

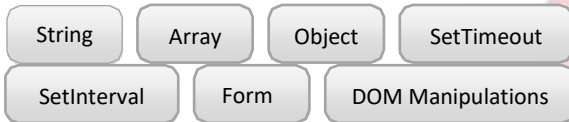
REAL-TIME PROJECTS



Array, String, Number Manipulation

Doing different challengeable Machine tests which are conducted in top MNCs.

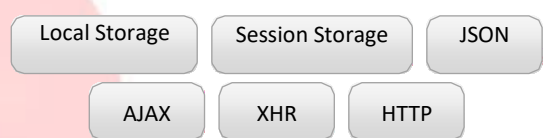
Skills:



Movie, Jokes & User APIs calls

An app with Registration, login, access different APIs via AJAX, logout.

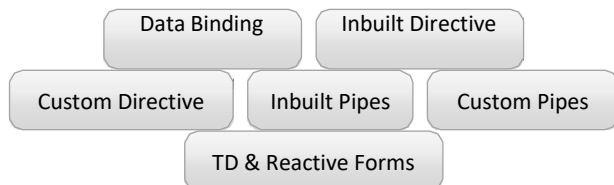
Skills:



Product Management in Angular

A real-time project with Product Management which includes, Add, Edit, Delete and Search!

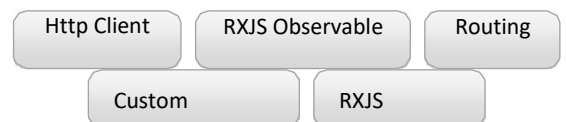
Skills:



Auth, Weather, News & Zomato App

A Project which accesses your JSON data and authenticate the credentials. Call Weather, News & Zomato APIs.

Skills:



REAL-TIME PROJECT - SHOPPING CART APPLICATION

A real-time Shopping cart app from very scratch to Advanced level. The project includes Register, Login, Client- side Authorization, Server-side Authorization, Add Products, List products, Filter products by category wise, Add to cart, View cart, Update cart, Delete cart items, Search, View Product details, Logout.

Technologies



How to convert HTML app into Angular App, How to load external CSS, JS & JQuery, Routing, Route Guard, Dynamic Routing, Connect REST API, Observable, Subject, Behaviour Subject, Event Emitter, HTTP,Interceptor, Child Routes, File upload, Split app into different Modules, Eager Loading, Lazy Loading, Async Loading.



Integrate Angular Material Module and UI Components

TYPESCRIPT COURSE CONTENT

SECTION 1: TYPESCRIPT INTRODUCTION

- What is TypeScript?
- Features of TypeScript.
- How to compile TypeScript into JS?
- What is Transpilation?

Hands-On Practicals:

- Installing Typescript Engine in Node.
- Compiling TS code (source) to JS code(source).

SECTION 2: NODE INTRODUCTION

- What is Node?
- Node Environment setups.
- A Brief Node Introduction.
- What is Node Modules?
- What are the Types of Node Modules available?
- What is Local & Global Module?

SECTION 3: TYPESCRIPT ENVIRONMENT SETUPS

- Install TypeScript Engine in Node.
- Execute the 1stTypeScriptcode.
- What is watcher?
- Introduction of ts-nodeNPM.

Hands-On Practicals:

- Practicing Watcher mode.

SECTION 4: VARIABLES IN TS

- Difference between Static & Dynamic type checking.
- How to Declare & Define Variables inTS?
- What is any inTS?
- Difference between let &var.
- Template string introduction.

SECTION 5: DATATYPES IN TS

- Brief Introduction of below types,
 - String
 - Number
 - Boolean
 - Array
 - Object
 - Tuple



- EnumAny
- Void
- Never
- Null
- Undefined

Hands-On Practicals:

- Practicing different types of Data types in TS.

SECTION 6: DIFFERENT TYPES OF PARAMETERS IN TS

- Difference between JS parameter & TS Parameter.
- Let vs var vs const
- Passing Parameters to the functions.
- How to pass optional parameter?
- What is Default Parameter?
- How to use default parameter before required parameter?
- Rest Parameter overview.

Hands-On Practicals:

- Passing all above parameters in a single function.

SECTION 7: FEATURES IN TS.

- Arrow functions.
- Where to use Arrow functions?
- Destructuring an Array.
- Destructuring and Object.

SECTION 8: OOPS IN TYPESCRIPT

- What is Class?
- What is Class definition??
- What is Object??
- How to Create Object?
- Brief introduction about,
 - Property
 - Method
 - Constructor
- What is Inheritance?
- Different types of Inheritance in TS.?
- Access modifiers in TS.?
- Read only in TS.?
- Static Property & Methods inTS.
- Interface in TypeScript.

Hands-On Practicals:

- Practicing all opps in a real-time examples.
- Converting JS function into TS Class with oops concepts.

ANGULAR 11 TRAINING COURSE CONTENT

SECTION 1: INTRODUCTION TO ANGULAR

- What is Angular?
- Difference between framework & Library?
- History of Angular and its versions.
- Why Angular?
- Features of Angular.
- What is Single Page Application?
- Difference between SPA & Traditional Application.
- What is MVC?
- How MVC works in Client & Server sides?

SECTION 2: ANGULAR ENVIRONMENT SETUPS

- What is Angular CLI?
- Purpose of the CLI.
- Angular CLI installation.
- Create a Basic Angular App by using CLI.
- Executing the Angular Application in the browser.
- How an Angular app Bootstrapping?
- Angular Libraries
- Brief explanation about the structure of the Angular App.

Hands-On Practicals:

- Create a New Angular App using CLI
- Run Angular app in different ports

SECTION 3: MAIN BUILDING BLOCKS OF ANGULAR

- An Overview of the below Building blocks of Angular
 - Modules
 - Components
 - Decorators
 - Templates
 - Metadata
 - Data binding –Directives
 - Services
 - Dependency Injection.
- What is Module & its importance?
- What is Root Module and Feature Module?
- How to import & Export Module?

Hands-On Practicals:

- Practicing Root Module and Core Module workflows.
- Practicing on Different MetaData properties.

SECTION 4: COMPONENTS IN ANGULAR

- What is Component?
- What is Root Component?
- How to create a Component manually and through CLI?
- What is Component decorator and its Meta data?
- Component's Structure overview.
- Component in an Action
- What is Instance of the Component?
- What are the ways to call a Component in a view?
- Component Lifecycle Hooks.
- What is Nested or Parent Child relation Component?
- Components Interactions overview.
- How to pass Data from Parent to Child & Child to Parent?

Hands-On Practicals:

- Create and Configure a Component without CLI.
- Practice on Nested Components
- Call a Component as Element, Attribute & Class

SECTION 5: DATA BINDING, PROPERTY BINDING, EVENT BINDING AND 2 -WAY DATA BINDING

- What is Data Binding?
- String Interpolation.
- What is Property Binding?
- String Interpolation VS Property Binding
- What is Two-way Data Binding?
- How to implement 2-way Data Binding?
- Style Binding, Class Binding.
- Overview of Event Binding.
- Element reference and \$event service in Angular.
- Event Filtering

Hands-On Practicals:

- Practice on Event Binding with certain conditions.
- Implement few Property Bindings.
- Applying 2-way Data Binding in a form.

SECTION 6: SERVICES IN ANGULAR

- What is Services?
- Why should we use Services?
- How to create Services in Angular?
- What is Dependency Injection?
- How to use Dependency Injection?
- Singleton Object -Overview
- Singleton Obj VS Regular Obj
- What is providers?

- What is @Injectable()?
- Registering a Service in Providers VS @Injectable() Decorators
- What is HierarchicalDI?

Hands-On Practicals:

- Create a Service and use some Business Logics.
- Practice on Dependency Injection & Singleton Object.
- Use Services for sharing Data between Components.

SECTION 7: DIRECTIVES & CUSTOM DIRECTIVES IN ANGULAR

- What is Directives?
- Component VS Directives
- What are the Different kind of Directives available in Angular?
- Difference between Structural & Attribute Directives.
- Overview of All Structural & Attribute Directives.
 - ngIf
 - ngFor
 - ngSwitch
 - hidden
 - ngClass
 - ngStyle - ngNonBindable
 - ngTemplate
 - ngContent
 - ng-container
- Difference between ngIf & hidden directives.
- How to create Custom Directives?
- How to pass Input property to the custom directive?
- How to receive Input property from the custom directive?
- What is @Input decorator and its methods to use?
- What is ElementRef and its purpose?

Hands-On Practicals:

- Practice on All kind of Directives.
- Create a Simple Custom Directive
- Create a Complex Custom directive with accept data from outside.
- Using Custom Directive change the Form values.

SECTION 8: PIPES IN ANGULAR

- What is Pipe in Angular?
- Purpose of the Pipes.
- Difference between Pipes & Directives.
- Detailed explanation of the below Pipes,
 - Lowercase
 - Uppercase
 - Titlecase
 - Slice
 - Json

- Number
- Percent
- Currency
- Date

- What is Pure Pipe & Impure Pipe?
- How to create Custom Pipes?
- What is chaining Pipes?
- What is Parameterized Pipe?

Hands-On Practicals:

- Practice on All kind of default Pipes.
- Transform the form & table data with 7 different custom Pipes.

SECTION 9: COMPONENTS COMMUNICATION

- Overview of Components Interaction
- Share data between components by using Service
- Component Interaction from Parent to Child.
- Component Interaction from Child to Parent.
- @ViewChild decorator overview
- @Input & @Output decorator overview
- Custom Event Binding
- Components Interaction by using Event Emitter & Subject and observable

Hands-On Practicals:

- Practice on Pass data from Parent to Child and vice versa.
- Create Custom Events and Emit Data.

SECTION 10: EVENT EMITTER AND OTHER SPECIAL DIRECTIVES

- What is Event Emitter?
- When to use Event Emitter?
- How to transfer data with Event Emitter?
- How to create Custom Events and Trigger?
- View Encapsulation Overview.
- What is ng-content and its purpose?
- ng-container Directive overview.

Hands-On Practicals:

- Applying View Encapsulation in a real-time Project.
- Trigger custom events from component and subscribing it from another component.

SECTION 11: TEMPLATE DRIVEN FORM IN ANGULAR

- What is Template Driven Form?
- Purpose of Form.
- How to setup a form in Angular?
- What is form control and how to set it?

- What is ngForm?
- How to Access form values?
- What are the form states available in Angular?
- TD Form Validations.
- HTML5 Validation vs Angular form Validation.
- How to set prepopulate values in form fields?
- Setup One-way data binding in Angular.

Hands-On Practicals:

- Create a TD form and get values in Form submission.
- Apply all kind of form validations.
- Get and Set values from Form fields.

SECTION 12: REACTIVE FORM IN ANGULAR

- What is Reactive Form?
- Difference between Template Drive & Reactive Form
- What is form group & form control?
- How to sync view & Reactive formTS?
- What is Patch Values & Set Values?
- How to attach Dynamic Validations?
- How to get Reactive from Values?
- Create Dynamic Form by using Reactive Form

Hands-On Practicals:

- Create a Dynamic form.
- Login form using Reactive Form.

SECTION 13: SERVICE AND DEPENDENCY INJECTION IN ANGULAR

- What is Service in Angular?
- Create & configure Service in Angular.
- How to do Dependency Injection in Angular?
- What is Singleton Object?
- Multiple DI in single Class.

SECTION 14: ROUTING IN ANGULAR

- What is Routing?
- How Routing makes our App into SPA?
- How to configure Routing in an Application?
- Load our components dynamically based on url path.
- How to create Child Route?
- Navigating to other links programmatically.
- Passing Parameter to the Routes.
- Static Route vs Dynamic Route.
- Fetching Route Params.
- Router Directives overview
- Client-side authorization using Route Guard

Hands-On Practicals:

- Create a SPA.
- Practice on Client-side Authorization.

SECTION 15: HTTP & OBSERVABLE IN ANGULAR

- HTTP Client in Angular
- REST API Overview
- How to establish HTTP request to Server side.
- How HTTP Mechanism works?
- What are the methods available in HTTP?
- What is Observable & Observer?
- What are the call back methods available in Observable?
- How to create a CustomObservable?
- What is next(), error() and complete()?
- How to send Query Params & Custom Headers?
- How to connect any backend & APIs?
- What is CORS?
- How to resolve CORS errors?

Hands-On Practicals:

- A real-time HTTP Project by using connecting 4 APIs

SECTION 16: AUTHORIZATION IN ANGULAR & JWT

- Client side Authorization vs Server side Authorization.
- Server side Authorization by using JWT Token.
- Set JWT Key expiry time.
- HTTP interceptors-Overview
- How to configure HTTP interceptors?

Hands-On Practicals:

- Create and Verify JWT key in real-time project.
- Validate the request using HTTP interceptors.

SECTION 17: FILE UPLOAD IN ANGULAR

- Angular form setups for File Upload.
- FormData() in Angular
- Server side setups for File Upload.
- Multer NPM in node.
- How to use Multer NPM as middleware in Node?
- Attach the Multer middleware in server-side routing.

Hands-On Practicals:

- Upload Product Images with validations.
- Move the uploaded files in a directory and save data in DB.

SECTION 18: REAL-TIME PROJECT IN ANGULAR

During the course we will take one Real-time E-commerce application and apply all the above sections into the project.

In the project Front-end will be in Angular and Back-end will be in Node JS. On top of the Node JS we will write Express JS as a REST Api. For Database, we choosed MongoDB for CURD Operations.



CREDO SYSTEMZ