

Document Generated: 01/24/2026

Learning Style: Virtual Classroom

Technology:

Difficulty: Intermediate

Course Duration: 5 Days

Next Course Date: **February 9, 2026**

Mastering Angular 20 Boot Camp (TT4168)



About this course:

Geared for experienced web developers, our Mastering Angular 20 Boot Camp is a five-day, comprehensive hands-on program that explores the latest features and benefits Angular has to offer. Throughout the course you'll learn the practical "real-world" foundational and advanced Angular skills required to solve the usual challenges you might face when developing modern web applications with JavaScript.

Kicking off with an in-depth introduction to Angular CLI and TypeScript, the course propels you into the world of component-driven development, touching on vital topics like data-binding, directives, and services. As you advance through the curriculum, you'll explore SPA routing, the RxJS library, forms, and unit testing, along with HTTP/REST protocols for server interaction.

By the end of this course, you'll have built a comprehensive application that brings together all these elements: website structuring through components, dynamic interactivity via directives, and custom pipes for content refinement. You'll know how to employ reactive forms for active user engagement and SPA routing for fluid navigation, fortified by efficient data management through services. All of this is underscored by a strong emphasis on unit testing, as well as practical REST API interactions for modern server communication.

You will leave the course equipped to construct robust, scalable, and feature-rich web applications using Angular 18, able to implement seamless user experiences, integrate with various data sources through REST APIs, and employ best practices in unit testing for sustainable, high-quality code.

Course Objective:

The lab intensive course includes plenty of hands-on lab work designed to immerse you in the skills required to get you up and running with Angular right away. The course is rich with hands-on activities, challenge labs, knowledge checks, valuable discussions and focused projects.

Guided by our engaging, highly-experienced web development instructor, you will learn how to:

- Build Powerful Components: Master creating and organizing Angular components, leveraging dependency injection and lifecycle hooks for scalable apps.
- Harness TypeScript's Power: Become proficient in TypeScript, using its advanced features like classes, interfaces, and decorators to supercharge your Angular development.
- Create Dynamic, Reactive Forms: Learn to build complex forms with Angular's reactive approach, ensuring seamless validation and user

interaction.

- Connect with RESTful Services: Integrate Angular apps with RESTful services using HttpClient, and handle asynchronous data streams with RxJS for efficient server communication.
- Optimize with CLI and DevTools: Use Angular CLI for easy project setup and deployment, and harness DevTools for debugging and optimizing your applications.
- Master SPA Navigation: Implement smooth and efficient Single Page Application (SPA) routing with nested routes and route guards for a top-notch user experience.
- Utilize Third-Party Libraries: Learn to effectively integrate and manage third-party libraries within Angular to extend functionality and streamline development.
- Implement Effective Testing: Gain skills in using tools like Jest and Web Test Runner for unit testing, ensuring your Angular applications are robust and error-free.

Audience:

- The content is appropriate for experienced web developers new to Angular. In order to be successful in the course, you should have solid, current hands-on experience developing basic web applications, and be well versed in HTML5, CSS3 and JavaScript.

Prerequisite:

- The content is appropriate for experienced web developers new to Angular. In order to be successful in the course, you should have solid, current hands-on experience developing basic web applications, and be well versed in HTML5, CSS3 and JavaScript.

Course Outline:

1. Overview of Angular Architecture

- Angular Versioning
- Data Binding
- Component Architecture

- Show concepts: how to render the content in the browser
- Components and Dependency Injection
- Services and Events
- Common Component Lifecycles and Hooks

2. TypeScript

- Angular, ES6 and TypeScript
- Typing and Classes
- Abstract Classes and Interfaces
- Interface patterns
- Annotations
- Generics
- Optional Chaining
- Nullish coalescing
- Decorators
- Functional vs Procedural JavaScript
- Functional Approach
- Lab: Quick Hands-on TypeScript

3. Bootstrapping with Angular CLI

- Angular CLI Overview
- New Projects with CLI
- Testing and Generating with CLI

4. Angular Basics

- Configuration Files
- Working with angular.json

- Top-Level Directories
- Contents of app folder
- Angular DevTools
- ng build
- Lab: Creating Projects with Angular CLI

Working with Angular

5. Components and Events

- Data Binding and Components
- Event Binding
- Custom Events
- Parent/Child Events
- EventEmitter/emit()
- New Input/Output function
- Lab: Nested Components

6. Standalone Components

- Project Setup and Configuration
- Folder Structure
- Components
- Services
- Routing
- Versus non-standalone components (overview)
- Versus modules (overview)

7. Third Party Libraries

- Angular and the Need for Libraries

- Options for adding Libraries
- Working with npm install
- Lab: Third-Party Libraries

8. Dynamic Views - Directives

- View Encapsulation
- Structural/Attribute Directives
- Conditional Styling
- @if/@else
- @for
- @defer
- Lab: Working with Directives and control flow blocks

9. Pipes

- Overview of Pipes
- Built-in Pipes
- Formatting and Conversions
- Parameterizing Pipes
- Pure vs. Impure Pipes
- Customizing Pipes
- Lab: Built-in and Custom Pipes

10. Signals and State

- What are Signals?
- How to read the value of a signal?
- How to modify the value of a signal?
- The update signal API

- Read-only signals
- The computed() Signal API
- How do we subscribe to a signal?
- Using effects
- Lab: Signals

Angular Forms

11. Forms and the Forms API focus on Reactive Forms

- Template Forms Overview
- User Input/Interactions
- Model-Driven Reactive Forms
- FormBuilder API
- Reactive Form Validation
- Forms new control state change events
- Lab: Reactive Forms

Single Page Applications and Routes

12. SPA - Routing

- Overview of the SPA Concept
- Routing for Page Display
- Working with ActivatedRoute
- Location Strategies
- Nested Routes
- Route redirects as functions
- Lab: Routing

13. Services (Dependency Injection)

- Angulars DI Framework
- Components and Injectables
- Tree-shakeable providers
- Services Architecture
- Service Interfaces and DI
- Constructor Injection
- Inject function
- Scenarios for use
- Helper Services
- Fascade Services
- Lab: Services

14. Modules

- Overview of Modules
- Feature Modules
- Feature Module Routing
- Shared Modules
- Lab: Modules

Using RESTful Services

15. Angular and REST

- REST in Angular
- Promises, Observables, Subject
- HttpClient and Observables
- Deprecation of HttpClientModule
- Async Pipes and HttpClient Interceptors

- Angular working with axios

- Lab: Rest with axios

14a. Standalone Components vs Modules

- Review of Standalone components
- A non-standalone component
- Using standard directives in standalone components
- practical benefits of standalone components
- Using standalone components in NgModule-based components
- Using feature NgModule-based components in standalone components
- Lazy loading with standalone components
- 3 steps to migrating to standalone components
- Lab: Using Standalone components alongside modules

Using RESTful Services

15. Angular and REST

- REST in Angular
- Promises, Observables, Subject
- HttpClient and Observables
- Deprecation of HttpClientModule
- Async Pipes and HttpClient Interceptors
- Angular working with axios
- Lab: Rest with axios

Reactive Programming in Angular

16. Working with RxJS

- Working with RxJS in Angular

- RxJS Operators
 - switchMap, tap, map, forkJoin, combineLatest, etc
 - Most used operators
- Asynchronous operations
- Manipulate stream
- Enhanced Lab: Working with RxJS

17. Testing and Angular

- Testing Dependencies
- Options for Testing in Angular
- Using Jest in Angular
- Using the Web Test Runner
- Enhanced Lab: Unit Testing

Bonus Content / Time Permitting

18. Zoneless

- The Need for Zoneless Angular
 - Performance considerations
- Zone Change Detection
- Advantages and Trade-offs

Errors, Debugging and Standalone

19. Angular Error Handling and Debugging

- With Chrome DevTools
- With VS Code
- Http client request errors
- Lab: Debugging Angular

20. Angular coding style guide and Best Practices

- Angular 16 to Angular 20 Upgrade (Optional)

Credly Badge:



Display your Completion Badge And Get The Recognition You Deserve.

Add a completion and readiness badge to your LinkedIn profile, Facebook page, or Twitter account to validate your professional and technical expertise. With badges issued and validated by Credly, you can:

- Let anyone verify your completion and achievement by clicking on the badge
- Display your hard work and validate your expertise
- Display each badge's details about specific skills you developed.

Badges are issued by QuickStart and verified through Credly.

[Find Out More](#) or [See List Of Badges](#)