Angular Notes - Full Stack Exam

1. Angular vs AngularJS

AngularJS (1.x) is the older JavaScript-based framework by Google using the MVC architecture. Angular (2+) is a modern, TypeScript-based framework with better performance, modularity, and mobile support.

Differences between AngularJS and Angular

AngularJS	**Angular**
JavaScript-based	TypeScript-based
Uses controllers and scope	Uses components and services
Two-way data binding by default	Supports both one- and two-way data binding
Not mobile friendly	Built for mobile-first
No CLI officially	Comes with Angular CLI

2. Angular Architecture

Angular applications are based on a component-based architecture. The core building blocks are:
- Modules
- Components
- Templates
- Services
- Dependency Injection
- Routing
- Metadata

These parts work together to build scalable and maintainable applications.

3. Building Blocks of Angular

Angular's main building blocks include:
- Components
- Templates
- Modules
- Metadata
- Services
- Dependency Injection

4. Angular CLI and Commands

Angular CLI (Command Line Interface) is a tool to initialize, develop, scaffold, and maintain Angular apps.

Example commands:

- ng new my-app
- ng serve
- ng generate component name
- ng build

5. Angular Modules

Angular modules group components, directives, services, etc. into cohesive blocks of functionality. Root module is AppModule. You can also have feature modules.

6. Angular Directives

Directives are classes that add behavior to elements in Angular templates. Types:

- Structural Directives (*ngIf, *ngFor)
- Attribute Directives (ngClass, ngStyle)

7. Angular Forms

Two types:

- Template-driven Forms: Easier, suitable for simple forms
- Reactive Forms: More control, scalable for complex forms

8. Data Binding

Data binding connects component data to the UI. Types:

1. Interpolation: {{ value }}
2. Property Binding: [property]="value"
3. Event Binding: (event)="handler()"
4. Two-Way Binding: [(ngModel)]="value"

9. Component Interaction (@Input/@Output)

@Input() passes data from Parent to Child.
@Output() uses EventEmitter to send data from Child to Parent.

10. Angular Animations

Angular provides the @angular/animations module to create transitions and effects using the trigger(), state(), style(), animate() functions.

Angular Notes - Full Stack Exam

11. Component Lifecycle Hooks

Angular components follow a lifecycle:

- ngOnChanges
- ngOnInit
- ngDoCheck
- ngAfterContentInit
- ngAfterContentChecked
- ngAfterViewInit
- ngAfterViewChecked
- ngOnDestroy