**1. What is Angular?**

**Answer:**  
Angular is a **front-end web application framework** developed by Google. It’s used to build **single-page applications (SPAs)** using **HTML, CSS, and TypeScript**.

**2. What is a Single Page Application (SPA)?**

**Answer:**  
A Single Page Application loads a **single HTML page** and dynamically updates the content without refreshing the whole page.

**3. What is TypeScript and how is it used in Angular?**

**Answer:**  
TypeScript is a **superset of JavaScript** that adds **static typing**. Angular is built using TypeScript because it makes the code more structured and easier to debug.

**4. What are components in Angular?**

**Answer:**  
A component is the **basic building block** of an Angular application. It controls a **part of the UI** and consists of:

* A TypeScript class (logic)
* An HTML template (view)
* CSS styles

**5. What is a module in Angular?**

**Answer:**  
A module is a container for a group of components, directives, pipes, and services. Every Angular app has a **root module** called AppModule.

**6. What is the difference between a component and a module?**

**Answer:**

* A **component** controls a view.
* A **module** groups multiple components and helps organize the app.

**7. What is data binding in Angular?**

**Answer:**  
Data binding is the mechanism to **synchronize data between the model (class) and view (template)**.  
Angular supports:

* Interpolation ({{ }})
* Property binding ([value])
* Event binding ((click))
* Two-way binding ([(ngModel)])

**8. What is interpolation in Angular?**

**Answer:**  
Interpolation is using {{ }} to bind a variable from the class to the HTML.

**Example:**

html

CopyEdit

<h1>{{ title }}</h1>

**9. What is a directive in Angular?**

**Answer:**  
Directives are instructions in the DOM.  
Types:

* **Structural** (\*ngIf, \*ngFor)
* **Attribute** (like [ngClass])
* **Custom** (user-defined)

**10. What is ngIf and ngFor in Angular?**

**Answer:**

* \*ngIf: Conditionally shows/hides elements.
* \*ngFor: Loops over a list to display each item.

**11. What is a service in Angular?**

**Answer:**  
A service is a class used to **share data or logic** across multiple components. Services are usually injected using **Dependency Injection**.

**12. What is Dependency Injection (DI) in Angular?**

**Answer:**  
DI is a design pattern where **services or dependencies are injected** into components rather than being created inside them. Angular has a built-in DI system.

**13. What is routing in Angular?**

**Answer:**  
Routing in Angular allows **navigation between different components/pages** without refreshing the browser.

**14. What is the purpose of @Component decorator?**

**Answer:**  
It marks a class as a **component** and provides metadata like:

* selector
* template URL
* style URLs

**15. What is the purpose of @NgModule decorator?**

**Answer:**  
It marks a class as an **Angular module** and provides metadata like:

* declarations
* imports
* providers
* bootstrap

**16. What is a template in Angular?**

**Answer:**  
A template is the **HTML part** of a component that defines what the user sees on the screen.

**17. What is ngModel in Angular?**

**Answer:**  
ngModel is used for **two-way data binding** between the input field and the component class.

**Example:**

html

CopyEdit

<input [(ngModel)]="username">

**18. What are lifecycle hooks in Angular?**

**Answer:**  
Lifecycle hooks are **methods** that allow you to act at different stages of a component’s life.  
Common ones:

* ngOnInit()
* ngOnDestroy()

**19. What is the difference between ngOnInit() and constructor?**

**Answer:**

* **Constructor**: Used to initialize the class.
* **ngOnInit()**: Called by Angular after the component is created and inputs are initialized. Used for logic that requires component setup.

**20. How do you pass data between components in Angular?**

**Answer:**

* **Parent to child**: Use @Input()
* **Child to parent**: Use @Output() with EventEmitter