to Segoe WP, Regular, Size 22 *(ref" Figure 5-15).*

**Set Radio Button Properties**  Click the Pine RadioButton object and change its Checked property from False to True. Pine is the most commonly used wood by this cabinetmaker *(ref: Figure 5-20).*

*The rectangle and radio buttons are included on the frmWoodCabinetEstimate Form object*

*(Figure* 5-94). *The light gray background of the Rectangle object helps it to stand out on the form. The radPine radio button is selected because it is the most widely used wood type.*

**Add Estimate and Cost Text blocks** Drag two more Text block objects below the Rectangle object. Align these text blocks by their tops using snap lines. Name the first text block lblCostEstimateLabel and change its Text property to *Cost Estimate:* and resize the Text block object to view the text. Name the second text block lblCostEstimate and set its Text property to *0000.00*. These placement zeros allow you to view the Text block object when it is not selected. The placement zeros will be cleared using code when the form is loaded. Change the font for both Text block objects to Segoe WP, Regular, Size 25. Horizontally center the text blocks as a unit on the frmWoodCabinetEstimate Form object.

**Add Calculate and Clear Buttons** Drag two Button objects onto the form. Align the tops of the Button objects using snap lines. Name the first Button object btnCalculate and change its Content property to Calculate. Name the second Button object on the right btnClear and change its Content property to Clear.



**Code the Comments** Double-click the btnCalculate Button object on the frmWoodCabinetEstimate Form object to open the code editing window and create the btnCalculate\_Click Event Handler. Close the Toolbox. Click in front of the first words, Partial Public Class MainPage, and press the ENTER key to create a blank line. Insert the first four standard comments. Insert the Option Strict On command at the beginning of the code to turn on strict type checking

'Program Name: Wood Cabinet Estimate for Windows Phone

'Author: YOUR NAME HERE

'Date: January 29, 2012

'Purpose : This mobile application computes the estimated cost

' of wood cabinets based on the number of linear feet of

' cabinets and the following cost per linear foot:

' Pine -$100.00 per linear foot; Oak -$150.00 per

' linear foot; Cherry -$250.00 per linear foot.

**Comment btnCalculate\_Click Event Handler** Enter a com ment to describe the purpose of the btnCalculate Click event. **This goes in the place where you place code to calculate.**

Private Sub btnCalculate\_Click(sender As System.Object, e As System.Windows.RoutedEventArgs) Handles btnCalculate.Click

'The btnCalculate event handler calculates the estimated cost of 16 cabinets based on the linear feet and the wood type.

'Declaration Section

**Declare and Initialize the Variables** This application requires six variables: 1) decLinearFeet: Holds the estimated linear footage of the cabinets. 2) decCostPerFoot: Holds the cost per linear foot based on the wood type; 3) decCostEstimate: Is assigned the calculated final estimated cost; 4) decPineCost: Is assigned the value 100.00; 5) decOakCost: Is assigned the value 150.00; 6) decCherryCost: Is assigned the value of 250.00. Declare and initialize these six variables.

'Declaration Section

Dim decLinearFeet As Decimal

Dim decCostPerFoot As Decimal

Dim decCostEstimate As Decimal

Dim decPineCost As Decimal = 1000

Dim decOakCost As Decimal = 150D

Dim decCherryCost As Decimal = 250D

Dim input As String = txtLinearFeet.Text

Dim Letter As String

Dim SelectionInder As Integer = txtLinearFeet.SelectionStart

**Dim charactersAllowed As String = "0123456789."**

Dim charactersDisallowed As String = "abcdefghijklmopqrstuvwxyz!@#$%^&\*()\_+"

**Write the IfStatement to Test for Numeric Data** When the user clicks the Calculate button, the program must first ensure that the user entered a valid numeric value in the txtLinearFeet TextBox object. If the user has entered a valid numeric value, the value must be converted from a string value into a decimal data type. Write the If statement and conversion statement required for this process *(rel Figure* 5-82).

**Write the If Statement to Test for Positive**

**Number** If the value is numeric, then the converted numeric value must be checked to ensure

it is a positive number. Write the If statement to

check if the converted numeric value is greater

than zero (ref: Figure 5-83).

'Is Linear Feet greater than zero

If decLinearFeet > 0 Then

**Write the If Statements to Determine Cost**

**Per Linear Foot** When the value is greater than

zero, the cost per linear foot is determined by

checking the status of the RadioButton objects

and placing the appropriate cost per linear foot

in the decCostPerFoot variable. Using the

If ... Then ... ElseIf structure, write the statements

to identify the checked radio button and place

the appropriate cost in the decCostPerFoot vari­

able (ref: Figure 5-57).

Private Sub btnCalculate\_Click(sender As System.Object, e As System.Windows.RoutedEventArgs) Handles btnCalculate.Click

'The btnCalculate event handler calculates the estimated cost of 16 cabinets based on the linear feet and the wood type.

'Declaration Section

Dim decLinearFeet As Decimal

Dim decCostPerFoot As Decimal

Dim decCostEstimate As Decimal

Dim decPineCost As Decimal = 1000

Dim decOakCost As Decimal = 150D

Dim decCherryCost As Decimal = 250D

Dim charactersAllowed As String = "0123456789."

Dim charactersDisallowed As String = "abcdefghijklmopqrstuvwxyz!@#$%^&\*()\_+"

Dim input As String = txtLinearFeet.Text

Dim Letter As String

Dim SelectionInder As Integer = txtLinearFeet.SelectionStart

'Did user enter a numeric value?

For X As Integer = 0 To txtLinearFeet.Text.Length - 1

Letter = txtLinearFeet.Text.Substring(X, 1)

If charactersAllowed.Contains(Letter) Then

decLinearFeet = Convert.ToDecimal(Me.txtLinearFeet.Text)

'Is Linear Feet greater than zero

If decLinearFeet > 0 Then

'Determine cost per foot of wood

If Me.radPine.IsChecked Then

decCostPerFoot = decPineCost

ElseIf Me.radOak.IsChecked Then

decCostPerFoot = decOakCost

ElseIf Me.radCherry.IsChecked Then

decCostPerFoot = decCherryCost

End If

'Calculate and display the cost estimate

decCostEstimate = decLinearFeet \* decCostPerFoot

Me.lblCostEstimate.Text = decCostEstimate.ToString("C")

Else

'Display error message if user entered a negative value

MessageBox.Show("You entered" & decLinearFeet.ToString() & \_

" . Enter a Number Greater Than Zero. Input Error")

Me.txtLinearFeet.Text = ""

Me.txtLinearFeet.Focus()

End If

End If

If charactersDisallowed.Contains(Letter) Then

' Display error message if user entered a nonnumeric value

MessageBox.Show("Enter the Linear Feet of the Cabinets. Input Error")

Me.txtLinearFeet.Text = Me.txtLinearFeet.Text

Me.txtLinearFeet.Focus()

End If

Next

End Sub

The Final Code is below.

'Program Name: Wood Cabinet Estimate

'Author: Corinne Hoisington

'Date: January 29, 2012

'Purpose : This mobile application computes the estimated cost

' of wood cabinets based on the number of linear feet of

' cabinets and the following cost per linear foot:

' Pine -$100.00 per linear foot; Oak -$150.00 per

' linear foot; Cherry -$250.00 per linear foot.

Option Strict On

Partial Public Class MainPage

Inherits PhoneApplicationPage

' Constructor

Public Sub New()

InitializeComponent()

End Sub

Dim charactersAllowed As String = "0123456789."

Dim charactersDisallowed As String = "abcdefghijklmopqrstuvwxyz!@#$%^&\*()\_+"

Private Sub btnCalculate\_Click(sender As System.Object, e As System.Windows.RoutedEventArgs) Handles btnCalculate.Click

'The btnCalculate event handler calculates the estimated cost of 16 cabinets based on the linear feet and the wood type.

'Declaration Section

Dim decLinearFeet As Decimal

Dim decCostPerFoot As Decimal

Dim decCostEstimate As Decimal

Dim decPineCost As Decimal = 1000

Dim decOakCost As Decimal = 150D

Dim decCherryCost As Decimal = 250D

Dim input As String = txtLinearFeet.Text

Dim Letter As String

Dim SelectionInder As Integer = txtLinearFeet.SelectionStart

'Did user enter a numeric value?

For X As Integer = 0 To txtLinearFeet.Text.Length - 1

Letter = txtLinearFeet.Text.Substring(X, 1)

If charactersAllowed.Contains(Letter) Then

decLinearFeet = Convert.ToDecimal(Me.txtLinearFeet.Text)

'Is Linear Feet greater than zero

If decLinearFeet > 0 Then

'Determine cost per foot of wood

If Me.radPine.IsChecked Then

decCostPerFoot = decPineCost

ElseIf Me.radOak.IsChecked Then

decCostPerFoot = decOakCost

ElseIf Me.radCherry.IsChecked Then

decCostPerFoot = decCherryCost

End If

'Calculate and display the cost estimate

decCostEstimate = decLinearFeet \* decCostPerFoot

Me.lblCostEstimate.Text = decCostEstimate.ToString("C")

Else

'Display error message if user entered a negative value

MessageBox.Show("You entered" & decLinearFeet.ToString() & \_

" . Enter a Number Greater Than Zero. Input Error")

Me.txtLinearFeet.Text = ""

Me.txtLinearFeet.Focus()

End If

End If

If charactersDisallowed.Contains(Letter) Then

' Display error message if user entered a nonnumeric value

MessageBox.Show("Enter the Linear Feet of the Cabinets. Input Error")

Me.txtLinearFeet.Text = Me.txtLinearFeet.Text

Me.txtLinearFeet.Focus()

End If

Next

End Sub

Private Sub btnClear\_Click(sender As System.Object, e As System.Windows.RoutedEventArgs) Handles btnClear.Click

Me.txtLinearFeet.Text = ""

Me.lblCostEstimate.Text = ""

Me.radPine.IsChecked = True

Me.radOak.IsChecked = False

Me.radCherry.IsChecked = False

Me.txtLinearFeet.Focus()

End Sub

Private Sub Estimate\_Loaded(sender As System.Object, e As System.Windows.RoutedEventArgs) Handles MyBase.Loaded

Me.txtLinearFeet.Focus()

Me.radPine.IsChecked = True

lblCostEstimate.Text = ""

End Sub

End Class