

**NAME: ADEOSUN TAIWO OLASUNKANMI**

**MATRIC NO: H/CS/23/1069**

**C# ASSIGNMENT ONE**

- 1. Write a short note on the evolution of .Net Framework and C#(100 words)**
- 2. Explain the following terms ;**  
**Mono, Xamarin , COM , .Net Core,Unity C#,REST**
- 3. Critically,explain ANY three key functions of CLR(50 words)**

**Answer**

The .NET Framework is an open-source platform for developing Windows-based applications, often referred to as Microsoft .net. The .NET Framework includes a variety of developer tools and class libraries. The .NET Framework works with applications developed in C#, F#, Visual Basic, and other popular programming languages.

.NET Framework and C# are both technologies developed by Microsoft, and they have seen significant changes and improvements over the years. The .NET Framework was first released in 2002, and it has evolved to be more powerful and feature-rich, with new versions released every few years.

C#, the programming language used with the .NET Framework, has also evolved over time, adding new features and improvements to make it easier and more efficient for developers to use. Overall, the evolution of these technologies has been impressive, and they continue to be widely.

Explain the following terms;

**Mono:** Mono is an open-source project that aims to provide a cross-platform implementation of the .NET Framework. In other words, it's a way to run .NET-based applications on platforms other than Windows, such as macOS and Linux.

**XAMARIN:** Xamarin is a software platform that allows developers to build applications using C# and .NET, and then deploy them on multiple platforms, including iOS, Android, and Windows. Xamarin was originally created by Xamarin Inc.,. With Xamarin, developers can write the code once and then deploy it on multiple platforms without having to write separate code for each platform.

**COM:** This is an acronym for Component Object Model, is a Microsoft technology that allows software components to communicate with each other. COM is a binary standard, meaning that it defines how components interact with each other at the binary level.

**.Net Core:** .NET Core is a cross-platform and open-source implementation of the .NET Framework. Unlike the original .NET Framework, which is only available on Windows, .NET Core can be run on multiple operating systems, including macOS, Linux, and Windows. .NET Core is a modular platform, meaning that developers can pick and choose the features they need for their applications. .NET Core is a relatively new technology, but it's growing in popularity, especially among developers who want to create cross-platform applications.

**Unity C# :** C sharp, is a general-purpose programming language developed by Microsoft. It's a strongly-typed, object-oriented language that can be used to create a wide variety of applications, including web applications, mobile apps, desktop apps, and more. C# is a powerful language that's known for its ease of use and its performance. It's also highly compatible with other Microsoft technologies, like the .NET Framework and Visual Studio. C# is a popular choice among developers, and it's often compared to other languages like Java and Python.

**REST:** This simply means Representational State Transfer and is a software architectural style that's commonly used for building APIs. It's based on HTTP, and it defines a set of rules that govern how data is transmitted between a client and a server. With REST, each resource on the server is uniquely identified by a URL, and the client can perform operations on the resource by sending HTTP requests to the server. REST is known for its simplicity and scalability, and it's used by many popular APIs, including Twitter, GitHub, and Instagram.

3.

- Garbage collection: The CLR automatically deallocates memory that is no longer being used, freeing up resources for other tasks.
- JIT compilation: The CLR compiles code at runtime, rather than pre-compiling it, which can result in faster execution and more efficient use of resources.
- Reflection: The CLR can introspect the types and metadata that make up an application, making it easy to work with dynamic data.
- Exception handling: The CLR provides a comprehensive exception handling model, allowing developers to gracefully handle errors and prevent applications from crashing.