**What is DotNet Framework?**

It is defined as platform which supports various Languages (C#, VB, F#...), Technologies (ASP.NET) and Databases (SQLSERVER, ORACLE...) in order to develop network and distributed applications. An user can develop, execute and deploy an application in a single framework.

There are two main components namely CLR (Common Language Runtime Environment) and BCL (Base Class Library)/FCL (Framework Class Library).

**What is MSIL?**

MSIL stands for **Microsoft Intermediate Language**. Generally, execution of a DotNet application happens in two steps.

Of which,

1). High Level Language is converted into **Intermediate Language** (IL) using the corresponding Language Compiler in which the application is developed. This Intermediate Language is known as MSIL.

2.) This MSIL is then converted into native code which is understandable by underlying Operating System and Hardware.

**What is CLR?**

CLR stands for **Common Language Runtime** Environment which is the runtime environment for any DotNet application. When a user installs a DotNet on computer, CLR will automatically get installed.

It is the heart and soul of a DotNet application since without which a user cannot execute or run an application.

It is the interface between DotNet application and Operating System.

In DotNet application, CLR contains JIT (Just in Time) Compiler which converts Intermediate Language into native code which is understandable by particular Operating System on which the application is installed. Another important feature provided by CLR is garbage collection i.e; CLR cleans the objects that are not in use. We, programmers are not responsible for memory management.

**What is JIT?**

JIT Compiler stands for **Just in Time Compiler**.

It resides in CLR and is responsible for conversion of Intermediate Language into native code which is understandable by underlying operating system and hardware.

**What is portability in DotNet?**

In Computer Language, an application is said to be portable when it is executable on different platforms (platform independent).

Coming to DotNet, portability is achieved by CLR (Common Language Runtime Environment) because the JIT Compiler converts any application’s Microsoft Intermediate Language or Intermediate Language into native code which is understandable by underlying operating system and hardware. The main advantage of portability is the developer need not to rewrite the source code for each platform.

**What is GAC?**

GAC stands for **Global Assembly Cache**.

It is a folder in which the assemblies which are commonly used by almost all applications are registered. When a user wants to make use of these assemblies he can directly access those without copying them locally.

**What is the difference between DLL and EXE?**

These are extensions of Assemblies.

Depending on the type of application, .DLL or .EXE will be generated. For example,

.EXE will be generated for **Console** or **Windows** application.

.DLL will be generated For **Class Library** or **Web** application.

**The main differences are:**

|  |  |
| --- | --- |
| **DLL** | **EXE** |
| 1) DLL stands for Dynamic Link Library. | 1) EXE stands for Executable File. |
| 2) DLL is supportive file. It can be reused in other applications. | 2) EXE will generates an ouptut of an application. |
| 3) It does not have Main Method due to which it cannot be executed. It can only be Compiled. | 3) It has Main Method. So, it can be executed and an output will be generated. |