

Angular 10 FT103

40 hours

Course Outline:

Single page applications are web applications that are run and managed by JS code, the result is a very fast, desktop like, high performance web application. Building these kinds of applications means that we will have to deal with a lot of difficulties in our JS code, from dealing with forms, routing, page templates, backend-frontend communication, etc. A good frontend framework can help us deal with these challenges, thus Angular.js was born. But Angular.js failed with bad performance, especially on mobile phones, and bad initial load time and SEO. Angular is a remake of the popular Angular.js . A framework built with typescript, helps us solve all our web challenges while maintaining a super fast web application.

This course will teach you how to create a web site using Angular. We will go over the basic challenges and how we can solve them with Angular, and at the end of the course you will be able to build a professional single page application using Angular.

On completion, students will be able to:

- Write Typescript code
- Built web sites using angular

Who should attend the course

- Software developers with OOP experience

Prerequisites:

- Basic JS
- HTML
- CSS



Course Contents:

Module 1 - Typescript

- What is Typescript
- tsconfig.json
- variables - const, let, var
- Typescript Basic Types
- Typescript advanced types
- OOP
- Teaching the Typescript compiler

Module 2 - @angular/cli

- Our first angular application
- Going over project files
- Thinking in monorepository, libraries and infrastructure.

Module 3 - Angular Architecture

- Modules
- Services
- Components
- Directives
- Dependency injection
- Pipes

Module 4 - Component and template

- Component and template
- Passing data from class to template
- Passing events from template to component
- 2 way binding ex.
- Common builtin directives
- NgIf
- NgForOf
- Common builtin pipes
- async, json

Module 5 - RXJS

- How to properly deal with async code.
- Promise
- Promise chaining
- Async Await
- Reactive programming
- RXJS
- Observables
- Observers
- Subscription
- Operators
- Dealing with async errors
- Promise to observable
- Observable to promise

Module 6 - REST server communication

- Services
- HttpClient
- Authentication
- Authorization
- JWT
- Interceptors

Module 7 - @angular/forms

- How to deal with forms using Angular
- Reactive forms, Template driven forms
- FormControl / NgModel
- FormGroup / NgForm
- FormBuilder
- NgSubmit
- Validation
- Building infrastructure and custom NgModel and custom validation

Module 8 - @angular/router

- How to deal with routing with angular.
- Routing architecture
- RouterModule and Route
- SEO and best practices
- Using the routes to pass data
- Lazy loading.