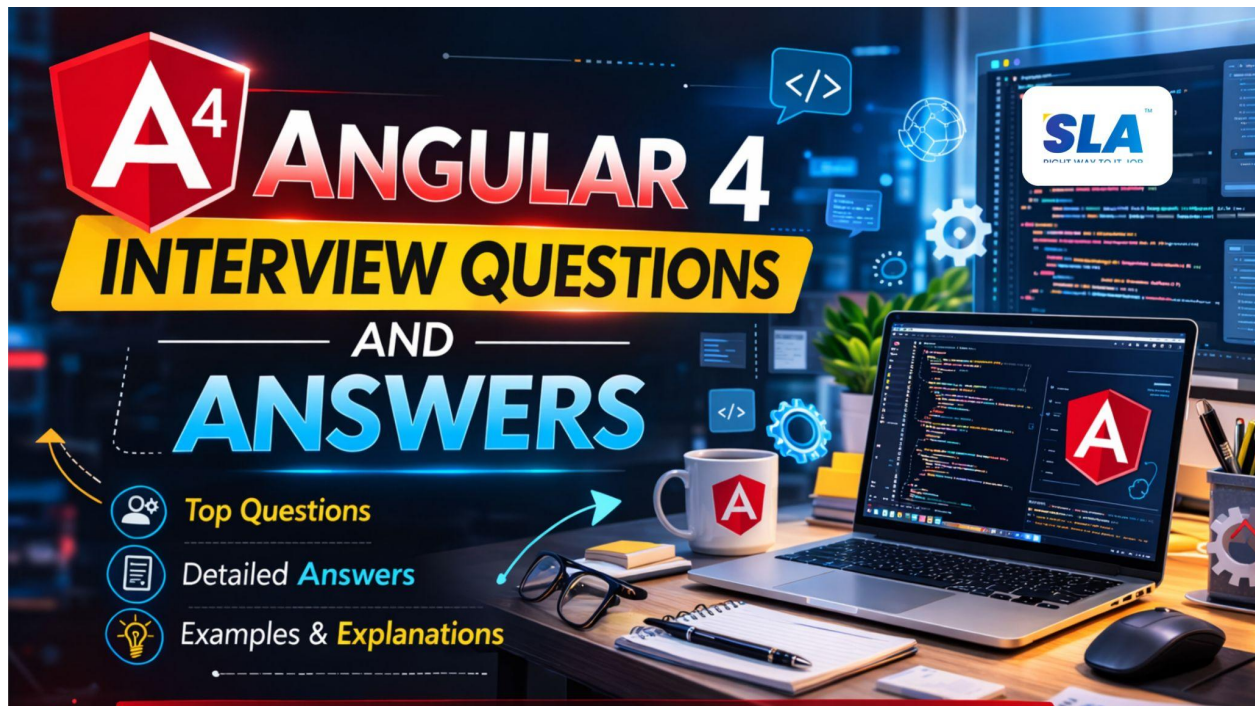




Angular 4 Interview Questions and Answers

Introduction

You plan an Angular developer interview. Interview panels test Angular knowledge. Topics include components, modules, routing, services, and project structure. Clear understanding of these areas helps you explain development tasks and project flow.

Angular 4 Interview Questions and Answers support focused preparation before technical rounds. You review common interview topics used in Angular development roles. This practice improves recall during interview discussions.

The following **Angular 4 Interview Questions and Answers** present topics often used by recruiters while evaluating Angular developer candidates.

List of Angular 4 Interview Questions for Freshers

1. What is Angular 4?
2. What is binding?
3. Explain the various binding types in Angular 4.
4. What makes Angular 4 faster than the previous version?
5. What is TypeScript?
6. What are decorators in Angular?
7. List some benefits of using Angular
8. What are Templates in Angular?
9. What are Annotations in Angular?
10. What are Components in Angular?
11. What is Angular Router and use of this?
12. What are the Modules in Angular 4?
13. Describe the features of Angular 4?
14. What is Routing?
15. What is JIT?

Angular 4 Technical Interview Questions and Answers for Freshers

1. What is Angular 4?

Angular 4, created by Google, is a tool for building dynamic websites. It follows Angular 2 and comes before Angular 5. The aim of Angular 4 is to improve website functionality while reducing the amount of code needed. It simplifies tasks such as connecting data in both directions, injecting dependencies, and organizing a website's structure. Many people find Angular 4 helpful because it makes the process of building modern websites easier and more efficient.

2. What is binding?

Binding in web development refers to the dynamic connection between different elements of a webpage. It involves linking data or properties of elements in the user interface with corresponding values in the underlying code or data source.

3. Explain the various binding types in Angular 4.

In Angular 4, different binding types enable dynamic changes on webpages:

- Interpolation: Embed expressions in double curly braces to display changing content.
- Property Binding: Adjust HTML element properties by using square brackets, linking data from the component to the view.
- Event Binding: Respond to view events, like clicks, triggering actions in the component.

- **Two-Way Binding:** Combine property and event binding for synchronized updates between the view and component. These methods significantly enhance interactivity in Angular 4 web applications.

4. What makes Angular 4 faster than the previous version?

Angular 4 is faster than its predecessor, Angular 2, because of some key improvements:

- **Smaller Size:** Angular 4 creates smaller and more efficient code bundles, making it quicker to download and run.
- **Ahead-of-Time Compilation (AOT):** Angular 4 uses AOT compilation by default, translating the code during build, which speeds up how the application runs.
- **Better Change Detection:** Angular 4 has improved how it detects changes, making it more efficient and reducing the overall workload.
- **Separate Animation Package:** Animations are now in a separate package, so they only get included if needed, avoiding unnecessary work for applications without animations.
- **RxJS 5.0 Integration:** Angular 4 includes RxJS 5.0, which helps with handling asynchronous operations more efficiently.
- **Improved Core Directives:** **Core** directives like `NgIf` and `NgFor` are better, making rendering and data binding more efficient.
- **Bug Fixes and General Improvements:** Angular 4 includes many fixes and improvements, addressing issues to make the framework perform better.

5. What is TypeScript?

TypeScript is a programming language created by Microsoft, expanding on JavaScript by adding static typing. It enables developers to specify and

enforce variable types, function parameters, and return values. TypeScript enhances coding with features like better navigation and autocompletion. It compiles into regular JavaScript, making it compatible with existing code. TypeScript is beneficial for creating scalable and maintainable applications, particularly in larger projects where static typing helps identify errors early in development.

Check out: [Web Development Training in Chennai](#)

6. What are decorators in Angular?

In Angular, decorators are like labels that you put on classes to give them special instructions. They add extra information to the classes, making them behave in a certain way. For instance, the `@Component` decorator is used to set up and configure a component in Angular. Decorators use the `@decorator` syntax, and they help Angular organize and understand different parts of your code, like components, services, and modules, making it easier to work with and extend your application. Angular has various types of decorators, each serving a specific purpose:

- **@Component:** Configures and sets up a component, defining details like its template and styles.
- **@Directive:** Enhances elements on the webpage by adding extra features.
- **@Pipe:** Creates custom filters for transforming data in templates.
- **@Injectable:** Informs Angular to provide dependencies to a service.
- **@NgModule:** Defines and configures groups of related parts in your Angular application.

7. List some benefits of using Angular

Angular offers several advantages that contribute to its popularity in web development:

- Angular is a top choice for web development with many benefits.
- It simplifies the creation of reusable parts for a more scalable and manageable website.
- Two-way data binding keeps data and display in sync without excessive code.
- Built-in dependency management ensures modular and easily testable code.
- Directives like `ngIf` and `ngFor` simplify complex tasks in development.
- Angular seamlessly handles compatibility across different browsers.
- TypeScript integration improves code quality with static typing.
- The Angular CLI streamlines project management tasks.
- Overall, Angular excels in building strong and flexible web applications.

**Take your Knowledge
Test Report**

Check your Score



8. What are Templates in Angular?

In Angular, templates act as guides for web page appearance and behavior, combining HTML with Angular instructions to define data presentation. They facilitate view structure, and Angular uses them to update the webpage dynamically with changing data. This separation of templates and components simplifies application development, promoting an organized and scalable approach.

9. What are Annotations in Angular?

In Angular, annotations mean adding special information to classes using decorators like `@Component` or `@Injectable`. These decorators provide details that help Angular work with classes more efficiently. For example, `@Component` sets up a component, and `@Injectable` signals that a class can be injected as a service. These annotations are crucial for Angular to manage different parts of an application, making it simpler to handle components, services, and other aspects of the code.

10. What are Components in Angular?

In Angular, components are the fundamental building blocks that encapsulate a part of an application's logic and user interface. A component consists of a TypeScript class defining behavior, an HTML template specifying the view, and metadata offering configuration details. Components use selectors to embed custom HTML elements in templates. Data binding facilitates communication between the class and template, and lifecycle hooks enable actions at specific stages. This modular approach supports the creation of reusable and maintainable code for constructing web applications in a structured manner.

Check out: [MERN Full Stack Developer Course](#)

11. What is Angular Router and use of this?

Angular Router, a pivotal module in Angular, facilitates seamless navigation and routing in web applications. Its features simplify the development of single-page applications, providing a structured approach to handle routing and views. For additional insights, consider exploring more about Angular Routing.

Key aspects of Angular Router include:

- **Declarative Routing:** Developers define navigation paths through route configurations, mapping URL patterns to specific components.
- **Dynamic Views:** The router dynamically updates views based on the current URL, ensuring a smooth and responsive user experience.
- **Nested Routing:** Support for nested routes allows the creation of intricate application layouts with multiple views.
- **Route Parameters:** Routes can include parameters for passing dynamic data through the URL.
- **Route Guards:** Custom guards enable control over access to routes, enforcing authentication or authorization criteria.
- **Lazy Loading:** Angular Router supports lazy loading, improving application performance by loading components only when necessary.
- **Navigation Events:** Developers can subscribe to events emitted by the router, allowing for custom logic execution based on navigation events.
- **Router Outlet:** The <router-outlet> directive acts as a placeholder in the application's template, dynamically rendering components associated with the current route. This feature enhances flexibility and maintainability in building interactive web applications.

12. What are the Modules in Angular 4?

In Angular 4, modules help organize and structure apps. The main module, like AppModule, starts the app. Feature modules, group related parts, and shared modules let you reuse code. Core modules handle global services. Third-party modules add extra features. This modular setup makes Angular 4 apps easier to manage and expand.

13. Describe the features of Angular 4?

Angular 4 boasts several features that enhance development:

- **Optimized View Engine:** Angular 4 enhances performance by reducing generated code size in the view engine.
- **Animation Package Separation:** Animation features are modularized in a separate package, allowing for efficient bundle size management.
- **Template Enhancements:** Improved ngIf and ngFor functionality, introducing the 'as' keyword and 'else' support in ngIf.
- **Global HTTP Interceptor:** Angular 4 introduces HTTP interceptors for globally modifying requests and responses.
- **Streamlined AOT Compilation:** Ahead-of-Time compilation is refined, resulting in faster application startup.
- **Router ParamMap:** The router now provides ParamMap for simplified extraction of route parameters.
- **FormGroupDirective and FormControlDirective:** These directives offer cleaner access to form-related functionality in templates.

14. What is Routing?

Routing is the mechanism in web development that manages navigation between pages or views in an application. It involves defining URL paths for specific components, facilitating dynamic content changes without reloading the entire page.

15. What is JIT?

JIT, or Just-in-Time compilation in Angular 4, dynamically translates and optimizes code during runtime, right before execution. This on-the-fly compilation boosts performance as the program runs.

List of Angular 4 Interview Questions for Experienced

- 16. What is ChangeDetectionStrategy.OnPush in Angular and when do you use it?
- 17. What is the difference between Observables and Promises in Angular?
- 18. How do you optimise performance in a large Angular application?
- 19. What is the purpose of TrackBy in ngFor?
- 20. How do you handle HTTP requests in Angular?
- 21. What are Guards in Angular Routing?
- 22. What is the difference between Smart Components and Presentational Components?

Angular 4 Technical Interview Questions and Answers for Experienced

16. What is ChangeDetectionStrategy.OnPush in Angular and when do you use it?

Angular normally checks every component during change detection. This behavior may reduce performance in large applications. OnPush strategy limits change detection to specific conditions such as input property changes or event triggers. Developers use this strategy to reduce unnecessary checks and improve performance in complex applications.

17. What is the difference between Observables and Promises in Angular?

Observables support multiple values over time. They allow cancellation and support many operators for data processing. Promises return a single value and resolve once. Angular commonly uses Observables with HTTP requests and asynchronous data streams through RxJS.

18. How do you optimise performance in a large Angular application?

Performance optimisation often includes several practices.

- Lazy loading modules
- Using OnPush change detection
- Reducing unnecessary DOM updates
- TrackBy function in ngFor loops
- Using Ahead of Time compilation
- Minimizing bundle size

These steps improve application speed and reduce rendering workload.

**Take your Knowledge
Test Report**

Check your Score



19. What is the purpose of TrackBy in ngFor?

Angular recreates DOM elements when a list changes. This process affects performance when handling large lists. TrackBy allows Angular to identify

items using a unique key. Angular updates only the changed items instead of re-rendering the entire list.

20. How do you handle HTTP requests in Angular?

Angular uses the HttpClient service to communicate with APIs. Developers inject the service into components or services and send requests such as GET, POST, PUT, and DELETE. The service returns Observables, which allow developers to manage asynchronous responses efficiently.

Check out: [MEAN Full Stack Developer Course](#)

21. What are Guards in Angular Routing?

Route guards control access to routes. They check conditions before navigation occurs. Guards support tasks such as authentication and authorisation.

Common guard types include:

- CanActivate
- CanDeactivate
- CanLoad
- Resolve

These help secure application routes.

22. What is the difference between Smart Components and Presentational Components?

Smart components manage application logic and data fetching. Presentational components focus on displaying data and handling UI

interactions. This separation improves maintainability and reusability in large applications.

Conclusion

Angular skills support many front end developer roles. Recruiters check your understanding of components, routing, services, and application structure during interviews. Clear knowledge of these topics helps you explain development tasks with confidence.

Review **Angular 4 Interview Questions and Answers** and practice concepts used in real projects. Regular study improves technical understanding and helps you perform better during developer interviews.

Build strong development skills through guided learning. Join our [Angular JS Training in Chennai](#). You practice project development, learn core Angular concepts, and prepare for front end developer job roles.