



IT EDUCATION
CENTRE

Angular 17



Introduction

- ▶ Introduction to Angular
- ▶ Introduction to Typescript
- ▶ SPA and Local Setup Typescript

Typescript

- ▶ Typescript Data Types and Functions
- ▶ Classes and Interface
- ▶ Typescript decorators and tsconfig file Angular Installation and Basics

Angular Installation and Basics

- ▶ Angular Installation and Binding Syntax Component Communication
- ▶ Built-in Directives
- ▶ Built-in Pipes
- ▶ Adding Bootstrap CSS to App

Lifecycle Hook and Component Communication

- ▶ `ngOnInit` and Component Communication using Input and Output
- ▶ Change Detection and `ngOnChanges`
- ▶ `ngDoCheck`
- ▶ `ViewChild`, `ViewChildren` and `AfterViewInit`
- ▶ Content Projection, `AfterContentinit` and `OnDestroy`

Dependency Injection

- ▶ Introduction
- ▶ Resolution Modifiers.
- ▶ Value Providers

HttpClient and RxJs

- ▶ Setting Up `HttpClientModule`
- ▶ `HttpService`, RxJs observables and `http get`
- ▶ RxJs Observable and Observer
- ▶ `Http Put` and `Delete`
- ▶ `Http Request`
- ▶ `ShareReplay` RxJs Operators
- ▶ `Async Pipe`

- ▶ **catchError operator**
- ▶ **map operator**
- ▶ **Http Interceptor**
- ▶ **APP INITIALIZER**

Routing Basics

- ▶ **Angular Router and default Route**
- ▶ **Adding Angular material and navigation**
- ▶ **Wild card, dynamic route and ActivatedRoute service**
- ▶ **ParamMap and Activate Route Service**

Template Driven Forms

- ▶ **Introduction**
- ▶ **Validation**
- ▶ **Pristine, Dirty State and Reset Form**
- ▶ **Custom Directives and Custom Validation**

Advanced Routing

- ▶ **Navigation using Router Service**
- ▶ **Feature and Routing Module**
- ▶ **Nested and Child Route**

- ▶ **Lazy Loading**
- ▶ **Configure Lazy Loading using CLI**
- ▶ **Using Provided in Any**
- ▶ **Router Events**
- ▶ **Listening to Router Events**

Route Guards

- ▶ **CanActivate**
- ▶ **CanActivateChild**
- ▶ **CanLoad**

Reactive Forms

- ▶ **Introduction**
- ▶ **Using Material Controls**
- ▶ **Nested Form Controls.**
- ▶ **Dynamic Forms**
- ▶ **Built-in Validators**
- ▶ **Reset Form**
- ▶ **Control Level Validation**
- ▶ **SetValue and PatchValue**
- ▶ **ValueChanges and UpdateOn**
- ▶ **map operator with Form**
- ▶ **Custom Validator**

CanDeactivate Guard

- ▶ CanDeactivate Guard and Form

Custom Pipe

- ▶ Custom Pipe

Resolve Guard

- ▶ Resolve Guard

Global Error Handling

- ▶ Error Handling

Testing Basics

- ▶ Introduction
- ▶ First test
- ▶ Testing Component and Service

Deployment and CI/CD

- ▶ Using Netlify for Deployment
- ▶ GitHub Actions to Automate Tasks

GitHub

- ▶ Creating a Git Account
- ▶ Cloning the repository
- ▶ Adding the file
- ▶ Committing the file
- ▶ Git push
- ▶ Removing the file