Angular Interview Questions and Answers

# Core Angular Questions

1. What is Angular?

Angular is a TypeScript-based open-source web application framework developed by Google. It is used to build client-side single-page applications (SPAs).

1. What are the main building blocks of Angular?

Components,

Modules,

Templates,

Services and Dependency Injection,

Directives,

Routing.

1. What is a Component in Angular?

A component controls a part of the screen called a view. Each component has

a TypeScript class (logic),

an HTML template,

a CSS file (style).

1. What is a Module in Angular?

Modules group related components, services, and other code together. Every Angular app has at least one module: AppModule.

1. What is a Service in Angular?

A service is a reusable class that provides logic not directly related to views (e.g., data fetching, logging).

1. What is Dependency Injection?

A design pattern in which components are given their dependencies instead of creating them themselves.

1. What is Two-way Binding?

It allows synchronization of data between the component class and the view using [(ngModel)].

1. What are Directives in Angular?

Directives add behavior to HTML elements:

Structural Directives (\*ngIf, \*ngFor) a

Attribute Directives ([ngClass], [ngStyle]).

1. What is the difference between ngOnInit and Constructor?

Constructor is used for dependency injection.

ngOnInit is used for initialization logic after the component is initialized.

1. How does Routing work in Angular?

Angular uses the RouterModule to navigate between views. Define routes in app-routing.module.ts.

# Advanced Angular Questions

1. What is Lazy Loading in Angular?

Lazy loading loads feature modules only when needed, improving performance.

1. What is the difference between Observable and Promise?

Observable can emit multiple values over time (used in RxJS),

Promise handles a single async event.

1. What is Change Detection in Angular?

Angular automatically checks for changes in the component and updates the DOM using Zone.js and ChangeDetectorRef.

1. What is Angular CLI?

A command-line interface tool to scaffold and manage Angular apps.

1. What is Ahead-of-Time (AOT) Compilation?

AOT compiles the application at build time, improving performance and reducing runtime errors.

# Scenario-Based Angular Questions

1. How would you handle a failed API call in a service?

Use catchError from RxJS and display error messages.  
  
this.http.get('/api/books').pipe(catchError(err => { console.error('API failed', err); return throwError(() => err); }));

1. How do you share data between two unrelated components?

Use a shared Service with Subject or BehaviorSubject.

1. You need to optimize an Angular app’s performance. What steps would you take?

Use Lazy loading, AOT, OnPush change detection, avoid unnecessary DOM manipulations, use trackBy in \*ngFor.

1. How would you implement role-based access in routes?

Use Route Guards like CanActivate and check user role in the guard.

1. How would you create a reusable modal component?

Create a dynamic component using ComponentFactoryResolver or use ng-template and ng-container.

# Testing in Angular

1. What is TestBed in Angular Testing?

TestBed is the primary API for writing unit tests for Angular applications.

1. What is Jasmine and Karma?

Jasmine is a testing framework. Karma is a test runner to run tests in browsers.

1. What is the purpose of async and fakeAsync?

They are used to simulate asynchronous code execution in tests.

# Other Useful Topics

Difference between Reactive Forms and Template-driven Forms

Template-driven uses directives in HTML, Reactive uses explicit code (FormBuilder, FormGroup).

1. How to implement Interceptors in Angular?

Use interceptors to modify requests and responses.  
  
@Injectable() export class AuthInterceptor implements HttpInterceptor { intercept(req: HttpRequest<any>, next: HttpHandler) { const cloned = req.clone({ headers: req.headers.set('Authorization', 'Bearer TOKEN') }); return next.handle(cloned); } }

### ****1. What is Angular?****

**Answer:**  
Angular is a TypeScript-based open-source web application framework developed by Google. It is used to build single-page applications (SPAs) with a component-based architecture.

### ****2. What are Components in Angular?****

**Answer:**  
Components are the basic building blocks of an Angular application. Each component consists of:

* A **TypeScript class** (business logic),
* An **HTML template** (view),
* And a **CSS file** (styling).

They control a part of the screen and are declared using the @Component decorator.

### ****3. What is a Module in Angular?****

**Answer:**  
Modules are containers for a cohesive block of code dedicated to an application domain. The root module is usually AppModule. They are defined using the @NgModule decorator and can include components, services, directives, and pipes.

### ****4. What is a Service in Angular?****

**Answer:**  
Services are used to encapsulate business logic, data access logic, or reusable logic. They are typically injected into components using Angular's **Dependency Injection** system.

### ****5. What is Dependency Injection (DI)?****

**Answer:**  
Dependency Injection is a design pattern used to implement IoC (Inversion of Control). Angular uses DI to provide components with the services or objects they need.

### ****6. What are Directives in Angular?****

**Answer:**  
Directives are classes that add additional behavior to elements in Angular.

* **Structural Directives** (like \*ngIf, \*ngFor) change the DOM layout.
* **Attribute Directives** (like ngClass, ngStyle) change the appearance or behavior.

### ****7. What is Two-way Data Binding?****

**Answer:**  
Two-way data binding means that changes in the UI automatically update the component class and vice versa. It’s implemented using [(ngModel)].

### ****8. What is Routing in Angular?****

**Answer:**  
Routing in Angular allows navigation between different views or components. It is configured in the RouterModule using route definitions like:

ts

CopyEdit

const routes: Routes = [

{ path: 'home', component: HomeComponent },

{ path: '\*\*', redirectTo: 'home' }

];

### ****9. What are Lifecycle Hooks in Angular?****

**Answer:**  
Angular provides lifecycle hooks to tap into key moments in the component lifecycle:

* ngOnInit() – called once after the first ngOnChanges()
* ngOnChanges()
* ngDoCheck()
* ngAfterViewInit(), ngAfterContentInit(), etc.
* ngOnDestroy()

### ****10. What is the Angular CLI?****

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

bash

CopyEdit

ng new my-app

ng serve

ng generate component my-component

### ****11. How do you communicate between components?****

**Answer:**

* **Parent to Child**: Use @Input() decorator.
* **Child to Parent**: Use @Output() with EventEmitter.
* **Unrelated Components**: Use a shared service with RxJS Subjects/BehaviorSubjects.

### ****12. What is the purpose of ngOnInit()?****

**Answer:**  
ngOnInit() is a lifecycle hook that is called after Angular has initialized all data-bound properties of a component. It’s the best place to fetch data or perform setup.

### ****13. What are Pipes in Angular?****

**Answer:**  
Pipes transform data in the template.  
Example: {{ amount | currency }}  
You can also create custom pipes using @Pipe().

### ****14. What is HttpClient in Angular?****

**Answer:**  
HttpClient is a service used to make HTTP requests. It returns an Observable which can be subscribed to.

ts

CopyEdit

this.http.get('api/books').subscribe(data => this.books = data);

### ****15. What is a Reactive Form vs Template-Driven Form?****

**Answer:**

* **Template-driven forms** use HTML with directives like ngModel.
* **Reactive forms** use a model-driven approach with FormGroup, FormControl.

### ****16. What is Lazy Loading in Angular?****

**Answer:**  
Lazy loading is a technique where Angular loads feature modules only when they are needed, reducing the initial load time of the app.

### ****17. How do you handle errors in Angular HTTP calls?****

**Answer:**  
Using RxJS catchError() operator in a service:

ts

CopyEdit

catchError(error => {

// handle error

return throwError(() => new Error('Something went wrong'));

})

### ****18. What is the difference between Observable and Promise?****

**Answer:**

| **Observable** | **Promise** |
| --- | --- |
| Can handle multiple values over time | Handles a single value |
| Supports operators like map, filter | No operators |
| Can be canceled | Not cancelable |

### ****19. What is the purpose of ngFor and ngIf?****

**Answer:**

* \*ngFor: Loops through an array and displays items.
* \*ngIf: Conditionally includes a template in the DOM.

### ****20. What is Change Detection?****

**Answer:**  
Change Detection in Angular is the process of synchronizing the DOM with the component model. It checks for changes in data and updates the view accordingly.

## ✅ ****Core Angular Questions****

### 1. ****What is Angular?****

Angular is a TypeScript-based open-source web application framework developed by Google. It is used to build client-side single-page applications (SPAs).

### 2. ****What are the main building blocks of Angular?****

* **Components**
* **Modules**
* **Templates**
* **Services and Dependency Injection**
* **Directives**
* **Routing**

### 3. ****What is a Component in Angular?****

A component controls a part of the screen called a **view**. Each component has:

* A TypeScript class (logic)
* An HTML template
* A CSS file (style)

### 4. ****What is a Module in Angular?****

Modules group related components, services, and other code together. Every Angular app has at least one module: AppModule.

### 5. ****What is a Service in Angular?****

A service is a reusable class that provides logic not directly related to views (e.g., data fetching, logging). It is usually injected using Angular’s **dependency injection system**.

### 6. ****What is Dependency Injection?****

Dependency Injection (DI) is a design pattern in which components are given their dependencies instead of creating them themselves. Angular uses DI for services.

### 7. ****What is Two-way Binding?****

Two-way binding allows synchronization of data between the component class and the view ([(ngModel)]).

### 8. ****What are Directives in Angular?****

Directives add behavior to HTML elements:

* **Structural Directives** (\*ngIf, \*ngFor)
* **Attribute Directives** ([ngClass], [ngStyle])

### 9. ****What is the difference between**** ngOnInit ****and Constructor?****

* **Constructor** is used for dependency injection.
* **ngOnInit** is used for initialization logic when the component is initialized.

### 10. ****How does Routing work in Angular?****

Angular uses the RouterModule to navigate between views. Define routes in app-routing.module.ts.

## 🔍 ****Advanced Angular Questions****

### 11. ****What is Lazy Loading in Angular?****

Lazy loading loads feature modules only when needed, improving performance.

### 12. ****What is the difference between**** Observable ****and**** Promise****?****

* Observable can emit multiple values over time (used in RxJS).
* Promise handles a single async event.

### 13. ****What is Change Detection in Angular?****

Angular automatically checks for changes in the component and updates the DOM. It uses Zone.js and ChangeDetectorRef.

### 14. ****What is Angular CLI?****

Angular CLI is a command-line interface tool to scaffold and manage Angular apps.

### 15. ****What is Ahead-of-Time (AOT) Compilation?****

AOT compiles the application at build time, improving performance and reducing runtime errors.

## 🧠 ****Scenario-Based Angular Questions****

### 16. ****Scenario: How would you handle a failed API call in a service?****

Use catchError from RxJS in the service and display appropriate error messages in the component.

ts

CopyEdit

this.http.get('/api/books').pipe(

catchError(err => {

console.error('API failed', err);

return throwError(() => err);

})

);

### 17. ****Scenario: How do you share data between two unrelated components?****

Use a shared **Service with Subject/BehaviorSubject**.

### 18. ****Scenario: You need to optimize an Angular app’s performance. What steps would you take?****

* Use **Lazy loading**
* Enable **AOT**
* Use **OnPush change detection**
* Avoid unnecessary DOM manipulations
* Use **trackBy** in \*ngFor

### 19. ****Scenario: How would you implement role-based access in routes?****

Use **Route Guards** like CanActivate, and check the user role in the guard.

### 20. ****Scenario: How would you create a reusable modal component?****

Create a dynamic component using ComponentFactoryResolver or use ng-template and ng-container.

## 🧪 ****Testing in Angular****

### 21. ****What is TestBed in Angular Testing?****

TestBed is the primary API for writing unit tests for Angular applications.

### 22. ****What is Jasmine and Karma?****

* **Jasmine**: Testing framework for writing test cases.
* **Karma**: Test runner to run tests in browsers.

### 23. ****What is the purpose of**** async ****and**** fakeAsync****?****

Used in Angular testing to simulate asynchronous code execution.

## 🧰 ****Other Useful Topics****

### 24. ****Difference between Reactive Forms and Template-driven Forms****

* **Template-driven**: Uses directives in HTML.
* **Reactive**: Uses explicit code in the component (FormBuilder, FormGroup).

### 25. ****How to implement Interceptors in Angular?****

Used to modify requests and responses (e.g., add auth token).

ts

CopyEdit

@Injectable()

export class AuthInterceptor implements HttpInterceptor {

intercept(req: HttpRequest<any>, next: HttpHandler) {

const cloned = req.clone({ headers: req.headers.set('Authorization', 'Bearer TOKEN') });

return next.handle(cloned);

}

}