

35 Angular Interview Questions and Answers



SAMSUNG



McKESSON

STARTUPS FROM



Whether you're an entrepreneur, a project manager in a large enterprise, or a CTO, you are well aware that the success of your project rests on your ability to find top developers.

In this guide, you'll find example interview questions and answers you can refer to when seeking a new Angular developer to build your web applications. You'll also get some practical advice on how to use these questions to reliably identify the best people for the job.

These interview questions are for Angular versions 2, 4, 5, 6 (simply referred to as 'Angular'). If you are looking for developers for the original *AngularJS* JavaScript framework, [click here for the AngularJS interview questions](#).

First Things First: Select Your Job Requirements

The most common mistake interviewers make is to start questioning candidates without a clear definition of the type of developer they are looking for. Building Angular applications requires many different skills, and you need the developers with the right ones for your unique web app. You also need to define any 'soft' skills they will need to work effectively in your team.

Some example requirements include:

- **Essential web development skills** – E.g. HTML, CSS, JavaScript
- **Angular-specific skills** – E.g. TypeScript, Building Single Page Applications
- **Library/toolkit experience** – E.g. Jasmine and Karma
- **Design skills** – Performance optimization, Application security
- **Communication skills** – Discussing problems and constraints
- **Initiative** – If they'll have to figure out solutions by themselves

Avoid making a laundry list of non-essential skills for your perfect Angular developer. Instead, focus on what your candidate will actually be doing day-to-day. Keep your requirements list as short as

possible. Cut anything they can do without or learn on the job.

With clearly stated requirements, you'll be able to choose the right coding interview questions and have a much better idea of what kinds of responses you are looking for.

The questions in this guide are broken down into two sections: Basic Interview Questions and Advanced Interview Questions. You should use the basic Angular programming interview questions in this section if you're looking for a junior or entry level developer with less experience.

Skip to the [Advanced](#) developer sections for Angular interview questions and answers for experienced developers)

Angular Basic Interview Questions

These basic questions are for interviewing junior developers. It's important to know before hiring a Junior Angular developer that they will need guidance from more experienced developers. A great junior dev will be able to execute tasks given by senior developers. Skip to the advanced section if you need someone who can lead your Angular project.

Skill Requirements for Junior Java Developers

- Basic HTML, CSS, JavaScript and TypeScript skills
- Foundational Angular knowledge
- Learning on the job
- Following instructions and receiving feedback
- Thinking like a programmer

Example Angular Basic Interview Questions and Answers

Note: Important keywords are underlined in the answers. Bonus points if the candidate mentions them!

Question 1: Write an example of a simple HTML document with some header information and page content.

Requirement: Basic HTML skills

Answer: HTML documents are all different, but they follow a basic structure of head and body. Here you're checking the candidate has a good grasp of HTML document structure and basic tags such as DOCTYPE, html, head, title, meta, body, h1, p, etc.

For example:

```
<!DOCTYPE html>

<html>

  <head>

    <title>Page Title</title>

    <meta charset="UTF-8">

    <meta name="description" content="Page description">

  </head>

  <body>

    <h1>Interview Example Web Page</h1>

    <p>Some content goes here</p>

  </body>

</html>
```

Question 2: Briefly explain the CSS box model. Write some code snippets to describe show what you mean.

Requirement: Basic CSS skills

Answer: CSS is the language that describes how webpages look. Every front-end developer should have good CSS knowledge. Good candidates will be able to describe CSS concepts concisely.

The CSS box model refers to the layout and design of HTML elements. It's a box shape that wraps around each HTML element. A box is made up of its content, padding, border and margin.

- Content of the box
- Padding
- Border
- Margin

(the same padding on all 4 sides)

```
padding: 25px;
```

(padding for the top, right, bottom, left)

```
padding: 25px 50px 75px 100px;
```

(top/bottom padding 25 pixels, right/left padding 50 pixels)

```
padding: 25px 50px;
```

Question 3: In JavaScript, how can the style of an HTML element be changed?

Requirement: Basic JavaScript skills

Answer: For example, to change the font size:

```
document.getElementById("someElement").style.fontSize = "20";
```

Question 4: Write some code for a basic class in TypeScript with a constructor and a method.

Requirement: Basic TypeScript skills

Answer: Here's a simple class that is created with a greeting message which can be retrieved with the `greet()` function

```
class Greeter {
    greeting: string;
    constructor(message: string) {
        this.greeting = message;
    }
    greet() {
        return "Hello, " + this.greeting;
    }
}

let greeter = new Greeter("world");
```

Question 5: What are Single Page Applications? How do they work in Angular?

Requirement: Foundational Angular knowledge

Answer: Single Page Applications (SPAs) are web applications that use only one HTML page. As the user interacts with the page, new content is dynamically updated on that master page. Navigation between pages happens without refreshing the whole page. Angular uses AJAX and to dynamically update HTML elements. Angular Routing can be used to make SPAs. The result is an application that feels more like a desktop app rather than a webpage.

Question 6: What's the basic syntax of a Decorator in Angular?

Requirement: Foundational Angular Knowledge

Answer: @() with optional parameters.

Question 7: What is [(ngModel)] used for?

Requirement: Foundational Angular Knowledge

Answer: Two-way data binding.

Question 8: What are the basic parts of an Angular application?

Requirement: Foundational Angular Knowledge

Answer: Modules, Component, Data Binding, Template, Directives, Dependency Injection, Services, Routing.

Question 9: Tell me about a time you received feedback on a task.

Requirement: Following instructions and receiving feedback

Answer: This is a typical open-ended question. The candidate should demonstrate they can accept, understand and act on feedback.

Question 10: Describe how you would approach solving (some problem) on a high level?

Requirement: Thinking like a programmer

Answer: In this question, the problem should be directly related to the work the candidate will actually be doing. You aren't looking for a perfect answer or even necessarily a correct answer. Instead, listen to how they approach solving a problem, their ability to break a problem down into parts, and if they can anticipate problems.

Question 11: What are some advantages of using Angular framework for building web applications?

Requirement: Expert Angular knowledge

Answer: Advantages of using the Angular framework include:

- Angular does lots of things for you under the hood. It saves time for developers by doing a lot of the work for them like writing tedious DOM manipulation tasks
- TypeScript and the Angular framework allow you to catch errors much earlier
- In many cases has faster performance than traditional web development techniques
- Can give web apps the feel of a desktop application
- It separates out the code of an application to make it easier for multiple developers to work on an app and easier to test
- More consistent code base that's easy to maintain
- Big developer community

Question 12: What function is called when an object is created in TypeScript? What is its basic syntax in TypeScript code?

Requirement: TypeScript knowledge

Answer: The constructor function is called. Its syntax is: `Constructor()`

Question 13: In Angular, how can you interact between Parent and Child components?

Requirement: Expert Angular knowledge

Answer: When passing data from Parent to Child component, you can use the @Input decorator in the Child component. When passing data from Child to Parent component, you can use the @Output decorator in the Child component.

Question 14. Write an example usage of ngFor for displaying all items from an array 'Items' in a list with .

Requirement: Expert Angular knowledge

Answer:

```
<li *ngFor="let item of Items">
  {{item}}
</li>
```

Question 15. What is the sequence of Angular Lifecycle Hooks?

Requirement: Foundational Angular knowledge

Answer: OnChange() – OnInit() – DoCheck() – AfterContentInit() – AfterContentChecked() – AfterViewInit() – AfterViewChecked() – OnDestroy().

Question 16: If you provide a service in two components' "providers" section of @Component decorator, how many instances of service shall get created?

Requirement: Foundational Angular knowledge

Answer: 2

Question 17: What is the main difference between constructor and ngOnInit?

Requirement: Foundational Angular knowledge

Answer: The constructor is a feature of the class itself, not Angular. The main difference is that Angular will launch ngOnInit after it has finished configuring the component. Meaning, it is a signal through which the @Input() and other binding properties and decorated properties are available in ngOnInit, but are not defined within the constructor by design.

Angular Advanced Interview Questions

Here are some more advanced and technical interview questions and answers for experienced Angular developers. Use them to pick out the right Angular developers with the skills to build your web app.

An expert Angular developer has to know the Angular framework inside and out. They will also be able to design efficient applications, write clean and robust code, work effectively with your team, and pass on their experience to junior developers. Remember to list your requirements before you choose your questions.

Skill Requirements for Senior Angular JS Developers

- Expert Angular knowledge and its different versions (2, 4, 5, 6, 7)
 - Component based architecture
 - Designing for specific requirements (e.g. security, scalability, optimization)
 - Maintaining and upgrading applications
 - Experience in frameworks/toolkits/libraries you use
 - Efficient programming and clean code
 - Debugging
 - End-to-end testing and unit testing
 - Leadership skills
 - Clear communication skills
 - Mentoring less experienced developers
-

Example Angular Advanced Interview Questions and Answers

Note: Important keywords are underlined in the answers. Listen out for them!

Question 18: What modules should you import in Angular to use [(ngModel)] and reactive forms?

Requirement: Middle Angular knowledge, Tools/libraries

Answer: FormsModule and Reactiveforms Module.

Question 19: How similar is AngularJS to Angular 2?

Requirement: Middle Angular knowledge

Answer: Both are front-end frameworks maintained by Google, but Angular 2 is not a simple update of AngularJS, it is a new framework written from scratch. Updating an app from AngularJS to Angular 2 would require a complete rewrite of the code.

Question 20: What were some features introduced in the different versions of Angular (2, 4, 5 and 6)?

Requirement: Expert Angular knowledge, Component-based architecture

Answer:

Angular 2:

- Complete rewrite of the Angular framework
- Component-based rather than controllers/view/\$scope. This allows more code to be reused, easier communication between components and easier testing
- Much faster
- Support for mobile devices
- More language choices such as TypeScript

Angular 4:

- An update to Angular 2, not a complete rewrite. Updating from Angular 2 to 4 just requires updating the core libraries
 - Improvements to Ahead-of-time (AOT) generated code
 - Support for new versions of TypeScript
 - Animation packages are removed from the core package
 - Else block
-

Angular 5:

- Focused on making Angular smaller and faster to use
- Http is depreciated and HttpClient API client is now recommended for all apps
- Supports TypeScript 2.3
- Introduction of a build optimizer
- Angular Universal State Transfer API
- Improvements to the Angular Compiler
- Router Lifecycle Events
- Better cross-browser standardization

Angular 6:

- Better service worker support
- Better URL serialization
- Ivy rendering engine
- ng update and ng add
- <template> element completely removed
- Angular Elements/Custom Elements
- Form validation changes
- Schematics

Question 21: What is Transpiling in Angular?

Requirement: Middle Angular knowledge, TypeScript

Answer: Transpiling means converting the source code of one programming language into another. In Angular, that usually means converting TypeScript into JavaScript. You can write the code for your Angular application in TypeScript (or another language such as Dart) that is then transpiled to JavaScript for the application. This happens internally and automatically.

Question 22: What is AOT Compilation?

Requirement: Expert Angular knowledge, Optimization

Answer: AOT refers to Ahead-of-time compilation. In Angular, it means that the code you write for your application is compiled at build time before the application is run in a browser. It's an alternative to Just-in-time compilation, where code is compiled just before it is run in the browser. AOT compilation can lead to better application performance.

Question 23: What are HTTP Interceptors?

Requirement: Middle Angular knowledge

Answer: Interceptor is just a fancy word for a function that receives requests/responses before they are processed/sent to the server. You should use interceptors if you want to pre-process many types of requests in one way. For example, you need to set the authorization header Bearer for all requests:

```
token.interceptor.ts

import { Injectable } from '@angular/core';
import { HttpInterceptor, HttpRequest, HttpHandler, HttpEvent } from
'@angular/common/http';
import { Observable } from 'rxjs/Observable';

@Injectable()
export class TokenInterceptor implements HttpInterceptor {

    public intercept(req: HttpRequest<any>, next: HttpHandler):
Observable<HttpEvent<any>> {
        const token = localStorage.getItem('token') as string;

        if (token) {
            req = req.clone({
                setHeaders: {
                    'Authorization': `Bearer ${token}`
                }
            });
        }
        return next.handle(req);
    }
}
```

And register the interceptor as singleton in the module providers:

```
app.module.ts

import { NgModule } from '@angular/core';
import { BrowserModule } from '@angular/platform-browser';
import { HTTP_INTERCEPTORS } from '@angular/common/http';
```

```

import { AppComponent } from './app.component';
import { TokenInterceptor } from './token.interceptor';

@NgModule({
  imports: [
    BrowserModule
  ],
  declarations: [
    AppComponent
  ],
  bootstrap: [AppComponent],
  providers: [{
    provide: HTTP_INTERCEPTORS,
    useClass: TokenInterceptor,
    multi: true // < - - - - an array of interceptors can be registered
  }]
})

export class AppModule {}

```

Question 24: How many Change Detectors can there be in the whole application?

Requirement: Expert Angular knowledge

Answer: Each component has its own ChangeDetector. All Change Detectors are inherited from AbstractChangeDetector.

Question 25: What change detection strategies do you know?

Requirement: Expert Angular knowledge

Answer: There are two strategies – Default and OnPush. If all components use the default strategy, Zone checks the entire tree regardless of where the change occurred. To inform Angular that we will comply with the performance improvement conditions, we need to use the onpush change detection strategy. This will tell Angular that our component depends only on the input and any object that is passed to it should be considered immutable. This is all built on the Principle of the mile automaton, where the current state depends only on the input values.

Question 26: What is Change Detection, how does Change Detection Mechanism work?

Requirement: Expert Angular knowledge

Answer: Change Detection is the process of synchronizing a model with a view. In Angular, the flow of information is unidirectional, even when using the ng Model to implement two-way binding, which is syntactic sugar on top of a unidirectional flow.

Change Detection Mechanism-moves only forward and never looks back, starting from the root (root) component to the last. This is the meaning of one-way data flow. The architecture of an Angular application is very simple — the tree of components. Each component points to a child, but the child does not point to a parent. One-way flow eliminates the need for a \$digest loop.

Question 27: How do you update the view if your data model is updated outside the 'Zone'?

Requirement: Expert Angular knowledge

Answer: Using the ApplicationRef.prototype.tick method, which will run change detection on the entire component tree.

1. Using NgZone.prototype.run method, which will also run change detection on the entire tree. The run method under the hood itself calls tick, and the parameter takes the function you want to perform before tick.
2. Using the ChangeDetectorRef.prototype.detectChanges method, which will launch change detection on the current component and its children.

Question 28: Why do we need lazy loading of modules and how is it implemented?

Requirement: Middle Angular knowledge

Answer: Lazy loading of modules is needed to break the code into pieces. When downloading the app in the browser, it doesn't load all of the application code. During the transition to the route with lazy loading, the module has to load the code into a browser.

Exemple for using lazy loading modules:

```
{ path: 'example', loadChildren: './example/example.module#ExampleModule', component:
PublicComponent },
```

Question 29: What are Core and Shared modules for?

Requirement: Middle Angular knowledge

Answer: A *Shared module* serves as a generic module for all modules, components, directives, pipes, etc., which are not required to be in a single copy for the application but need to be imported into many different modules.

A *Core module* is a place to store services that you need to have in the form of singleton for the entire application (for example, a user authorization service with data storage about it).

Question 30: What are some points to consider when optimizing an Angular 6 application for performance?

Requirement: Application performance optimization

Answer: There are many ways, some ideas include: AOT compilation, bundling and uglifying the application, tree shaking, lazy loading, separating dependencies and devDependencies, Using OnPush and TrackBy, removing unnecessary 3rd party libraries and import statements, avoid computing values within the template.

Question 31: What are some important practices to secure an Angular application?

Requirement: Designing for security

Answer: Some basic guidelines include:

- Check that all requests come from within your own web app and not external websites
- Sanitize all input data
- Use Angular template instead of DOM APIs
- Content Security Policies
- Validate all data with server-side code
- Use an offline template compiler
- Avoid including external URLs in your application
- Make JSON responses non-executable
- Keep all libraries and frameworks up-to-date

Question 32: What's the difference between unit testing and end-to-end testing? What are some testing tools you would use for an Angular application?

Requirement: End-to-end and unit testing

Answer: Unit testing is a technique to test that isolated segments of code are functioning properly. End-to-end testing involves checking that entire sets of components to make sure they are working together properly and that the application is working as you would expect. End-to-end tests often simulate user interactions to test that an app is functioning as it should. Jasmine and Karma are all great testing tools.

Open-Ended Questions

Once you've established that your developer is an expert with some Angular coding interview questions, you should ask some less technical questions. These should spark a discussion and you should tailor them to fit your own job requirements. Don't be afraid to ask follow up questions!

Question 33: Describe a time you fixed a bug/error in an application. How did you approach the problem? What debugging tools did you use? What did you learn from this experience?

Requirement: Debugging, Breaking down a problem into parts

Debugging is one of the key skills for any software developer. However, the real skill is in breaking the problem down in a practical way rather than finding small errors in code snippets. Debugging often takes hours or even days, so you don't have time in an interview setting. Asking these questions will give you an idea of how your candidate approaches errors and bugs.

Answer: In the candidate's response you should look out for things like:

- A measured, scientific approach
- Breaking down the problem into parts
- Finding out how to reproduce the error
- Expressing and then testing assumptions
- Looking at stack traces
- Getting someone else to help/take a look
- Searching the internet for others that have had the same problem
- Writing tests to check if the bug returns
- Checking the rest of the code for similar errors
- Turn problems into learning experiences

Question 34: What's the most important thing to look for or check when reviewing another team member's code?

Requirement: Mentoring less experienced developers, Leadership skills

Answer: Here you're checking for analysis skills, knowledge of mistakes that less experienced developers make, keeping in mind the larger project and attention to detail.

A good answer might mention code functionality, readability and style conventions, security flaws that could lead to system vulnerabilities, simplicity, regulatory requirements, or resource optimization.

Question 35: What tools & practices do you consider necessary for Continuous Integration and

Delivery of an Angular application?

Requirement: DevOps systems design, Maintaining and upgrading applications

Summary

Hiring the right people for your development team is critical to the success of your project. Remember that you should not be aiming to hire the perfect Angular developer, but rather the right person for the job at hand.

With the help of this information and sample interview questions on Angular, you can ensure that the hiring process goes smoothly and successfully – allowing you to hire a great programming team to get your project completed on time and on budget.

Finally, you can simply print out this ready-to-use cheat sheet of questions and answers and bring with you to the interview with the candidates.