1.What is .Net Core?

.Net core is a completely NEW framework, which is a Free and Open-source platform developed and maintained by Microsoft.

**2. What is the role of startup.cs file?**

1. **Startup constructer** will set the configuration related thing in IConfiguration interface.

2. **ConfigruationServices** methods configures the SERVICES which are required by the application.

3**. CONFIGURE** method defines the application REQUEST HANDLING PIPLINE as a series of middleware components.

**3. What are the advantages of .NET Core over .NET framework?**

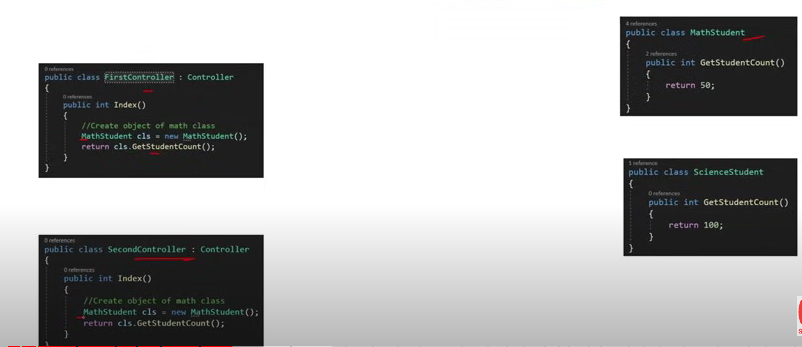
|  |  |  |  |
| --- | --- | --- | --- |
| **Cross Platform** | **Open Source** | **Hosting** | **Build in Dependency Injection** |
| .Net Core Supports Windows, Linux, Mac OS | .Net Core free to use, modify, distribute | .Net Core IIS, Kestrel, Nginx | .Net Core is Loosely coupled design,  Reusability, Testability |
| .Net framework only Supports Windows | .Net framework is Paid | .Net framework supports only IIS | .Net framework do not have built in dependency injection |

**4. What is dependency injections in ASP.Net Core?**

**Dependency injections (DI)** is a software **design pattern** that allow us to develop **Loosely coupled** application.

Constructer Injection, Property Injection, Method Injection

With out DI



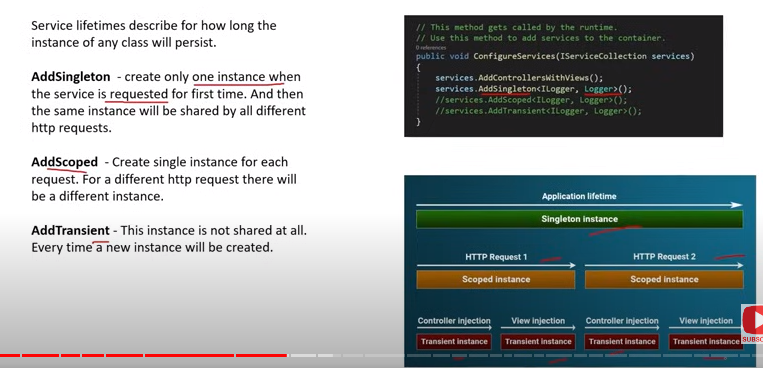
With DI



**5. What are the types of Service Lifetimes of an object/ instance in ASP.NET Core?**

Services lifetimes describe for how long the instance of any class will persist.

Types of services: 1. AddSingleton 2. AddScoped 3. AddTransient

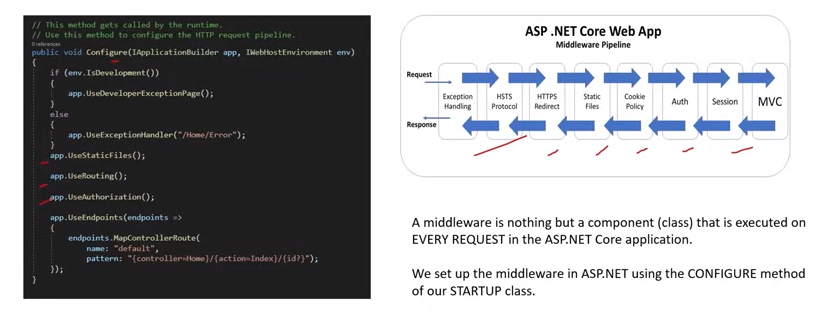


**6. How to use Dependency Injection in Views in ASP.NET Core?**

A service can be injected into a view using the @inject directive.

7**. What is Middleware in ASP.NET Core?**

A middleware is a component that is executed on EVERY REQUEST in the ASP.NET Core application.



**8. What is Run (), Use () and Map () method?**

**Use method** will execute next middleware or line in sequence.

**Run Method** will Terminate the chain,

No other middleware method will run after this,

should be placed at the end of any pipeline.

**MAP Method** extension is used to match request delegates based on a request URL path.

**9. What is Request Delegate in .Net Core?**

**`** Request delegates are used to build the request pipeline. Request delegates handle request pipeline by using RUN, MAP and USE Methods.

**10. What is Routing? Explain attribute routing in ASP.NET Core?**

**Routing** is used to handle incoming HTTP request based on the URL.

Attribute based routing is the ability to manipulate the behaviour of URL by route Attribute.

11**. What are the main JSON files available in ASP.NET Core?**

**Launchsettings.json –** when the application is launching or starting.

Ex: set development or production environment in this file

**Appsettings.json –** Configuration setting like database connection string can be set in this file. Like web.config in ASP.Net

**Project.json –** ASP.Net Core uses project.JSON file for storing all project level configuration setting. Ex:: the nugget packages you have installed in the project.

**Globale.json -** define the solution level setting in global.json file. Ex: Application name and version.

**12. What is Metapackage? What is the name of Metapackage provided by ASP.NET Core?**

Metapackage is a consolidated package of many dependencies.

* + Microsoft.AspNetCore.Hosting
  + Microsoft. AspNetCore.Routing
  + Microsoft. AspNetCore.Logging

Metapackage present in ASP.NET Core project by default.

**13. Default project structure in ASP.Net Core Application?**

**WWWRoot –** To store static files of the application like js/css/images.

**Program.cs –** Entry point of the application

**Startup.cs –** Configure services and request pipeline for the application.

**Appsettings.json –** Configuration settings like database connection strings and other things are saved here.

14. **What is the role of program.cs file??**

* Program.cs file is the entry point of application via Main() method.
* It also registers Startup.cs class
* It also registers IISIntegration and Create a host using an instance of IWebHostBuilder.

**15. What is Middleware?**

**A Middleware** is nothing but a component (Class) that is executed on **Every Request** in the ASP.Net Core application.

We set up the middleware in ASP.Net using the CONFIGURE method of our STARTUP class.

16. What is HOST in ASP.Net Core

The Host Configures a server like kestrel or IIS and a request processing pipeline.

17. What are the various way to save configuration settings in ASP.Net Core?

* + Appsetting.json (default)(Mostly used)
  + Azure Key Vault (Mostly used)
  + Environment variables
  + In-memory .Net objects
  + Command Line Arguments
  + Custom providers

18. How Routing works in ASP.Net Core?

Routing is used to handle incoming HTTP request for the app based on the URL.

http//localhost:1234/home/index/100

19. Session and State Management in ASP.Net Core/

**Cookies-** Cookies store data in the user browser. Browsers send cookies with every request.

**SESSION**: Session state helps in transferring data bw different pages. It takes the help of cookies for this purpose.

**TEMPDATA:** Asp.Net MVC Core application tempdata is used to transfer data from controlled to view.

**QUERY STRING:**  It is used to transfer limited data by appending the url of the request.

**HIDDEN FIELDS:** are used to persist data in same page during postbacks.

20. What is **CORS**?

**CORS** (Cross-Origin Resource Sharing) in .NET Core is a security feature that controls how resources on a web server are shared with client-side applications hosted on a different origin (domain, scheme, or port). It allows the server to specify which domains are permitted to access its resources, helping to protect against cross-site request forgery (CSRF) and similar attacks.

When a web page makes a request to a resource on a different domain (like from http://localhost:3000 to http://localhost:5000), the browser sends a **preflight request** (an OPTIONS request) to check if the server permits access from the client’s origin.s