**INTRODUCTION TO .NET EXERCISE**

**Q1**. What is the main component in the .Net Framework? (T)

**Solution:- .net framework has two main components:-**

1. **CLR(Common Language Runtime)-**The common language runtime is the foundation of the .NET Framework. Runtime as an agent that manages code at execution time, providing core services such as memory management, thread management, and remoting,while also enforcing strict type safety and other forms of code accuracy that promote security and robustness.

In fact, the concept of code management is a fundamental principle of the runtime. Code that targets the runtime is known as **managed code**, while code that does not target the runtime is known as **unmanaged code.**

**CLR COMPONENTS:-**

**Class Loader :** The compilers of the framework compile the source code into intermediate code and this code consists of IL code and Meta data. These are contained in a portable executable (PE) file. The class loader is loading this data into the run time.

**Code Manager:** This component is managing the above code during execution. This takes the responsibility of allocating memory to the objects also.

**Garbage collector:** This provides automatic garbage collection of the object when the object Is no longer in use. To achieve this, Garbage collector is performing the periodical checks in the heap from where the object gets memory.

**Security Checker:** One of the important components of CLR is security checker. This engine restricts the access to system resources such as hard disk and enforcing the restriction on the MSIL code.

**Type Checker:** This ensures the datatype checking of the variable. This is also checking the valid operations on the corresponding datatype. It means that integer value must be assigned to integer datatype and valid operations are allowed in that type.Otherwise it raises the exception.

**Thread support:** Multithreading is a very important feature of any programming language. Threads are playing an very important role in developing the application in this framework The application can contain one or more threads. These threads are managed by the CLR.

**Exception Manager :** The Net framework follows the structured exception handling in all its compliant languages.The exception might rise from managed code as well as unmanaged code. This manager provides the support to handle these type of Exceptions.

**Debug Engine** : Debug means finding and removing the bug from the programs, The application written in any supported framework languages are debugged by this engine.

**Base class library support** – It provides the types that the applications need at run time.

1. .**net framework class library-**The class library, the other main component of the .NET Framework, is a comprehensive, applications ranging from traditional command-line or graphical user interface (GUI) applications to applications based on the latest innovations provided by ASP.NET, such as Web Forms and XML Web services.

**Q2**. Current .Net Standard framework and Current core framework?`(T)

**Solution:-**

Current .Net standard framework is 4.8 and current core framework is 3.1

**Q3**. Difference between Managed and Unmanaged code? (T)

**Solution:-**

|  |  |
| --- | --- |
| **Managed Code** | **Unmanaged code** |
| It is executed by managed runtime environment or managed by the CLR. | It is executed directly by the operating system. |
| It provides security to the application written in .NET Framework. | It does not provide any security to the application. |
| Memory buffer overflow does not occur. | Memory buffer overflow may occur. |
| It provide runtime services like Garbage Collection, exception handling, etc. | It does not provide runtime services like Garbage Collection, exception handling, etc. |
| The source code is complied in the intermediate language know as *IL or MSIL or CIL* | The source code directly compile into native language. |
| It does not provide low-level access to the programmer. | It provide low-level access to the programmer. |

**Q4**. What do you mean by MSIL Code? (T)

**Solution:-**

SIL stands for Microsoft Intermediate Language. We can call it as Intermediate Language (IL) or Common Intermediate Language (CIL). During the compile time , the compiler convert the source code into Microsoft Intermediate Language (MSIL) .Microsoft Intermediate Language (MSIL) is a CPU-independent set of instructions that can be efficiently converted to the native code. During the runtime the [Common Language Runtime](http://vb.net-informations.com/framework/common_language_runtime.htm) (CLR)'s [Just In Time](http://vb.net-informations.com/framework/just_in_time_compiler.htm) (JIT) compiler converts the Microsoft Intermediate Language (MSIL) code into native code to the Operating System.

When a compiler produces Microsoft Intermediate Language (MSIL), it also produces [Metadata](http://vb.net-informations.com/framework/metadata.htm). The Microsoft Intermediate Language (MSIL) and Metadata are contained in a portable executable (PE) file . Microsoft Intermediate Language (MSIL) includes instructions for loading, storing, initializing, and calling methods on objects, as well as instructions for arithmetic and logical operations, control flow, direct memory access, exception handling, and other operations

**Q5**. What is the difference between .Net and .Net Core Frameworks? (T)

**Solution:-**

|  |  |
| --- | --- |
| **.Net Framework** | **.Net Core** |
| A high-performance and scalable system without UI | .NET Core is much faster. |
| Time is not a problem. Experiments are acceptable. No rush to deployment. |  |
| Stable version for an immediate need to build and deploy | .NET Framework has been around since 2001. .NET Core is just a baby. |
| Already have a pre-configured environment and systems | .NET Framework is better. |
| Windows client applications using Windows Forms and WPF | .NET Framework |
| User interface centric Web applications | NET Framework is better now until .NET Core catches up. |
| Using microservices | Both, but .NET Core is designed to keep today's needs in mind. |
| Cross-platform needs | .NET Core |
| Heavily rely on the command line | NET Core has better support. |
| Docker containers support | Both, but .NET Core is born to live in a container. |

**Q6**. Explain the execution process of managed code in .Net? (T)

**Solution:-**

**Execution process of managed code in .net is as follows:-**

**Choosing the right compiler-** .Net Framework is a multilanguage execution environment, the runtime supports a wide variety of data types and language features. In order to obtain the full benefits provided by the common language runtime , you should use one or more language (VB.Net, C# etc.) compilers that target the runtim**e**

**Compiling the code to MSIL-** Unlike the execution style of compiling source code into machine level code, .Net language compilers translates the source code into Microsoft Intermediate Language. This ensures language interoperability because no matter which language has been used to develop the application, it always gets translated to Microsoft Intermediate Language. During the compile time the compiler produces metadata, that contains description of the program like dependencies, versions etc

**Compiling MSIL to native code-** Before the program execution , Jist In Time compiler (JIT) compiles the MSIL into native code and stores it in a memory buffer. During JIT compilation, the code is also checked for type safety. Type safety ensures that objects are always accessed in a compatible way. The compiled native code is in memory and is not persisted. So every time we run our application this whole thing has to happen again.

**Excecution of Code-**After translating the IL into native code, it is sent to .Net runtime manager. The .Net runtime manager executes the code. During execution, managed code receives services such as garbage collection, security, interoperability with unmanaged code, cross-language debugging support, and enhanced deployment and versioning support.

**Q7**. Write a sample program to understand CLS and CTS? (P)

**Solution:-In git Repository**

**Q8**. Create a Console Application and Class Library using Visual Studio IDE? (P)

**Solution:-In git Repository**

**Q9**. Create a Windows Application using Visual Studio IDE? (P)

**Solution:-In git Repository**

**Q10**. Create a Web Application and MVC and WEB APIs Application using Visual Studio IDE? (P)

**Solution:-In git Repository**