

Introduction to C#.NET

by Derek Peacock

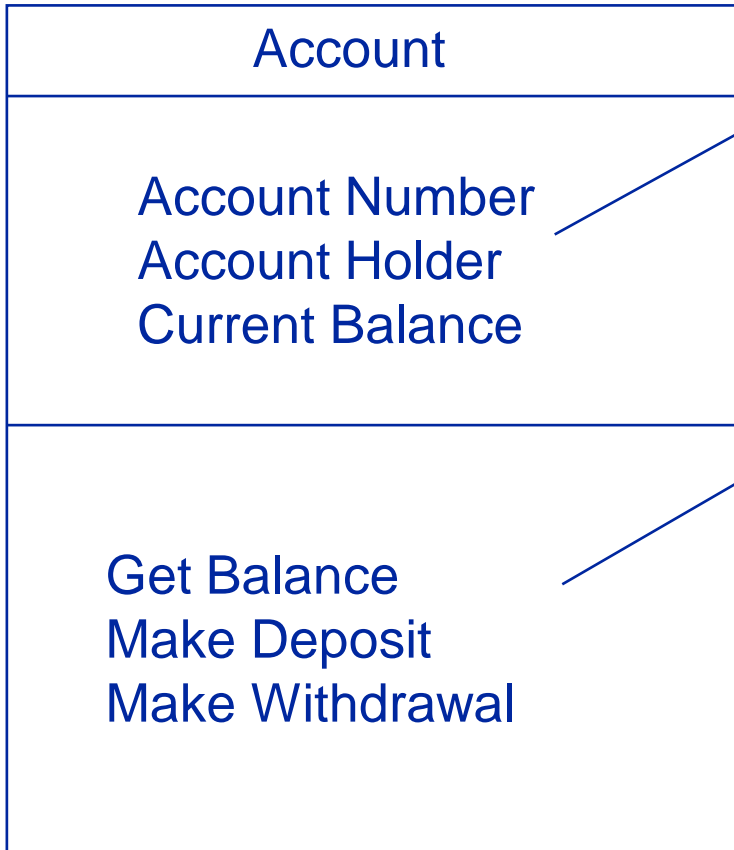


Classes & Objects

- ☞ A program is a set of interacting classes
- ☞ A class contains methods and properties
- ☞ A class represents one specific object in the problem domain (e.g. Car, Student, Product)
- ☞ Classes are used as a blueprint for creating objects which are used when the program runs.



Example Class



Private Attributes

Account.CurrentBalance = £1,000,000

Public Methods

balance = Account.GetBalance();



Programming in C#.NET

```
namespace ConsoleCsExamples
{
    class Program
    {
        static void Main(string[] args)
        {
            ConvertToEuros();

            /// <summary>
            /// A simple method to ask the user to enter an
            /// amount in pounds (GBP) and convert it to
            /// Euros (EUR) using the current exchange rate
            /// </summary>
            static void ConvertToEuros()
            {
                const decimal ExchangeRate = 1.24710M;

                decimal pounds; // GBP
                decimal euros; // EUR
            }
        }
    }
}
```

Class Program

Executable statements

Function (Method)

Variables (Attributes)



Program Errors

☞ Syntax errors

- ▮ Mistakes in using the language
- ▮ Program will not execute (run)
- ▮ Mistakes indicated by wavy underline

☞ Semantic errors

- ▮ Mistakes in the meaning or logic
- ▮ Program executes but does wrong thing.
- ▮ Program might crash (fatal error)



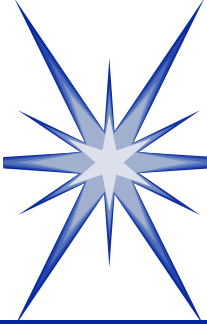
C# Solutions

☞ Solutions

- ☐ Name of solution = name of main folder
- ☐ Contains 1 or more projects

☞ Projects

- ☐ Contain one or more classes
- ☐ Might contains one or more forms or web pages



Visual Studio Projects

☞ Store Apps
(WPF)

☞ Windows Forms

☞ Class Library

☞ Web Forms

☞ MVC

☞ Windows Azure

☞ MS Office

☞ SharePoint

☞ Reporting

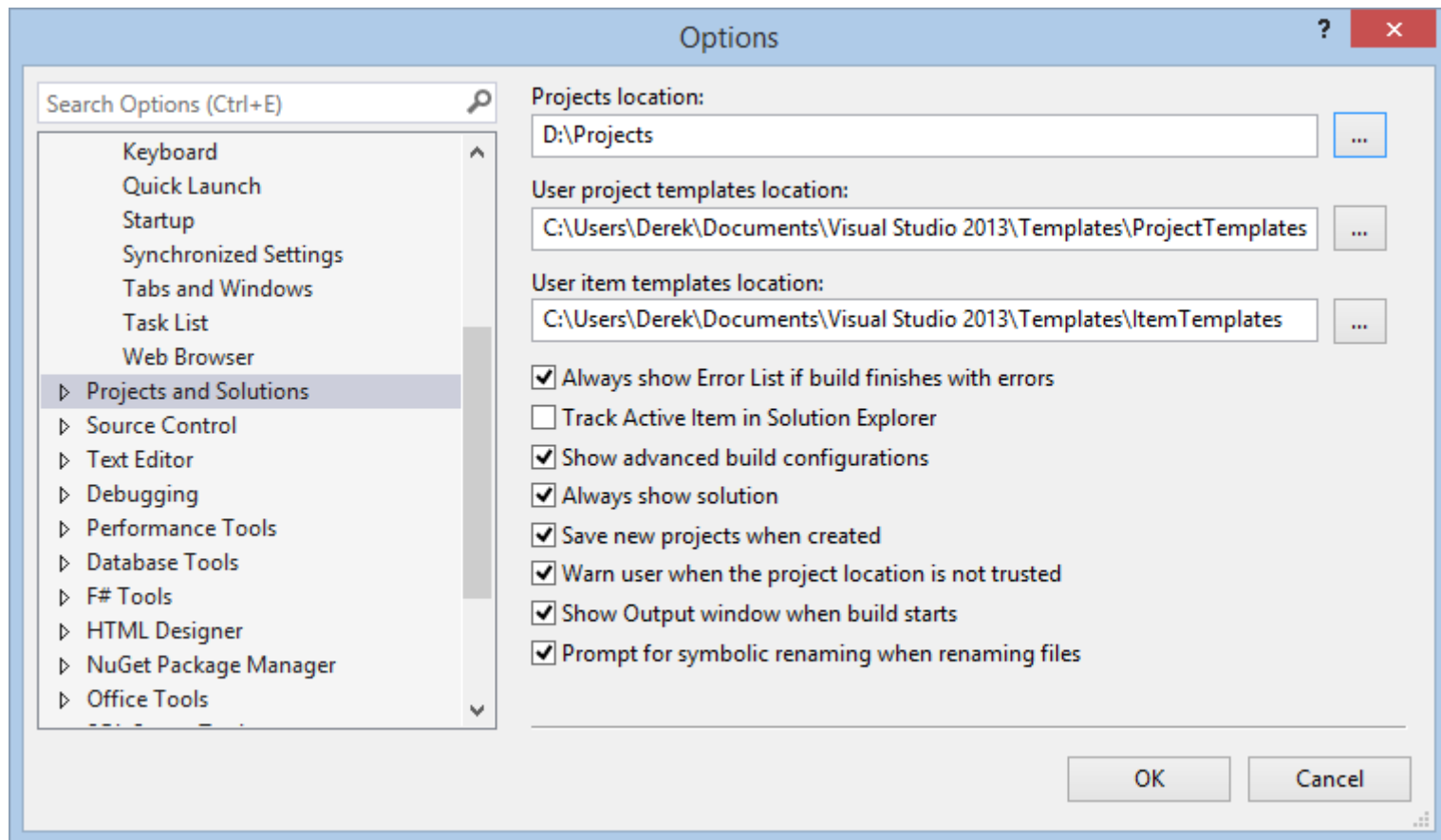
☞ SilverLight

☞ Test Projects

☞ WCF

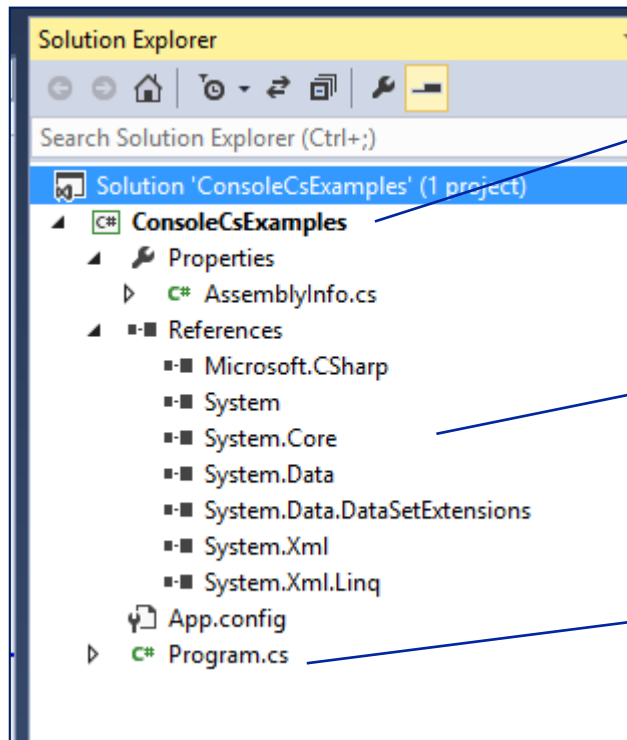


Visual Studio Main Folder





Solution Explorer



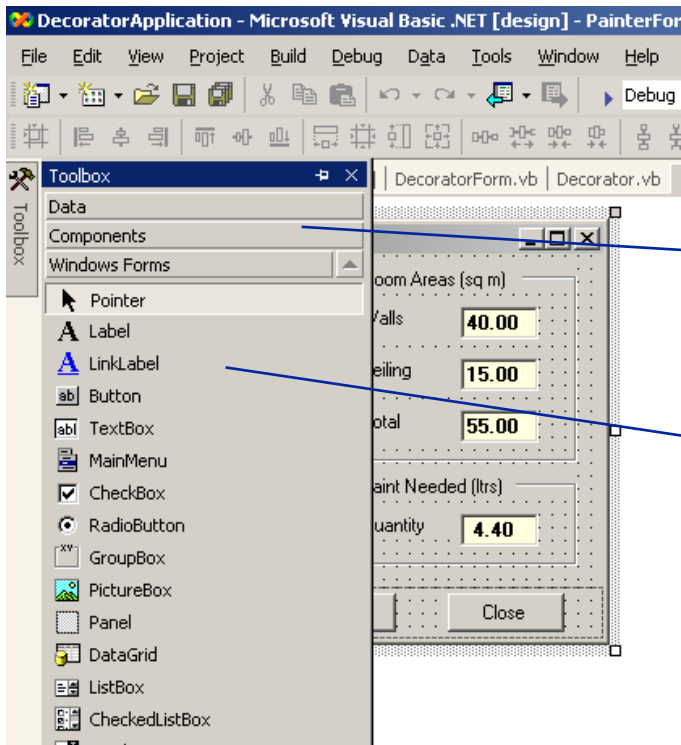
ConsoleCsExamples
Solution

Libraries
(Packages)

Program Class
(File)



Toolbox of GUI Controls



Other controls

Windows GUI Controls



Form Control Properties

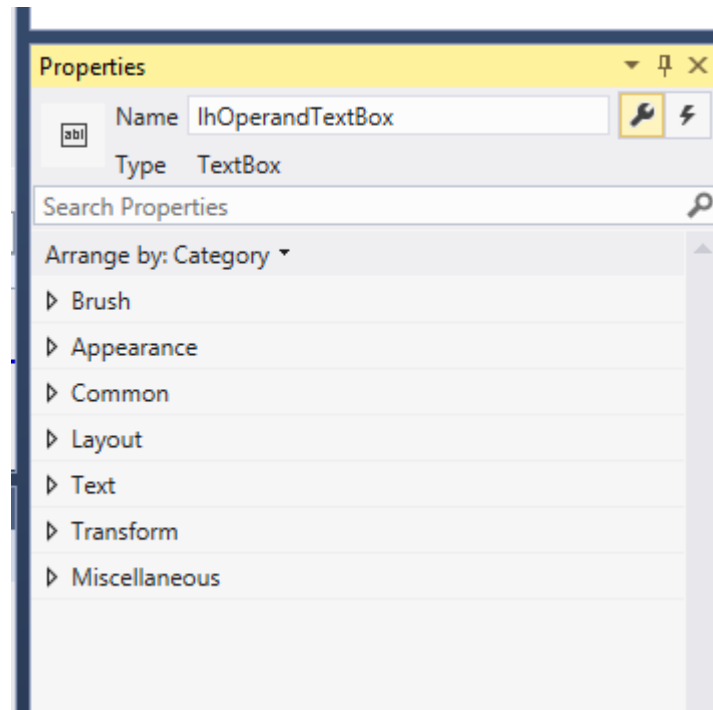
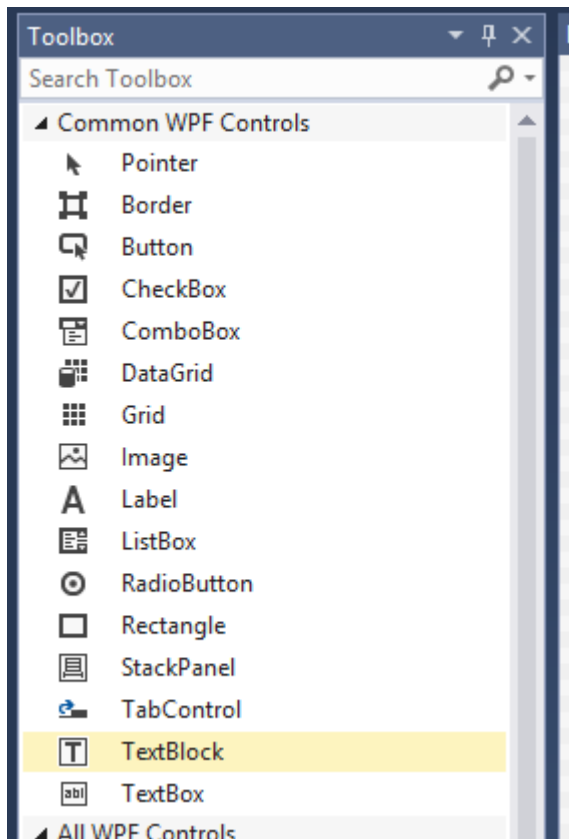
Property	Value
BackColor	LightYellow
BorderStyle	Fixed3D
Cursor	IBeam
Font	Microsoft Sans Serif, 8.25pt
ForeColor	WindowText
Lines	String[] Array
RightToLeft	No
ScrollBars	None
Text	40.00
TextAlign	Left

Text
The text contained in the control.

Selected control
(a text box)

Text property
(value set to "40.00")

WPF Controls



HTML Controls

- HTML
 - Pointer
 - Input (Button)
 - Input (Reset)
 - Input (Submit)
 - Input (Text)
 - Input (File)
 - Input (Password)
 - Input (Checkbox)
 - Input (Radio)
 - Input (Hidden)
 - Textarea
 - Table
 - Image
 - Select
 - Horizontal Rule
 - Div

Properties	
	
(id)	
accesskey	
alt	
aria-activedescendant	
aria-atomic	False
aria-autocomplete	none
aria-busy	False
aria-checked	undefined
aria-controls	
aria-describedby	
aria-disabled	False
aria-dropeffect	none
aria-expanded	undefined
aria-flowto	
(id)	



GUI - Code Only

```
@ViewBag.Message</h2>  
- <p>  
  This site has been re-mounte  
  not working even though it i  
</p>  
- <p>  
  The Valerian Project Managem  
  any kind of project (profess
```



References

☞ “Microsoft Visual C# Step by Step”
by John Sharp (2014)
Microsoft Press £19 (£14 Kindle)

☞ Chapter 1