

MODULE :- 2 .net core

Badgu Nikhilesh

1 :- What is .net Core?

Ans :- .NET is a free and open-source, managed computer software framework for Windows, Linux, and macOS operating systems. It is a cross-platform successor to .NET Framework. The project is primarily developed by Microsoft employees by way of the .NET Foundation, and released under the MIT License.

2 :- What is Core CLR?

Ans :- CoreCLR is the . NET execution engine in . NET Core, performing functions such as garbage collection and compilation to machine code. . NET Core is a modular implementation of . NET that can be used as the base stack for a wide variety of scenarios, today scaling from console utilities to web apps in the cloud.

3 :- What is role of client?

Ans :- he client is the person for whom the project is carried out. In the case of notifiable projects, clients must appoint a Principal Designer and a Principal Contractor.

4 :- What is role of server?

Ans :- 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.

5 :- What is client server architecture?

Ans :- Client-server model is a distributed application structure that partitions tasks or workloads between the providers of a resource or service, called servers, and service requesters, called clients.

6 :- What is role of compiler?

Ans :- NET Compiler Platform, empowers the C# compiler on . NET Core and allows developers to leverage the rich code analysis APIs to perform code generation, analysis and compilation.

7 :- What is difference between Compiler and Interpreter?

Ans :- Scans the entire program and translates it as a whole into machine code. Interpreters usually take less amount of time to analyze the source code. However, the overall execution time is comparatively slower than compilers. Compilers usually take a large amount of time to analyze the source code.

8 :- How is .Net Core different from .Net Framework?

Ans :- . Net Core does not support desktop application development and it rather focuses on the web, windows mobile, and windows store. . Net Framework is used for the development of both desktop and web applications as well as it supports windows forms and WPF applications.

9 :- What is MVC?

Ans :- Model–view–controller is a software architectural pattern commonly used for developing user interfaces that divide the related program logic into three interconnected elements. This is done to separate internal representations of information from the ways information is presented to and accepted from the user.

10 :- What is communication protocol and what is difference between HTTP and HTTPS?

Ans :- The only difference between the two protocols is that HTTPS uses TLS (SSL) to encrypt normal HTTP requests and responses, and to digitally sign those requests and responses. As a result, HTTPS is far more secure than HTTP. A website that uses HTTP has http:// in its URL, while a website that uses HTTPS has https://.

11 :- Explain Services in .Net Core?

Ans :- ASP.NET Core includes a built-in dependency injection (DI) framework that makes configured services available throughout an app. For example, a logging component is a service. Code to configure (or register) services is added to the Startup.ConfigureServices method. For example: C# Copy.

12 :- Explain Middleware Components?

Ans :- Its primary components are in-memory and enterprise application servers, as well as web servers and content management. Platform middleware includes tools that support application development and delivery, such as web servers, application servers and content management systems.

13 :- Explain What is Request Pipeline?

Ans :- The Request Pipeline is the mechanism by which requests are processed beginning with a Request and ending with a Response. The pipeline specifies how the application should respond to the HTTP request. The Request arriving from the browser goes through the pipeline and back.

14 :- Explain MVC application life cycle.

Ans :- The MVC framework handles converting the route data into a concrete controller that can handle requests. After the controller has been created, the next major step is Action Execution. A component called the action invoker finds and selects an appropriate Action method to invoke the controller.

15 :- Where do you find the Middleware in .Net Core?

Ans :- A middleware is nothing but a component (class) which is executed on every request in ASP.NET Core application. In the classic ASP.NET, HttpHandlers and HttpModules were part of request pipeline. Middleware is similar to HttpHandlers and HttpModules where both needs to be configured and executed in each request.

16 :- What is difference between Asp.Net webform and .Net Core?

Ans :- ASP.Net Core is a new version of asp.net. It is a free open source which can run on different OS like Mac, Windows and Linux.