

week2

January 20, 2018

1 Development Environment

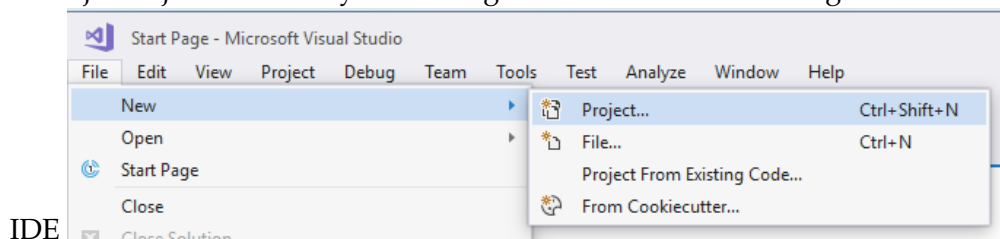
- You can use text editors like notepad to write your programs
- Alternatively you can use Integrated Development Environments (IDE) to make your life easier

2 Visual Studio (VS)

- We will use visual studio as our IDE
- Available for both Mac and PC
- Since we are working on VB, you have to use the PC version on mac
 - Use [parallels](#) to get it to work
- Get Visual Studio Community edition for free [here](#)

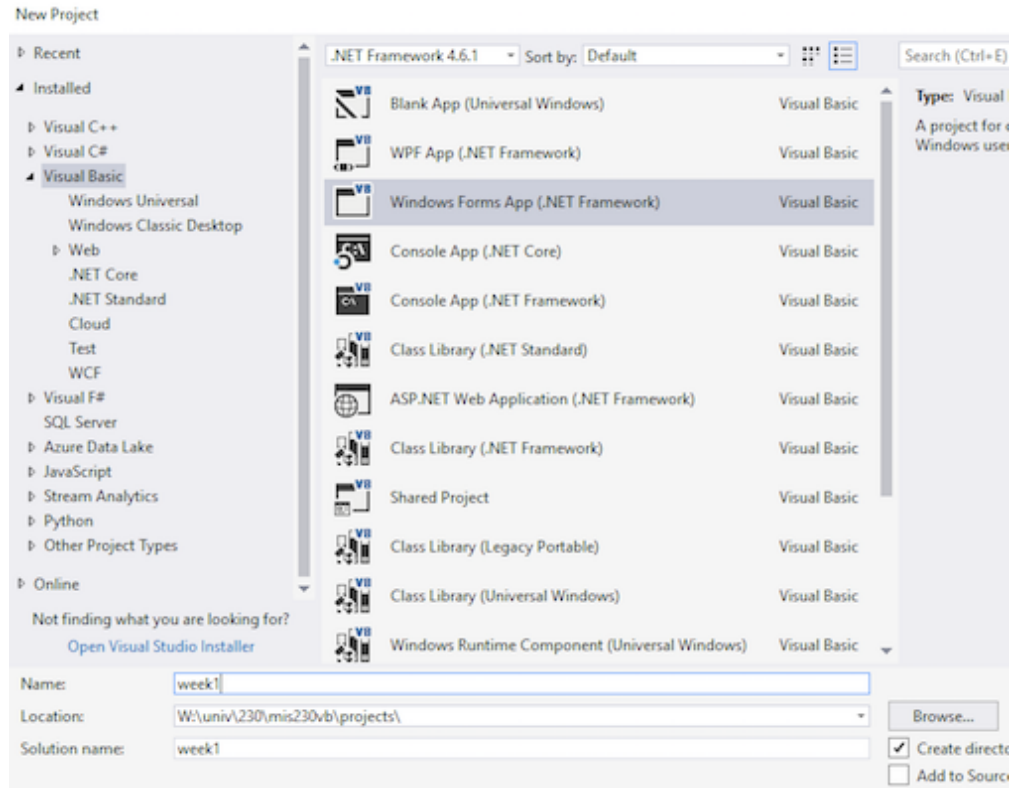
3 Starting Projects in VS

- You create programs by creating projects in VS
- A Project is just a directory containing the source files and configuration files needed by the



4 Project Types

- You set the program type by selecting the language, and type of project
- Change *Name* to set the name of your project
- ***Location** is where you can find your project directory
- The project directory will have the same name as the project
- You can share your project by simply sharing the directory of the project



5 Types of Projects (Examples)

- Console
- Windows Forms App (.NET Framework)
- ASP.NET Web Application
- ... etc

6 Console

- Simplest form of program, computer doesn't do anything unless you instruct it to
- Used mostly to perform quick tasks and create server applications
- User interaction is not expected
- Will perform instructions sequentially
- Use this to get quick output from the computer

7 Windows Forms App (.NET Framework)

- A Windows based application the uses GUI for user interaction
- Event driven
- Making the window, buttons, and controls is handled for you
- You tell the computer what to do if an event occurs
- An event example is a button click or text input

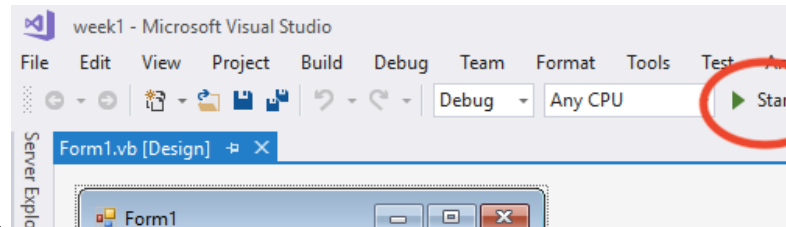
8 ASP.NET Web Application

- A browser based application
- Expected to run on a web server
- Client/Server Request/Response model
- Not covered this course

8.1 Keep in mind that programs come in different forms!

9 Our First Windows Forms Program

- Create a *Windows Forms* project named "HelloWorld"



- Start the program by clicking the *Start* button.

9.1 Can you explain what is happening?

10 Our Second Program

- Create a **Console** project and name it **HelloConsole**
- What is different about this project?
- Start the project and see what happens.
- Can you explain what is going on?

11 The Console Program

```
Module Module1
```

```
    Sub Main()
```

```
    End Sub
```

```
End Module
```

12 The Console Program Cont.

- The program does nothing!
- **Main** is the starting point for the program
- All instruction you write inside **Main** and before **End Sub** is executed
- Nothing else is done by the computer
- Let's tell the computer to type something for us

13 Updated Console Program

```
Module Module1
```

```
    Sub Main()  
        Console.Out.WriteLine("Hello World!")  
    End Sub
```

```
End Module
```

Run it and explain what happend

14 Updated Console Program

```
Module Module1
```

```
    Sub Main()  
        Console.Out.WriteLine("Hello World!")  
        Console.In.ReadLine()  
    End Sub
```

```
End Module
```

Now What is happening?

15 The Statements

15.1 Console.Out.WriteLine

- This statement is used to output text
- You put the text inside the parantheses and the computer will display it

16 The Statements

16.1 Console.In.ReadLine

- This is an input statement
- It tells the computer to wait for the user to type some text on the keyboard
- Then what will it do with the text?

17 Challenge

- Try to display what the user types in the keyboard as output
- Can you add a message to let the user know what he/she needs to do?

18 Variables

- When getting input, we need to hold it somewhere to be able to use it
- This is where variables come in
- A Variable is like a bucket that holds data/input so we can use it elsewhere in the program
- Any input that is not placed in a variable is lost!

19 Using Variables in VB

1. You have to declare it first
2. You assign the input/data/result to the variable

20 What is a Declaration?

```
Dim myText as String
```

- **Dim** stands for **declare in memory**
- Tells the computer to keep some space in memory to store data in of a specific type (called **data type**), what is it here?
- Gives it a name so we can refer to it, what is it here?
- Where do you think we should place this statement in the program?

21 What is an Assignment?

```
myText = Console.In.ReadLine()
```

- Store whatever is on the right of the = operator, in the variable on the left
- Where should we place this statement?
- What do you think is the type of data returned by **ReadLine**?
- Data types must match or can be converted.

22 The Updated Program

- No need to show the sub main part:

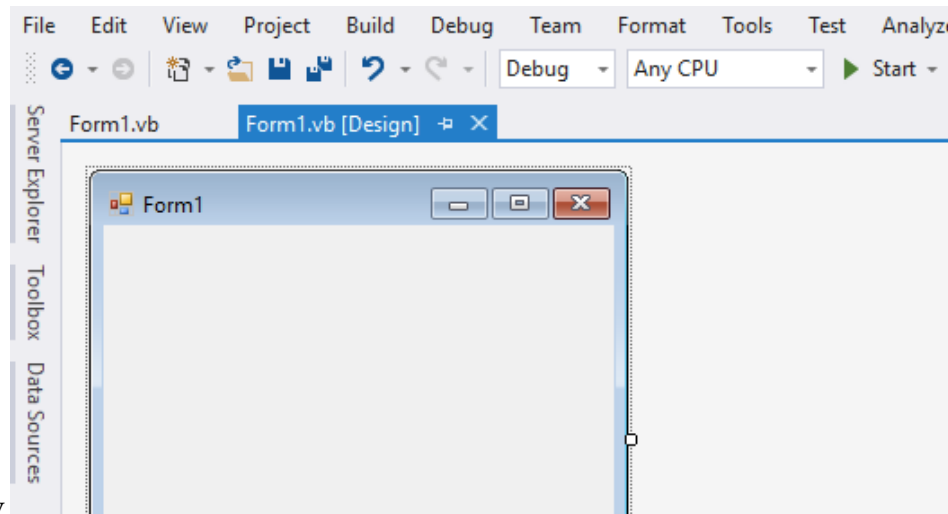
```
Dim myText As String
Console.Out.WriteLine("Please type message:")
myText = Console.In.ReadLine()
Console.Out.WriteLine(myText)
Console.In.ReadLine()
```

23 Review of Concepts

- Console vs Windows Forms vs Web apps
- Project creation and execution
- Variables and data types
- Declaration and assignment

24 Hello World in Windows Forms

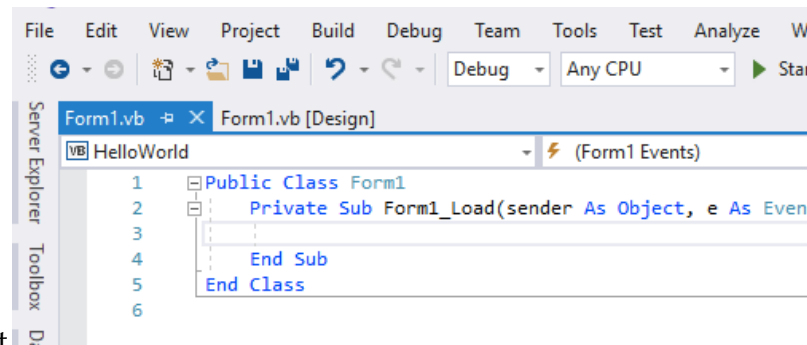
- Open your old windows form project, or create a new one named **Hello World**
- Look at form1



- This is known as the design view

25 Hello World in Windows Forms

- Double click anywhere on Form1



- This will open the coding view of the project

26 What is Form1_load?

- This is known as an event sub
- Remember **Main()** from console projects?
- Form1_load is similar to it
- It is executed only when Form1 is loaded
- This is known as an event sub and it is part of event based programming

27 What is Form1_load?

- In event based programming you do not execute the execution of the application
- The windows, buttons, forms, and all other GUI objects are already built for you

- Instead, as a programmer, you write subs that are executed when the user does specific actions

28 What is Form1_load?

- For example, in Form1_load, the user starts the application, and Form1 is loaded
- The event sub always consists of two parts, the name of the object/control, and the action that the user must do for the program to execute this code

29 Displaying Messages

- MsgBox can be used to display a message notification for the user in a windows form application
- Let's add a message to display "hello world" when form1 is loaded
- Double click on Form1 in design view, then inside the form1_load sub, type:

```
MsgBox("Hello World!")
```

30 Hello World in Windows forms

Your code should look like this:

```
Public Class Form1
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        MsgBox("Hello World!")
    End Sub
End Class
```

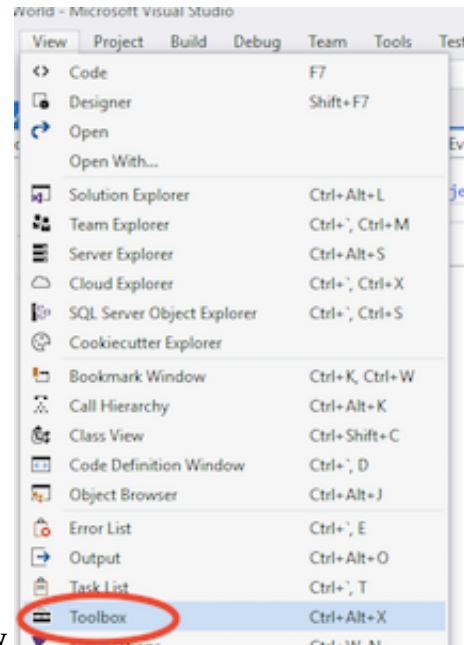
30.1 Now start your program!

31 The running program

- Did you notice what MsgBox did?
- How is hello world different in windows forms compared to console

32 What Are Controls?

- they are all the graphical components that you can use to build an application



- Select Toolbox from View Menu while in **design view**

33 The Controls

You can place these in forms to design the application by dragging them

Controls might perform input or output or both

Double click the control to override the main behavior

Hover over control to know what it is

34 The Controls

Use properties window to discover what properties are available

Use it to modify appearance and behavior of control

name and text are two of the most frequently used properties

35 The Button Control

- Let's add a button to the project
- Go to the design view
- Drag a button from the toolbox and place the button anywhere on Form1
- Double click on the button in the design view

36 Your Code Now

```
Public Class Form1
```

```
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
        MsgBox("Hello World!")
```

```
    End Sub
```



```

        Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        End Sub
End Class

```

36.0.1 What Changed?

37 Button1_click

- This is the button1 click event sub
- When do you think this is executed?
- Add a statement to display a msgbox with the message:

The Button Has Been Clicked!

38 The Solution

```

Public Class Form1
    Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
        MsgBox("Hello World!")
    End Sub

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        MsgBox("The Button Has Been Clicked!")
    End Sub
End Class

```

39 Moving Forward

- I will not display complete code from the code view
- Instead, I will do the following:

```

    Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
        MsgBox("The Button Has Been Clicked!")
    End Sub

```

- Do you think there is any difference if Button1_Click came before Form1_load? Try moving them

40 Challenge

- Is the MsgBox an Input or Output control?
- Is the Button an Input or Output control?
- When do we use it?
- Try adding 2 more buttons
- For each button display a different message