Top 50 Interview Questions and Answers in Angular

1-What is angular?

Angular is a platform that makes it easy to build applications with the web. Angular combines declarative templates, dependency injection, end to end tooling, and integrated best practices to solve development challenges. Angular empowers developers to build applications that live on the web, mobile, or the desktop

2-What is directive? What are the types of directives in angular?

A directive is a custom HTML element that is used to extend the power of HTML.

```
ng generate directive <name> [options]
ng g directive <name> [options
```

There are three types of directives in Angular:

1. Components directives with a template.
2. Structural directives change the DOM layout by adding and removing DOM elements.
3. Attribute directives change the appearance or behavior of an element, component, or another directive.

3-What is component? How to create component in angular?

Components are like the basic building block in an Angular application. Components are defined using the @component decorator. A component has a selector, template, style and other properties, using which it specifies the metadata required to process the component.
Syntax: ng generate component componentName

```
ng generate component <name> [options]
ng g component <name> [options]
```

```
import { Component } from '@angular/core';

@Component({
  selector: 'app-root',
  templateUrl: './app.component.html',
  styleUrls: ['./app.component.css']
})
export class AppComponent {
  title = 'app';
}
```

4-What is module? How to create component?

Module is collection or set of component, directives, services, pipes etc. An application can have more than one module in angular.

```
ng generate module <name> [options]
ng g module <name> [options]
```

5-What are the new features added in angular 7?

- CLI prompts
- Angular material & component dev kit (CDK)
- Drag & drop
- Virtual scrolling
- Application performance improvements
- Updated Dependencies
- Ivy Renderer
- Upgrading

6-What are the new features added in Angular 6?

- ng add
- ng update
- Animations
- Registering Providers
- ngModelChange
- Angular Material + CDK
- Angular Material Starter Components
- Material Sidenav
- Material Dashboard
- Material Data Table
- Upgrading to Angular 6.0
- RxJS
- Angular Element

7-What is angular CLI? List the command to install angular CLI.

Angular cli stand for command line interface. CLI is command line tool to create angular application faster and with great consistency. It creates the boiler plate code for angular features like component, directives, pipes and services etc. it also create boiler plate code for TypeScript features like classes, interface and enums.

Command to install CLI globally

```
npm install -g @angular/cli
```

Command to install CLI Locally:

```
npm install @angular/cli
```

Command to install CLI with specify version:

```
npm install -g @angular/cli@6.1.1
```

Command to check version:

```
 npm --v
```

8-How to update or upgrade angular cli version

In order to upgrade angular-cli package that was installed globally in your system, you need to run following commands

```
npm uninstall -g angular-cli
npm cache clean or npm cache verify (if npm > 5)
npm install -g @angular/cli@latest
```

Instead of upgrading global version of angular-cli you can also upgrade the local version for specific project y running following commands:

```
rm -rf node_modules
npm uninstall --save-dev angular-cli
npm install --save-dev @angular/cli@latest
npm install
```

9-How to create a new project in angular using CLI.

```
ng new <name> [options]
ng n <name> [options]
```

10-How to run Angular application locally during development?

Builds and serves your app, rebuilding on file changes.

```
cd PROJECT-NAME
ng serve <project> [options]
ng s <project> [options]
```

You can configure the default HTTP host and port used by the development server with two command-line options:

```
ng serve --host 0.0.0.0 --port 4201
```

11-What is npm (node package manager)?

Npm is the package manager for JavaScript and the world’s largest software registry. Discover packages of reusable code and assemble them in powerful new ways.

12-What is typescript? List the command to install typescript?

TypeScript is a superset of JavaScript which primarily provides optional static typing, classes and interfaces. One of the big benefits is to enable IDEs to provide a richer environment for spotting common errors *as* you type the code.

```
npm install -g typescript

tsc -v
```

13-List inbuilt pipes available in Angular

- AsyncPipe
- CurrencyPipe
- DatePipe
- DecimalPipe
- DeprecatedCurrencyPipe
- DeprecatedDatePipe
- DeprecatedDecimalPipe
- DeprecatedPercentPipe
- I18nPluralPipe
- I18nSelectPipe
- JsonPipe
- KeyValuePipe
- LowerCasePipe
- PercentPipe
- SlicePipe
- TitleCasePipe
- UpperCasePipe

14-What are Services and how to create a service in Angular?

Services help us in not repeating the code. With the creation of services, we can use the same code from different components. Here is the command to create a service in angular.

```
ng generate service <name> [options]
ng g service <name> [options]
```

15-What is Routing in Angular?

The Angular Router enables navigation from one view to the next as users perform application tasks. Routing helps a user in navigating to different pages using links.

16-What is RouterOutlet?

The RouterOutlet is a directive from the router library that is used like a component. It acts as a placeholder that marks the spot in the template where the router should display the components for that outlet.

```
<router-outlet></router-outlet>
```

17-What are Decorators?

Decorators are functions that adds metadata to class members and functions. It was proposed in ES2016 and implemented in Typescript.

18-How do we import a module in Angular?

Simply use below syntax to import a module in Angular.

```
import { Component } from '@angular/core';
```

19-What is *ngFor directive used for?

*ngFor directive is used for Iterating over a list of items and for generating a new DOM element for each one.

20-Explain Webpack?

Webpack is module bundler for Angular or above. It bundles, minifies and transpiles an Angular application.

21-Explain component life cycle in Angular?

In Angular component life cycle in Angular goes through the following stages.

- Create
- Render
- Create and render children
- Check for bound data changes and re-render
- Destroy

22-Explain NgModule?

NgModule is a decorator function in Angular that takes a single metadata object whose properties describe the module.

23-What is Router links

Now you have routes configured and a place to render them, but how do you navigate? The URL could arrive directly from the browser address bar. But most of the time you navigate as a result of some user action such as the click of an anchor tag.

```
<h1>Angular Router</h1>
<nav>
     <a routerLink="/crisis-center" routerLinkActive="active">Crisis Center</a>
     <a routerLink="/heroes" routerLinkActive="active">Heroes</a>
</nav>
<router-outlet></router-outlet>
```

24-How to handle Events in Angular?

Any activity (button click, mouse click, mouse hover, mouse move, etc) of a user on a frontend/web screen is termed as an event. Such events are passed from the view (.HTML) page to a typescript component (.ts).

25-Explain the usage of {{}}?

The set of brackets {{}} when used with an HTML tag, represent data from a component. For example, on a HTML page which has <h1>{{variableName}}</h1>, here the ‘variableName’ is actually typescript (component) data representing its value on the template; i.e., HTML. This entire concept is called String Interpolation.

26-What is data binding and types of data binding in angular?

Data Binding is a process that creates a connection between the application’s UI and the data. When the data changes its value, the UI elements that are bound to the data, will also change.

- Interpolation
- Property Binding
- Event Binding
- Two-way binding

27-How Single page application different from the tradition web application?

In traditional web technology, the client requests for a web page (HTML/JSP/asp) and the server sends the resource (or HTML page), and the client again requests for another page and the server responds with another resource. The problem here is a lot of time is consumed in the requesting/responding or due to a lot of reloading. Whereas, in the SPA technology, we maintain only one page (index.HTML) even though the URL keeps on changing.

28-What are ngModel and how do we represent it?

ngModel is a directive which can be applied on a text field. This a two-way data binding. ngModel is represented by [()]

29-What does a Subscribe method do in Angular?

It is a method which is subscribed to an observable. Whenever the subscribe method is called, an independent execution of the observable happens.

30-What is Transpiling in angular?

Transpiling is a process of converting code from one language to another. In Angular, Traceur compiler is used for converting TypeScript to JavaScript so that browsers can understand.

31-What is the different between observables and promises?

Observable: observable are lazy which means nothing happens until a subscription is made. Observable is a stream in which passing of zero or more events is possible and the callback is called for each event.

Promises: Promises are eager; which means as soon as a promise is created, the execution takes place. Promise handles a single event.

32-What is an AsyncPipe in Angular?

When an observable or promise returns something, we use a temporary property to hold the content. Later, we bind the same content to the template. With the usage of AsyncPipe, the promise or observable can be directly used in a template and a temporary property is not required.

33-Explain Authentication and Authorization.

Authentication: The user login credentials are passed to an authenticate API (on the server). On the server side validation of the credentials happens and a JSON Web Token (JWT) is returned. JWT is a JSON object that has some information or attributes about the current user. Once the JWT is given to the client, the client or the user will be identified with that JWT.
Authorization: After logging in successfully, the authenticated or genuine user does not have access to everything. The user is not authorized to access someone else’s data, he/she is authorized to access some data.

34-What is AOT Compilation?

Every angular application gets compiled internally. The angular compiler takes JavaScript code, compiles it and produces JavaScript code again. Ahead-of-Time Compilation does not happen every time or for every user, as is the case with Just-In-Time (JIT) Compilation.

35-What is Redux?

It is a library which helps us maintain the state of the application. Redux is not required in applications that are simple with the simple data flow, it is used in Single Page Applications that have complex data flow.

36-What are the Pipes?

This feature is used to change the output on the template; something like changing the string into uppercase and displaying it on the template. It can also change Date format accordingly.

37-Differentiate between ng-Class and ng-Style.

In ng-Class, loading of CSS class is possible; whereas, in ng-Style we can set the CSS style.

38-Name the building blocks of Angular.

1. Modules
2. Component
3. Template
4. Directives
5. Data Binding
6. Services
7. Dependency Injection
8. Routing

39-Which component life cycle execute when data-bound input value updates?

ngOnChanges is the life cycle hook that gets executed whenever a change happens to the data that was bound to an input.

40-Differentiate between Components and Directives in Angular.

Components break up the application into smaller parts; whereas, Directives add behavior to an existing DOM element.

41-What is the use of @Input and @Output?

When it comes to the communication of Angular Components, which are in Parent-Child Relationship; we use @Input in Child Component when we are passing data from Parent to Child Component and @Output is used in Child Component to receive an event from Child to Parent Component.

42-What does a router.navigate do?

When we want to route to a component we use router.navigate.
Syntax: this.router.navigate(['/component_name’]);

43-What is ng-content Directive?

The HTML elements like p (paragraph) or h1 (heading) have some content between the tags. For example, <p>this is a paragraph</p> and <h1>this is a heading</h1>. Now, similar to this, what if we want to have some custom text or content between the angular tags like <app-tax>some tax-related content</app-tax>. This will not work the way it worked for HTML elements. Now, in such cases, the <ng-content> tag directive is used.

44-What is ViewEncapsulation?

ViewEncapsulation decides whether the styles defined in a component can affect the entire application or not. There are three ways to do this in Angular:

- Emulated: styles from other HTML spread to the component.
- Native: styles from other HTML do not spread to the component.
- None: styles defined in a component are visible to all components.

45-What is Dependency Injection in Angular?

Dependency injection (DI), is an important application design pattern. Angular has its own DI framework, which is typically used in the design of Angular applications to increase their efficiency and modularity. When a component is dependent on another component the dependency is injected/provided during runtime.

46-What are the directives in CommonModule?

- NgClass
- NgComponentOutlet
- NgForOf
- NgIf
- NgPlural
- NgPluralCase
- NgStyle
- NgSwitch
- NgSwitchCase
- NgSwitchDefault
- NgTemplateOutlet

47-What are the directives in FormsModule?

- CheckboxControlValueAccessor
- CheckboxRequiredValidator
- DefaultValueAccessor
- EmailValidator
- FormArrayName
- FormControlDirective
- FormControlName
- FormGroupDirective
- FormGroupName
- MaxLengthValidator
- MinLengthValidator
- NgControlStatus
- NgControlStatusGroup
- NgForm
- ~~NgFormSelectorWarning~~
- NgModel
- NgModelGroup
- NgSelectOption
- PatternValidator
- RadioControlValueAccessor
- RequiredValidator
- SelectControlValueAccessor
- SelectMultipleControlValueAccessor

48-What are the directives in RouterModule?

- RouterLink
- RouterLinkActive
- RouterLinkWithHref
- RouterOutlet

49-How to create custom pipe in angular?

```
import { Pipe, PipeTransform } from '@angular/core';

@Pipe({name: 'reverseStr'})
export class ReverseStr implements PipeTransform {
  transform(value: string): string {
    let newStr: string = "";
    for (var i = value.length - 1; i >= 0; i--) {
      newStr += value.charAt(i);
    }
    return newStr;
  }
}
```

Then you'd include the custom pipe as a declaration in your app module

```
import { BrowserModule } from '@angular/platform-browser';
import { NgModule } from '@angular/core';
import { FormsModule } from '@angular/forms';
import { HttpModule } from '@angular/http';

import { AppComponent } from './app.component';
import { ReverseStr } from './reverse-str.pipe.ts';

@NgModule({
  declarations: [
    AppComponent,
    ReverseStr
  ],
  imports: [
    BrowserModule,
    FormsModule,
    HttpModule
  ],
  providers: [],
  bootstrap: [AppComponent]
})
export class AppModule { }
```

```
{{ user.name | reverseStr }}
```

50- How to handle error in angular?

Using Angular's HttpClient along with `catchError` from RxJS, we can easily write a function to handle errors within each service. HttpClient will also conveniently parse JSON responses and return a javascript object in the observable. There are two categories of errors which need to be handled differently:

- Client-side: Network problems and front-end code errors. With HttpClient, these errors return ErrorEvent instances.
- Server-side: AJAX errors, user errors, back-end code errors, database errors, file system errors. With HttpClient, these errors return HTTP Error Responses.

```
import { Injectable } from '@angular/core';
import { HttpClient, HttpHeaders } from '@angular/common/http';
import { Observable, throwError } from 'rxjs';
import { retry, catchError } from 'rxjs/operators';
import { User } from './user.model';

@Injectable({
 providedIn: 'root'
})
export class UserService {
 private apiUrl = 'https://localhost:8080/api/users';

 constructor(private http: HttpClient) { }

 getUsers(): Observable<User[]> {
   return this.http.get<User[]>(this.apiUrl)
     .pipe(
       retry(1),
       catchError(this.handleError)
     );
 }

 handleError(error) {
   let errorMessage = '';
   if (error.error instanceof ErrorEvent) {
     // client-side error
     errorMessage = `Error: ${error.error.message}`;
   } else {
     // server-side error
     errorMessage = `Error Code: ${error.status}\nMessage: ${error.message}`;
   }
   window.alert(errorMessage);
   return throwError(errorMessage);
 }
}
```