ProTech Professional Technical Services, Inc.

ProTech
protechtraining.com

Advanced Angular 7
Course Summary

Description

Use Angular 7 to easily build web applications that interacts with the user by dynamically rewriting the current page rather than loading entire new pages from a server. Learn how to build these applications using the newest JavaScript language features by leveraging ES6, TypeScript, and modern front-end tools including npm and Webpack. Understand application architecture and design best practices in Angular. Get up to speed on how to authenticate, unit test, and manage application state in an Angular application.

Objectives

- After taking this course, students will be able to:
- Understand how single-page web application architectures are different than traditional web application architectures
- Use new JavaScript (ES6) language features including Classes, Modules, and Arrow Functions
- Use new TypeScript language features including Types, Decorators, Interfaces, and Generics
- Learn Angular coding and architecture best practices including project layout and using container and presentation components
- Understand and use Angular model-driven forms, observables, dependency injection, and routing
- Communicate with a backend server using Angular's HttpClient to load and save data
- Configure the router and navigate between components
- Unit test all parts of an application including Components, Services, and Pipes

- Understand RxJS and Observables and where they can be used
- Implement Authentication and Authorization in an Angular Application
- Optimize Angular Performance by changing Change Detection Strategies
- Setup new projects from scratch using the Angular CLI
- Scaffold modules, components, services, models, routes, and unit tests in accordance with best practices using the Angular CLI
- Build and deploy an application to production using the Angular CLI
- Write End-to-End Tests (optional; taught only if this applies to your group)
- Upgrade an existing application from AngularJS to Angular 7 (optional; taught only if this applies to your group)

Topics

- Unit Testing
- RxJS and Observables
- Change Detection
- Advanced Routing

- Advanced Dependency Injection
- Pipes
- Two optional topics
- Conclusion

Prerequisites

All Angular training students must have substantial prior experience developing with JavaScript. If attendees will not have prior JavaScript experience, we would be delighted to precede this class with a one- or two-day intensive JavaScript primer.

Duration

Two Days

ProTech Professional Technical Services, Inc.

Advanced Angular 7
Course Outline

I.   Unit Testing
- A.   Tools: Jasmine, Karma
- B.   Jasmine Syntax: describe, it, beforeEach, afterEach, matchers
- C.   Setup and your First Test
- D.   Testing Terminology: Mock, Stub, Spy, Fakes
- E.   Angular Testing Terminology: TestBed, ComponentFixture, debugElement, async, fakeAsync, tick, inject
- F.   Simple Component Test
- G.   Detecting Component Changes
- H.   Testing a Component with properties (inputs) and events (outputs)
- I.   Testing a Component that uses the Router
- J.   Testing a Component that depends on a Service
- K.   Testing a Service and Mocking its Http requests
- L.   Testing a Pipe

II.   RxJS and Observables
- A.   What is an Observable?
- B.   Creating Observables
- C.   What is an Observer?
- D.   Observer Example
- E.   Operators: map, switchMap, debounceTime, distinctUntilChanged
- F.   Practical Application of using RxJS
- G.   Subject
- H.   Subject Example
- *I.*   EventEmitter or Observable
- *J.*   *Security*
- K.   Best Practices
- L.   Preventing Cross-site Scripting (XSS)
- M.   Trusting values with the DOMSanitizer
- N.   HTTP Attacks (CSRF and CSSI)
- O.   Authentication using JSON Web Tokens (JWT)
- P.   Authorization: Router Guards

III.   Change Detection
- A.   Understanding Zone.js and Change Detection
- B.   Change Detection Strategies Default and OnPush

IV.   Advanced Routing
- A.   Lazy-loading Angular Modules
- B.   Nested or Child Routes

V.   Advanced Dependency Injection
- A.   Providers
- B.   Hierarchical Injection

VI.   Pipes
- A.   Creating a custom Pipe using PipeTransform
- B.   Understanding Pure and Impure Pipes

VII.   Choose any two optional topics. If desired, the course can be customized to include more than two of these topics if other topics are scaled back or removed.
- A.   npm QuickStart
  1. Installing Dependencies
  2. Understanding package.json and package-lock.json
  3. Using npm as a Build Tool
- B.   Managing Shared Application State using ngrx and Redux
  1. Benefits Overview
  2. Three Principles of Redux: Single Source of Truth, State is Read-Only, Pure Functions
  3. Examples of Pure Functions
  4. Reducers
  5. Simple ngrx Example
  6. Time-traveling with Redux Devtools
  7. Full ngrx Example Application
- C.   Upgrade Strategies from AngularJS
  1. High-level Approaches
  2. Concept Mapping AngularJS to Angular
  3. UpgradeAdapter

Course Outline (cont.)

4. What can be Upgraded or Downgraded
5. What cannot be Upgraded or Downgraded
6. UpgradeAdapter and Dependency Injection
D. End-to-End Testing
   1. What is Protractor?
   2. Why Protractor?
   3. Using Locators
   4. Page Objects
   5. Debugging E2E Tests
E. Communicating with the Server using WebSockets
F. Creating Custom Attribute Directives

1. Creating a custom Attribute Directive using ElementRef and Render
G. Webpack Guide
   1. Installation
   2. Building/Bundling
   3. Development Builds
   4. Production Builds
H. Template-driven Forms
   1. NgSubmit Directive
   2. FormsModule
   3. NgForm, NgModel, and NgModelGroup Directives
   4. Validation Directives
I. Introduction to Angular Material Design

VIII.   Conclusion