***Angular Interview Questions:***

**1Q: Whate are the Directives in Angular?**

Directives are instructions in the DOM. They specify how to place your components and business logic in the Angular.

Directives are js class and declared as @directive. There are 3 directives in Angular.

* Component Directives
* Structural Directives
* Attribute Directives

**Component Directives:** Component directives are used in main class. They contain the detail of how the component should be processed, instantiated and used at runtime.

**Structural Directives:** Structural directives start with a \* sign. These directives are used to manipulate and change the structure of the DOM elements. For example, \*ngIf and \*ngFor.

**Attribute Directives:** Attribute directives are used to change the look and behavior of the DOM elements. For example: ngClass, ngStyle etc.

## **2Q:** **Difference between Attribute Directive and Structural Directive**

|  |  |
| --- | --- |
| **Attribute Directives** | **Structural Directives** |
| Attribute directives look like a normal HTML Attribute and mainly used in databinding and event binding. | Structural Directives start with a \* symbol and look different. |
| Attribute Directives affect only the element they are added to. | Structural Directives affect the whole area in the DOM. |

**3Q: What is Databinding in Angular?**

Databinding is a powerful feature of Angular. Angular Databinding is used for communication. It is used to communicate between your TypeScript code (your business logic) and the other component which is shown to the users i.e. HTML layout.

Databinding is necessary because when we write the code in TypeScript, it is compiled to JavaScript and the result is shown on HTML layout. Thus, to show the correct and spontaneous result to the users, a proper communication is necessary. That's why databinding is used in Angular.

**4Q: How many type of databinding in Angular?**

**One-way databinding**

One way databinding is a simple one way communication where HTML template is changed when we make changes in TypeScript code.

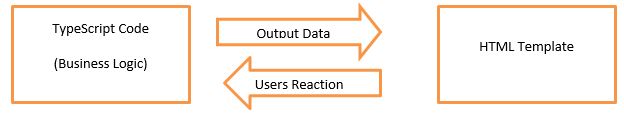
Or

In one-way databinding, the value of the Model is used in the View (HTML page) but you can't update Model from the View. Angular Interpolation / String Interpolation, Property Binding, and Event Binding are the example of one-way databinding.

**Two-way databinding**

In two-way databinding, automatic synchronization of data happens between the Model and the View. Here, change is reflected in both components. Whenever you make changes in the Model, it will be reflected in the View and when you make changes in View, it will be reflected in Model.

This happens immediately and automatically, ensures that the HTML template and the TypeScript code are updated at all times.



**5Q: What is Event Binding in Angular? How to use data with Event Binding?**

Angular facilitates us to bind the events along with the methods. This process is known as event binding. Event binding is used with parenthesis ().

We will create a method named "onUpdateServerName" and add an event with it.

|  |
| --- |
| **<input type="text"**  **class="form-control"**  **(input)="OnUpdateServerName($event)">** |

**6Q: What is Property Binding in Angular? How property binding works?**

In Angular , property binding is used to pass data from the component class (component.ts) and setting the value of the given element in the user-end (component.html).

Property binding is an example of one-way databinding where the data is transferred from the component to the class.

The main advantage of property binding is that it facilitates you to control elements property.

|  |
| --- |
| <button class="btn btn-primary" disabled>Add Server</button> |

|  |
| --- |
| export class Server2Component implements OnInit {  allowNewServer = false;    constructor() {  setTimeout(() =>{  this.allowNewServer = true;  }, 5000);  } |

**7Q: What is String Interpolation in Angular?**

In Angular, String interpolation is used to display dynamic data on HTML template (at user end). It facilitates you to make changes on component.ts file and fetch data from there to HTML template (component.html file).

**8Q: Defference between String Interpolation vs Property Binding in Angular?**

String Interpolation and Property binding both are used for same purpose i.e. one-way databinding. But the problem is how to know which one is best suited for your application.

String Interpolation and Property Binding doth are about one-way data binding. They both flow a value in one direction from our components to HTML elements.

String Interpolation is a special syntax which is converted to property binding by Angular. It's a convenient alternative to property binding.

When you need to concatenate strings, you must use interpolation instead of property binding.

**9Q: What is pipe in Angular and types?**

In Angular 1, filters are used which are later called Pipes onwards Angular2. In Angular 7, it is known as pipe and used to transform data. It is denoted by symbol |.

Pipe takes integers, strings, arrays, and date as input separated with |. It transforms the data in the format as required and displays the same in the browser.

Built-in Pipes in Angular:

1. Lowercasepipe
2. Uppercasepipe
3. Datepipe
4. Currencypipe
5. Jsonpipe
6. Percentpipe
7. Decimalpipe
8. Slicepipe

**10Q: How to create a custom pipein Angular?**

To create a custom pipe, create a new ts file and use the code according to the work you have to do. You have to import Pipe, PipeTransform from Angular/Core. Let's create a sqrt custom pipe.

|  |
| --- |
| export class SqrtPipe implements PipeTransform {  transform(val : number) : number {  return Math.sqrt(val);  }  } |

|  |
| --- |
| <h1>Example of Custom Pipe</h1>  <h2>Square root of 625 is: {{625 | sqrt}}</h2><br/>  <h2>Square root of 169 is: {{169 | sqrt}}</h2> |

**11Q: What is use of Forms in Angular?**

Angular forms are used to handle user's input. We can use Angular form in our application to enable users to log in, to update profile, to enter information, and to perform many other data-entry tasks.

In Angular , there are 2 approaches to handle user's input through forms:

1. Reactive forms
2. Template-driven forms

Both approaches are used to collect user input events from the view, validate the user input, create a form model and data model to update, and provide a way to track changes.

## **12Q: Difference between Reactive Forms and Template-driven Forms in Angular?**

|  |  |  |
| --- | --- | --- |
| **Comparison Index** | **Reactive Forms** | **Template-driven Forms** |
| Setup (form model) | Reactive forms are more explicit. They are created in component class. | Template-driven forms are less explicit. They are created by directives. |
| Data model | Structured | Unstructured |
| Predictability | Synchronous | Asynchronous |
| Form validation | Functions | Directives |
| Mutability | Immutable | Mutable |
| Scalability | Low-level API access | Abstraction on top of APIs |

**13Q: What is string interpolation in Angular?**

Ans: String interpolation 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. {{ }}.

**14Q: How to generate a service in Angular using CLI?**

Ans:ng generate service Mservice

**15Q: What is Two-way Binding?**

Ans: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

**16Q: What is @NgModule?**

Ans:An NgModule class describes how the application parts fit together

Every application has at least one NgModule, the root module that we

bootstrap to launch the application.

**17Q: What are all the metadata properties of NgModule? And what are they**

**used for?**

Ans:@NgModule accepts a metadata object that tells Angular how to compile and

launch the application. The properties are:

imports – Modules that the application needs or depends on to run like,

the BrowserModule that every application needs to run

in a browser.

declarations – the application's components, which belongs to the

NgModuleclass. We must declare every component in

an NgModule class. If we use a component without declaring

it, we'll see a clear error message in the browser console.

bootstrap – the root component that Angular creates and inserts into the

index.html host web page. The application will be launched

by creating the components listed in this array.

**18Q: What is the command to create new angular child component?**

Ans: ng generate component childcomponent

or

ng g c childcomponent

**19Q: What is the role of @Component?**

Ans : @component is a decorator which includes selector, template

and style urls.

@componnet is given as meta data or decorator to class.

@Component({

selector: 'app-root',

templateUrl: './app.component.html',

styleUrls: ['./app.component.css']

})

**20Q: What are all the uses of a service?**

Ans : Services encapsulates business logic and separates them from UI

concerns or the controller concerns, which governs them both.

**21Q: What are the angular pipes or filters?**

Ans:

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

**22Q: How to generate a module in Angular?**

In order to generate a module in Angular, cd to the current project

directory and below command.

**ng g module module\_name**