

Angular 9.x Development

32 hours

Course Overview:

Today JavaScript is the language of the web. On the client, on the server, on the mobile, everywhere. But let's face it: JavaScript still has challenges.

Angular brings an entirely new approach to web application development. We need to understand TypeScript, RxJS, and a new template syntax just to get started!

This course provides a developer-level introduction along with more advanced and useful features of JavaScript and Angular.

Who Should Attend:

Experienced Developers in any programming language

Required Skills:

Basic knowledge of HTML and Javascript

Course Contents:

Angular Fundamentals

- The Challenge with Web Apps (SPA)
- Directives & Components
- Pipes
- Forms
- Modules
- Routes

Template Syntax

- Interpolation
- Template expressions
- Binding syntax
- Input and Output Properties
- Template Expression Operators
- Property Binding
- Attribute, Class and Style Bindings
- Event Binding
- Two-way data binding with NgModel
- Built-in Directives
- * and template
- Local template variables

Components

- Component Overview
- Component Metadata (@component)
- Lifecycle Hooks
- Component encapsulation
- @Input and @Output
- @ContentChildren & @ContentChild
- @ViewChildren & @ViewChild
- Component change detection strategy

NAYA College

הנדיב 71, הרצליה | טלפון: 0732865544 | פקס: 0732865417 | www.naya-college.co.il

Directives

- Directive Overview
- Directive Metadata (@Directive)
- Structural Directive
- TemplateRef & ViewContainerRef Classes
- Renderer Class
- Attribute Directive
- @HostBinding
- @HostListener

Forms in Angular

- Forms Overview
- User Input
- Template-driven Forms
- Reactive Forms
- FormBuilder Service
- Dynamic Forms

Angular Pipes

- Using Pipes
- Built-in pipes
- Parameterizing a Pipe
- Chaining pipes
- Custom Pipes
- Stateful Pipes
- Async Pipe

Routing and Navigation

- Navigation flow
- Routes Tree
- RouterLink Directive
- RouterOutlet Component
- Router class
- Router lifecycle hooks & guards
- Lazy loading
- Preloading
- Dynamic Routes

Server Communication (HttpClient service)

- Installation
- Enable RxJS Operators
- Using HttpClient To Request Data
- Using HttpClient To Send Data
- HttpClient API's
- Http Default Options
- Error handling
- Configuring other parts of the request
- Interceptors
- JSON Web Token (JWT)
- Security: XSRF Protection
- Communication with JSONP

- Events (HttpEvent class)

Dependency Injection (DI)

- Why Dependency Injection?
- Angular Dependency Injection
- Configuring the Injector
- Services
- @Injectable() & @inject
- Injector Providers
- The Injector Tree
- Component Injectors

Change Detection & Lifecycle Hooks

- ngZone
- Change Detection Algorithm
- Lifecycle Hooks
- Tips and Tricks to improve the performance

NGModules

- Template Context
- Providers
- Core modules
- Share modules
- Feature modules
- TypeScript Barrels pattern

Async & Parallel programming

- Async programming problems and solutions
 - Callbacks
 - Micro tasks in ECMAScript 2016
 - Promises
 - Async & await
 - Rxjs library
- Parallel programming problems and solutions
 - Multithreading with Workers
 - Operative.js library
 - Parallel.js library
 - Rxjs schedulers
- Angular CLI – Adding Web Worker
- Angular async pipe

NGRX – Reactive State for Angular

- NGRX Architecture
- Action
- Reducers
- Selectors
- @ngrx/store-devtools
- Effects
- @ngrx Entity

Angular Techniques

- Security

- Internationalization (i18n)
- Accessibility
- PWA and Service Worker
- Angular Universal
- ng deploy
- Custom Schematics

Upgrading From 1.X (optional)

- Preparation for migrating.
- Component method vs. Directive method.
- Upgrading with The Upgrade Adapter.