

```

' *****
' Surname, Initials: Kuni, D
' Student Number: 201320596
' Practical: P10
' Class name: frmIG
' *****
Option Strict On
Option Explicit On
Option Infer Off
Imports System.IO
Public Class frmIG

#Region "Variables"
    Private Structure FacilityRec
        Public ID As Integer ' is 4 bytes
        <VBFixedString(20)> Public name As String 'is 2 * 20 bytes
        Public lockers As Boolean 'is 2 bytes
        Public Popularity As Integer 'is 4 bytes
    End Structure
    Private blankRec, Facilites As FacilityRec
    Private FS As FileStream
    Private Const FileName As String = "S201320596_P10.IFM"
    Private Const RecSize As Integer = 4 + (20 * 2) + 4 + 4 'total record size
    Private Const numbRecs As Integer = 30
    Private Const lastrec As Integer = 29 'last record position
    Private Stadiums() As cStadium
    Private numberOfStadiums As Integer
#End Region
#Region "File Options"
    Private Sub tsmAddrec_Click(sender As Object, e As EventArgs) Handles tsmAddrec.Click
        ' Allows the user to add a facility record to file

        Try
            Facilites.ID = CInt(InputBox("Please enter The ID (0 to 100 ) of the
Facility"))
            Facilites.lockers = CBool(InputBox("Is there a Need For lockers, true or
false? "))
            Facilites.name = InputBox("Please enter the name of the Facility")
            Facilites.Popularity = CInt(InputBox("Please enter the popularity of the
Facility (0 to 100)"))
            writetofile(Facilites.ID)
        Catch ex As InvalidCastException
            MsgBox("Sorry the Datatype you entered was incorrect , please try again " +
vbNewLine + "Error Details: " + vbNewLine + ex.Message)
        Catch ex As Exception
            MsgBox(" It is the true or false question, enter true or false for it ,
nothing else , no spaces")
        End Try
    End Sub
    Private Sub tsmCreate_Click(sender As Object, e As EventArgs) Handles tsmCreate.Click
        'creates an empty record file
        FS = New FileStream(FileName, FileMode.Create, FileAccess.Write)
        FS.SetLength(RecSize * numbRecs)
        Dim bw As New BinaryWriter(FS)
        For i As Integer = 0 To lastrec
            FS.Seek(i * RecSize, SeekOrigin.Begin)
            bw.Write(blankRec.ID)
        Next
    End Sub
End Class

```

```

        bw.Write(blankRec.lockers)
        bw.Write(blankRec.name)
        bw.Write(blankRec.Popularity)
    Next
    FS.Flush()
    FS.Close()
    FS = Nothing
    bw = Nothing
    MsgBox("File Creation Successful!")
End Sub
#End Region
#Region "Methods"
    Private Sub writetofile(key As Integer) 'method to write a single record to file via
a hash function
        FS = New FileStream(FileName, FileMode.Open, FileAccess.Write)
        Dim bw As New BinaryWriter(FS)
        FS.Seek(GetFilePosition(key), SeekOrigin.Begin)
        bw.Write(Facilites.ID)
        bw.Write(Facilites.lockers)
        bw.Write(Facilites.name)
        bw.Write(Facilites.Popularity)
        FS.Flush()
        FS.Close()
        FS = Nothing
        bw = Nothing
    End Sub
    Private Function readfromfile(ByVal key As Integer) As FacilityRec 'method to read a
single file from records
        FS = New FileStream(FileName, FileMode.Open, FileAccess.Read)
        Dim br As New BinaryReader(FS)
        FS.Seek(GetFilePosition(key), SeekOrigin.Begin)
        readfromfile.ID = br.ReadInt32
        readfromfile.lockers = br.ReadBoolean
        readfromfile.name = br.ReadString
        readfromfile.Popularity = br.ReadInt32
        FS.Close()
        FS = Nothing
        br = Nothing
    End Function
    Private Function GetFilePosition(ByVal Key As Integer) As Integer
        Return (Key Mod 29) * RecSize '29 is the greatest prime number below 30
    End Function
#End Region
    Private Sub frmIG_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        'set blank record
        blankRec.ID = -1
        blankRec.lockers = False
        blankRec.name = "nothing"
        blankRec.Popularity = -1
    End Sub
#Region "Stadium Options "
    Private Sub tsmNumbStadiums_Click(sender As Object, e As EventArgs) Handles
tsmNumbStadiums.Click
        'gets the number of stadiums
        Try
            numberOfStadiums = CInt(InputBox("Please enter the number of stadiums "))
            ReDim Stadiums(numberOfStadiums)
            MsgBox("setup done")
        Catch
        End Try
    End Sub

```

```

        Catch inex As InvalidCastException
            MsgBox("Something wrong with the value you entered, make sure its an
numerical value " + vbNewLine + "Error Details: " + vbNewLine + inex.Message)
        Exit Sub
    Catch ex As Exception
        MsgBox("Error Details: " + vbNewLine + ex.Message)
    Exit Sub
End Try
End Sub

#End Region

Private Sub tsmSDetails_Click(sender As Object, e As EventArgs) Handles
tsmSDetails.Click
    'allows the user to input details of each stadium
    Dim numbfacts As Integer
    Dim name As String
    Try
        For i As Integer = 1 To numberOfStadiums
            name = InputBox("Please enter the name of stadium " + CStr(i))
            numbfacts = CInt(InputBox("Please enter the number of facilities for
stadium " + CStr(i)))
            Stadiums(i) = New cStadium(numbfacts)
            Stadiums(i).Name = name
        Next
        MsgBox("All done")
    Catch ex As Exception
        MsgBox("Error Details: " + vbNewLine + ex.Message)
    Exit Sub
End Try
End Sub
Private Sub tsmAssign_Click(sender As Object, e As EventArgs) Handles tsmAssign.Click
    'sets objects of a stadium of a specific index from file records
    Try
        Dim id As Integer = 0
        Dim stadiumnumb As Integer = CInt(InputBox("Please enter the number of which
stadium desired to assign desired facilities"))
        If Stadiums(stadiumnumb) Is Nothing Then Throw New NullReferenceException
        Dim count As Integer = 1
        MsgBox("There is " + CStr(Stadiums(stadiumnumb).NumberOfFacilities) + "
facilities in the stadium , Ensure that you have enter enough records to assign to the
desired stadium")
        While count <= Stadiums(stadiumnumb).NumberOfFacilities
            id = CInt(InputBox("Enter ID number of facility " + CStr(count)))
            Facilites = readfromfile(id)
            If Facilites.ID = -1 Then
                MsgBox("Its a Blank Record, try again please")
            Else
                Stadiums(stadiumnumb).Facilities(count) = New
cFacility(Facilites.name, Facilites.ID, Facilites.Popularity, Facilites.lockers)
                count += 1
            End If
        End While
    Catch ex As NullReferenceException
        MsgBox("There is no such Stadium" + vbNewLine + "Error Details: " + vbNewLine
+ ex.Message)
    Catch ex As Exception
        MsgBox("Error Details: " + vbNewLine + ex.Message)
    End Try
End Sub

```

```

        Exit Sub
    End Try
End Sub

Private Sub tsmDisplay_Click(sender As Object, e As EventArgs) Handles
tsmDisplay.Click
    'diplays stadiums data and extra information that is needed
    Dim count As Integer = 0
    txtextra.Clear()
    txtIG.Clear()
    Try
        For i As Integer = 1 To numberOfStadiums
            txtIG.Text += Stadiums(i).Display + vbNewLine +
"===== " + vbNewLine
            If Stadiums(i).DetermineOverallRating < 60 Then
                txtextra.Text += "Name Of Stadium : " + Stadiums(i).Name + vbNewLine
                txtextra.Text += "Stadiums Overall Rating : " +
CStr(Stadiums(i).DetermineOverallRating) + vbNewLine +
"===== " + vbNewLine
            End If
        Next
    Catch ex As Exception
        MsgBox("Error Details" + vbNewLine(2) + ex.Message)
        txtextra.Clear()
        txtIG.Clear()
    End Try
End Sub
#Region "Window Options"
Private Sub tsmExtra_Click(sender As Object, e As EventArgs) Handles tsmExtra.Click
    tcStadium.SelectedTab = tpExtra
End Sub
Private Sub tsmSwindow_click(sender As Object, e As EventArgs) Handles
tsmSwindow.Click
    tcStadium.SelectedTab = tpNormal
End Sub
#End Region
End Class

```

```

' *****
' Surname, Initials: Kuni, D
' Student Number: 201320596
' Practical: P10
' Class name: cStadium
' *****

Option Strict On
Option Explicit On
Option Infer Off
Public Class cStadium
#Region "Attributes"
    Private _Name As String
    Private _NumberofFacilities As Integer
    Private _Facilities() As cFacility
#End Region

```

```

#Region "Properties"
    Public Property Name As String
        Get
            Return _Name
        End Get
        Set(value As String)
            _Name = value
        End Set
    End Property
    Public ReadOnly Property NumberOfFacilities As Integer
        Get
            Return _NumberofFacilities
        End Get
    End Property
    Public Property Facilities(index As Integer) As cFacility
        Get
            Return _Facilities(index)
        End Get
        Set(value As cFacility)
            _Facilities(index) = value
        End Set
    End Property
#End Region
#Region "methods"
    Public Function Display() As String
        Display = "Name Of Stadium : " + _Name + vbNewLine
        Display += "Number Of Facilities : " + CStr(_NumberofFacilities) + vbNewLine
        Display += "Overall Rating : " + CStr(Math.Round(DetermineOverallRating, 2)) +
vbNewLine
        For i As Integer = 1 To NumberOfFacilities
            Display += "Facility Number " + CStr(i) + " details " + vbNewLine
            Display += _Facilities(i).display
        Next
    End Function
    Public Function DetermineOverallRating() As Double
        Dim total As Double = 0
        For i As Integer = 1 To NumberOfFacilities
            total += _Facilities(i).DeterminePopularityFactor
        Next
        Return total / _Facilities.Length
    End Function
    Public Sub New(NumberOfFacilities As Integer)
        _NumberofFacilities = NumberOfFacilities
        ReDim _Facilities(_NumberofFacilities)
    End Sub
#End Region
End Class

```

```

' *****
' Surname, Initials: Kuni, D
' Student Number: 201320596
' Practical: P10
' Class name: cfacility
' *****

```

```
Option Strict On
```

```

Option Explicit On
Option Infer Off
Public Class cFacility
#Region "Attributes"
    Private _Name As String
    Private _ID As Integer
    Private _Popularity As Integer
    Private _Lockers As Boolean
#End Region
    Public Function DeterminePopularityFactor() As Double
        If IsOdd() Then
            Return _Popularity * 1.5
        Else
            Return _Popularity
        End If
    End Function
    Private Function IsOdd() As Boolean
        If _ID Mod 2 = 0 Then
            Return False
        Else
            Return True
        End If
    End Function
    Public Sub New(_Name As String, _ID As Integer, _Popularity As Integer, _Lockers As
Boolean)
        Me._Name = _Name
        Me._ID = _ID
        Me._Popularity = _Popularity
        Me._Lockers = _Lockers
    End Sub
    Public Function display() As String
        display = "ID Of Facility : " + CStr(_ID) + vbNewLine
        display += "Name Of Facility : " + _Name + vbNewLine
        display += "Needs Lockers : " + NeedsLocker() + vbNewLine
        display += "Popularity : " + CStr(_Popularity) + vbNewLine
    End Function
    Private Function NeedsLocker() As String
        If _Lockers Then
            Return "Yes"
        Else
            Return "No"
        End If
    End Function
End Class

```