**1.what is role of client?**

**Client computer provide an interface to allow a computer user to request services of the server and to display the results the server returns. Servers wait for requests to arrive from clients and then respond to them.**

**2.what is role of server?**

**The role of a server is to share data as well as to share resources and distribute work.a server computer can serve its own computer programs as well; depending on the scenario.this could be part of a quid pro quo transaction, or simply a technical possibility.**

**3.what is client server architecture?**

**Client-server architecture,architecture of a computer network in which many clients(remote processors) request and receive service from a centralized server (host computer). Client computer provide an interface to allow a computer user to request services of the server and to display the results the server resturns.**

**4.what is role of compiler?**

**Compiler analyze and convert source code written in languages such as java, c++, c# or swift.they’re commonly used to generate machine code or bytecode that can be executed by the target host system.**

**5.what is difference between compiler and interpreter?**

**(i)interpreter**

**Translate programe one statement at a time.**

**Interpreters usually take less amount of time to analyze the source code. However, the overall execution time is comparatively slower than compiler.**

No Object Code is generated, hence are memory efficient.

Programming languages like JavaScript, Python, Ruby use interpreters.

**(ii)compiler**

Scans the entire program and translates it as a whole into machine code.

Compilers usually take a large amount of time to analyze the source code. However, the overall execution time is comparatively faster than interpreters.

Generates Object Code which further requires linking, hence requires more memory.

Programming languages like C, C++, Java use compilers.

6.what is mvc?

**MVC** (Model-View-Controller) is a pattern in software design commonly used to implement user interfaces, data, and controlling logic. It emphasizes a separation between the software's business logic and display. This "separation of concerns" provides for a better division of labor and improved maintenance. Some other design patterns are based on MVC, such as MVVM (Model-View-Viewmodel), MVP (Model-View-Presenter), and MVW (Model-View-Whatever).

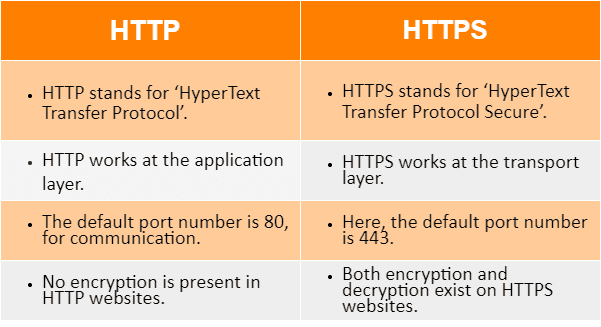
The three parts of the MVC software-design pattern can be described as follows:

1. Model: Manages data and business logic.
2. View: Handles layout and display.
3. Controller: Routes commands to the model and view parts.



**7.what is communication protocol and what is difference between http and https?.**

Communication protocols **allow different network devices to communicate with each other**. They are used in both analog and digital communications and can be used for important processes, ranging from transferring files between devices to accessing the internet.



8.what is .net?

**.NET**

.NET is a free, cross-platform, open source developer platform for building many different types of applications.

With .NET, you can use multiple languages, editors, and libraries o build for web, mobile, desktop, games, IoT, and more.

**Languages**

You can write .NET apps in C#, F#, or Visual Basic.

* C# is a simple, modern, object-oriented, and type-safe programming language.
* F# is a programming language that makes it easy to write succinct, robust, and performant code.
* Visual Basic is an approachable language with a simple syntax for building type-safe, object-oriented apps.

## Cross Platform

Whether you're working in C#, F#, or Visual Basic, your code will run natively on any compatible operating system. You can build many types of apps with .NET. Some are cross-platform, and some target a specific set of operating systems and devices.

9.what is CLR?

The Common Language Runtime (CLR) is **programming that manages the execution of programs written in any of several supported languages, allowing them to share common object-oriented classes written in any of the languages**. It is a part of Microsoft's . NET Framework.

**The runtime provides the following benefits:**

* Performance improvements.
* The ability to easily use components developed in other languages.
* Extensible types provided by a class library.
* Language features such as inheritance, interfaces, and overloading for object-oriented programming.

10.what is difference between cls and cts?

CTS and CLS are parts of .NET CLR and are responsible for type safety within the code. Both allow cross-language communication and type safety. In this article, I would like to expose the relationship between these two.

## CTS

CTS stands for Common Type System. It defines the rules which Common Language Runtime follows when declaring, using, and managing types. The CTS generally deals with the data types. So normally we have access to multiple languages and each and every language has its own set of rules and tweaks regarding the data type where the one language's data type cannot be interpreted by other languages but the .NET Framework language would be able to understand all the data types certainly.

The common type system performs the following functions:

1. It enables cross-language integration, type safety, and high-performance code execution.
2. It provides an object-oriented model for the implementation of many programming languages.
3. It defines rules that every language must follow which runs under the [**.NET framework**](https://www.dotnettricks.com/learn/netframework). It ensures that objects are written in different .NET languages like C#, VB.NET, F#, etc. can interact with each other.

## CLS

[**CLS**](https://simple.wikipedia.org/wiki/Common_Language_Specification) stands for Common Language Specification and it is a subset of CTS. It defines a set of rules and restrictions that every language must follow which runs under the .NET framework. The languages which follow this set of rules are said to be CLS Compliant. In simple words, CLS enables cross-language integration.

The CLS is a specification that defines the rules for supporting the language integration in a certain way that the programs are written in any language, still, it can interoperate with the one another seamlessly while taking the full advantage of concepts such as exceptions handling, inheritance, polymorphism, and other features accordingly. These CLS rules and the specification are documented in the ECMA proposed standard document.

**For example**, one rule is that you cannot use multiple inheritances within .NET Framework. As you know C++ supports multiple inheritances but; when you will try to use that C++ code within C#, it is not possible because C# doesn’t support multiple inheritances.

One another rule is that you cannot have members with the same name with case difference only i.e. you cannot have to add() and Add() methods. This easily works in C# because it is case-sensitive but when you will try to use that C# code in VB.NET, it is not possible because VB.NET is not case-sensitive.

**11.what is difference between compiler and interpreter?**

**(i)interpreter**

**Translate programe one statement at a time.**

**Interpreters usually take less amount of time to analyze the source code. However, the overall execution time is comparatively slower than compiler.**

No Object Code is generated, hence are memory efficient.

Programming languages like JavaScript, Python, Ruby use interpreters.

**(ii)compiler**

Scans the entire program and translates it as a whole into machine code.

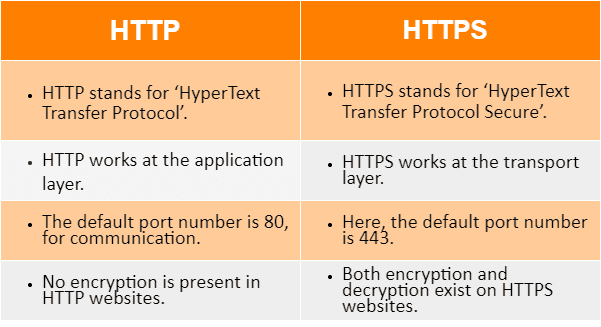
Compilers usually take a large amount of time to analyze the source code. However, the overall execution time is comparatively faster than interpreters.

Generates Object Code which further requires linking, hence requires more memory.

Programming languages like C, C++, Java use compilers.

**12.what is communication protocol and what is difference between http and https?.**

Communication protocols **allow different network devices to communicate with each other**. They are used in both analog and digital communications and can be used for important processes, ranging from transferring files between devices to accessing the internet.



13.what do you mean by design pattern?

In software development, a pattern (or design pattern) is **a written document that describes a general solution to a design problem that recurs repeatedly in many projects**. Software designers adapt the pattern solution to their specific project.

**14.what is client server architecture?**

**Client-server architecture,architecture of a computer network in which many clients(remote processors) request and receive service from a centralized server (host computer). Client computer provide an interface to allow a computer user to request services of the server and to display the results the server resturns.**

15.what is difference between asp.net and mvc.net?

ASP.NET

* Asp.Net Web Form follow a traditional event-driven development model.
* Asp.Net Web Form has server controls.
* Asp.Net Web Form supports view state for state management at the client side.
* Asp.Net Web Form has file-based URLs means file name exist in the URLs must have its physical existence.
* Asp.Net Web Form follows Web Forms Syntax
* In Asp.Net Web Form, Web Forms(ASPX) i.e. views are tightly coupled to Code behind(ASPX.CS) i.e. logic.
* Asp.Net Web Form has Master Pages for a consistent look and feels.
* Asp.Net Web Form has User Controls for code re-usability.
* Asp.Net Web Form has built-in data controls and best for rapid development with powerful data access.
* Asp.Net Web Form is not Open Source.

MVC.NET

* Asp.Net MVC is a lightweight and follows MVC (Model, View, Controller) pattern based development, model.
* Asp.Net MVC has HTML helpers.
* Asp.Net MVC does not support view state.
* Asp.Net MVC has route-based URLs means URLs are divided into controllers and actions and moreover it is based on controller not on physical file.
* Asp.Net MVC follow customizable syntax (Razor as default)
* In Asp.Net MVC, Views and logic are kept separately.
* Asp.Net MVC has Layouts for a consistent look and feels.
* Asp.Net MVC has Partial Views for code re-usability.
* Asp.Net MVC is lightweight, provide full control over markup and support many features that allow fast & agile development. Hence it is best for developing an interactive web application with the latest web standards.
* Asp.Net Web MVC is an Open Source.