

```
1  ''' <summary>
2  ''' This application will open a text files with a list of students
3  ''' marks, convert each mark to a grade and then work out the
4  ''' percentage of students who have a particular grade
5  '''
6  ''' College:      Alyesbury
7  ''' Course:      OCR Level 2 IT
8  ''' Unit 27:     Developing Program Solutions
9  ''' Assessment 3: Code Student Grades Program
10 ''' Date:       19th March 2018
11 ''' </summary>
12 Public Class GradesForm
13
14     ' Maximum number of student marks that the program
15     ' can load and process set a the largest class size
16     Dim MaxNoStudents As Integer = 20
17
18     ' Arrays for the student names, marks and grades
19
20     Dim marks(MaxNoStudents) As Integer
21     Dim grades(MaxNoStudents) As String
22     Dim students(MaxNoStudents) As String
23
24     Dim filename As String = "MarksExample.csv"
25
26     ' F = Fail, P = Pass, M = Merit, D = Distinction
27     Dim gradeNames() As String = {"F", "P", "M", "D"}
28     Dim percentages(3) As Integer
29
30
31     Private Sub LoadFile(sender As Object, e As EventArgs) Handles MyBase.Load
32
33     End Sub
34
35
36     ''' <summary>
37     ''' Load all the student marks from the external file
38     ''' and store them in the listbox. The first line of the
39     ''' file are the column headings
40     ''' </summary>
41     Private Sub OpenFile(sender As Object, e As EventArgs) Handles
42         OpenToolStripMenuItem.Click
43
44         Dim inputline As String
45         Dim index As Integer = 0
46
47         OpenFileDialog1.ShowDialog()
48         filename = OpenFileDialog1.FileName
49
50         FileOpen(1, filename, OpenMode.Input)
51
52         Do While Not EOF(1)
53
54             inputline = LineInput(1)
55
56             If index > 0 Then ' not titles
```

```
56         marksListBox.Items.Add(inputline)
57         GetStudent(index - 1, inputline)
58     End If
59
60     index = index + 1
61
62     Loop
63
64     FileClose(1)
65     MaxNoStudents = index - 1
66
67     marksListBox.Items.Add("")
68     marksListBox.Items.Add("No of Marks = " & MaxNoStudents)
69
70
71 End Sub
72
73
74 ''' <summary>
75 ''' Display each student with their name, mark and grade in
76 ''' the grades list box
77 ''' </summary>
78 Private Sub DisplayGrades()
79
80     Dim index As Integer = 0
81     Dim line As String = Nothing
82
83     For Each student In students
84
85         If (index < MaxNoStudents) Then
86             line = String.Format("Mark = {0:##}, Grade = {1} for {2}",
87                 marks(index), grades(index), student)
88             gradesListBox.Items.Add(line)
89         End If
90
91         index = index + 1
92     Next
93
94 End Sub
95
96 ''' <summary>
97 ''' Calculate and display the percentage of the students
98 ''' that have acheived a particular grade and display them
99 ''' in the percentageListBox
100 ''' </summary>
101 Private Sub DisplayPercentages()
102
103     Dim index As Integer = 0
104     Dim line As String
105     Dim percent As Integer
106
107     For Each number In percentages
108
109         line = "Grade " & gradeNames(index) & vbTab
110
```

```
111         percent = (number * 100) / MaxNoStudents
112
113         percentagesListBox.Items.Add(line & percent & "%")
114         index = index + 1
115
116     Next
117
118 End Sub
119
120 ''' <summary>
121 ''' Take one line of input and split it into the student
122 ''' name and associated mark. Add it to the marks
123 ''' array and return the mark
124 ''' </summary>
125 ''' <param name="index">to array of students and marks</param>
126 ''' <param name="line">One line of input</param>
127 ''' <returns>the mark between 0 and 100</returns>
128 Function GetStudent(index As Integer, line As String) As Integer
129
130     Dim dataArray() As String
131     Dim mark As Integer
132     Dim name As String
133
134     dataArray = line.Split(",")
135
136     name = dataArray(0)
137     mark = dataArray(1)
138
139     students(index) = name
140     marks(index) = mark
141
142     Return mark
143
144 End Function
145
146
147 ''' <summary>
148 ''' Close the form and quit the application
149 ''' </summary>
150 Private Sub CloseForm(sender As Object, e As EventArgs) Handles ↗
151     closeButton.Click
152     Close()
153 End Sub
154
155 ''' <summary>
156 ''' Prompt the user for a filename, and then save the student names
157 ''' and grades in a new csv file
158 ''' </summary>
159 Private Sub SaveFile(sender As Object, e As EventArgs) Handles ↗
160     SaveToolStripMenuItem.Click
161
162     Dim outputline As String
163     Dim index As Integer = 0
164
165     SaveFileDialog1.Filter = "csv Files|*.csv"
166     SaveFileDialog1.ShowDialog()
```

```
165     filename = SaveFileDialog1.FileName
166
167     FileOpen(1, filename, OpenMode.Output)
168
169     For index = 0 To MaxNoStudents - 1
170
171         outputline = String.Format("{0}, {1}{2}", students(index), grades ↗
            (index), vbCrLf)
172         Print(1, outputline)
173
174     Next
175
176     FileClose()
177
178     MessageBox.Show(filename & " saved!", "Save File", ↗
        MessageBoxButtons.OK)
179
180 End Sub
181
182
183 ''' <summary>
184 ''' Count how many marks fall within a particular range
185 ''' for each of the grades F, P, M, and D
186 ''' </summary>
187 Private Sub CalculateGrades(sender As Object, e As EventArgs) Handles ↗
    CalculateButton.Click
188
189     Dim index As Integer = 0
190
191     For Each mark In marks
192
193         If (index < MaxNoStudents) Then
194             If (mark < 40) Then
195                 grades(index) = gradeNames(0)
196                 percentages(0) = percentages(0) + 1
197             ElseIf (mark < 65) Then
198                 grades(index) = gradeNames(1)
199                 percentages(1) = percentages(1) + 1
200             ElseIf (mark < 80) Then
201                 grades(index) = gradeNames(2)
202                 percentages(2) = percentages(2) + 1
203             ElseIf (mark <= 100) Then
204                 grades(index) = gradeNames(3)
205                 percentages(3) = percentages(3) + 1
206             End If
207         End If
208
209         index = index + 1
210
211     Next
212
213     DisplayGrades()
214     DisplayPercentages()
215
216 End Sub
217
```

218 End Class

219