

Angular Advanced: Mastering the framework

Master the advanced concepts and techniques of Angular development for building robust and scalable web applications.

Category: Programming · Duration: 3 days

WHO IT IS FOR

- Front-end developers
- Developers of any background
- Practitioners who already use it daily and want to go further

This comprehensive training program is designed to take your Angular skills to the next level. Following the Angular Fundamentals training course , with a focus on advanced concepts and techniques, you'll learn how to build robust and scalable web applications using Angular. From advanced component design patterns to routing and state management, this course will equip you with the skills needed to excel in Angular development. Last features such as signal, standalone components and so on are included in this course.

Prerequisites

Prior knowledge of Angular fundamentals and experience with web development is required.

Syllabus

Angular Refresher

- Application architecture
- Component-level Lazy Loading
- Automating documentation
- Manual Data Binding with ChangeDetectorRef
- Analyze and optimize an existing application

Angular Internal Mechanics

- ZoneJS: the concept
- Optimizing rendering cycles, running outside ZoneJS
- Why choose RxJS
- Using observables
- Route guards and resolve mechanisms
- CanActivate, CanActivateChild: route protection
- Pre-fetching with resolve

Signals

- Introduction to the Signals API: definition, reading, writing
- Read-only signals
- Connection with observables
- Computed signals
- Effects
- Hands-on lab
- Rewrite a component using signals instead of observables

Angular Standalone

- Why this new architecture?
- Setting up a standalone component
- A new module-free architecture
- Migrating from modular to fully standalone architecture

Internationalization

- i18n markers
- Managing translatable text
- Pluralization handling

Dynamic Forms: FormBuilder

- Creating dynamic forms with ReactiveFormsModule
- FormControl, FormGroup, AbstractControl, FormArray
- Custom validation and error handling
- Creating data models
- Using FormBuilder
- Dynamically generating templates
- Abstracting form business logic into components
- Implementing form generation and advanced validation cycles

Unit Testing: Best Practices and Tools

- Jest vs Karma
- Integration testing with Cypress
- Code coverage
- Behaviour Driven Development (BDD), Test Driven Development (TDD)
- Test cases: pipes, components, services, etc.
- TDD: Develop an application starting from unit tests

NgRx & State Management

- Introduction to NgRx: store, actions, reducers, effects
- Architecture and unidirectional data flow
- Selectors and entity adapters
- DevTools and debugging
- Lazy-loaded state modules
- Integrate NgRx into an Angular feature module

Progressive Web Apps (PWA)

- What is a PWA?
- Setting up Angular PWA support
- Service workers and caching strategies
- Offline mode and update mechanism
- Manifest configuration
- Convert the existing application into a PWA

Performance Optimization & Debugging Tools

- Using Angular DevTools and browser profiling
- Change detection debugging with ng.profiler
- trackBy and structural directive performance
- Memory leak detection and optimization
- Profile performance before and after refactoring

Advanced Dependency Injection

- InjectionToken and @Inject
- Multi-providers and useClass/useFactory
- DI scopes: root, platform, any
- Tree-shakable providers
- Standalone DI contexts
- Create a flexible and reusable service setup

Security and Best Practices

- Template and DOM XSS protection
- Authentication tokens in HttpInterceptor
- Role-based route guards
- Safe API interactions and backend trust
- CSP and Angular sanitizer
- Secure an application end-to-end with guards and interceptors