BeckAddisonCH6Lab

## Shape Class

Public Class Shape

'Shape Class:

'- Single Instance Variable of Name

Protected m\_name As String

'- Constructor method with a required Name input parameter

Public Sub New(p\_name)

m\_name = p\_name

End Sub

'- GetArea method – defined as MustOverride, returns a Double

Public Overridable Function GetArea() As Double

Return 0

End Function

'- ToString method should return the name of the shape

Public Overrides Function ToString() As String

Return m\_name

End Function

End Class

## Rectangle Class

Public Class Rectangle

Inherits Shape

Private m\_width As Integer

Private m\_height As Integer

Public Property Width() As String

Get

Return m\_width

End Get

Set(value As String)

m\_width = value

End Set

End Property

Public Property Height() As String

Get

Return m\_height

End Get

Set(value As String)

m\_height = value

End Set

End Property

Sub New()

MyBase.New("Rectangle")

m\_height = 0

m\_width = 0

End Sub

Sub New(ByVal p\_height, ByVal p\_width)

MyBase.New("Rectangle")

m\_height = p\_height

m\_width = p\_width

End Sub

Public Overrides Function GetArea() As Double

Return (m\_width \* m\_height)

End Function

Public Overrides Function ToString() As String

Return "Shape: " & MyBase.m\_name & vbCrLf & "Height: " & CStr(m\_height) & vbCrLf & "Width: " & CStr(m\_width) & vbCrLf & "Area: " & GetArea()

End Function

End Class

## Circle Class

Public Class Circle

Inherits Shape

Private m\_radius As Double

Public Property Radius() As Double

Get

Return m\_radius

End Get

Set(value As Double)

m\_radius = value

End Set

End Property

Public Sub New()

MyBase.New("Circle")

m\_radius = 0

End Sub

Public Sub New(p\_radius)

MyBase.New("Circle")

m\_radius = p\_radius

End Sub

Public Overrides Function GetArea() As Double

Dim radius As Double = ((m\_radius \* m\_radius) \* Math.PI)

Return Math.Round(radius, 2)

End Function

Public Overrides Function ToString() As String

Return "Shape: " & MyBase.m\_name & vbCrLf & "Radius: " & m\_radius & vbCrLf & "Area: " & GetArea()

End Function

End Class

## Form Code

Public Class HomePage

Private Sub rdbRectangle\_CheckedChanged(sender As Object, e As EventArgs) Handles rdbRectangle.CheckedChanged

txtHeight.ReadOnly = False

txtWidth.ReadOnly = False

txtRadius.ReadOnly = True

txtRadius.Text = ""

End Sub

Private Sub rdbCircle\_CheckedChanged(sender As Object, e As EventArgs) Handles rdbCircle.CheckedChanged

txtHeight.ReadOnly = True

txtWidth.ReadOnly = True

txtRadius.ReadOnly = False

txtHeight.Text = ""

txtWidth.Text = ""

End Sub

Private Sub btnAdd\_Click(sender As Object, e As EventArgs) Handles btnAdd.Click

If rdbCircle.Checked = True Then

Dim radius As Double = CDbl(txtRadius.Text)

Dim myCircle As New Circle(radius)

txtOutput.Text = myCircle.ToString()

ElseIf rdbRectangle.Checked = True Then

Dim height As Integer = txtHeight.Text

Dim width As Integer = txtWidth.Text

Dim myRectangle As New Rectangle(height, width)

txtOutput.Text = myRectangle.ToString

End If

End Sub

End Class

## Screenshots

