```
' ************************************************************************
' Surname, Initials: Kuni, D
' Student Number: 201320596
' Practical: P07
' Class name: (this is the name of your class)
' ************************************************************************
Option Strict On
Option Explicit On
Option Infer Off
Public Class frmArties
    Private productions() As Production
    Private p As Integer 'number of productions
    Private choice As Integer 'type of production
    Private Sub btnsetup_Click(sender As Object, e As EventArgs) Handles btnsetup.Click
        p = CInt(InputBox("Please enter the number of Productions")) - 1
        ReDim productions(p)
    End Sub
    Private Sub btnaddPro_Click(sender As Object, e As EventArgs) Handles btnaddPro.Click
        For i As Integer = 0 To p
            choice = CInt(InputBox("Enter the number 1 for a normal Production, Enter the
number 2 for a Comedy Production or Enter the number 3 for a Drama Production", "Choose
Type of production"))
            If choice <> 1 And choice <> 2 And choice <> 3 Then
                'changes the choice if incorrect to a standard production type
                MsgBox("Error!!!, the value you have enter is incorrect, your choice will
be changed to a normal production", MsgBoxStyle.OkOnly, "ERROR!!!")
                choice = 1
            End If
            Select Case choice
                Case 1
                    Dim pro As Production
                    Dim name As String = InputBox("The name of the production ",
"Production " + CStr(i + 1))
                    Dim arrsize As Integer = CInt(InputBox("How many shows were performed
from the production?", "Production " + CStr(i + 1))) - 1
                        Dim arr(arrsize) As Integer
                        For r As Integer = 0 To arrsize
                        arr(r) = CInt(InputBox("Enter the Audience number for show " +
CStr(r + 1), "Production " + CStr(i + 1)))
                        Next
                        pro = New Production(name, arr)
                        productions(i) = pro
                Case 2
                        Dim pro As ComedyProduction
                    Dim name As String = InputBox("The name of the production ",
"Production " + CStr(i + 1))
                    Dim arrsize As Integer = CInt(InputBox("How many shows were performed
from the production?", "Production " + CStr(i + 1))) - 1
                        Dim arr(arrsize) As Integer
                        For r As Integer = 0 To arrsize
                        arr(r) = CInt(InputBox("Enter the Audience number for show " +
CStr(r + 1), "Production " + CStr(i + 1)))
                        Next
                        Dim comedians, awards, prevAwards As Integer
                    comedians = CInt(InputBox("Please enter the number of comedians for
the production ", "Production " + CStr(i + 1)))
                    awards = CInt(InputBox("Please enter the number of awards the
production's staff recieved ", "Production " + CStr(i + 1)))
```

```
                    prevAwards = CInt(InputBox("Please enter the number of previous
awards given to the production", "Production " + CStr(i + 1)))
                        pro = New ComedyProduction(name, comedians, awards, prevAwards,
arr)
                        productions(i) = pro
                Case 3
                        Dim pro As DramaProduction
                    Dim name As String = InputBox("The name of the production ",
"Production " + CStr(i + 1))
                    Dim arrsize As Integer = CInt(InputBox("How many shows were performed
from the production?", "Production " + CStr(i + 1))) - 1
                        Dim arr(arrsize) As Integer
                        For r As Integer = 0 To arrsize
                        arr(r) = CInt(InputBox("Enter the Audience number for show " +
CStr(r + 1), "Production " + CStr(i + 1)))
                        Next
                        Dim themes, twists As Integer
                    themes = CInt(InputBox("Please enter the number of themes for the
production", "Production " + CStr(i + 1)))
                    twists = CInt(InputBox("Please enter the number of twists in the
production ", "Production " + CStr(i + 1)))
                        pro = New DramaProduction(name, twists, themes, arr)
                        productions(i) = pro
            End Select
        Next
    End Sub

    Private Sub btnDisplay_Click(sender As Object, e As EventArgs) Handles
btnDisplay.Click
        'displaying of data
        txtave.Text = ""
        txtDisplay.Text = ""
        txthigh.Text = ""
        Dim lining As String = Environment.NewLine +
"========================================" + Environment.NewLine
        Dim numbofcoms As Integer = 0
        Dim total As Double = 0
        Dim high As Double = 0
        Dim com As ComedyProduction
        For i As Integer = 0 To p
            txtDisplay.Text += productions(i).DisplayDetails + lining ' this is
polymorphism
            If TypeOf productions(i) Is ComedyProduction Then 'checks for comedy
productions
                numbofcoms += 1 ' gets the number of comedy productions
                com = DirectCast(productions(i), ComedyProduction) 'downcasting
                total += com.CalculateScore
                If high < com.CalculateScore Then
                    high = com.CalculateScore
                End If
                productions(i) = com 'upcasting
            End If
        Next
        txthigh.Text = CStr(Math.Round(high, 2))
        txtave.Text = CStr(Math.Round(total / numbofcoms, 2))
    End Sub
    Private Sub btntest_Click(sender As Object, e As EventArgs) Handles btntest.Click
        p = 6
```

```
        ReDim productions(p)
        productions(0) = New Production("Show A", 125, 100, 89, 75)
        productions(1) = New ComedyProduction("Show B", 10, 5, 111, 90, 50, 120)
        productions(2) = New ComedyProduction("Show C", 5, 5, 90, 90, 85, 77, 90, 55, 72)
        productions(3) = New DramaProduction("Show D", 125, 100, 89, 75)
        productions(4) = New DramaProduction("Show E", 4, 8, 15, 20, 15, 45, 75)
        productions(5) = New DramaProduction("Show F", 10, 20, 189, 275, 30)
        productions(6) = New ComedyProduction("Show G", 25, 10, 9, 25, 44, 34, 25, 15,
32, 78)
    End Sub
End Class
```

```
' ************************************************************************
' Surname, Initials: Kuni, D
' Student Number: 201320596
' Practical: P07
' Class name: Production
' ************************************************************************
Option Strict On
Option Explicit On
Option Infer Off
Public Class Production
    Private _name As String
    Protected AveAudience As Double
    Public ReadOnly Property Name As String
        Get
            Return _name
        End Get
    End Property
    Public Overridable Function DisplayDetails() As String
        ' displays all neccessary data
        Return "The Name of the production : " + Name + Environment.NewLine + "The
average audience numbers : " + CStr(Math.Round(AveAudience, 2)) + Environment.NewLine
    End Function
    Private Function CalculateAveAudience(ByRef arr() As Integer) As Double
        For Each Val As Integer In arr
            If Val < 0 Then
                Val = 0
            End If
            CalculateAveAudience += Val
        Next
        CalculateAveAudience = CalculateAveAudience / arr.Length
    End Function
    Public Sub New(Name As String, ParamArray AudienceNumbers() As Integer)
        'sets all neccessary input data
        _name = Name 'set name at instantiation
        AveAudience = CalculateAveAudience(AudienceNumbers)
    End Sub
End Class
```

```
' ************************************************************************
' Surname, Initials: Kuni, D
```

```
' Student Number: 201320596
' Practical: P07
' Class name: ComedyProduction
' ******************************************************************************
Option Strict On
Option Explicit On
Option Infer Off
Public Class ComedyProduction
    Inherits Production
    Implements iWinning
#Region "Attributes"
    Private _NumberOfComedians As Integer
    Private _AwardWinningStaff As Integer
    Private _PreviousAwards As Integer
#End Region
   public Function CalculateScore() As Double Implements iWinning.CalculateScore
        Return AveAudience * ((_AwardWinningStaff + _PreviousAwards) / 2) +
_NumberOfComedians
    End Function
    Public Property NumberOfComedians As Integer
        Get
            Return _NumberOfComedians
        End Get
        Set(value As Integer)
            If value >= 0 Then
                _NumberOfComedians = value
            End If
        End Set
    End Property
    Public Property NumberOfAwardWinningStaff As Integer Implements
iWinning.NumberOfAwardWinningStaff
        Get
            Return _AwardWinningStaff
        End Get
        Set(value As Integer)
            If value >= 0 Then
                _AwardWinningStaff = value
            End If
        End Set
    End Property
    Public Property NumberOfPreviousAwards As Integer Implements
iWinning.NumberOfPreviousAwards
        Get
            Return _PreviousAwards
        End Get
        Set(value As Integer)
            If value >= 0 Then
                _PreviousAwards = value
            End If
        End Set
    End Property
    Public Sub New(name As String, NumberOfComedians As Integer,
NumberOfAwardWinningStaff As Integer, NumberOfPreviousAwards As Integer, ParamArray
AudienceNumbers() As Integer)
        'sets all neccessary input data
        MyBase.New(name, AudienceNumbers)
        Me.NumberOfComedians = NumberOfComedians
        Me.NumberOfAwardWinningStaff = NumberOfAwardWinningStaff
```

```
        Me.NumberOfPreviousAwards = NumberOfPreviousAwards
    End Sub
    Public Overrides Function DisplayDetails() As String
        ' displays all neccessary data
        Return MyBase.DisplayDetails() + "Number Of Award Winning Staff : " +
CStr(NumberOfAwardWinningStaff) + Environment.NewLine + "Number Of Previous Awards : " +
CStr(NumberOfPreviousAwards) + Environment.NewLine + "Number Of Comedians : " +
CStr(NumberOfComedians) + Environment.NewLine
    End Function
End Class
```

```
' ****************************************************************************
' Surname, Initials: Kuni, D
' Student Number: 201320596
' Practical: P07
' Class name: iWinning
' ****************************************************************************
Option Strict On
Option Explicit On
Option Infer Off
Public Interface iWinning
    Property NumberOfAwardWinningStaff As Integer
    Property NumberOfPreviousAwards As Integer
    Function CalculateScore() As Double
End Interface
```

```
' ****************************************************************************
' Surname, Initials: Kuni, D
' Student Number: 201320596
' Practical: P07
' Class name: DramaProduction
' ****************************************************************************
Option Strict On
Option Explicit On
Option Infer Off
Public Class DramaProduction
    Inherits Production
    Private _Themes As Integer
    Private _PlotTwists As Integer
    Public Property NumberOfThemes As Integer
        Get
            Return _Themes
        End Get
        Set(value As Integer)
            If value >= 0 Then
                _Themes = value
            End If
        End Set
    End Property
    Public Property NumberOfPlotTwists As Integer
        Get
            Return _PlotTwists
```

```
        End Get
        Set(value As Integer)
            If value >= 0 Then
                _PlotTwists = value
            End If
        End Set
    End Property
    Public Overrides Function DisplayDetails() As String
        ' displays all neccessary data
        Return MyBase.DisplayDetails() + "Number of Themes : " + CStr(_Themes) +
Environment.NewLine + "Number of Plot Twists : " + CStr(_PlotTwists) +
Environment.NewLine
    End Function
    Public Sub New(name As String, NumberOfPlotTwists As Integer, NumberOfThemes As
Integer, ParamArray AudienceNumbers() As Integer)
        'sets all neccessary input data
        MyBase.New(name, AudienceNumbers)
        Me.NumberOfPlotTwists = NumberOfPlotTwists
        Me.NumberOfThemes = NumberOfThemes
    End Sub
End Class
```