1.Coding question

------------------------

Write a program to check palindrome.

Example - Malayalam

Output - True/False

2.NET (ASP.net core)

-----------------

- Difference between .Net framework and .NET core

- Middleware

- microservices architecture in .net core with aws

- Authentication / Authorization

    -JWT token

    - X-API

- Cache memory

- Program.cs and Startup.cs class (methods and uses)

- IIS server

- Dependency injection

- Custom middleware (exception middleware)

- Practical Questions

3. C# and OOPS

--------------------

Memory (Generic and non-generic)

Constant / Read-only

Garbage collector

 Encapsulation / abstraction

 Interface / Abstract

 Overriding / overloading (related question)

 Ienuramble / Iquerable

5. Database

---------------

Delete vs Truncate.

Store procedure

Trigger

Magic table

Joins

Drop, delete , truncate

Indexing

Asked queries and asked to optimize.

6:- EF core with Linq

------------------

Code First and Data first approach

Commands for connection

7- Azure

------------

Keyvault

App Service

Logic app

service bus

8:- Git

-------------

Git flow

Hitachi - Dunnhumby Interview Question -

-----------------------------

1.Coding question

------------------------

Write a program to print pair of largest and smallest numbers from an array

Example - [7,4,8,1,3,2,5,6]

Output - 8,1 7,2 6,3 5,4

using System;

using System.Linq;

class Program

{

static void Main(string[] args)

{

int[] numbers = { 7, 4, 8, 1, 3, 2, 5, 6 };

PrintPairsOfLargestAndSmallest(numbers);

}

static void PrintPairsOfLargestAndSmallest(int[] arr)

{

if (arr == null || arr.Length < 2)

{

Console.WriteLine("Array is empty or too small.");

return;

}

Array.Sort(arr);

int left = 0;

int right = arr.Length - 1;

while (left < right)

{

Console.WriteLine(arr[right] + "," + arr[left]);

left++;

right--;

}

}

}

2.NET (ASP.net core )

-----------------

- Difference between .Net frmawork and .NET core

- Middleware

- Authentication / Authorization

-JWT token

- X-API

- Cashe memory

- Program.cs and Startup.cs class ( methods and uses)

- IIS server ( Kestral )

- Depedency injection

- Service lifeTime ( Scope , Transient , Singletion)

- Custom middleware ( exception middleware)

- Request pipeline

- BackgroyndJOB ( Ihosted service)

- Few practical question

3. C# and OOPS

--------------------

Memory ( Generic and non generic)

Constant / Readonly

Garbage collector

Encapsulation / abstraction

Interface / Astract

Overriding / overrloading ( related question )

Ienuramble / Iquerable

And few conceptuial question based on my answer

4.Angular :-

---------------------

Life cycle hooks

Service

Data binading

Components

Selectors

Templetes

Difference - Obseravle / promises

Dff - Angualr js / angualr ( Js / Ts)

Communication between Components ( parents / child / siblings )

interceptor

Rxjs

Directives

5. Database

---------------

Diff - first or default

Store procesure

Trigger

Magic table

Joins

Drop , delete , trucate

Indexing

6:- EF core with Linq

------------------

Diff first and firstOrDefault

Code first approch

Commands for connection

How to connect with applictaion

7- Azure

------------

Keyvault

App service

Logic app

service bus

8:- Git

-------------

Commands

Fetch , Pull

Mearge , Rebase

8 :- Experince ( Project related question )

----------------------------------------------

Responsiblity

Project overview

Based on that few practrical question

Microservices - 30+ Microservices (20 in Dotnet , rest on Python , Scala)

Cloud - Good Pubsub and Rabit MQ

Dotnet Core

Request Delegate - Used in Customer Middle ware - Struggling to Answer

Ap.use

2) Service Life cycle (Transit, Scope and Single)

* Difference between Scope and Single ton
* If request comes from Different browser, will it be same instance or different instance.
* Castrol works, can we use IIS or Apache or another web server.
* I Application Builder
* Startup (Configure and xxxx)
* I Option
* API Gateway (When should we use the API Gateway)
* Ocelot

Microservice

1. What pattern you are following (
2. How Microservice connect with another Microservice (Event Driven Architecture)

Dotnet Core

1. Apsetting.json (Profiles, how can we edit, what values we can edit)
2. Circuit Breaker (Ex Polly)
3. Reverse Proxy (Any Example)
4. Engine X

Docker

1. Did u work on Docker?

C#

1. Await
2. Difference between Task and Thread
3. Asynchronous method without using return.
4. Parallelism
5. Once a thread is started, can I cancel a thread.
6. If a task is started, can I cancel the task (Cancellation token)
7. Async and Await
8. Delegates
9. Difference between Action Delegate and Function Delegate
10. Predictive Delegate , Multicast Delegate
11. Private constructor in a static class
12. Sealed Constructor
13. Method in sealed keyword
14. Can we create a constructor of
15. Constructor churning
16. Base class constructor from child class
17. Diff between Abstract and Interface
18. Diff bet Abstract and
19. Static vs singleton
20. Hoisting in JavaScript

React

1. State Management
2. Redux
3. Use Call Back
4. Use Memo
5. Local Storage and Session Storage
6. Useref

Testing Strategy

IEnumurable

Tupper

Settimeout

for(let i = 0; i < 3; i++) {  
setTimeout(() => console.log(i), 1000)  
}

for(var i = 0; i < 3; i++) {  
setTimeout(() => console.log(i), 1000)

Difference between Var and Let

Subject and Behaviour Subject