

## Related Tutorials

🔴 Angular

🌙 jQuery

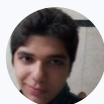
⚛️ React

TS TypeScript

🌱 Vue.js

# Angular Interview Questions (Frequently Asked)

Posted in [Angular](#), [Interview Questions](#)



Youssef Nader

Last Updated 27 Jan, 2020

## Jobs suited for you



### Technical Trainer/Instructor

>h Hackr.io • 📍 Gurugram

₹6,00,000/yr

🕒 posted today

### Senior Software Engineer

🚀 SquareBoat • 📍 Gurugram

₹8,00,000/yr

🕒 posted today

### Project Manager

🚢 SquareBoat • 📍 Gurugram

₹6,00,000/yr

🕒 posted today

## Engineering Manager

🚢 SquareBoat • 📍 Gurugram

₹20,00,000/yr

🕒 posted today

## Tech Lead

🚢 SquareBoat • 📍 Gurugram

₹16,00,000/yr

🕒 posted today

[View All →](#)

## Recommended Learning

Angular 8 - The Complete Guide  
(udemy.com)

Shaping up with AngularJS  
(pluralsight.com)

The Complete Angular Course:  
Beginner to Advanced (udemy.com)

[VIEW MORE](#)

**Advertise with Us!**

Share:



[29 Comments](#)

Preparing for getting your dream job of an Angular developer? Or, simply planning to switch to being an Angular developer? In this article, we're going to look at some of the frequently asked angular questions that you can expect to be coming your way during an Angular interview for the role of an Angular developer.

In addition to these concept-based questions, there will be a requirement for a (or maybe multiple) coding test(s). So, keep up your programming practice for the interview. If you wish to hone your Angular development skills further, here are some of the best [Angular tutorials](#) that can help you out.

## Angular Interview Questions

So, ready to see how well your preparation is going? Without further ado, here we present you with some important Angular interview questions that you might expect in your interview. These questions are also valid for your Angular 6 interview questions.

### Question: What is Angular?

**Answer:** Angular is a TypeScript-based open-source web application framework, developed and maintained by Google. It offers an easy and powerful way of building front end web-based applications.

Angular integrates a range of features like declarative templates, dependency injection, end-to-end tooling, etc. that facilitates web application development.

### Question: Define the ng-content Directive?

**Answer:** Conventional HTML elements have some content between the tags. For instance:

<p>Put your paragraph here</p>

Now consider the following example of having custom text between angular tags:

We have a variety of advertising options which would give your courses an instant visibility to a very large set of developers, designers and data scientists.

[VIEW PLANS](#)

<app-work>This won't work like HTML until you use ng-content Directive</app-work>

However, doing so won't work the way it worked for HTML elements. In order to make it work just like the HTML example mentioned above, we need to use the ng-content Directive. Moreover, it is helpful in building reusable components.

Know more about the [ng-content directive](#).

**Question: Please explain the various features of Angular.**

**Answer:** There are several features of Angular that makes it an ideal front end JavaScript framework. Most important of them are described as follows:

- **Accessibility Applications**

Angular allows creating accessible applications using ARIA-enabled components, built-in a11y test infrastructure, and developer guides.

**Interested in Angular?  
Enquire for the best  
course suited for you.**

**SUBMIT**

- **Angular CLI**

Angular provides support for command-line interface tools. These tools can be used for adding components, testing, instant deploying, etc.

- **Animation Support**

Angular's intuitive API allows the creation of high-performance, complex animation timelines with very little code.

- **Cross-Platform App Development**

Angular can be used for building an efficient and powerful desktop, native, and progressive web apps. Angular provides support for building native mobile applications using Cordova, Ionic, or NativeScript.

Angular allows creating high performance, offline, and zero-step installation progressive web apps using modern web platform capabilities. The popular JS framework can also be used for building desktop apps for Linux, macOS, and Windows.

- **Code Generation**

Angular is able to convert templates into highly-optimized code for modern JavaScript virtual machines.

- **Code Splitting**

With the new Component Router, Angular apps load quickly. The Component Router offers automatic code-splitting so that only the code required to render the view that is requested by a user is loaded.

- **Synergy with Popular Code Editors and IDEs**

Angular offers code completion, instant errors, etc. with popular source code editors and IDEs.

- **Templates**

Allows creating UI views with a simple and powerful template syntax.

- **Testing**

Angular lets you carry out frequent unit tests using Karma. The Protractor allows running faster scenario tests in a stable way.

**Question: Demonstrate navigating between different routes in an Angular application.**

**Answer:** Following code demonstrates how to navigate between different routes in an

Angular app dubbed “Some Search App”:

```
import {Router} from "@angular/router";
.
.
.
@Component({
  selector: 'app-header',
  template: `
<nav class="navbar navbar-light bg-faded">
  <a class="navbar-brand" (click)="goHome()">Some Search App</a>
  <ul class="nav navbar-nav">
    <li class="nav-item">
      <a class="nav-link" (click)="goHome()">Home</a>
    </li>
    <li class="nav-item">
      <a class="nav-link" (click)="goSearch()">Search</a>
    </li>
  </ul>
</nav>
`
})
class HeaderComponent {
  constructor(private router: Router) {}
  goHome() {
    this.router.navigate(['']);
  }
  goSearch() {
    this.router.navigate(['search']);
  }
}
```

**Question:** Could you explain services in Angular?

**Answer:** Singleton objects in Angular that get instantiated only once during the lifetime of an application are called services. An Angular service contains methods that maintain the data throughout the life of an application.

The primary intent of an Angular service is to organize as well as share business logic, models, or data and functions with various components of an Angular application.

The functions offered by an Angular service can be invoked from any Angular component, such as a controller or directive.

**Question: Discuss the advantages and disadvantages of using Angular?**

**Answer:** Following are the various advantages of using Angular:

- Ability to add a custom directive
- Exceptional community support
- Facilitates client and server communication
- Features strong features, such as Animation and Event Handlers
- Follows the MVC pattern architecture
- Offers support for static template and Angular template
- Support for two-way data-binding
- Supports dependency injection, RESTful services, and validations

Disadvantages of using Angular are enumerated as follows:

- Complex SPAs can be inconvenient and laggy to use due to their size
- Dynamic applications do not always perform well
- Learning Angular requires a decent effort and time

**Question: Enumerate some salient features of Angular 7.**

**Answer:** Unlike the previous versions of Angular, the 7th major release comes with splitting in @angular/core. This is done in order to reduce the size of the same. Typically, not each and every module is required by an Angular developer. Therefore, in Angular 7 each split of the @angular/core will have no more than 418 modules.

Also, Angular 7 brings drag-and-drop and virtual scrolling into play. The latter enables loading as well as unloading elements from the DOM. For virtual scrolling, the latest

version of Angular comes with the package. Furthermore, Angular 7 comes with a new and enhanced version of the ng-compiler.

**Question: What is string interpolation in Angular?**

**Answer:** Also referred to as moustache syntax, string interpolation in Angular refers to a special type of syntax that makes use of template expressions in order to display the component data. These template expressions are enclosed within double curly braces i.e. {{ }}.

The JavaScript expressions that are to be executed by Angular are added within the curly braces and the corresponding output is embedded into the HTML code. Typically, these expressions are updated and registered like watches as a part of the digest cycle.

**Question: Explain Angular Authentication and Authorization.**

**Answer:** The user login credentials are passed to an authenticate API, which is present on the server. Post server-side validation of the credentials, a JWT (JSON Web Token) is returned. The JWT has information or attributes regarding the current user. The user is then identified with the given JWT. This is called authentication.

Post logging-in successfully, different users have a different level of access. While some may access everything, access for others might be restricted to only some resources. The level of access is authorization.

Here is a detailed post on Angular 7 - JWT Authentication Example & Tutorial: <http://jasonwatmore.com/post/2018/11/16/angular-7-jwt-authentication-example-tutorial>

**Question: Can you explain the concept of scope hierarchy in Angular?**

**Answer:** Angular organizes the \$scope objects into a hierarchy that is typically used by views. This is known as the scope hierarchy in Angular. It has a root scope that can further contain one or several scopes called child scopes.

In a scope hierarchy, each view has its own \$scope. Hence, the variables set by a view's view controller will remain hidden to other view controllers. Following is a typical representation of a Scope Hierarchy:

- Root \$scope
  - \$scope for Controller 1
  - \$scope for Controller 2
  - ...
  - ..
  - .
  - \$scope for Controller n

**Question: How to generate a class in Angular 7 using CLI?**

**Answer:**

```
ng generate class Dummy [options]
```

This will generate a class named Dummy.

**Question: Explain what is the difference between Angular and backbone.js?**

**Answer:** Following are the various notable differences between Angular and Backbone.js

- **Architecture**

Backbone.js makes use of the MVP architecture and doesn't offer any data binding process. Angular, on the contrary, works on the MVC architecture and makes use of two-way data binding for driving application activity.

- **Community Support**

Being backed by Google greatly ups the community support received by the Angular framework. Also, extensive documentation is available. Although Backbone.js has a good level of community support, it only documents on Underscore.js templates, not much else.

- **Data Binding**



Angular uses two-way data binding process and thus is a bit complex. Backbone.js, on the contrary, doesn't have any data binding process and thus, has a simplistic API.

- **DOM**

The prime focus of Angular JS is upon valid HTML and dynamic elements that imitate the underlying data for rebuilding the DOM as per the specified rules and then works on the updated data records.

Backbone.js follows the direct DOM manipulation approach for representing data and application architecture changes.

- **Performance**

Thanks to its two-way data binding functionality, Angular offers an impactful performance for both small and large projects.

Backbone.js has a significant upper hand in performance over Angular in small data sets or small webpages. However, it is not recommended for larger webpages or large data sets due to the absence of any data binding process.

- **Templating**

Angular supports templating via dynamic HTML attributes. These are added to the document to develop an easy to understand application at a functional level. Unlike Angular, Backbone.js uses [Underscore.js](#) templates that aren't fully-featured as Angular templates.

- **The Approach to Testing**

The approach to testing varies greatly between Angular and Backbone.js due to the fact that while the former is preferred for building large applications the latter is ideal for developing smaller webpages or applications.

For Angular, unit testing is preferred and the testing process is smoother through the framework. In the case of Backbone.js, the absence of a data binding process allows for a

swift testing experience for a single page and small applications.

- **Type**

Angular is an open-source JS-based front-end web application framework that extends HTML with new attributes. On the other hand, Backbone.js is a lightweight JavaScript library featuring a RESTful JSON interface and MVP framework.

**Question: How do Observables differ from Promises?**

**Answer:** As soon as a [promise](#) is made, the execution takes place. However, this is not the case with observables because they are lazy. This means that nothing happens until a subscription is made. While promises handle a single event, observable is a stream that allows passing of more than one event. A callback is made for each event in an observable.

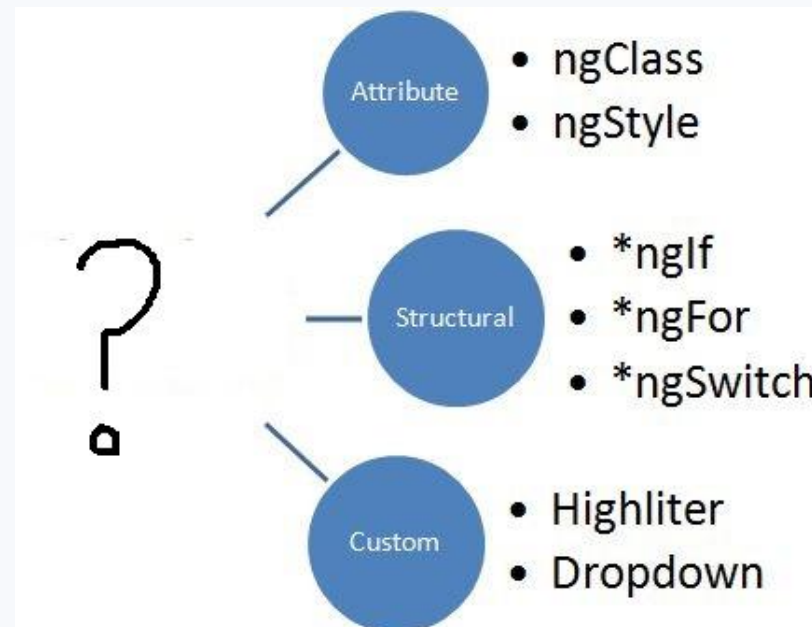
**Question: Please explain the difference between Angular and AngularJS?**

**Answer:** Various differences between Angular and AngularJS are stated as follows:

- **Architecture** - AngularJS supports the MVC design model. Angular relies on components and directives instead
- **Dependency Injection (DI)** - Angular supports a hierarchical Dependency Injection with unidirectional tree-based change detection. AngularJS doesn't support DI
- **Expression Syntax** - In AngularJS, a specific ng directive is required for the image or property and an event. Angular, on the other hand, use () and [] for binding an event and accomplishing property binding, respectively
- **Mobile Support** - AngularJS doesn't have mobile support while Angular does have
- **Recommended Language** - While JavaScript is the recommended language for AngularJS, TypeScript is the recommended language for Angular
- **Routing** - For routing, AngularJS uses `$routeProvider.when()` whereas Angular uses `@RouteConfig({...})`

- **Speed** - The development effort and time are reduced significantly thanks to support for two-way data binding in AngularJS. Nonetheless, Angular is faster thanks to upgraded features
- **Structure** - With a simplified structure, Angular makes the development and maintenance of large applications easier. Comparatively, AngularJS has a less manageable structure
- **Support** - No official support or updates are available for the AngularJS. On the contrary, Angular has active support with updates rolling out every now and then

**Question: Observe the following image:**



**What should replace the “?”?**

**Answer:** Directives. The image represents the types of directives in Angular; Attribute, structural, and custom.

**Question: Could you explain the concept of templates in Angular?**

**Answer:** Written with HTML, templates in Angular contains Angular-specific attributes and elements. Combined with information coming from the controller and model, templates are then further rendered to cater the user with the dynamic view.

**Question: Explain the difference between an Annotation and a Decorator in Angular?**

**Answer:** In Angular, annotations are used for creating an annotation array. They are only metadata set of the class using the Reflect Metadata library.

Decorators in Angular are design patterns used for separating decoration or modification of some class without changing the original source code.

**Question: What are directives in Angular?**

**Answer:** Directives are one of the core features of Angular. They allow an Angular developer to write new, application-specific HTML syntax. In actual, directives are functions that are executed by the Angular compiler when the same finds them in the DOM. Directives are of three types:

- Attribute Directives
- Component Directives
- Structural Directives

**Question: What are the building blocks of Angular?**

**Answer:** There are essentially 9 building blocks of an Angular application. These are:

1. **Components** – A component controls one or more views. Each view is some specific section of the screen. Every Angular application has at least one component, known as the [root component](#). It is bootstrapped inside the main module, known as the root module. A component contains application logic defined inside a class. This class is responsible for interacting with the view via an API of properties and methods.
2. **Data Binding** – The mechanism by which parts of a template coordinates with parts of a component is known as data binding. In order to let Angular know how to connect both

sides (template and its component), the binding markup is added to the template HTML.

3. **Dependency Injection (DI)** – Angular makes use of DI to provide required dependencies to new components. Typically, dependencies required by a component are services. A component's constructor parameters tell Angular about the services that a component requires. So, a dependency injection offers a way to supply fully-formed dependencies required by a new instance of a class.
4. **Directives** – The templates used by Angular are dynamic in nature. Directives are responsible for instructing Angular about how to transform the DOM when rendering a template. Actually, components are directives with a template. Other [types of directives](#) are attribute and structural directives.
5. **Metadata** – In order to let Angular know how to process a class, metadata is attached to the class. For doing so decorators are used.
6. **Modules** – Also known as NgModules, a module is an organized block of code with a specific set of capabilities. It has a specific application domain or a workflow. Like components, any Angular application has at least one module. This is known as the root module. Typically, an Angular application has several modules.
7. **Routing** – An Angular router is responsible for interpreting a browser URL as an instruction to navigate to a client-generated view. The router is bound to links on a page to tell Angular to navigate the application view when a user clicks on it.
8. **Services** – A very broad category, a service can be anything ranging from a value and function to a feature that is required by an Angular app. Technically, a service is a class with a well-defined purpose.
9. **Template** – Each component's view is associated with its companion template. A template in Angular is a form of HTML tags that lets Angular know that how it is meant to render the component.

**Question: Please explain the differences between Angular and jQuery?**

**Answer:** The single biggest difference between Angular and jQuery is that while the former is a JS frontend framework, the latter is a JS library. Despite this, there are some similarities between the two, such as both features DOM manipulation and provides support for animation.

Nonetheless, notable differences between Angular and jQuery are:

- Angular has two-way data binding, jQuery does not
- Angular provides support for RESTful API while jQuery doesn't
- jQuery doesn't offer deep linking routing though Angular supports it
- There is no form validation in jQuery whereas it is present in Angular

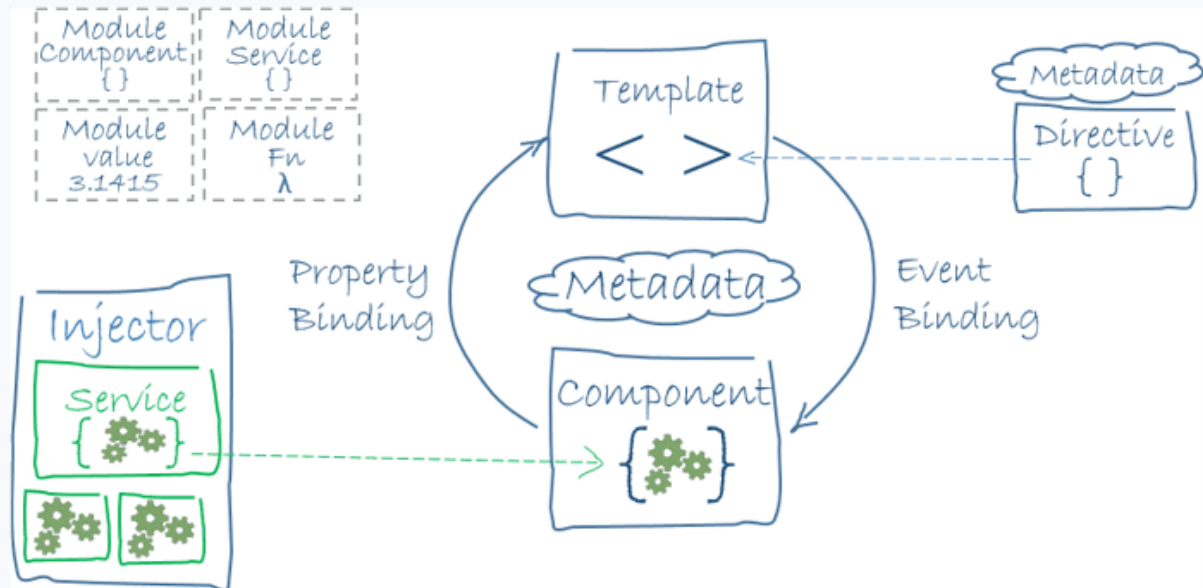
**Question: Could you explain the difference between Angular expressions and JavaScript expressions?**

**Answer:** Although both Angular expressions and JavaScript expressions can contain literals, operators, and variables, there are some notable dissimilarities between the two. Important differences between Angular expressions and JavaScript expressions are enlisted below:

- Angular expressions support filters while JavaScript expressions do not
- It is possible to write Angular expressions inside the HTML tags. JavaScript expressions, contrarily, can't be written inside the HTML tags
- While JavaScript expressions support conditionals, exceptions, and loops, Angular expressions don't

**Question: Can you give us an overview of Angular architecture?**

**Answer:** You can draw some like this:



Here is Angular Architecture in detail: <https://angular.io/guide/architecture>

**Question: What is Angular Material?**

**Answer:** It is a UI component library. [Angular Material](#) helps in creating attractive, consistent, and fully functional web pages as well as web applications. It does so while following modern web design principles, including browser portability and graceful degradation.

**Question: What is AOT (Ahead-Of-Time) Compilation?**

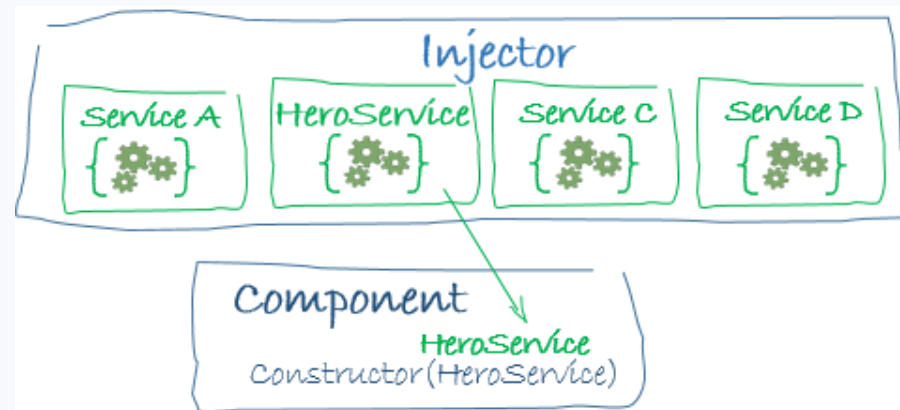
**Answer:** Each Angular app gets compiled internally. The Angular compiler takes in the JS code, compiles it and then produces some JS code. This happens only once per occasion per user. It is known as AOT (Ahead-Of-Time) compilation.

**Question: What is Data Binding? How many ways it can be done?**

**Answer:** In order to connect application data with the DOM (Data Object Model), data binding is used. It happens between the template (HTML) and component (TypeScript). There are 3 ways to achieve data binding:

1. Event Binding – Enables the application to respond to user input in the target environment
2. Property Binding – Enables interpolation of values computed from application data into the HTML
3. Two-way Binding – Changes made in the application state gets automatically reflected in the view and vice-versa. The ngModel directive is used for achieving this type of data binding.

**Question: What is demonstrated by the arrow in the following image?**



**Answer:** This represents a dependency injection or DI.

**Question: Can you draw a comparison between the `service()` and the `factory()` functions?**

**Answer:** Used for the business layer of the application, the `service()` function operates as a constructor function. The function is invoked at runtime using the `new` keyword.

Although the `factory()` function works in pretty much the same way as the `service()` function does, the former is more flexible and powerful. In actual, the `factory()` function are design patterns that help in creating objects.



**Question: Please explain the digest cycle in Angular?**

**Answer:** The process of monitoring the watchlist in order to track changes in the value of the watch variable is termed the digest cycle in Angular. The previous and present versions of the scope model values are compared in each digest cycle.

Although the digest cycle process gets triggered implicitly, it is possible to start it manually by using the *\$apply()* function.

**Question: Could you explain the various types of filters in AngularJS.**

**Answer:** In order to format the value of expression so that it can be displayed to the user, AngularJS has filters. It is possible to add these filters to the controllers, directives, services, or templates. AngularJS also provides support for creating custom filters.

Organizing data in such a way so that it is displayed only when certain criteria are fulfilled is made possible using filters. Filters are added to the expressions using the pipe '|' character. Various types of AngularJS filters are enumerated as follows:

- *currency* – Formats a number to the currency format
- *date* – Formats a data to some specific format
- *filter* – Selects a subset of items from an array
- *json* – Formats an object to a JSON string
- *limitTo* – Limits an array or string into a specified number of characters or elements
- *lowercase* – Formats a string to lowercase
- *number* – Formats a number to a string
- *orderBy* – Orders an array by an expression

**Question: What is new in Angular 6?**

**Answer:** Here are some of the new aspects introduced in Angular 6:

- Angular Elements – It allows converting Angular components into web components and embeds the same in some non-Angular application

- Tree Shakeable Provider – Angular 6 introduces a new way of registering a provider directly inside the @Injectable() decorator. It is achieved by using the providedIn attribute
- RxJS 6 – Angular 6 makes use of RxJS 6 internally
- i18n (internationalization) – Without having to build the application once per locale, any Angular application can have “runtime i18n”

**Question: What is ngOnInit()? How to define it?**

**Answer:** ngOnInit() is a lifecycle hook that is called after Angular has finished initializing all data-bound properties of a directive. It is defined as:

```
Interface OnInit {  
  ngOnInit() : void  
}
```

**Question: What is SPA (Single Page Application) in Angular? Contrast SPA technology with traditional web technology?**

**Answer:** In the SPA technology, only a single page, which is index.HTML, is maintained although the URL keeps on changing. Unlike traditional web technology, SPA technology is faster and easy to develop as well.

In conventional web technology, as soon as a client requests a webpage, the server sends the resource. However, when again the client requests for another page, the server responds again with sending the requested resource. The problem with this technology is that it requires a lot of time.

**Question: What is the code for creating a decorator?**

**Answer:** We create a decorator called Dummy:

```
function Dummy(target) {  
  dummy.log('This decorator is Dummy', target);  
}
```

**Question: What is the process called by which TypeScript code is converted into JavaScript code?**

**Answer:** It is called Transpiling. Even though TypeScript is used for writing code in Angular applications, it gets internally transpiled into equivalent JavaScript.

**Question: What is ViewEncapsulation and how many ways are there do to do it in Angular?**

**Answer:** To put simply, ViewEncapsulation determines whether the styles defined in a particular component will affect the entire application or not. Angular supports 3 types of ViewEncapsulation:

- Emulated – Styles used in other HTML spread to the component
- Native – Styles used in other HTML doesn't spread to the component
- None – Styles defined in a component are visible to all components of the application

**Question: Why prioritize TypeScript over JavaScript in Angular?**

**Answer:** TypeScript is a superset of Javascript as it is Javascript + Types or extra features like typecasting for variables, annotations, variable scope and much more. The typescript is designed in a way to overcome Javascript shortcomings like typecasting of variables, classes, decorators, variable scope and many more. Moreover, Typescript is purely object-oriented programming that offers a "Compiler" that can convert to Javascript-equivalent code.

**Question: Discuss the lifecycle designed for directive and components in Angular JS especially for the newly introduced version 6.0?**

**Answer:**

Components and directive of Angular JS follow the following typical lifecycle.

- `ngOnInit`
- `ngDoCheck`
- `ngOnDestroy`
- `Constructor`

- ngOnChanges
- ngAfterContentInit (only for components)
- ngAfterContentChecked (only for components)
- ngAfterViewInit (only for components)
- ngAfterViewChecked (only for components)

**Question: Write the features for Angular 6 over Angular 5.**

**Answer:** Following are the features:

### 1. Added ng update

CLI command updates angular project dependencies to their latest versions. The ng update is a normal package manager tool to identify and update in normal package manager tools to identify and update other dependencies.

### 2. Uses RxJS6

This is the third party library (RxJS) and introduces two important changes as compared to RxJS5.

1. Introduces a new internal package structure.
2. Operator concept

To update of RxJS6, run the following command:

```
npm install --save rxjs@6
```

To update your existing Angular Project, run the following:

```
npm install --save rxjs-compat
```

### 3. The <ng-template>

Angular 6 uses <ng-template> instead of <template>

#### **4. Service Level Changes**

If in an earlier version, the user wanted to provide a service to the entire application, the user was required to add it to providers in the AppModule but it is not required in the case of Angular6.

#### **5. Renamed Operators**

Angular 6 has renamed following operators:

- do() => tap()
- catch() => catchError()
- finally() => finalize()
- switch()=>switchAll()
- throw() => throwError

**Question: Do these questions show relevancy to Angular 6 Interview Questions?**

**Answer:** Yes, they can be considered as Angular 6 Interview Question.

#### **Additional Tips to Win the Interview!**

You need to be confident while giving the interview. Also, try avoiding hogwash in case you're asked a question that you don't know about. A simple no is better than giving some random non-answer and adding unnecessary details. Candidates that are straightforward and honest are preferred over those pretending to be know-it-alls!

Further, don't disburse your personal details until asked for. The interviewer is more interested in knowing you as a technical person. So, all the very best! Do let us know the Angular questions you faced in the interview that are not covered here so that we can add those here for the benefit of the Angular community.

Are these questions also relevant as Angular JS Interview Questions? Yes, these questions are helpful while preparing for the AngularJS interview.

### People are also reading:

- [Top Angular Alternatives](#)
- [Head to Head Comparison Between Angular vs React](#)
- [8 Best Angular IDE & Editor](#)
- [Top 50 Javascript Interview Questions & Answers](#)
- [Top 30 JQuery interview Questions](#)
- [Top React Interview Questions and Answers](#)
- [Top Bootstrap Interview Questions & Answers](#)
- [Top SQL Interview Questions & Answers](#)
- [SQL Cheat Sheet for Quick Reference](#)
- [Top Java Interview Questions & Answers](#)
- [Top Programming Interview Questions & Answers](#)

[Angular](#)[Angular Interview Questions](#)[Interview Questions](#)

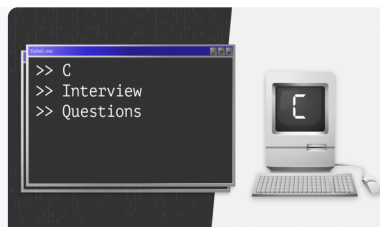
Share:



#### Youssef Nader

Youssef Nader, Computer Engineering Student at Cairo University. Technology technical writer and blogger, full-stack Web developer, specializes in rails and node. Founder of Yadawy, an E-commerce platform under construction. AI enthusiast, loves reading, traveling and martial arts. [View all posts by the Author](#)

### Related Posts



### 40+ Best C Interview Questions

[Read More](#)



### MVC Interview Questions

[Read More](#)



### Top 50 JSP Interview Questions & Answers

[Read More](#)

## Leave a comment

**Email address\***

Enter email

Your email will not be published

**Name\***

Name

**Comment\***

**SUBMIT**

Aravind

NgOnchnages will come first in page life cycle.

[Reply](#)

prakash

Awesome work, thanks!

[Reply](#)

Esteban Ocampo

Thanks for the post. Very interesting, just one thing, following is repeated

What should replace the “?”?

Answer: Directives. The image represents the types of directives in Angular; Attribute, structural, and custom.

[Reply](#)

sravanthireddy

what is the difference between routing and lazy routing?

[Reply](#)

chaman jain

As you said "JS code doesn't follow the Prototype Pattern"

can you please explain by example

how ts support Prototype Pattern and js not support

[Reply](#)

**My Interview Practice**

I glad to read your amazing blog "Angular Interview Questions". Thanks for sharing your knowledge with us. I highly recommend it to all!

[Reply](#)



**stephen**

It's useful for developer and programmers

[Reply](#)

**Eduardo Antonio Fernandez Diaz**

One learns a lot when reading these questions, thanks for compiling these questions, Youssef.

[Reply](#)

**Vijay Singh**  312 Points

Thanks for Recommending.

[Reply](#)

**seemant**

Nice post

[Reply](#)

**Akbar Sheikh**

its really helpful for angular technical interview

[Reply](#)

**Tom**

Thanks a lot, it was very helpfull

[Reply](#)

**Rajendra**

Hi,

I need Angular project like IMDB.com which can have only 1 producer for a Movie & many Actors & show card view. search box can able to find with Movie name, Producer Name, Actors Name, Year of release

[Reply](#)

priyanka yuvraj chaudhari

difference between template driven forms and reactive forms?

how to override custom components?

[Reply](#)



**Saddam Hussain**

Template Driven Forms Features

Easy to use

Suitable for simple scenarios and fails for complex scenarios

Similar to AngularJS

Two way data binding(using `[(NgModel)]` syntax)

Minimal component code

Automatic track of the form and its data(handled by Angular)

Unit testing is another challenge

Reactive Forms Features

More flexible, but needs a lot of practice

Handles any complex scenarios

No data binding is done (immutable data model preferred by most developers)

More component code and less HTML markup

Reactive transformations can be made possible such as

Handling a event based on a debounce time

Handling events when the components are distinct until changed

Adding elements dynamically

Easier unit testing

[Reply](#)



**Ranjan**

First create your customer component like a compound component or a standalone component,  
second step: export this component by using the public\_api.ts.  
third step: create a package by using : npm run package  
fourth: install this package wherever you want to consume this component .  
fifth: in the page where you need this component , just use the selector and use it.  
Note: keep in mind while creating the component , you need the component data in the page as well for save the data or update.

[Reply](#)

**Kathryn Newton**

What are the differences between bootstrap and angular.js?

[Reply](#)



**Joesph Draper**

AngularJS was developed by Google for front end development, which provides a set of components help to structure the application. Bootstrap was developed by Twitter with very common components like CSS, JavaScript, styles etc. enables the developer to develop fast.  
AngularJS provides AngularUI bootstrap, Angular foundation, Ionic framework, and Mobile Angular UI. Bootstrap provides structure layouts, CSS, and styles. Bootstrap can't help in making a Single Page Application, AngularJS can help you a lot in doing so.  
These differences are very less to count. There are many more differences list available for both bootstrap and angular.js.

[Reply](#)

**Nichole Fleming**

What are the features of AngularJS?

[Reply](#)



**Arline Crain**

AngularJS is fastest growing and popular JavaScript framework and this is obviously for its features. AngularJS features includes MVC framework, User interface with HTML, Less code need to write, Filters, POJO (Plain Old JavaScript Objects) model, Unit testing, Context aware communications, DOM manipulation, Service providers, Behavior with Directives etc. Also, cross browser complaint application can be built using AngularJS.

[Reply](#)



**John**

Angular JS is not fastest growing. It's actually dying. The latest release was in mid 2018 and it is currently in long term support until july 2021. That means development stopped except for critical bug fixes, which get fixes during the lts period

[Reply](#)

**Mabel Banks**

Is AngularJS MVVM or MVC?

[Reply](#)



**Anissa Whited**

AngularJS is fastest JavaScript framework. AngularJS supports both MVC (Model View Controller) and MVVM (Model View View Model) design patterns that are essential to build modern websites. The AngularJS framework allows us data binding, form validation, directives, communication with server, controller and so on. MVC defines the way that objects communicate with each other. MVVM used to represent an abstract view of the user interface.

[Reply](#)

**William Hughes**

Can an HTML page have multiple NG App directive?

[Reply](#)



**Serita Keel**

NO, The NG App directive is used to auto-bootstrap an AngularJS application. According to AngularJS Documentation, only one AngularJS application can be auto-bootstrapped per HTML document. Only one ng-app directive will be automatically instantiated and initialized by the Angular framework, while it is technically possible to have multiple applications per page. The reason for this is that only one AngularJS application can be automatically bootstrapped per HTML document.

[Reply](#)

**Marty Spencer**

What is Link function in AngularJS directive?

[Reply](#)



**Corey August**

Link function for the directives in AngularJS is AngularJS Directive's link. We can define directive's API & functions using link function that can then be used by directive to perform some business logic. For registering DOM listeners as well as updating the DOM, the link function is responsible. Parameters which are passed into the link function is function link (scope, element, attrs). Where scope is an Angular scope object, element is the jqLite-wrapped element that this directive matches and attrs is an object with the normalized attribute names and their corresponding values.

[Reply](#)

**Glen Paul**

What is default scope in a AngularJS directive?

[Reply](#)



**Valentin Flynn**

In AngularJS, by default directives do not create new scope or their own scope, instead they use their parent scope, which is usually a controller (within the

scope of which the directive is defined). AngularJS allows us that default scope can be changed using the scope field of the Data Definition Object (DDO). The default scope of directives can be changes by passing a configuration object known as directive definition object.

[Reply](#)

**Toby Bass**

Do you know scope in AngularJS?

[Reply](#)



**Jeraldine Settle**

\$scope is a built-in object in AngularJS. \$scope contains application data and methods. Whatever function you apply in View, that is accessed in controller using scope only. You can create properties to a \$scope object inside a controller function and assign a value or function to it. The view can display \$scope data using an expression, ng-bind directive or ng-model.

[Reply](#)

**Leo Benson**

What technique AngularJS uses for two way binding?

[Reply](#)



**Dominique Sommers**

Automatic synchronization of data between the model and view components is known as the data binding in Angular JS. Angular JS creates a two way binding between the select element and the \$ctrl.orderProp model. \$ctrl.orderProp is then used as the input in the next for the orderBy filter.

[Reply](#)

**Aubrey Cannon**

Do AngularJS provide reusable components?

[Reply](#)



**Doloris Utley**

Yes, Angular JS enables developers to create reusable components. These components can be used within the whole application. Angular JS even provide their own custom components also. Angular JS comes with basic built-in directives. These directives can be applies as an attribute, class, element, and even as component also.

[Reply](#)

**Michele Morton**

What are the pros and cons of angular+ionic vs angular+bootstrap?

[Reply](#)



**Kieth Temple**

The Angular+Ionic is used when we require for Hybrid Application development, whereas angular+bootstrap is used for web application development. Bootstrap UI elements can be customized but its execution is slow as compared to Ionic. Ionic elements are a close copy of UI of Android and iOS. Ionic gives better webview support for Android and iOS.

[Reply](#)

**Josephine Kim**

Which Javascript framework should I go for React or Angular?

[Reply](#)

**Hackr Team** ★ 0 Points

We have written a post on detailed comparison of Angular vs React:

<https://hackr.io/blog/angular-vs-react-2018>

[Reply](#)

Virgil Beck

Which version of angular should I learn in 2019?

[Reply](#)



Ambrose Lindstrom

Learn the latest version of Angular in 2019. Right now Angular 5 and Angular 6 are running. Angular JS is the different than other versions of Angular. Angular 2 and Angular 4 are most preferable for the enterprise applications in companies. Angular 5 and Angular 6 release recently and have backward compatibility with the previous versions Angular 2 and Angular 4. Angular 4 is faster and smaller than Angular 2.

[Reply](#)

Herman Lamb

Which is easy to learn, AngularJS or Angular 2 or Angular 4?

[Reply](#)



Dorinda Angel

Angular JS is basic for all Angular versions. Angular 2 and Angular 4 are the latest frameworks to learn. Angular 4 is the newest one, which has many new modules and documentations are not descriptive enough. Although, Angular 4 is the latest one and developers are moving towards it than Angular 2 and Angular JS.

[Reply](#)

Earl Robertson

Are Angular and AngularJS the same?

[Reply](#)



Mauricio Zielinski



Both are same and the versions of Angular. Angular is based on TypeScript and Angular JS is based on JavaScript. Angular can be learned without knowing Angular JS. Angular JS is the version less than 2 and from Angular version 2 it is known as Angular. Angular JS is developed by the search engine giant Google.

[Reply](#)



Alan

Indeed they are not the same. Google is the owner of the Angular entire project, but I can say they are 95% different one against the other.

[Reply](#)

Daisy Turner

Which filter will be executed one or more times during the each \$Digest cycle?

[Reply](#)



Shantel Croteau

The stateful filter will be executed one or more times during the each \$Digest cycle whenever needs. However you need to write a stateful filter, you have to use this as \$stateful, this means that it will be executed one or more times during the each \$digest cycle.

[Reply](#)

Maxine Payne

What is the difference between Angular and jQuery?

[Reply](#)



Ivette Carrion

Both the Angular and jQuery works at different levels.

Angular – Angular is a powerful JavaScript framework used to create internet applications. Angular JS applications are cross browser compliant. Angular gives you various concepts and structures that can apply to the projects like directives and controllers.

jQuery –jQuery is a JavaScript library which is fast and concise. jQuery make it easy to DOM manipulation, animation, event handling. jQuery is lightweight

library and supports latest technology. jQuery itself used in multiple ways to do the same thing as Angular.

[Reply](#)



**Imtiyaz**

Jquery is Just a library it can be helpful only at script side .Angular is framework can be develop application using javascript ,css ,html etc.

[Reply](#)

## Top Blogs

What is Data Science?  
Programming Languages 2020  
Web Development Frameworks  
Python Books  
Numpy Matrix Multiplication  
Java IDE  
IoT Applications  
Javascript Map  
Features of Java  
Normalization in DBMS

## Interview Questions

Angular Interview Questions  
Python Interview Questions  
C++ Interview Questions  
Selenium Interview Questions  
SQL Interview Questions  
Java Interview Questions  
Javascript interview Questions  
PHP Interview Questions  
Spring Interview Questions  
AWS Interview Questions

## Top Courses/Certifications

Python Course  
Data Analytics Course  
Javascript Course  
Web Developer Course  
C Course  
SQL Certification  
Java Certification  
Python Certification  
Machine Learning Certification  
AWS Certification

[About Us](#)

[Programming Tips](#)

[Help & FAQ](#)

[We ❤️ Feedback](#)

[Contact Us](#)

[Advertise With Us](#)

[We Are Hiring](#)

