

Document Generated: 09/12/2025

Learning Style: Virtual Classroom

Technology:

Difficulty: Intermediate

Course Duration: 5 Days

Next Course Date: **October 20, 2025**

## Mastering Angular 18 Boot Camp (TT4168)



## About this course:

**Mastering Angular** covers and provides practical and hands-on experience with a great option for building rich web applications. If you are building for the web, learning Angular will change the way you write web interfaces, working across multiple browsers and platforms. Angular provides the maximum performance possible on today's web platform.

The average salary of an Angular developer salary is **\$120,000** per year.

## Course Objective:

This course provides indoctrination in the practical use of the umbrella of technologies that are on the leading edge of modern web development.

Working within in an engaging, hands-on learning environment, guided by our expert team, attendees will learn to:

- What Angular is and why should you use it
- How Angular reduces the amount of code that you must write to add rich functionality to both existing and new web pages
- What TypeScript is, why it is useful, and how to use it with Angular
- How to facilitate development and deployment using Angular CLI
- How to work with the various aspects of the Angular architecture to implement clean, responsive web interfaces
- How Routers can support navigation within a Single Page Application
- What the best practices are for using Angular so that it works unobtrusively and performs well
- Working with the Ahead of Time compiler including its impact of developers and the development process
- How to facilitate testing and running Angular applications using various tools and techniques
- How to use Angular with HTTP to support JSON, REST, and other services
- Best practices for Angular so that it works unobtrusively and performs well
- How to meet huge data requirements by building data models on RxJS
- Simplifying server-side rendering
- How to use Angular with HTTP to support JSON, REST, and other services
- How to facilitate unit testing
- Understanding the components of the MEAN Stack
- How to use the MEAN stack to deploy, support, and run Angular applications

## Audience:

- This is an **introductory level** Angular development course, designed for experienced web developers that need to further extend their skills in web development. In order to be successful in this class.

## **Prerequisite:**

- Attendees are required to have current, hands-on experience in developing basic web applications, and be versed in HTML5, CSS3 and JavaScript.

## **Course Outline:**

### **Module 1: Angular Overview**

#### **Lesson: HTML5, CSS3, RWD, and jQuery**

- Leading to Angular
- Evolution of HTML and CSS
- Responsive Web Design
- jQuery Under the Covers
- Bootstrapping to Browsers

#### **Lesson: Angular Framework**

- Frameworks vs Libraries
- AngularJS to Angular2
- Angular Release Schedules
- Angular 4/5

#### **Lesson: Overview of Angular Architecture**

- Architectural Elements and Concepts
- TypeScript
- Model-View-\*
- Data Binding
- Components and Services
- Factories and Events
- Exercise: Hello Angular

### **Module 2: Getting Started with Angular**

#### **Lesson: Getting Started with TypeScript**

- Why Use Typescript
- Typescript Module System
- Types/Keywords/Concepts
- Transpilers
- Angular and TypeScript
- Exercise: Hello TypeScript

#### **Lesson: Bootstrapping with Angular CLI**

- Challenges in Setting up Projects

- Angular CLI Overview
- New Projects with CLI
- Testing and Generating with CLI
- Ahead of Time (AOT) Compilation
- Implications of using AOT

### **Lesson: Angular Project Structure**

- Top-Level Files and Directories
- Folders” src and src/app
- Exercise: Creating Projects with Angular CLI?

### **Lesson: Components and Events**

- Usable Classes in Angular
- HTML, CSS, and TypeScript in Components
- Data Binding and Events
- Exercise: Building Components

### **Lesson: Third Party Libraries**

- Angular and the Need for Libraries
- Options for adding Libraries
- Working with npm install
- Working with angular-cli.json
- ng2 Wrappers
- Exercise: Using Libraries

## **Module 3: Views**

### **Lesson: Dynamic Views**

- Overview of Directives
- Built-in Directives
- ngIf and else
- ngFor and ngSwitch
- Conditional Styling
- Exercise: Working with Directives

### **Lesson: Pipes**

- Overview of Pipes
- Built-in Pipes
- Formatting and Conversions
- Parameterizing Pipes
- Exercise: Working with Pipes

## **Module 4: Angular Forms**

## **Lesson: Forms and the Forms API**

- Forms and NgModel
- NG Form Groups
- NG Form Validation
- Built-in Validators
- Exercise: Basic Forms with Angular
- Exercise: Powerful Forms with Angular

## **Lesson: Single Page Applications**

- Overview of SPA Concept
- Routing in SPAs
- Creating Routes
- Routing Options
- Exercise: Routing in a Single Page Application

## **Lesson: Injectables and Local Storage**

- Components and Injectables
- Services
- Service Interfaces
- Local Storage
- Exercise: Local Storage

## **Module 5: Unit Testing with Angular**

### **Lesson: TDD, BDD, and ADD**

- Test-Driven Development
- Testing Types
- Behavior-Driven Development
- Acceptance-Driven Development

### **Lesson: Testing and Angular**

- Testing dependencies
- Options for Testing in Angular
- Components and Injectables
- Services
- LocalStorage
- Exercise: Unit Tests with Angular, Karma, and Jasmine
- Exercise: Local Storage Service

### **Lesson: Angular Best Practices**

- Upgrading from Angular 1.x
- Angular Style Guide
  - Single Responsibility Principle

- Naming Conventions
- Coding Conventions
- Application Structure
- Components, Directives, Services
- Lifecycle Hooks
- Routing

## **Module 6: Using RESTful Services**

### **Lesson: Overview of REST**

- REpresentational State Transfer
- REST and HTTP
- REST/HTTP: Representation-Oriented
- REST Design Principles

### **Lesson: Angular and REST**

- REST in Angular
- Promises
- Observables
- Http Module
- Putting it all Together
- Exercise: Using RESTful Services in Angular?

## **Module 7: Working with Applications**

### **The MEAN Stack**

- MEAN Components and Overview
- Working with MongoDB
- Working with Node.js
- Working with Express
- Creating a Node.js Web Server
- Exercise: Implementing a MEAN Application
- Exercise: Working with the Database

## **Module and Dependency Management**

- Node.js Projects and Modules
- Making a Module Executable
- Core Modules
- Dependency Management Using NPM
- Setting Up Dependencies
- Installing, Updating, and Uninstalling Packages
- Exercise: NPM Cache Clean
- Exercise: NPM Init
- Exercise: Package.json

## 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](#)