Full Stack Engineer (.NET + Angular) - Interview Questions and Answers

Angular (Frontend)

1. What is Angular?

Angular is a TypeScript-based open-source web application framework developed by Google, used to build single-page applications (SPA).

2. What are components in Angular?

Components are building blocks of an Angular application. Each component has:

- A TypeScript class (logic)
- An HTML template
- A CSS style
- A selector (used in HTML)

3. What is a service in Angular?

Services are used to write reusable business logic and perform tasks like HTTP calls. They are injected using Angular's Dependency Injection system.

4. What is data binding in Angular?

Types:

- Interpolation: {{ value }}
- Property binding: [property]="value"
- Event binding: (event)="handler()"
- Two-way binding: [(ngModel)]="value"

5. What is RxJS and how is it used in Angular?

RxJS (Reactive Extensions for JavaScript) is used for reactive programming with observables. It helps in handling asynchronous operations like HTTP requests.

6. How does Angular handle routing?

Using @angular/router, define routes in app-routing.module.ts. Components are rendered based on the URL.

.NET Core (Backend)

1. What is .NET Core / .NET 5/6/7?

.NET Core is a cross-platform, high-performance framework for building modern, cloud-based apps. As of .NET 5+, it's unified as just ".NET".

2. What is Dependency Injection in .NET Core?

It's a built-in feature used to inject services into controllers via constructor injection. You register services in Startup.cs or Program.cs.

3. Explain Middleware in .NET Core.

Middleware are components that process requests/responses in the pipeline.

4. How do you handle exceptions globally in .NET Core?

Using app.UseExceptionHandler() in Startup.cs or writing custom middleware.

5. What is Entity Framework Core?

EF Core is an ORM that allows working with a database using .NET objects. Supports LINQ queries, migrations, and more.

Integration Questions

1. How do you connect Angular with .NET Core API?

- .NET Core exposes REST APIs via controllers ([ApiController])

- Angular uses HttpClient to call the API

- Handle CORS using services.AddCors() in .NET

2. How do you secure your full-stack application?

- Use JWT tokens for authentication

- Implement role-based authorization

- Use HTTPS, input validation, and anti-forgery tokens

Common Coding Questions

1. Write an API in .NET Core to return list of products.

```
[HttpGet("products")]

public async Task<IActionResult> GetProducts() {

    var products = await _context.Products.ToListAsync();

    return Ok(products);

}
```

2. Write Angular code to call this API.

```
this.http.get<Product[]>('https://api.example.com/products')
```

```
  .subscribe(data => this.products = data);
```

Behavioral / Project Questions

1. Describe a recent project using .NET + Angular.

Talk about architecture, use of components, services, API integration, auth mechanism, challenges solved, etc.

2. How do you optimize performance in Angular and .NET?

- Angular: Lazy loading, OnPush change detection, trackBy, AOT
- .NET: Caching, async methods, connection pooling, logging