

p150258

by Lloyd Thomas Pennington

FILE	2671_LLOYD_THOMAS_PENNINGTON_P150258_113992_1460428139.DOCX		
TIME SUBMITTED	11-MAY-2018 10:39PM (UTC+0100)	WORD COUNT	1166
SUBMISSION ID	89029190	CHARACTER COUNT	6705

Contents

Process	2
Creating Forms	2
Menu Form	2
Add Form	2
All Form	3
Find Form	3
Evaluation of VB	4
Appendix	5

Process

The process that has been taken to combat this scenario was as follows;

Creating Forms

The first step was to produce all the forms, this is so all the different functions had a place to be implemented within the program. This also allowed the use of multiple forms; Menu, Add, All and Find. The use of multiple forms allowed the structure to be chosen in an orderly fashion, i.e. everything was not on one form and everything was not all cramped together.

Menu Form

The second step was to work on adding all the features and code onto the menu form. This being because it is the simplest form to complete, all that was needed for the features were three command buttons; these were to link to the other forms.

Appendix 1 shows the design of the form. This form is designed in a simple way, as this is only the menu form and its main purpose is to link to the other forms, so not a lot of features are needed on this form.

The code that was used for this was simply for the buttons. The code, as shown in appendix 2 shows the layout of the code and the different coding techniques used. The reasoning for this code is because it was simple and easy to use.

Add Form

The third step was to work on the features and the code aspects of the add form. This is because this is the main form that will be used by the user, as this form allows the user to add information to the excel document that is linked with the program.

Appendix 3 shows the overall design of the form. This form was designed in way that allows the user to simply add the information that is needed to the excel document. As is shown, there are text boxes, labels and a data grid; these are all for the user to know where to insert the information into the program. There are also command buttons; these command buttons are there to add the information to the spreadsheet and to export it from the spreadsheet.

The coding techniques that have been used are different than the previous form in all cases apart from the buttons. The buttons generally have the same task, when pressed they will have a purpose – in this case the purpose is to either add or export information to or from the spreadsheet. This piece of code is shown in appendix 4

There is code linking the program to the spreadsheet. This is done by the “Dim Exellocation” and the “Dim my ss” code. The datagridview code allows the information to be placed into the spreadsheet by comparing the cells from the datagrid to the cells in the excel document.

The export button just shows the code for the exporting of the information inserted in the excel document to the program. This is done by the code technique “Dim ColumnsCount”. This piece of code adds the information from each cell into the corresponding cell of the data grid on the program. This piece of code is shown in appendix 5.

All Form

The fourth step was to add the features and implement the code into the “All” form. This form was created next because it uses the information from the previous form, the “Add” form, and shows the information in a clear order.

Appendix 6 shows the overall design of the form. The design of the form is done in the way that allows the user to press a button and print off the necessary information, in this case the necessary information was to show the amount of medals that have been achieved by the members of the swimming club; this is shown in the list boxes and have been named for the user to know where the information is going to be shown.

The coding techniques that have been used are shown in appendix 7. There are the same code techniques as the other forms with the command button, i.e. the user presses the button and then the information gets printed off, the only difference being that the location of the information being printed; in this case it will be printed off in each of the list boxes.

Find Form

The last form that was designed and had the code implemented was the “Find” form. This form was done last as this form is used to find the relevant data and present information from the data, in this case the information being presented was the percentages of what gender got each medal

Appendix 8 shows the overall design of the form. This form was designed in the way that the user could select the gender and then add the medals for each of the gender. This then allows the user to press the calculate button and this then works out the total and is presented in the txtCalculate text box. This information is stored and will be presented in either the male or the female text box, depending on what the user selected. This has been done in the way that it allows the user to easily distinguish which gender got more medals as it shows the overall percentage.

The coding techniques are shown in appendix 9. The techniques that have been used in this form are mainly to do with calculations and printing off the information. The calculations are done by the use of the “If” statement code and the holding and printing off the information is mainly done with the use of the “cmdTotal” code.

Evaluation of VB

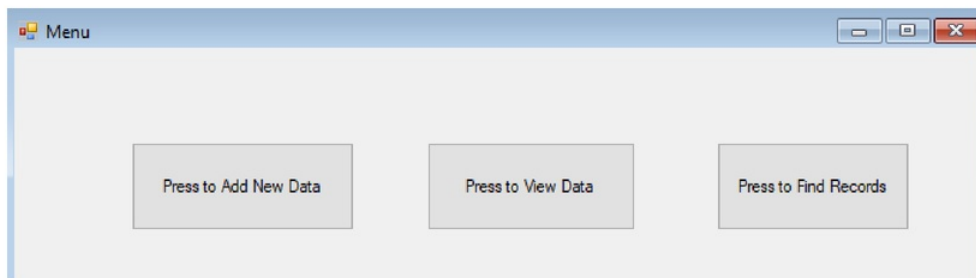
VB seems to be an overall simple coding interface to use, this is mainly due to the fact that when there is an error it will show and not run the program. This is useful, for beginner and experienced coders, as it allows people to go over and check to see if the code is correct and, if not, what needs to be changed.

Another plus of VB is that when writing a prefix or syntax there will be option available to be chosen, this is a help for people who do not really know what they are doing when it comes to coding as it can help them to learn new aspects of coding and learning different features.

The main issues that was observed is that VB sometimes uses the older files when running the program, this did happen when trying to run this program. What this means is that when the program is running it will not load any of the assets on the page, so no buttons or code will load, even though all the assets are there and the code is correctly implemented; this made testing the program overall an impossible task.

Appendix

1



2

```
0 references
Private Sub Menu_Load(sender As Object, e As EventArgs) Handles MyBase.Load
:
End Sub

0 references
Private Sub cmdAdd_Click(sender As Object, e As EventArgs) Handles cmdAdd.Click
:
Add.Show()
End Sub

0 references
Private Sub cmdAll_Click(sender As Object, e As EventArgs) Handles cmdAll.Click
:
All.Show()
End Sub

0 references
Private Sub cmdFind_Click(sender As Object, e As EventArgs) Handles cmdFind.Click
:
Find.Show()
End Sub
End Class
```

3

The screenshot shows a Windows application window titled "Add". On the left, there are four text input fields labeled "ID", "Sex", "Swimming Style", and "Time Taken to Complete". To the right of these fields is a DataGrid with four columns: "ID", "Sex", "Swimming Style", and "Time Taken". The first row of the DataGrid is highlighted. Below the input fields and the DataGrid are two buttons: "Add" and "Export".

4

```
Imports Excel = Microsoft.Office.Interop.Excel
1 reference
Public Class Add

    Dim excellocation As String = "C:\Users\Lloyd\Desktop\Business-Programming\add"

    Dim myss As String = "C:\Users\Lloyd\Desktop\Business-Programming\add"

    Dim ds As DataSet
    Dim tables As DataTableCollection
    Dim source1 As New BindingSource
    Dim APP As New Excel.Application
    Dim worksheet As Excel.Worksheet
    Dim workbook As Excel.Workbook

    Dim columnIndex As Integer = 0

    0 references
    Private Sub cmdAdd_Click(sender As Object, e As EventArgs) Handles cmdAdd.Click
        DataGridView1.Rows.Item(0).Cells("ID").Value = txtID.Text()
        DataGridView1.Rows.Item(0).Cells("Sex").Value = txtSex.Text
        DataGridView1.Rows.Item(0).Cells("SwimmingStyle").Value = txtSwimmingStyle.Text
        DataGridView1.Rows.Item(0).Cells("Competition 1 Medal").Value = txtTime.Text

        txtID.Clear()
        txtSex.Clear()
        txtSwimmingStyle.Clear()
        txtTime.Clear()
    End Sub

    0 references
    Private Sub DataGridView1_CellContentClick(sender As Object, e As DataGridViewCellEventArgs) Handles DataGridView1.CellContentClick
    End Sub
```

5

```
0 references
Private Sub Form1_Load(sender As System.Object, e As System.EventArgs) Handles MyBase.Load
    Dim xl = CreateObject("Excel.Application")
    workbook = APP.Workbooks.Open(excellocation)
    xl.workbooks.Open(myss)
    worksheet = workbook.Worksheets("sheet1")

    ds = New DataSet
    tables = ds.Tables

    Dim view As New DataView(tables(0))
    source1.DataSource = view
    DataGridView1.DataSource = view
    DataGridView1.AllowUserToAddRows = False
End Sub

0 references
Private Sub cmdExport_Click(sender As Object, e As EventArgs) Handles cmdExport.Click

    Dim columnsCount As Integer = DataGridView1.Columns.Count

    For Each column In DataGridView1.Columns
        worksheet.Cells(1, column.Index + 1).Value = column.Name
    Next
End Sub

End Class
```

6

7

```

1 reference
Public Class All
    Dim myss As String = "C:\Users\Lloyd\Desktop\Business-Programming\add"
    0 references
    Private Sub cmdMedal_Click(sender As Object, e As EventArgs) Handles cmdMedal.Click

        Dim row, col, finalrow, finalcol As Integer

        Dim xl = CreateObject("Excel.Application")
        xl.Workbooks.Open(myss)

        finalrow = xl.ActiveSheet.UsedRange.Rows.Count
        finalcol = xl.ActiveSheet.UsedRange.Columns.Count
        Dim vals(finalrow, finalcol) As String

        For row = 1 To finalrow
            For col = 1 To finalcol
                vals(row, col) =
                    Str(xl.ActiveSheet.Cells(row, col).Value)
                lstBronze.Items.Add(row)
                lstSilver.Items.Add(col)
                lstGold.Items.Add(vals(row, col))
            Next col
        Next row
        xl.Workbooks.Close()
        xl = Nothing
    End Sub

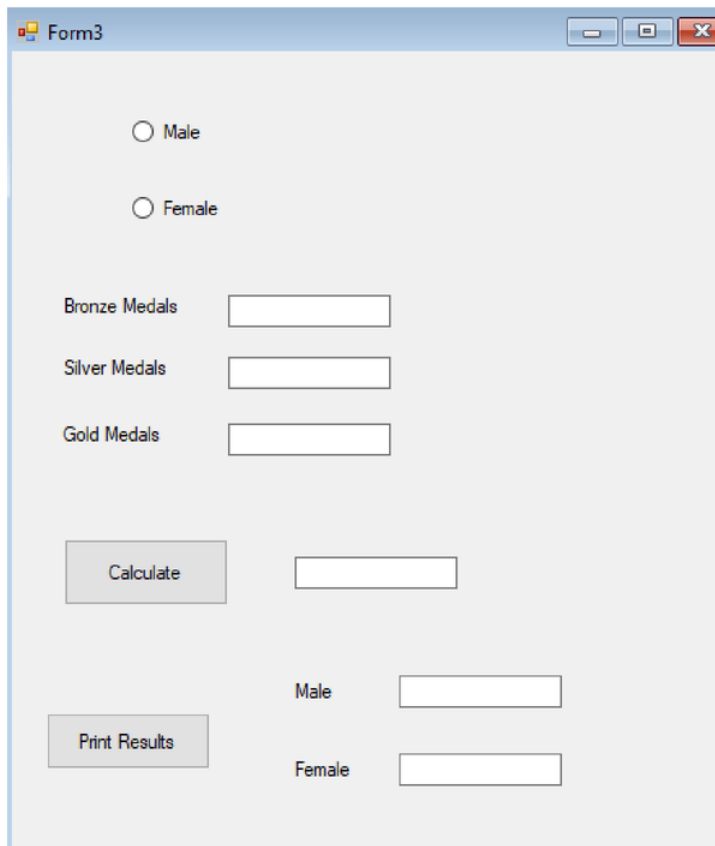
    0 references
    Private Sub lstSilver_SelectedIndexChanged(sender As Object, e As EventArgs) Handles lstSilver.SelectedIndexChanged
        ...
    End Sub

    0 references
    Private Sub Form2_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        ...
    End Sub
End Class

```

8

8



The screenshot shows a Windows application window titled "Form3". The window contains the following elements:

- Two radio buttons for gender selection: "Male" and "Female".
- Three text input fields labeled "Bronze Medals", "Silver Medals", and "Gold Medals".
- A "Calculate" button followed by a text input field.
- A "Print Results" button.
- Two more text input fields labeled "Male" and "Female" positioned to the right of the "Print Results" button.

9

```
1 reference
Public Class Find
    Dim sinMaleTotal
    Dim sinFemaleTotal

    0 references
    Private Sub cmdCalculate_Click(sender As Object, e As EventArgs) Handles cmdCalculate.Click
        Dim sinBronzemedals As Single
        Dim sinSilvermedals As Single
        Dim sinGoldmedals As Single
        Dim sinTotalmedals As Single

        If radMale.Checked Then
            sinMaleTotal = sinBronzemedals + sinSilvermedals + sinGoldmedals
        End If

        If radFemale.Checked Then
            sinFemaleTotal = sinBronzemedals + sinSilvermedals + sinGoldmedals
        End If
        txtCalculate.Text = sinTotalmedals
    End Sub

    0 references
    Private Sub cmdTotal_Click(sender As Object, e As EventArgs) Handles cmdTotal.Click
        txtMalesresult.Text = sinMaleTotal
        txtFemalesresult.Text = sinFemaleTotal
    End Sub

    0 references
    Private Sub Find_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    End Sub
End Class
```

FINAL GRADE

65/100

GENERAL COMMENTS

Instructor

Write a use guide that demonstrates the functionality of your system. 20%

A very good report and user guide. Demonstrating a suitable understanding of the requirements. 15/20

Demonstrate knowledge of the design principles used, when building a system using MS-Excel and VBA that will perform the procedures as requested by the user. 30%

Appropriate menu system. Good overall design principles and you have presented some well written code. 19/20

Facilities should be incorporated into the system that will automate the user activities. The success of the system build will be measured against the user requirements. 40%

All processes and functionality seem to work well. Maybe use more test data and show this within your report write-up 25/40

Overall presentation 10%

A good report presented to a high standard 7/10

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10
