Angular Fundamentals for Professionals: Build Maintainable SPAs with TypeScript

Class Duration

21 hours of live training delivered over 3-5 days to accommodate your scheduling needs

Student Prerequisites

- Good understanding of general programming concepts including asynchronous programming
- Experience with JavaScript or TypeScript programming
- Experience with web browser programming is recommended but not required

Target Audience

Designed for professional developers, tech leads, and solution architects building enterprise web apps who want a rigorous, practical upgrade to modern Angular. It suits teams moving from older Angular versions or other SPA frameworks, as well as managers and L&D leaders seeking a course that aligns with production workflows, enforces code quality, and accelerates delivery across online or in-person cohorts.

Description

This live, hands-on course equips software teams to ship reliable, maintainable Angular applications faster. Participants start by setting up a professional workflow with VS Code, Git, and the Angular CLI, then dive into core platform skills: TypeScript and JavaScript, project scaffolding and updates, linting, and configuration via angular.json and the Angular MCP server. You will build components with robust data and event binding, leverage signals for reactivity, and control UI flow with modern structural directives. The course then moves to real-world app architecture—routing with parameters, testable services, and data transformation with pipes—before integrating remote data using HttpClient and the AsyncPipe. Throughout, the emphasis is on patterns that reduce defects, speed delivery, and scale across teams.

To discuss this course and customizations:
Call: 434-509-5680 or Email: sales@cloudcontraptions.com

Learning Objectives

- Set up a professional Angular development environment with VS Code, Git, and Angular CLI.
- Apply modern JavaScript and TypeScript features to write clear, type-safe application code.
- Scaffold, configure, and update Angular projects, including linting and angular.json tuning.
- Design reusable components and implement data, event, and change-detection patterns, including signals.
- Use structural directives for conditions and loops and apply component styling with classes and inline styles.
- Configure client-side routing, define routes, and pass and consume route parameters.
- Architect and register Angular services and transform UI data with built-in and custom pipes.
- Integrate HttpClient for API calls and manage async data with services and the AsyncPipe.

Training Materials

All students receive comprehensive courseware covering all topics in the course. Courseware is distributed via GitHub in the form of documentation and extensive code samples. Students practice the topics covered through challenging hands-on lab exercises.

Software Requirements

Students will need a free, personal GitHub account to access the courseware. Students will need permission to install Node.js and Visual Studio Code on their computers. Also, students will need permission to install NPM Packages and Visual Studio Extensions. If students are unable to configure a local environment, a cloud-based environment can be provided.

Training Topics

Overview

- Overview of Angular
- Development Environment Setup
- Git Basics and Obtaining Course Demos & Solutions
- Basics of VS Code

- Switching branches in VS Code
- Create a Branch for your Course work

Setting Up Dev for Angular
- JavaScript Syntax
- Getting Started with TypeScript

Creating and Exploring Angular Projects
- Getting Started with Angular CLI
- Starting the Angular Project
- How Angular projects are loaded
- Linting in Angular
- Modifying the project angular.json
- Updating your projects with Angular CLI
- Angular MCP Server

Data and Event Binding
- Creating New Components
- Data Bindings and Template Expressions
- Triggering Change Detection
- Event Binding for Handling Browser UI Events
- Passing Data from Parent to Child - received as @Input()
- Passing Data from Child to Parent with @Output()
- Signals

Conditions, Looping and Styles
- Overview of Directives
  - @for
  - @if
- Adding Class and Style to Components

Routing
- Setting up Routing in Angular
- Defining and activating routes
- Passing Route Parameters

Services & Pipes
- Design Angular services
- Creating and registering services

- Using Pipes to Transform Data

HttpClient

- Starting a server for making HTTP calls
- Adding HttpClient in Services
- Using AsyncPipe