***Assessment of introduction of .net***

**Ques.1 What is the main component in the .Net Framework?**

**Answer** . Net Framework is a platform that provides tools and technologies to develop

Windows, Web and Enterprise applications. It mainly contains two components:

1. Common Language Runtime (CLR)

2. .Net Framework Class Library.

**Common language runtime:**

It provides an environment to run all the .Net Programs. The code which runs under the CLR is called as **Managed Code**. Programmers need not to worry on managing the memory if the programs are running under the CLR as it provides memory management and thread management.

Language Compilers (e.g. C#, VB.Net, J#) will convert the Code/Program to **Microsoft Intermediate Language** (MSIL) intern this will be converted to **Native Code** by CLR.

**.Net Framework Class Library:**

This is also called as Base Class Library and it is common for all types of applications

The following are different types of applications that can make use of .net class library.

1. Windows Application.

2. Console Application

3. Web Application.

4. XML Web Services.

5. Windows Services.

**Common Type System** (CTS)

It describes sets of data types that can be used in different .Net languages in common. CTS ensures that objects written in different .Net languages can interact with each other.

For Communicating between programs written in any .NET complaint language, the types have to be compatible on the basic level.

**Common Language Specification** (CLS)

It specifies a set of rules that needs to be satisfied by all language compilers targeting CLR. It helps in cross language inheritance and cross language debugging.

It describes the minimal and complete set of features to produce code that can be hosted by CLR. It ensures that products of compilers will work properly in .NET environment.

**Ques 2. Current .Net Standard framework and Current core framework?**

**Answer:**Current .net standard framework- 4.8 version

Current core framework- 3.1 version

**Ques 3: Difference between Managed and Unmanaged code?**

**Answer.**

**Managed code:**

1.It is executed by managed runtime environment or managed by the CLR.

2.It provides security to the application written in .NET Framework.

3.Memory buffer overflow does not occur.

4.It provide runtime services like Garbage Collection, exception handling, etc.

5.The source code is compiled in the intermediate language know as MSIL

**Unmanaged code:**

1.It is executed directly by the operating system.

2.It does not provide any security to the application.

3.Memory buffer overflow may occur.

4.It does not provide runtime services like Garbage Collection, exception handling, etc.

5.The source code direclty compile into native langugae.

**Ques4. What do you mean by MSIL Code?**

**Answer:**

**MSIL** stands for Microsoft Intermediate Language. ... During the compile time , the compiler convert the source code into Microsoft Intermediate Language . Microsoft Intermediate Language (MSIL) is a CPU-independent set of instructions that can be efficiently converted to the native code.

**Ques 5:What is the difference between .Net and .Net Core Frameworks?**

## 

## **Answer:**

## The difference between .net and .net core framework are:

**.NET Framework**

1. The .NET Framework is the first implementation of .NET which works on Windows only
2. Its source code is public but Microsoft doesn’t accept third party contributions for it
3. It has a very rich desktop top development framework for windows which include Windows Forms and WPF
4. A huge third-party packages library is also available for it
5. It doesn’t support the in-app deployment model
6. Although it can be used with a docker container, its image size is large and can only be deployed on Windows containers

**.NET Core**

1. .NET Core is the latest implementation of .NET which runs on Windows, Linux, and macOS
2. Its open-source and Microsoft accepts third party contribution to .NET Core
3. It supports desktop frameworks like Windows Forms and WPF from version 3.0
4. The .NET Core also has support for a large number of third party packages as well but still, it doesn’t compete with .NET Framework in this area
5. It does support in-app deployment model
6. It is the best choice to work with docker containers

**Ques 6: Explain the execution process of managed code in .Net?**

**Answer:**

Managed execution process includes the following steps:

* Choosing the right compiler. .Net Framework is a multilanguage execution environment, the runtime supports a wide variety of data types and language features
* Compiling the code to MSIL
* Compiling MSIL to native code
* Execution of Code.