

Assignment 1 - Business Programming - Report

by Luke James Evans

FILE	3783_LUKE_JAMES_EVANS_ASSIGNMENT_1_-_BUSINESS_PROGRAMMING_-_REPORT_113992_301834900.DOCX		
TIME SUBMITTED	11-MAY-2018 10:28PM (UTC+0100)	WORD COUNT	956
SUBMISSION ID	89028601	CHARACTER COUNT	5428

Assignment 1 – Business Programming – Luke Evans

Assignment 1 – Business Programming – Report

Luke Evans

BIT

Assignment 1 – Business Programming – Luke Evans

Contents

Assignment 1 – Business Programming – Report	1
Luke Evans.....	1
BIT	1
Program Documentation	3
Appendix	5

Program Documentation

The programming language chosen for this project was VB and the software used to create the program was Visual Studio using a Windows form Application.

Screenshot 1 in the Appendix shows a warning message that pops up when the program is launched. This issue will be addressed further on in the report. The program can still be run by clicking “Yes” on the pop up and runs completely fine.

Screenshot 2 in the Appendix shows what the user is greeted with when the program is properly opened, the Main Menu. The Main Menu offers 3 buttons for the user to choose from and each will take them to a new form. The “Add New Data” button will take them to a form where they can add new data to the spreadsheet, the “View All Data” button will essentially show them all the data that is stored in the spreadsheet in a presentable form and the “Find Records” form can help them with any specific queries that they have with the data.

Screenshot 3 in the Appendix shows the code that went into the making of the Main Menu. As you can see from the comments next to the code, each button was programmed to take the user to its respective form.

Screenshot 4 in the Appendix shows the Add data form which is where the user is taken after clicking that specific button on the main menu. The user is greeted by an empty data grid and text boxes to type in. The user is required to enter suitable information into these text boxes. Once the data has been added, the user can proceed to click the “Add Data” button.

Screenshot 5 in the Appendix shows the aftermath of the “Add Data” button; all information that the user has typed into the text boxes have been transferred to the data grid ready to be exported. The text boxes also get cleared after the button is clicked in case the user wishes to enter in different information.

Screenshot 6 in the Appendix shows the code that was used to make the “Add Data” button transfer the information from the text boxes to the data grid. As you can see, each labelled text box transferred its data to its respective column within the data grid. The code also shows the how the text boxes are cleared to make way for new information and it saves the user time because otherwise they would have to delete the information in each box 1 by 1.

Screenshot 7 in the Appendix shows the beginning of the code that is designed to export the information from the data grid to the excel spreadsheet which would happen after the “Export Data” button was clicked. Unfortunately, this is where the initial issue with the program comes in; this button does not work and has code which brings up that warning. The first part of the code is meant to find the spreadsheet; however, the second part which declares all variables necessary for the export is where the issue lies. Some of the variables come up as errors in the code and this is presumably an issue with excel. Due to this outcome, the code that is shown in Screenshots 7-9 in the appendix is what would have allowed the data to be exported if this issue didn’t occur.

Screenshot 10 in the Appendix show the View Data form which is accessed through the Main menu button. The user is greeted with 8 list boxes that represent the tables in the spreadsheet. The button

Assignment 1 – Business Programming – Luke Evans

“Get Data” is supposed to retrieve the data from the spreadsheet and display it for the user in the form.

Screenshot 11 in the Appendix shows the excel spreadsheet with limited data in it. Due to the issues with exporting data to the spreadsheet, example data had to be used just to show that data can be claimed back from the spreadsheet to view. Screenshot 12 in the Appendix simply shows this data in the spreadsheet being sent back to the program.

Screenshot 13 in the Appendix shows the code that was written for the data retrieval example. This code isn't the end product for this form due to it not being able to display letters, only digits; however, it still shows off the general aim of what the form should do once completed.

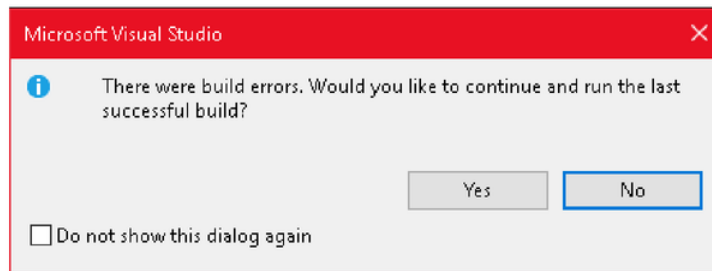
Screenshot 14 in the Appendix shows the Find Records form which doesn't seem to display properly. This form, when opened through the Main Menu, does not display what it should at all which is an issue that is yet to be explained. As you can see from the forms design view, what the form should do is calculate the medals percentage (Bronze, silver, gold) for all the male and female swimmers. The user gets to choose which gender they wish to make the query for. Unfortunately, no code has been added to this form to do the calculations due to the previous issues.

The majority of the code included in the program included suitable use of comments to explain what each part did.

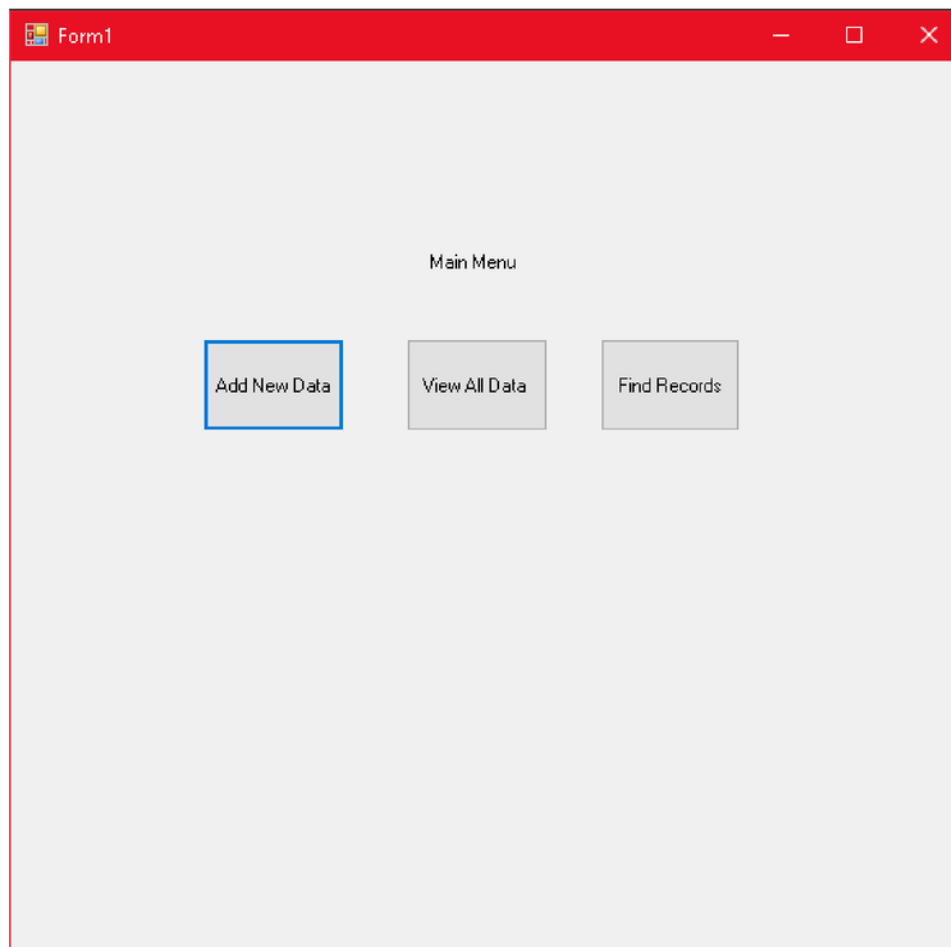
VB as a language, at least in this setting, is rather easy to use because for non technical people code is written after opening up a button in VB. It makes it a lot easier to start writing a program in comparison to other languages such as C#.

Appendix

Screenshot 1:



Screenshot 2:



Assignment 1 – Business Programming – Luke Evans

Screenshot 3:

```
1 Public Class Form1
2     Private Sub cmdAdd_Click(sender As Object, e As EventArgs) Handles cmdAdd.Click
3         'button to open the Add Data Form
4         Form2.Show()
5     End Sub
6
7     Private Sub cmdView_Click(sender As Object, e As EventArgs) Handles cmdView.Click
8         'Button to open the All Data Form
9         Form3.Show()
10    End Sub
11
12    Private Sub cmdFind_Click(sender As Object, e As EventArgs) Handles cmdFind.Click
13        'Button to open the Find Records Form
14        Form4.Show()
15    End Sub
16 End Class
17
```

Screenshot 4:

The screenshot shows a Windows application window titled 'Form2'. On the left, there is an 'Add Data' section with several text input fields: 'Swimmer Name', 'Sex', 'Swimming Club', 'Swimming Style', 'Swimmer Level', 'Number of Gold Medals', 'Number of Silver Medals', and 'Number of Bronze Medals'. To the right of these fields are two buttons: 'Add Data' and 'Export Data'. On the right side of the window is a data table with the following columns: 'Swimmer Name', 'Sex', 'Swimming Club', 'Swimming Style', 'Swimmer Level', 'Number of Gold Medals', 'Number Of Silver Medals', and 'Number Of Bronze Medals'. The table has a header row and one data row with a blue background. The data row contains the following values: 'John', 'Male', 'Swimming Club', 'Swimming Style', 'Swimmer Level', 'Number of Gold Medals', 'Number Of Silver Medals', and 'Number Of Bronze Medals'.

Swimmer Name	Sex	Swimming Club	Swimming Style	Swimmer Level	Number of Gold Medals	Number Of Silver Medals	Number Of Bronze Medals
John	Male	Swimming Club	Swimming Style	Swimmer Level	Number of Gold Medals	Number Of Silver Medals	Number Of Bronze Medals

Assignment 1 – Business Programming – Luke Evans

Screenshot 5:

Swimmer Name	Sex	Swimming Club	Swimming Style	Swimmer Level	Number of Gold Medals	Number of Silver Medals	Number of Bronze Medals
Luke Evans	Male	Gowerdon	Front Crawl	6	3	5	6

Screenshot 6:

```
17 Private Sub cmdAddData_Click(sender As Object, e As EventArgs) Handles cmdAddData.Click
18     'Button to add data to table
19     DataGridView1.Rows.Item(0).Cells("SwimmerName").Value = txtSwimmerName.Text
20     DataGridView1.Rows.Item(0).Cells("Sex").Value = txtSex.Text
21     DataGridView1.Rows.Item(0).Cells("SwimmingClub").Value = txtSwimmingClub.Text
22     DataGridView1.Rows.Item(0).Cells("SwimmingStyle").Value = txtStyle.Text
23     DataGridView1.Rows.Item(0).Cells("SwimmerLevel").Value = txtLevel.Text
24     DataGridView1.Rows.Item(0).Cells("NumberOfGoldMedals").Value = txtGold.Text
25     DataGridView1.Rows.Item(0).Cells("NumberOfSilverMedals").Value = txtSilver.Text
26     DataGridView1.Rows.Item(0).Cells("NumberOfBronzeMedals").Value = txtBronze.Text
27
28     'Clear the text boxes
29     txtSwimmerName.Clear()
30     txtSex.Clear()
31     txtSwimmingClub.Clear()
32     txtStyle.Clear()
33     txtLevel.Clear()
34     txtGold.Clear()
35     txtSilver.Clear()
36     txtBronze.Clear()
37
38 End Sub
```

Screenshot 7:

```
1 Public Class Form2
2
3     'Location of spreadsheet
4     Dim excellocation As String = "C:\Users\lukee\Desktop\Assignment 1 - Business Programming\Book.xlsx"
5
6     Dim myss As String = "C:\Users\lukee\Desktop\Assignment 1 - Business Programming\Book.xlsx"
7
8     'Declaring Variables
9     Dim ds As DataSet
10    Dim tables As DataTableCollection
11    Dim source1 As New BindingSource
12    Dim APP As New Excel.Application
13    Dim worksheet As Excel.Worksheet
14    Dim workbook As Excel.Workbook
15    Dim columnIndex As Integer = 0
```

Assignment 1 – Business Programming – Luke Evans

Screenshot 8:

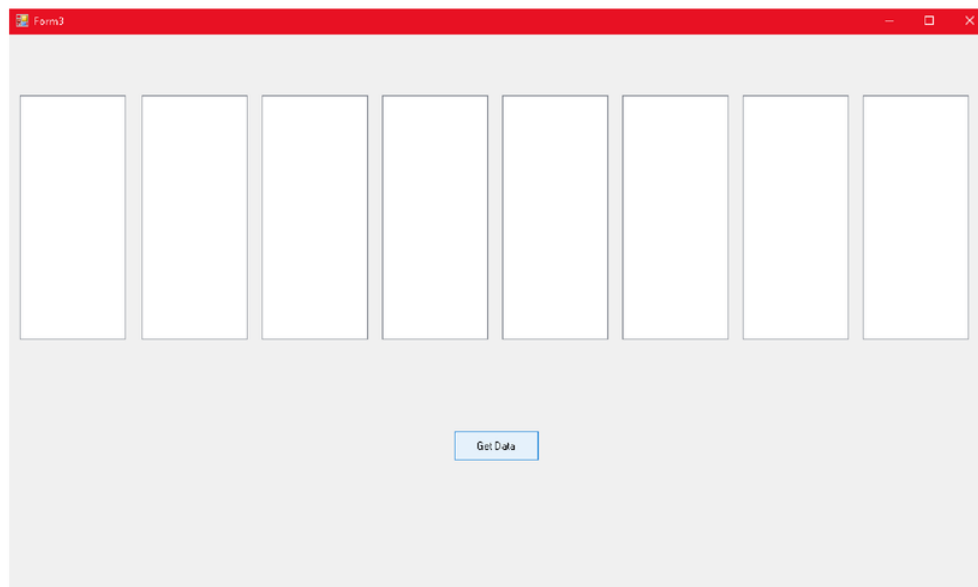
```
40 Private Sub Form2_Load(sender As System.Object, e As System.EventArgs) Handles MyBase.Load
41     Dim xl = CreateObject("Excel.Application")
42     workbook = APP.Workbooks.Open(excellocation)
43     xl.workbooks.Open(myss)
44     worksheet = workbook.Worksheets("sheet1")
45
46     ds = New DataSet
47     tables = ds.Tables
48
49     Dim view As New DataView(tables(0))
50     source1.DataSource = view
51     DataGridView1.DataSource = view
52     DataGridView1.AllowUserToAddRows = False
53
54 End Sub
55
56 Private Sub cmdExport_Click(sender As System.Object, e As System.EventArgs) Handles cmdExport.Click
57
58     Dim columnsCount As Integer = DataGridView1.Columns.Count
59
60     For Each column In DataGridView1.Columns
61         worksheet.Cells(1, column.Index + 1).Value = column.Name
62     Next
63
64     For i As Integer = 0 To DataGridView1.Rows.Count - 1
65
66         Dim columnIndex As Integer = 0
67         Do Until columnIndex = columnsCount
68
69             worksheet.Cells(i + 2, columnIndex + 1).Value = DataGridView1.Item(columnIndex, i).Value
70             columnIndex += 1
71
72         Loop
73     Next
74
75 End Sub
```

Screenshot 9:

```
77 Private Sub Form2_FormClosed(sender As Object, e As System.Windows.Forms.FormClosedEventArgs) Handles Me.FormClosed
78
79     workbook.Save()
80     workbook.Close()
81
82 End Sub
83
84
85 End Class
```

Assignment 1 – Business Programming – Luke Evans

Screenshot 10:



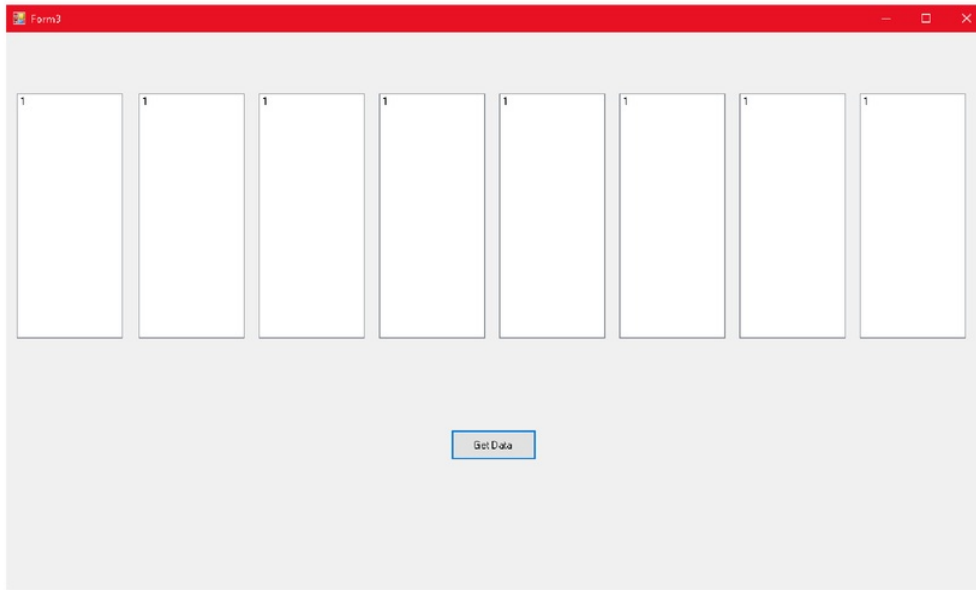
A screenshot of a web browser window displaying a form titled "Form3". The form consists of eight empty, vertically-oriented rectangular input fields arranged in a single row. Below these fields, centered horizontally, is a blue button with the text "Get Data". The browser window has a red title bar with standard minimize, maximize, and close buttons.

Screenshot 11:

	A	B	C	D
1	1			
2				
3				
4				
5				
6				

Assignment 1 – Business Programming – Luke Evans

Screenshot 12:



Screenshot 13:

```
1 Public Class Form3
2
3     Dim mySS As String = "C:\Users\lukee\Desktop\Assignment 1 - Business Programming\Book.xlsx"
4     Private Sub cmdGetData_Click(sender As Object, e As EventArgs) Handles cmdGetData.Click
5
6         Dim row, col, finalrow, finalCol As Integer
7         Dim xl = CreateObject("Excel.Application")
8         xl.Workbooks.Open(mySS)
9         finalrow = xl.ActiveSheet.UsedRange.Rows.Count
10        finalCol = xl.ActiveSheet.UsedRange.Columns.Count
11        Dim vals(finalrow, finalCol) As String
12
13        For row = 1 To finalrow
14            For col = 1 To finalCol
15                vals(row, col) =
16                    Str(xl.ActiveSheet.Cells(row, col).Value)
17                SwimmerName.Items.Add(col)
18                Sex.Items.Add(col)
19                Club.Items.Add(col)
20                Style.Items.Add(col)
21                Level.Items.Add(col)
22                NumberOfGoldMedals.Items.Add(col)
23                NumberOfSilverMedals.Items.Add(col)
24                NumberOfBronzeMedals.Items.Add(col)
25            Next col
26        Next row
27        xl.workbooks.Close()
28        xl = Nothing
29
30    End Sub
31 End Class
```

Assignment 1 – Business Programming – Luke Evans

Screenshot 14:

The screenshot shows a Windows application with two panes. The left pane, titled 'Form4', contains a 'Medal Percentages' section with radio buttons for 'Male' and 'Female', three text boxes for 'Bronze Medals', 'Silver Medals', and 'Gold Medals', a 'Calculate' button, and an empty text box. The right pane, titled 'Form4', contains a 'Label1' and a 'Button1'.

Assignment 1 - Business Programming - Report

GRADEMARK REPORT

FINAL GRADE

62/100

GENERAL COMMENTS

Instructor

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

20%

Good presentation and understanding demonstrated. 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%

You have discussed the program design well. 17/30

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%

Some of your code worked well and you provided a good menu system. 24/40

Overall presentation 10%

A very good write up and your report demonstrates that you have a suitable understanding of the development environment.

Your second form worked well it is shame you had some build errors. 6/10

PAGE 1

PAGE 2

PAGE 3

PAGE 4

PAGE 5

PAGE 6

PAGE 7

PAGE 8

PAGE 9

PAGE 10

PAGE 11
