**.Net Framework**

**Introduction**

The .Net framework is a software development platform developed by Microsoft.

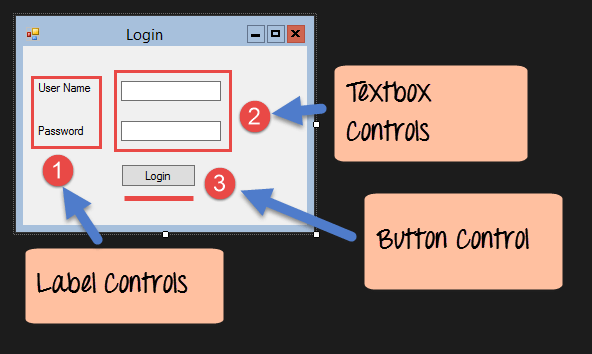
Microsoft .NET Framework provides a common platform to Execute or, Run the applications developed in various programming languages like C#, VB.Net etc. So, developers can select the language to develop the required application.

It is used to develop Form-based applications, Web-based applications, and Web services.

It is used to build applications for Windows, Mac,Linux,phone and web etc.

**Example of a simple Windows form application**.

It shows a simple Login screen, which is accessible by the user. The user will enter the required credentials and then will click the Login button to proceed. This is a collection of label controls which are normally used to describe adjacent controls.



**Web applications** include online forms, shopping carts, word processors, spreadsheets, video and photo editing, file conversion, file scanning, and email programs such as Gmail, Yahoo. Popular applications include Google Apps.

**Example of web services**

Amazon provides a web service that provides prices for products sold online via amazon.com

**.NET Components**

The architecture of the .Net framework is based on the following key components;

**1. Common Language Runtime**

The "Common Language Infrastructure" or CLI is a platform on which the .Net programs are executed.

In other words, .NET CLR is a run-time environment that manages the execution of code and provides services that make the execution of code easier.

“Runtime” means that code is running which means that code is being executed.

“Common Language” means that this runtime manages the execution of code written in several languages that share the services provided.

We call the code that uses the CLR managed code.

**2. Common Type System**

The Common Type System defines the types of data that managed code can use.

**3. Common Language Specification**

A Common Language Specification (CLS) defines the features that every language for developing managed code must provide. Programmers who use features in the common language specification can build an application combining programs in different languages.

**4. Base Class Library**

The Class Library is a comprehensive, object-oriented collection of reusable types

These class libraries can be used to develop applications that include:

Traditional command-line applications

Graphical user interface (GUI) applications

**5. .NET Framework Class Library**

The .NET Framework Class Library provides a large and very useful set of types that accelerate the development process. The library groups the types in namespaces that combine related types.

**Some namespaces from the library that we use are**

System -> Contains fundamental types

System.Collections -> Defines various collections of Objects

System.data -> Manages data from multiple sources including databases

System.Drawing -> Provides graphics

System.IO -> Allows reading and Writing

System.Net -> Provides an interface that computers use to communicate over networks

System.Runtime.Remoting ->Supports distributed applications

System.Text -> Handles Character encoding

System.Threading -> Enables multithreaded programming

System.Web -> Enables browser- Server Communication

System.Web. Services -> Enables the building and use of web services

System.Windows.Forms -> For user interfaces in Windows-based applications

System.Xml -> Provides support for processing XML

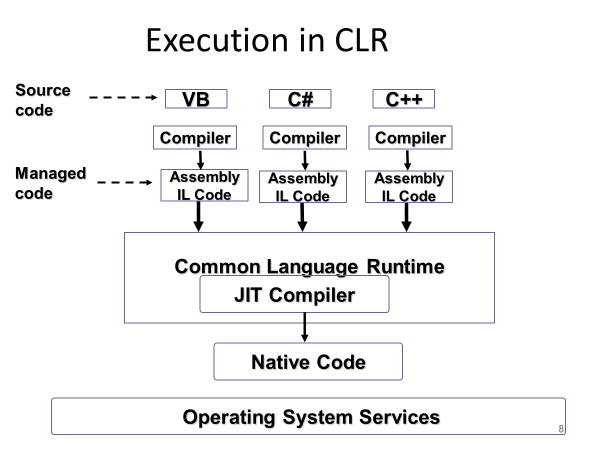
**Intermediate Language and Just in -Time Compilation**



The first step in the managed-code process is to compile a .NET project to IL (Intermediate Language), which also generates the required metadata.

At execution, a JIT (Just in -Time Compilation) compiler translates the IL code to native machine code.

During the runtime the CLR uses a JIT compiler to compile the MSIL code to native code for the device used.



**Garbage Collection**

Garbage collection means that memory is automatically managed. We instantiate and use objects, but we do not explicitly destroy them. The CLR takes care of releasing the memory used by objects when they are no longer referenced or used. The CLR's automatic garbage collection solves the problem of memory leaks.



Memory leak means failure to release memory after allocation.



**Microsoft .Net Assembly**



It is a logical unit of code, that contains code which the Common Language Runtime (CLR) executes. It is the smallest unit of deployment of a .net application and it can be a .dll or an exe. Assembly is really a collection of types and resource information that are built to work together and form a logical unit of functionality. It includes both executable application files that we can run directly from Windows without the need for any other programs (.exe files), and libraries (.dll files) for used by other applications.

