

BASE CLASS

Define the Properties

```
Public Class baseClass
    Private stockNumber As String
    Private movieTitle As String
    Private rating As String
    Private retailPrice As Decimal
    Private stockOnHand As Integer
    Private stockOnOrder As Integer
    Private lastReceivedDate As Date
    ,
```

Add the Accessor Methods

```
    ,
    Property movieID() As String
        Get
            Return stockNumber
        End Get
        Set(ByVal Value As String)
            If Len(Value) > 0 Then
                stockNumber = Value
            End If
        End Set
    End Property
    ,
    Property title() As String
        Get
            Return movieTitle
        End Get
        Set(ByVal Value As String)
            If Len(Value) > 0 Then
                movieTitle = Value
            End If
        End Set
    End Property
    ,
    Property viewingRating() As String
        Get
            Return rating
        End Get
        Set(ByVal Value As String)
            If Len(Value) > 0 Then
                rating = Value
            End If
        End Set
    End Property
    ,
```

```

Property price() As Decimal
    Get
        Return retailPrice
    End Get
    Set(ByVal Value As Decimal)
        If Value >= 0 Then
            retailPrice = Value
        End If
    End Set
End Property
'
Property onHandCount() As Integer
    Get
        Return stockOnHand
    End Get
    Set(ByVal Value As Integer)
        If Value >= 0 Then
            stockOnHand = Value
        End If
    End Set
End Property
'
Property onOrderCount() As Integer
    Get
        Return stockOnOrder
    End Get
    Set(ByVal Value As Integer)
        If Value >= 0 Then
            stockOnOrder = Value
        End If
    End Set
End Property
'
Property receivedDate() As Date
    Get
        Return lastReceivedDate
    End Get
    Set(ByVal Value As Date)
        lastReceivedDate = Value
    End Set
End Property

```

Add the Constructors

```
Public Sub New(ByVal movieID As String, ByVal title As String, _
ByVal viewingrating As String, ByVal price As Decimal, _
ByVal onhandcount As Integer, ByVal onordercount As Integer, _
ByVal receiveddate As Date)
    Me.movieID = movieID
    Me.title = title
    Me.viewingRating = viewingrating
    Me.price = price
    Me.onHandCount = onhandcount
    Me.onOrderCount = onordercount
    Me.receivedDate = receiveddate
    AddTotals()
End Sub
```

DERIVED CLASS

Action Class Example

```
Public Class actionClass
    Inherits baseClass

    Public Sub New(ByVal movieID As String, ByVal title As String, _
    ByVal viewingrating As String, ByVal price As Decimal, _
    ByVal onhandcount As Integer, ByVal onordercount As Integer, _
    ByVal receiveddate As Date)
        MyBase.new(movieID, title, viewingrating, price, _
        onhandcount, onordercount, receiveddate)
    End Sub
End Class
```

FORM

Define the Objects

```
Public Class Form1
    Inherits System.Windows.Forms.Form
    '
    Private anActionMovie As actionClass
    Private anAnimatedMovie As animatedClass
    Private aComedyMovie As comedyClass
    Private aDramaMovie As dramaClass
    Private aMusicalMovie As musicalClass
    Private aRomanceMovie As romanceClass
    Private aSciFiMovie As sciFiClass
    Private aWarMovie As warClass

```

Commit Button Click Event

```
Private Sub commitButton_Click(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles commitButton.Click
    ' Label7.Text = RatingComboBox.SelectedItem
    '
    If actionRadioButton.Checked Then
        Try
            anActionMovie = New actionClass(stockNumberTextBox.Text, _
                movieTitleTextBox.Text, RatingComboBox.SelectedItem, _
                Decimal.Parse(priceTextBox.Text), 0, 0, Today)
            actionCount += 1
        Catch ex As Exception

        End Try
    ElseIf animatedRadioButton.Checked Then
        Try
            anAnimatedMovie = New animatedClass(stockNumberTextBox.Text, _
                movieTitleTextBox.Text, RatingComboBox.SelectedItem, _
                Decimal.Parse(priceTextBox.Text), 0, 0, Today)
            animatedCount += 1
        Catch ex As Exception

        End Try
    ElseIf dramaRadioButton.Checked Then
        Try
            aDramaMovie = New dramaClass(stockNumberTextBox.Text, _
                movieTitleTextBox.Text, RatingComboBox.SelectedItem, _
                Decimal.Parse(priceTextBox.Text), 0, 0, Today)
            dramaCount += 1
        Catch ex As Exception

        End Try
    ElseIf comedyRadioButton.Checked Then

```

```

    Try
        aComedyMovie = New comedyClass(stockNumberTextBox.Text, _
            movieTitleTextBox.Text, RatingComboBox.SelectedItem, _
            Decimal.Parse(priceTextBox.Text), 0, 0, Today)
        comedyCount += 1
    Catch ex As Exception

End Try
ElseIf musicalRadioButton.Checked Then
    Try
        aMusicalMovie = New musicalClass(stockNumberTextBox.Text, _
            movieTitleTextBox.Text, RatingComboBox.SelectedItem, _
            Decimal.Parse(priceTextBox.Text), 0, 0, Today)
        musicalCount += 1
    Catch ex As Exception

End Try
ElseIf romanceRadioButton.Checked Then
    Try
        aRomanceMovie = New romanceClass(stockNumberTextBox.Text, _
            movieTitleTextBox.Text, RatingComboBox.SelectedItem, _
            Decimal.Parse(priceTextBox.Text), 0, 0, Today)
        romanceCount += 1
    Catch ex As Exception

End Try
ElseIf sciFiRadioButton.Checked Then
    Try
        aSciFiMovie = New sciFiClass(stockNumberTextBox.Text, _
            movieTitleTextBox.Text, RatingComboBox.SelectedItem, _
            Decimal.Parse(priceTextBox.Text), 0, 0, Today)
        sciFiCount += 1
    Catch ex As Exception

End Try
ElseIf warRadioButton.Checked Then
    Try
        aWarMovie = New warClass(stockNumberTextBox.Text, _
            movieTitleTextBox.Text, RatingComboBox.SelectedItem, _
            Decimal.Parse(priceTextBox.Text), 0, 0, Today)
        warCount += 1
    Catch ex As Exception

End Try
Else
    MessageBox.Show("Select a movie category", "Entry Error", _
        MessageBoxButtons.OK, MessageBoxIcon.Error)
End If
End Sub

```

BASE CLASS

Shared Variables

```
Private Shared totalStockonHand, totalStockonOrder As Integer
```

Property Procedures

```
Shared ReadOnly Property TotalOnHand()
```

```
Get
```

```
Return totalStockonHand
```

```
End Get
```

```
End Property
```

```
,
```

```
Shared ReadOnly Property TotalOnOrder()
```

```
Get
```

```
Return totalStockonOrder
```

```
End Get
```

```
End Property
```

```
,
```

Add Counts Procedure

```
Private Sub AddTotals()
```

```
totalStockonHand += 1
```

```
totalStockonOrder += 1
```

```
End Sub
```

FORM

List Box Selected Event

```
Private Sub categoryListBox_SelectedIndexChanged(ByVal sender As System.Object, ByVal e As System.EventArgs) Handles
categoryListBox.SelectedIndexChanged
    Select Case categoryListBox.SelectedIndex
        Case 0
            onHandLabel.Text = actionCount.ToString
        Case 1
            onHandLabel.Text = animatedCount.ToString
        Case 2
            onHandLabel.Text = comedyCount.ToString
        Case 3
            onHandLabel.Text = dramaCount.ToString
        Case 4
            onHandLabel.Text = musicalCount.ToString
        Case 5
            onHandLabel.Text = romanceCount.ToString
        Case 6
            onHandLabel.Text = sciFiCount.ToString
        Case 7
            onHandLabel.Text = warCount.ToString
    End Select

End Sub
End Class
```

```
Dim actionCount, animatedCount, comedyCount, dramaCount, _  
musicalCount, romanceCount, sciFiCount, warCount As Integer  
,  
Dim actionOrderCount, animatedOrderCount, comedyOrderCount, _  
dramaOrderCount, musicalOrderCount, romanceOrderCount, _  
sciFiOrderCount, warOrderCount As Integer
```