**clsStudent.VB**

Public Class clsStudent

'------------------------------------------------------------

'- File Name : clsStudent.vb -

'- Part of Project: StudentOOP -

'------------------------------------------------------------

'- Written By: Binh D. Dang -

'- Written On: April 7th, 2022 -

'------------------------------------------------------------

'- File Purpose: - -

'- This file contains the student object class

'------------------------------------------------------------

'- Program Purpose: -

'- The purpose of this program is showing the Student object class

'-having a constructor and getters and setters for each purpose

'------------------------------------------------------------

Private strFirstandMiddleIni As String

Private strLast As String

Private arrAssignment(4) As Single

Private dblExam As Double

'constructor for student

Public Sub New(ByVal FirstandMiddleInitial As String, ByVal LastName As String,

ByVal newArrAssignment As Single(), ByVal Exam As Double)

strFirstandMiddleIni = FirstandMiddleInitial

strLast = LastName

arrAssignment = newArrAssignment

dblExam = Exam

End Sub

'getter and setters

Public Property FirstandMiddleIni As String

Get

Return strFirstandMiddleIni

End Get

Set(value As String)

strFirstandMiddleIni = value

End Set

End Property

Public Property Last As String

Get

Return strLast

End Get

Set(value As String)

strLast = value

End Set

End Property

Public Property Assignments As Single()

Get

Return arrAssignment

End Get

Set(value As Single())

arrAssignment = value

End Set

End Property

Public Property Exam As Double

Get

Return dblExam

End Get

Set(value As Double)

dblExam = value

End Set

End Property

End Class

**Program.VB**

Imports System

Imports Microsoft.Office.Interop

Module Program

'------------------------------------------------------------

'- File Name : Program.vb -

'- Part of Project: StudentOOP -

'------------------------------------------------------------

'- Written By: Binh D. Dang -

'- Written On: April 6th, 2022 -

'------------------------------------------------------------

'- File Purpose: - -

'- This file contains the console that is able to populate

'- excel datasheet using a list

'------------------------------------------------------------

'- Program Purpose: -

'- The purpose of this program is to populate

'- excel datasheet using a list

'------------------------------------------------------------

Sub Main(args As String())

'------------------------------------------------------------

'- Subprogram Name: Main -

'------------------------------------------------------------

'- Written By: Binh Dang -

'- Written On: April 6, 2022 -

'------------------------------------------------------------

'- Subprogram Purpose: -

'- -

'- This subprogram started by adding the students to the students list

'-then later create an excel for populating

'------------------------------------------------------------

'- Local Variable Dictionary (alphabetically): -

'- anExcelDoc - Excel Application object

'- CheckExcel - Object to get excel appication

'- myStudents - List of Students

'-intColumn - int keep track the column

'- intRow - int keep Trach what row are on

'- strColumn() - array of column from C to I

'------------------------------------------------------------

Dim myStudents As New List(Of clsStudent)

'generating value for the list by creating a student object and add it in

myStudents.Add(New clsStudent("V.A.", "Borstellis", {25, 25, 25, 25}, 100.0))

myStudents.Add(New clsStudent("A.S.", "Reid", {20, 21, 20, 18}, 75.0))

myStudents.Add(New clsStudent("C.U.", "Tyler", {19, 20, 21, 24}, 75.5))

myStudents.Add(New clsStudent("H.A.", "Renee", {20, 23, 23, 25}, 80.5))

myStudents.Add(New clsStudent("I.A.", "Douglas", {24, 23, 25, 25}, 95.0))

myStudents.Add(New clsStudent("M.A.", "Elenaips", {23, 24, 23, 21}, 94.5))

myStudents.Add(New clsStudent("A.L.", "Emmet", {21, 19, 18, 15}, 73.0))

myStudents.Add(New clsStudent("S.U.", "James", {21, 24, 23, 22}, 87.5))

myStudents.Add(New clsStudent("S.H.", "Issacs", {23, 24, 21, 21}, 93.0))

myStudents.Add(New clsStudent("B.I.", "Opus", {23, 24, 25, 23}, 97.5))

myStudents.Add(New clsStudent("T.R.", "Alski", {24, 25, 25, 23}, 95.5))

myStudents.Add(New clsStudent("H.E.", "Zeus", {23, 24, 23, 23}, 77.0))

myStudents.Add(New clsStudent("S.C.", "Ustaf", {24, 23, 24, 25}, 91.0))

myStudents.Add(New clsStudent("K.I.", "Chrint", {23, 23, 24, 21}, 89.0))

myStudents.Add(New clsStudent("J.E.", "Yaz", {25, 24, 23, 24}, 92.5))

myStudents.Add(New clsStudent("F.R.", "Franks", {23, 19, 18, 23}, 88.5))

myStudents.Add(New clsStudent("W.I.", "Walton", {24, 23, 23, 19}, 90.0))

myStudents.Add(New clsStudent("K.A.", "Gilch", {24, 23, 25, 24}, 92.0))

myStudents.Add(New clsStudent("R.O.", "Little", {23, 24, 23, 24}, 94.0))

myStudents.Add(New clsStudent("S.A.", "Xerxes", {24, 23, 25, 23}, 94.0))

myStudents.Add(New clsStudent("W.I.", "Harris", {23, 24, 25, 23}, 92.0))

myStudents.Add(New clsStudent("T.I.", "Vargo", {24, 23, 25, 25}, 99.0))

myStudents.Add(New clsStudent("I.E.", "Interas", {24, 23, 25, 25}, 97.5))

myStudents.Add(New clsStudent("T.O.", "Kiliens", {23, 19, 18, 18}, 73.0))

myStudents.Add(New clsStudent("E.R.", "Manrose", {23, 24, 25, 23}, 84.0))

myStudents.Add(New clsStudent("W.A.", "Nelson", {23, 24, 25, 23}, 87.0))

myStudents.Add(New clsStudent("K.U.", "Quaras", {23, 24, 25, 23}, 96.5))

Dim CheckExcel As Object

Dim anExcelDoc As Excel.Application

Dim intRow As Integer = 2

'Check to see if Excel is already loaded in memory

Try

CheckExcel = GetObject(, "Excel.Application")

Catch Ex As Exception

'Excel was not running, so we got an error

End Try

If CheckExcel Is Nothing Then

'Create a new instance of Excel

anExcelDoc = New Excel.Application()

anExcelDoc.Visible = True

Else

anExcelDoc = CheckExcel

anExcelDoc.Visible = True

End If

'Add a new workbook and a new sheet

anExcelDoc.Workbooks.Add()

anExcelDoc.Sheets.Add()

'Put column header

anExcelDoc.Cells(1, 1) = "Initials"

anExcelDoc.Cells(1, 2) = "Name "

anExcelDoc.Cells(1, 3) = "Grade 1"

anExcelDoc.Cells(1, 4) = "Grade 2"

anExcelDoc.Cells(1, 5) = "Grade 3"

anExcelDoc.Cells(1, 6) = "Grade 4"

anExcelDoc.Cells(1, 7) = "Grade Total"

anExcelDoc.Cells(1, 8) = "Exam"

anExcelDoc.Cells(1, 9) = "Final Grade"

'run through the student list and print out info

For Each student In myStudents

anExcelDoc.Cells(intRow, 1) = student.FirstandMiddleIni

anExcelDoc.Cells(intRow, 2) = student.Last

anExcelDoc.Cells(intRow, 3) = student.Assignments(0)

anExcelDoc.Cells(intRow, 4) = student.Assignments(1)

anExcelDoc.Cells(intRow, 5) = student.Assignments(2)

anExcelDoc.Cells(intRow, 6) = student.Assignments(3)

anExcelDoc.Cells(intRow, 7) = "=SUM(C" + intRow.ToString + ":F" + intRow.ToString + ")"

anExcelDoc.Cells(intRow, 8) = student.Exam

anExcelDoc.Cells(intRow, 9) = "=G" + intRow.ToString + "\* 0.4 + H" + intRow.ToString + " \* 0.6"

intRow = intRow + 1

Next

intRow = intRow - 1 'getback to the latest row

Console.WriteLine("All data has been filled in" + vbCrLf + "Click Any Key to Continue!" + vbCrLf)

Console.ReadLine()

'Throw some statistical information on the sheet

anExcelDoc.Cells(intRow + 2, 2) = "Average:"

anExcelDoc.Cells(intRow + 3, 2) = "Standard Deviation:"

anExcelDoc.Cells(intRow + 4, 2) = "Min:"

anExcelDoc.Cells(intRow + 5, 2) = "Max:"

Dim strColumn() As String = {"C", "D", "E", "F", "G", "H", "I"}

Dim intColumn As Integer = 3

For Each col In strColumn

anExcelDoc.Cells(intRow + 2, intColumn) = "=AVERAGE(" + col + "2:" + col + intRow.ToString + ")"

anExcelDoc.Cells(intRow + 3, intColumn) = "=STDEV(" + col + "2:" + col + intRow.ToString + ")"

anExcelDoc.Cells(intRow + 4, intColumn) = "=MIN(" + col + "2:" + col + intRow.ToString + ")"

anExcelDoc.Cells(intRow + 5, intColumn) = "=MAX(" + col + "2:" + col + intRow.ToString + ")"

intColumn = intColumn + 1

Next

'Make the columns fit the data

anExcelDoc.Range("A1:I1").EntireColumn.AutoFit()

Console.WriteLine("All statistical data is now available" + vbCrLf + "Click Any Key to End!")

'Clean things up

anExcelDoc.Quit()

anExcelDoc = Nothing

Console.ReadLine()

End Sub

End Module