Introduction to .NET Framework

Introduction**- The **.NET Framework** is a ***software development platform that was introduced by ***Microsoft*** in the late **1990** on ***13th February 2002***, Microsoft launched the first version of the .NET Framework, referred to as the **.NET Framework 1.0**.

-Latest version of .NET framework is **.NET 4.8** that was released on **18 April 2019.**

-.NET stands for ***Network Enable Technology***.

- ***Framework in a programming is a tool that provides readymade components used to speed up S/W development***.

-**.NET Framework *is a virtual machine for compiling and executing programs written in different languages like C#, VB.Net, J#, F#,C++ etc.***

-***It is also used to create a form based, console-based, mobile and web-based application or services that are available in Microsoft environment.***

-Furthermore, the .NET framework is ***a pure object oriented***, that similar to the Java language. But it is ***NOT a platform independent*** as the Java. So**, its application runs only to the windows platform.**

- .NET framework supports **more than 60 programming languages** in which **11 programming languages** are designed & developed by **Microsoft**.

1. C#.NET
2. VB.NET
3. C++.NET
4. J#.NET
5. F#.NET
6. JSCRIPT.NET
7. WINDOWS POWERSHELL
8. IRON RUBY
9. IRON PYTHON
10. C OMEGA
11. ASML(Abstract State Machine Language

* .NET Framework Architecture-

The first three components from bottom are considered as the basic architecture of .NET framework and after this more components were added by Microsoft in the .NET Framework,

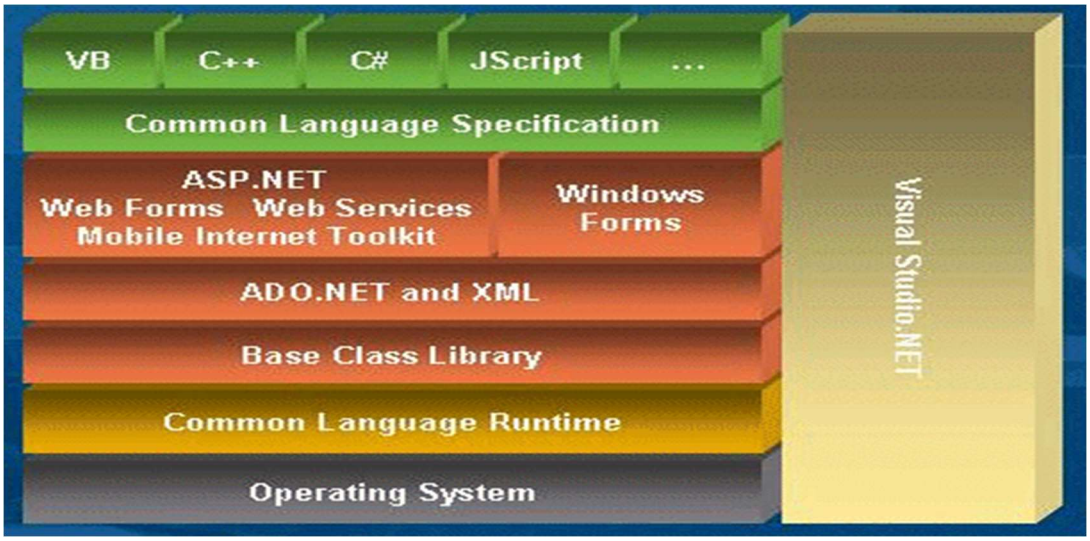


Fig- ***.NET Framework Architecture-***

Components of .NET Framework

There are following components of .NET Framework:

1. CLR (Common Language Runtime)
2. CTS (Common Type System)
3. BCL (Base Class Library)
4. CLS (Common Language Specification)
5. MSIL/CIL (Microsoft Intermediate Language/Common Intermediate Language)
6. FCL (Framework Class Library)
7. Garbage Collector.

1) CLR (Common Language Runtime)-

- CLR is heart and soul of .NET framework.

- Used to Execute programs which are written in different languages like C#, VB.Net, J#, F#,C++ etc.

***CLR provides a number of services that includes:**

1. Loading and executing of programs
2. Memory isolation for application
3. Memory management using automatic garbage collection
4. Providing metadata***(basic information about data)***
5. Verification of type safety
6. Security

7. Interoperability with other system
8. Compilation of IL into native executable code
9. Managing exception and errors

2) CTS (Common Type System)

CTS is built in CLR which provides Multiple Language Support.

It specifies a standard that represent what type of data and value can be defined and managed in computer memory at runtime.

A CTS ensures that programming data defined in various languages should be interact with each other to share information. For example, in C# we define data type as ***int***, while in VB.NET we define ***integer*** as a data type hence it does not require any type conversion.

3) BCL (Base Class Library)

The base class library has a rich collection of libraries features and functions that help to implement many programming languages in the .NET Framework, such as C #, F #, Visual C ++, and more. Furthermore, BCL divides into two parts:

1. **User defined class library**
    - **Assemblies –*it is a File which is automatically generated by compiler upon successful compilation***.
    - ***It is the collection of small parts of deployment an application's part.***
    - It contains either the ***.DLL (Dynamic Link Library) or .exe (Executable) file.***
2. **Predefined class library**
    - **Namespace** - ***It is the collection of predefined classes and methods that present in .NET.***
    - In other languages such as, C we used header files, in java we used package similarly we used "using system" in .NET, where ***using*** is a keyword and ***system*** is a namespace.

4) CLS (Common Language Specification)-

CLS is a subset of CTS. ***It defines a set of rules and restrictions that every language must follow which runs under the .NET framework.*** The languages which follow these set of rules are said to be ***CLS Compliant***. In simple words, **CLS enables cross-language integration or Interoperability.**

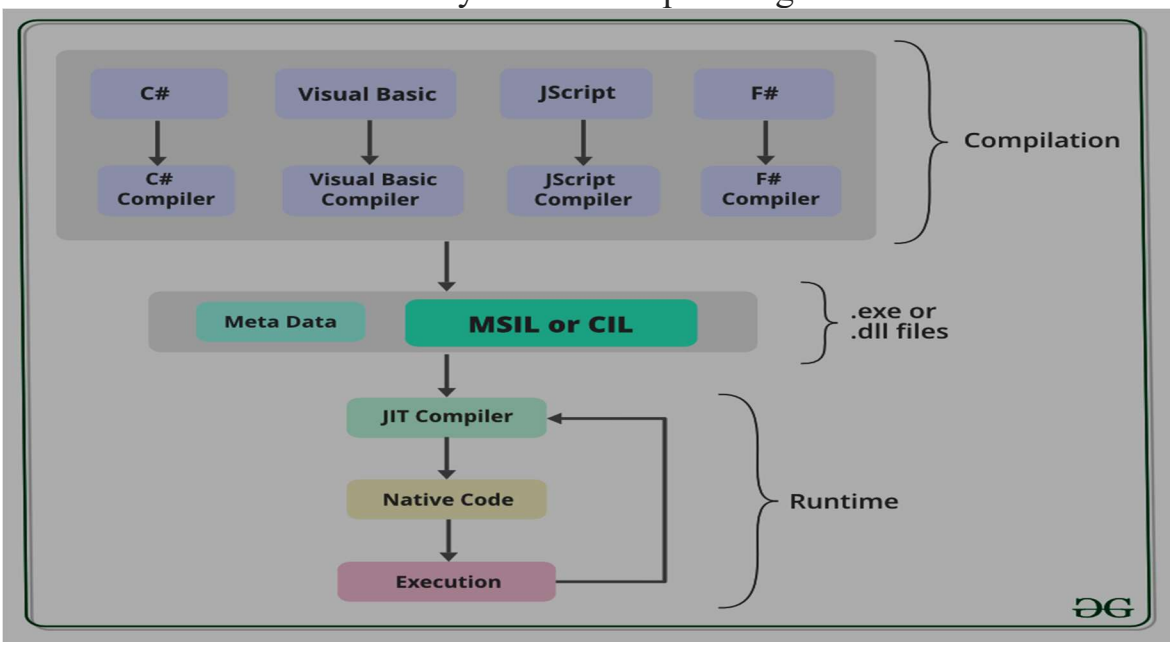
Ex. if we talk about C# and VB.NET then, in **C#** every statement must have to end with a semicolon. it is also called a statement Terminator, but in **VB.NET** each statement should not end with a semicolon(;).

5)MSIL/CIL(MicrosoftIntermediateLanguage/Common Intermediate Language) –

The Microsoft Intermediate Language (MSIL), also known as the Common Intermediate Language (CIL) is a set of instructions that are platform independent and are generated by the language-specific compiler from the source code.

The **MSIL** is platform independent and consequently, it can be executed on any of the Common Language Infrastructure supported environments such as the Windows .NET runtime.

The execution process that includes the creation of the MSIL and the conversion of the MSIL into machine code by the JIT compiler is given as follows:



- ***The source code is converted into the MSIL by a language-specific compiler in the compile time of the CLR. Also, along with the MSIL, metadata is also produced in the compilation. The metadata contains information such as the definition and signature of the types in the code, runtime information, etc.***
- ***A Common Language Infrastructure (CLI) assembly is created by assembling the MSIL. This assembly is basically a compiled code library that is used for security, deployment, versioning, etc. and it is of two types i.e. process assembly (EXE) and library assembly (DLL).***
- ***The JIT compiler then converts the Microsoft Intermediate Language (MSIL) into the machine code that is specific to the computer environment that the JIT compiler runs on. The MSIL is converted into the machine code on a requirement basis i.e. the JIT compiler compiles the MSIL as required rather than the whole of it.***
- ***The machine code obtained using the JIT compiler is then executed by the processor of the computer.***

6) FCL (Framework Class Library)-

The **Framework Class Library** or **FCL** is a superset of **BCL** ***which provides the system functionality in the .NET Framework as it has various classes, data types, interfaces, etc. to perform multiple functions and build different types of applications such as desktop applications, web applications, mobile applications,*** etc.

7) Garbage Collector

Garbage Collector is responsible for automatic memory management of objects. Also, the responsibility of the garbage collector is removing the object from memory that has no use.

Advantages of Garbage Collector

- Allows us to develop an application without having worry to free memory.
- Allocate memory for object efficiently on the managed Heap.
- Provide memory space by making sure that an object cannot use the content of another object