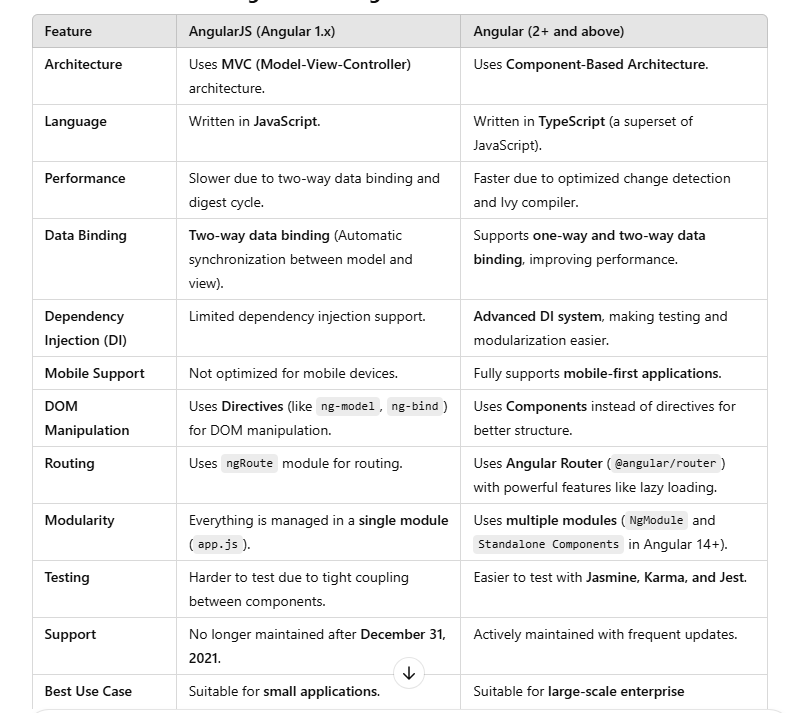
* **What is Angular and use of Angular?**

Angular is a JavaScript binding framework, which binds HTML UI with JavaScript model.

It is used to build SPA with the help of routing.

* **What are the advantages of Angular?**
* **Differentiate between AngularJS vs Angular?**



A table with text on it

AI-generated content may be incorrect.

* **Explain the importance of NPM and Node\_Modules folder?**

NPM is a package manager which makes installation of JavaScript library.

Node\_modules folder where all the packages are installed.

* **Explain the importance of Package.json file in Angular?**

It has all the reference of javascript references needed for the project.

* **What is typescript and why do we need it?**

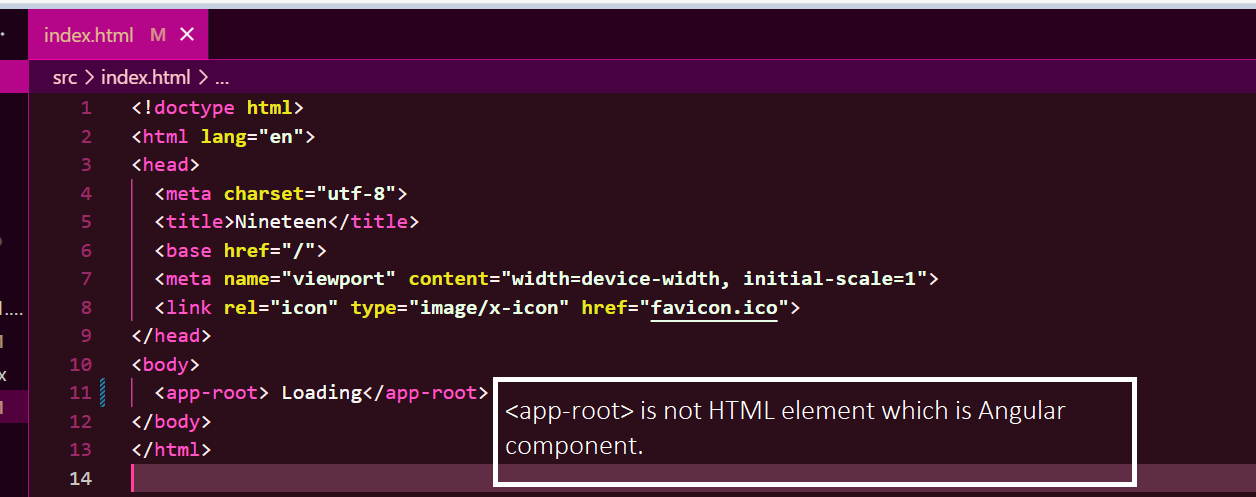
It adds types to JavaScript or superset of JavaScript.

* **What is Angular CLI and Explain importance of Angular CLI?**

Is a command line interface by which we can create Initial angular Project.

npm install @angilar/cli

* **What are Components in Angular?**
* **What is a Selector and Template?**
* **What is Module in Angular? What is app.module.ts file?**
* **How Angular App gets loaded and started? What are index.html, app-root, selector and main.ts?**
* When a client sends a request, it first loads the index.html file. This file includes a reference to main.js, which is the compiled JavaScript version of main.ts. (Angular CLI compiles and loads this JS file along with the HTML.)
* The main.ts file serves as the entry point of the Angular application. It compiles the web app and bootstraps the AppModule, allowing the app to run in the browser.
* The AppModule then bootstraps the AppComponent, which is the root component responsible for rendering the UI that is visible to the user.



* **What is a Bootstrapped Module & Bootstrapped Component?**
* **What is Event Binding in Angular? OR Data bindings in Angular with its types?**

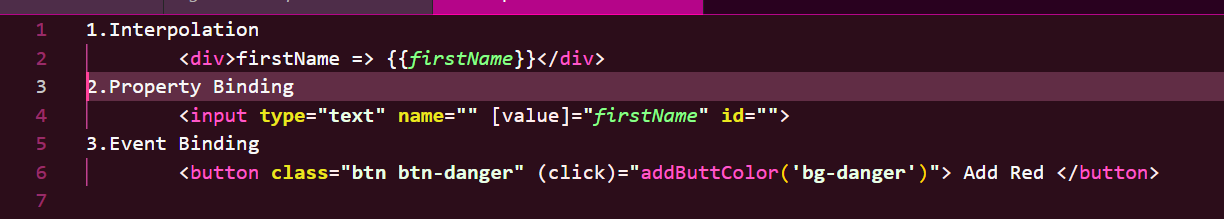
Databinding is the communication between the view and the component.

Event binding is used to handle events triggered by user actions, such as button clicks, on mouse events etc.

The TypeScript file (app.component.ts) contains the component logic, while the HTML template (app.component.html) represents the view.

Event binding follows a one-way data flow, where the event in the HTML template is linked to an expression in the component using the syntax:

(event) = "expression"



**Two-Way Binding** - [(ngModel)] → Two-way sync between component and UI

* **What is \*ngIf Directive?**
* **What is \*ngFor Directive?**
* **What is \*ngSwitch Directive?**

ngSwitch is a structural directive used in combination with \*ngSwitchCase and \*ngSwitchDefault, which are also structural directives.

Uses \*ngSwitchCase for different conditions.  
\*ngSwitchDefault ensures there's always a fallback.

Example:

<ul [ngSwitch]="studentList.length">

<li \*ngSwitchCase="0">Zero Student</li>

<li \*ngSwitchCase="1">One Student</li>

<li \*ngSwitchCase="2">Two Students</li>

<li \*ngSwitchDefault>More Students</li>

</ul>

* **What is [ngStyle] Attribute directive?**
* **What is [ngClass] Attribute directive?**
* **What are directives in Angular? and different types of Angular directives?**

Directive changes the behaviour of HTML DOM.

Angular syntax which can write inside HTML, so that we can change the behaviour of HTML DOM.

A black background with white text

AI-generated content may be incorrect.

Angular has three types of directives:

1. Component Directives (Most Common) → Customized user control. Directives with template (@Component)
2. Structural Directives → Change the DOM layout by adding or removing elements. (\*ngIf, \*ngFor, \*ngSwitch)
3. Attribute Directives → Modify the appearance or behaviour of an HTML element ([ngClass], [ngStyle], Custom directives)

* **What is Decorator?**

Defines what king of angular class it is..

@component denotes Angular component.

@NgModule denotes Angular Module

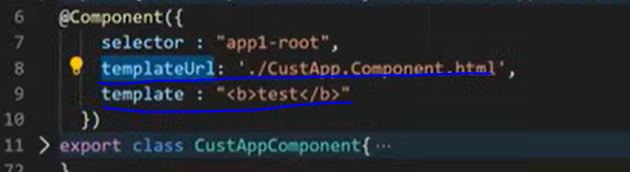
* **What are the types of Decorator?**
* **What are Pipes? What are the types of Pipes & Parameterized Pipes?**
* **What is Chaining Pipes?**
* **Explain the importance of Component and Modules?**

Components is where you write your binding code whereas Module logically groups the component.

* **What is a template?**

Template is an HTML view of Angular in which we can write directives.

Two types (inline and separate physical HTML file.)



* **Explain architecture of Angular?**

**⬇️ User Clicks a Button**

**┌──────────────────────────────────────────┐**

**│ COMPONENTS (UI + Logic) │**

**│ - app.component.ts (Controller) │**

**│ - app.component.html (View) │**

**└──────────────────────────────────────────┘**

**⬇️ Calls Service**

**┌──────────────────────────────────────────┐**

**│ SERVICES (Business Logic) │**

**│ - Fetch data from APIs │**

**│ - Perform calculations │**

**└──────────────────────────────────────────┘**

**⬇️ Uses DI to Fetch Data**

**┌──────────────────────────────────────────┐**

**│ DEPENDENCY INJECTION (DI) │**

**│ - Inject services into components │**

**└──────────────────────────────────────────┘**

**⬇️ Updates the UI**

**┌──────────────────────────────────────────┐**

**│ DIRECTIVES & PIPES │**

**│ - \*ngIf, \*ngFor (Structure) │**

**│ - ngClass, ngStyle (Styling) │**

**│ - Pipes (Data Transformation) │**

**└──────────────────────────────────────────┘**

* **What is SPA in Angular?**

SPA is single page Application where the main UI gets loaded ay once and the needed UI gets loaded on need.

* **How to implement SPA in Angular?**

We have to use Angular routing, Routing is a simple collection in Angular.

* **How to implement routing in Angular?**
* **Explain Lazy Loading ?**
* **How to implement Lazy Loading in Angular?**
* **Explain Services?**
* **How to create Servicein Angular?**
* **What is Depedency Injection?**
* **How to implement Depedency Injection?**
* **What’s the benefit of Depedency Injection?**
* **How to use Depedency Injection with services in Angular?**

1.Create a provider with service name.

2.Create a new proper inside a constructor assign the same service type.

3.this.propertyName.method()



* **What is Hierarchal Dependency Injector?**

If a child component requests a service:

It first looks for a provider inside its own component.

If not found, it checks the parent component.

If still not found, it checks module-level providers.

Finally, it looks in the root injector.



* **What is Provider in Angular?**
* **What is the role of @Injectable Decorator in a Service?**
* **What are Parent-Child Components?**
* **What are lifecycle hooks in Angular?**

A component from creation to destruction goes through several stages, and these stages are the life cycle hooks.

The stages will cover activities like:

* Component instantiating.
* Rendering the component HTML view.
* Creating the child components (if required).
* Destroying the component.

Lifecycle Hooks Order in Angular

* Constructor: It is a default method of the typescript class that is executed when the class is initiated
* ngOnChanges: called when the input property changes
* ngOnInit: it is called when the component is created.
* ngDoCheck: called for child component
* ngAfterContentInit: called for child component
* ngAfterContentChecked: called for child component
* ngAfterViewInit: called for child component
* ngAfterViewChecked: called for child component
* ngOnDestroy: called when the constructor is destroyed.
* **What is a Constructor in Angular?**
* **What is the difference between constructor and ngOnInit?**
* **What are Asynchronous operations?**
* **What is Type annotation?**
* **Differentiate between ng Serve and ng build?**
* **Explain the --prod parameter in ng build ?**
* **Promise and observable ?**

**Explaination**: Both are used to transfer the data asynchronously.

**Difference:** promise will provide the data only if whole data is ready to overcome that delay observable is works(chunks of data).

| **Observables** | **Promises** |
| --- | --- |
| 1. Emit **multiple** values over a period of time. Also called **streaming** of data. | Emit a **single** value at a time. |
| 2. Are **lazy**; they're not executed until we **subscribe** to them using the **subscribe()** method. | Are **not lazy**; execute **immediately** after creation. |
| 3. Have subscriptions that are **cancellable** using the **unsubscribe()** method. | Are **not cancellable**. |

* **What are angular forms and types of forms ?**

Angular forms are used to handle user's input.

Types of Forms:

1. Template Driven Forms
2. Reactive Forms

Flow of Data:

* HTML Template → Class (.ts) → Service → Server (API)
* Collect User Data → Bind Data → Send Data → Receive User Data

**Difference between Template and Reactive forms?**

| **Feature** | **Template Driven Forms** | **Reactive Forms** |
| --- | --- | --- |
| **Code & Validation** | Most code and validation logic is written in **HTML template**. | Most code and validation logic is written in **component TypeScript class file**. |
| **Module Required** | Need to add **FormsModule** in AppModule to activate it. | Need to add **ReactiveFormsModule** in AppModule to activate it. |
| **Usage** | Used when the application is **simple** and has **fewer controls**. | Used when the application is **complex** and has **more controls**. |

* **What is HTTP Interceptor and when and where to use?**

Interceptors are used to modify HTTP requests and responses globally.

Useful for authentication, error handling, logging, and modifying API calls.

Registered globally in app.module.ts using HTTP\_INTERCEPTORS.

Implemented using Angular’s HttpInterceptor interface.

* **How to access Child component from parent component with view child?**

child.component.ts

* +SendToParent()

app.component.ts

* @ViewChild(ChildComponent)
* ngAfterViewInit(): void { }
* **What are Built in/ Primitive and User-Defined/ Non-primitive Types in Typescript?**
* **What is “any” type in Typescript**
* **What is Enum type in Typescript?**
* **What is Type Assertion in Typescript?**
* **What are Arrow Functions in Typescript?**