Visual C# .Net using framework 4.5

Eng. Mahmoud Ouf

Lecture 01

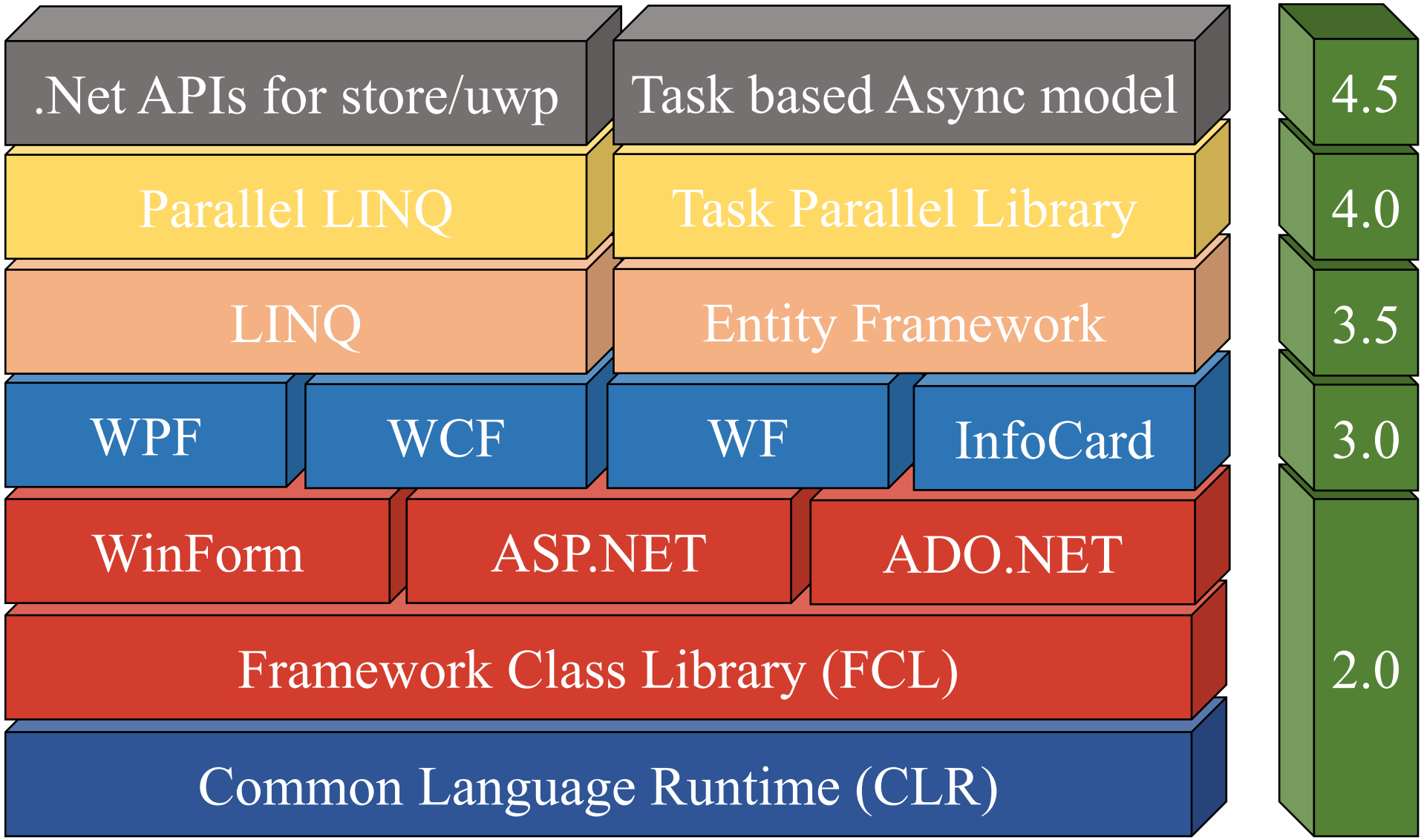
Introduction to .Net Framework

- The development of .Net Framework started the late 1990.
- First version was released early 2002.
- Through the 17 years it passes 8 upgrades
- Some of these upgrades was released with new version of Visual Studio.Net.

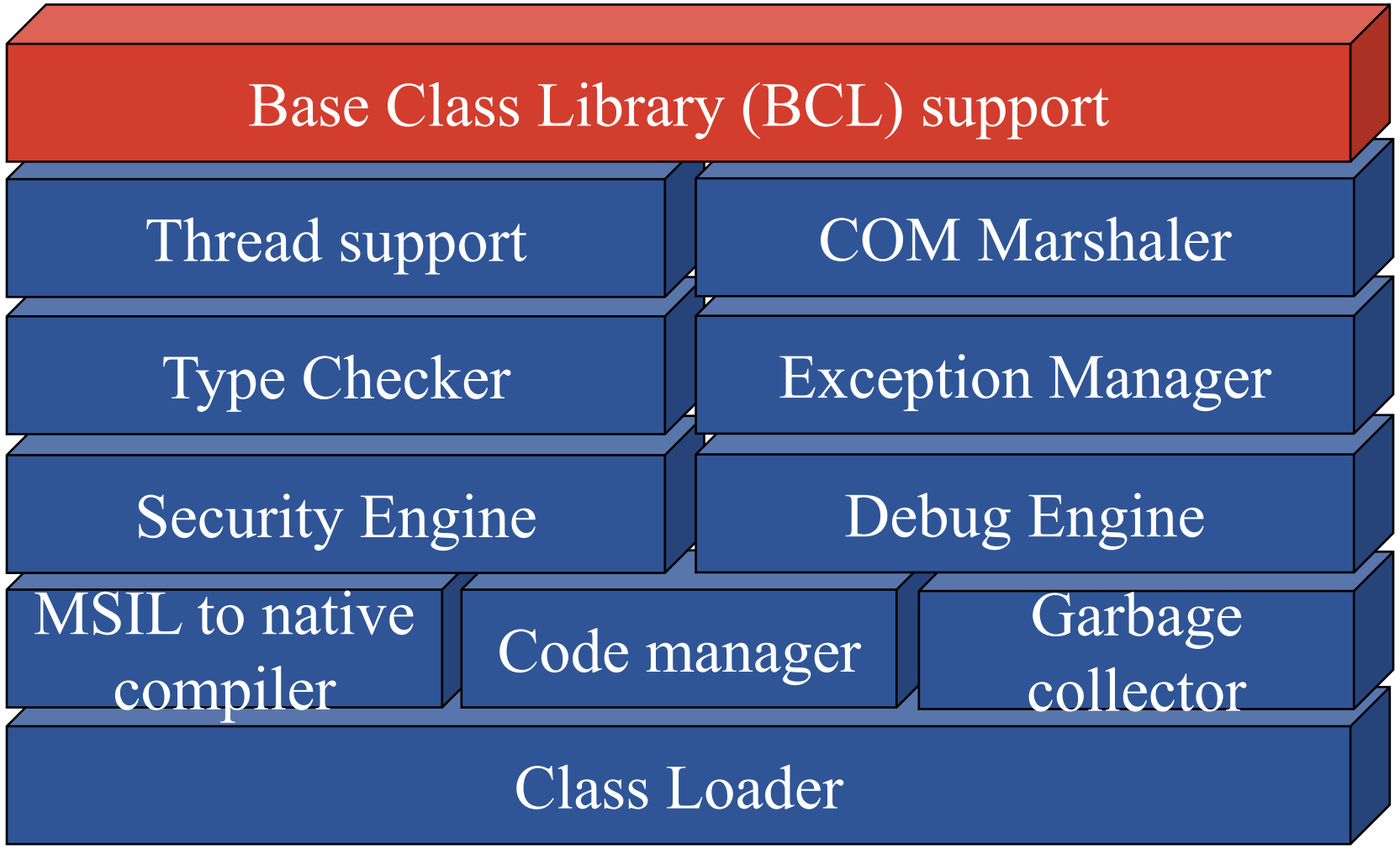
Introduction to .Net Framework

Framework version	Release date	Development tool
1.0	13/02/2002	Visual Studio .Net 2002
1.1	24/04/2003	Visual Studio .Net 2003
2.0	07/11/2005	Visual Studio .Net 2005
3.0	06/11/2006	Visual Studio .Net 2005
3.5	19/11/2007	Visual Studio .Net 2008
4.0	12/04/2010	Visual Studio .Net 2010
4.5	15/08/2012	Visual Studio .Net 2012
4.6	20/07/2015	Visual Studio .Net 2015

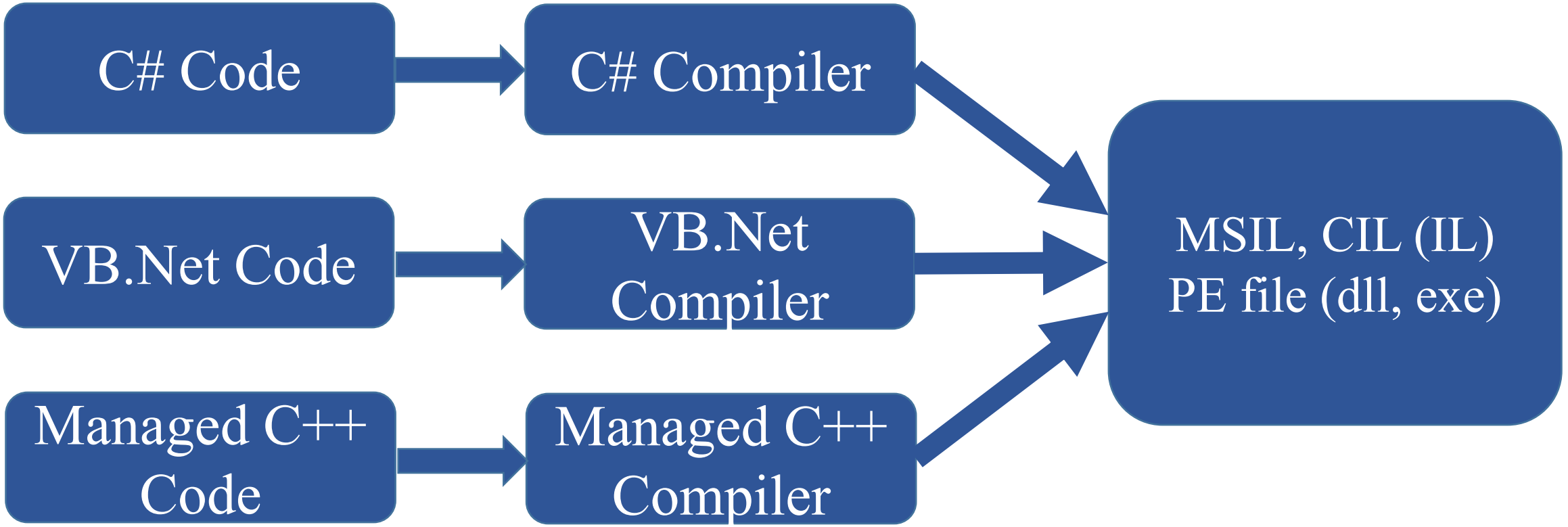
Overview of .Net Framework



Common Language Runtime (CLR)

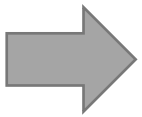


From source code to executable



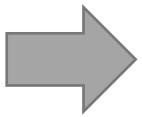
.Net Framework 1.1

1. Built in support for mobile ASP.Net controls
2. Enables Code Access Security in ASP.Net application
3. Built-in support for ODBC and Oracle Database
4. .Net Compact Framework
5. Support Internet Protocol version 6 (IPv6)



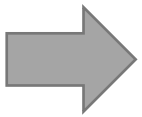
.Net Framework 2.0

1. Full 64-bit support
2. Numerous API changes
3. Microsoft SQL Server integration
4. Additional and improved ASP.Net web controls
5. New personalization features for ASP.Net
6. Partial classes
7. Nullable types
8. Anonymous methods



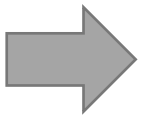
.Net Framework 3.0

1. Windows Presentation Foundation (WPF)
2. Windows Communication Foundation (WCF)
3. Windows Workflow Foundation (WF)
4. Windows CardSpace



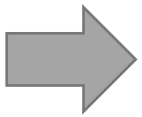
.Net Framework 4.0

1. Parallel Extension to improve support of parallel programming
2. New Visual basic and C# features
3. Include new types
4. Introduced Common Language Runtime (CLR) 4.0



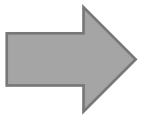
.Net Framework 4.5

1. .Net for Metro Style apps
2. Managed Extensibility Framework (MEF)
3. Core Features
4. ASP .Net
5. Networking



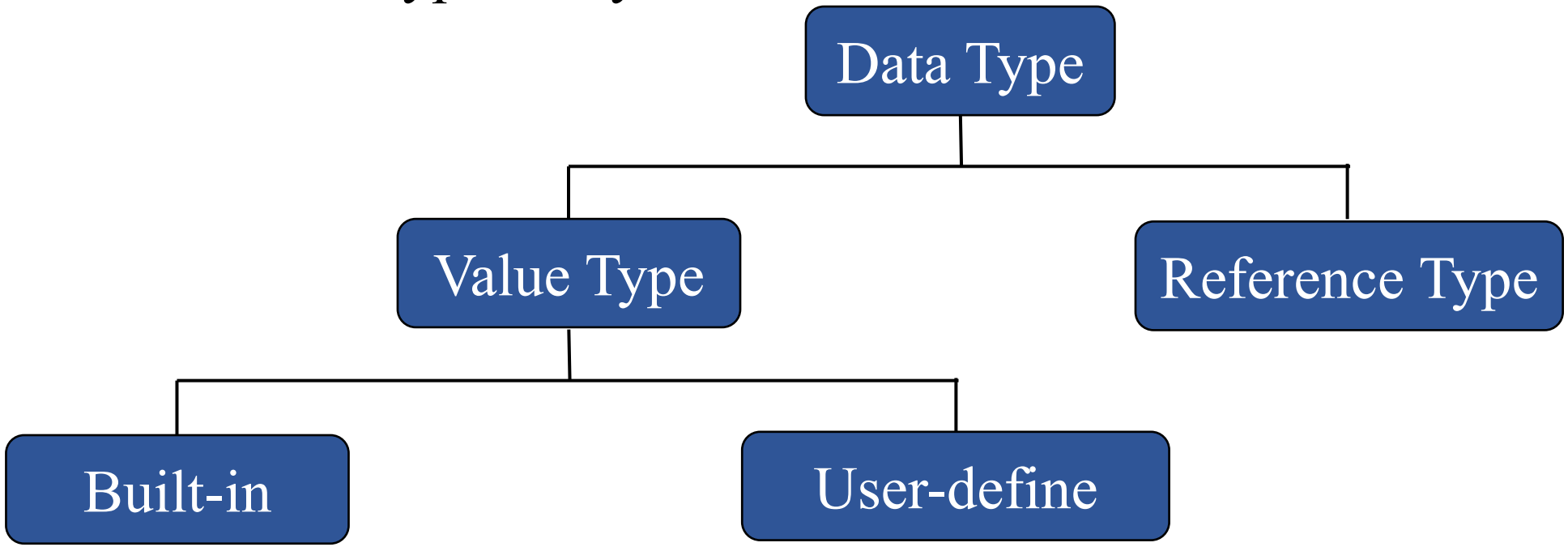
.Net Framework 4.6

1. Just In Time (jit) compiler for 64 bit system
2. WPF and Windows Forms both have been updated for high DPI scenarios
3. Support for TLS 1.1 and TLS 1.2 has been added to WCF ASP .Net
4. The cryptographic API in .NET Framework 4.6 uses the latest version of Windows CNG cryptography API.



Common Type System (CTS)

The CLR include Common Type System (CTS) that defines a set of built in data type that you can use.



Common Type System (CTS) (cont)

Value Type

The value type variables directly contain their data.

Reference Type

The reference type variable contains reference to their data. The data is stored in an object. It is possible for 2 reference type variables to reference the same object.

Note:

All data types are defined in System namespace.

All data types are derived from System.Object

Value types are derived from System.ValueType.

System.Object

System.ValueType

All value types directly contains data, and they can't be null

Common Type System (CTS) "Built-in"

Category	Class name (type in runtime)	C# data type	Range	Description
Integers	Byte	byte	0 to 255	8 bit unsigned integer(1byte)
	sByte	sbyte	-128 to 127	8 bit signed integer(1byte)
	Int16	short	-32,768 to 32,767	16 bit signed integer(2bytes)
	Int32	int	-2,147,483,648 to 2,147,483,647	32 bit signed integer(4bytes)
	Int64	long		64 bit signed integer(8bytes)
	UInt16	ushort	0 to 65,535	16 bit unsigned integer(2bytes)
	UInt32	uint	0 to 4,294,967,295	32 bit unsigned integer(4bytes)
	UInt64	ulong		64 bit unsigned integer(8byte)
Floating Points	Single	float	$1.5 * 10^{-45}$ to $3.4*10^{38}$	A single-precision (32-bit) floating-point number.
	Double	double	$5.0*10^{-324}$ to $1.7*10^{308}$	A double-precision (64-bit) floating-point number.
	Boolean	bool	true/false	
	Char	char		A Unicode (16-bit) character

Creating User-Defined Data Types:

Enumeration

Enumerators are useful when a variable can only have a specific set of values.

Defining : enum Color {Red, Green, Blue}

Using : Color ColorPalette = Color.Red; OR

Color ColorPalette = (Color) 0;

Console.WriteLine("{0}", ColorPalette);

Note :

The enumeration element are of type int and the first element has a value of 0, each successive element increase by 1. You can change the behavior as follow:

```
enum Color {Red = 102, Green, Blue};        //Green = 103, Blue = 104
enum Color {Red = 102, Green = 10, Blue = 97};
```