QUESTION 1.

The .NET Framework has come a long way since its inception. It started off as a Windows-only platform for building software applications. Over time, it evolved into a cross-platform framework, allowing developers to build applications for Windows, macOS, and Linux. As for C#, it has also evolved alongside the .NET Framework. It started as a simple, object-oriented language and has grown into a powerful and versatile language with features like async/await, LINQ, and more. Together, the .NET Framework and C# have become a popular choice for building a wide range of applications. C# 2.0 introduced generics, and so on.

QUESTION 2.

- Mono: Mono is an open-source implementation of the .NET Framework. It allows developers to run .NET applications on various platforms, including Linux, macOS, and Windows.

- Xamarin: Xamarin is a cross-platform development framework that allows developers to build native mobile applications using C# and the .NET Framework. With Xamarin, you can create apps for iOS, Android, and Windows using a shared codebase.

- COM: COM stands for Component Object Model. It is a Microsoft technology that enables software components to communicate with each other. COM allows for the integration of different software components, regardless of the programming language they were written in.

- .NET Core: .NET Core is an open-source, cross-platform development framework. It is the successor to the .NET Framework and is designed to be lightweight and modular. .NET Core allows developers to build applications that can run on Windows, macOS, and Linux.

- Unity: Unity is a popular game development platform that uses C# as its primary scripting language. It provides a range of tools and features for creating interactive and immersive games across various platforms.

- C#: C# is a modern, object-oriented programming language developed by Microsoft. It is widely used for building a wide range of applications, including web, mobile, and desktop applications. C# is known for its simplicity, readability, and strong integration with the .NET ecosystem.

- REST: REST stands for Representational State Transfer. It is an architectural style for designing networked applications. RESTful APIs (Application Programming Interfaces) use HTTP methods like GET, POST, PUT, and DELETE to perform operations on resources. It is commonly used in web development for building scalable and interoperable web services.

QUESTION 3.

1. Memory Management: CLR manages memory allocation and deallocation, including garbage collection, to optimize memory.

2. Code Execution: CLR compiles and executes .NET code into machine language, providing runtime environment .

3. Exception Handling: CLR provides a robust exception handling mechanism, allowing developers to handle and manage exceptions during runtime.