



5-DAY ANGULAR TRAINING OUTLINE

DAY 1: TYPESCRIPT, NODE JS and NPM

> Introduction to TypeScript

- Using the TSC compiler
- Basic types
- Usefulness of types
- Interfaces, classes and object inheritance
- Getters, setters and static properties
- Constructor syntax
- Generics and advanced types
- Modules: Export and import
- ES6-ES7 features: Arrow functions, backticks, template expressions, object rest and spread

> Introduction to Node JS and NPM

- What is Node JS?
- Example of server code with Node
- What is NPM?

- Npm install, npm update and other useful commands
- Semantic versioning and package.json

DAYS 2 to 4: ANGULAR

During these 3 days, all lab exercises are designed to make attendees build an entire web application step by step so that they can practice and understand how these Angular concepts come into play in an actual real life project.

Chapter 1 - Introduction to Angular

- > What is Angular?
- > Why using it?
- > What about Angular JS?
- > Why TypeScript?

LAB: Set-up

- > Introduction to Angular CLI

Chapter 2 - Building blocks of an Angular application

- > Modules and ngModule
- > Components and component libraries
- > Introduction to pipes, directives, and services

Chapter 3: Components

- > What are components?
- > Example of Angular component

LAB: Hello World

- > Template and expressions
- > Passing data to a component with @Input()
- > Practicing with Angular CLI

LAB: Creating components

Chapter 4: Data bindings

- > One-way data bindings
- > Expressions
- > Two-way data bindings with ngModel
- > Template syntax
- > Template reference variables

LAB: Bindings

Chapter 5: Control Flow and Directives

- > Blocks for control flow: @if, @for, @switch

- > What are directives for?
- > Different kinds of directives
- > ngFor
- LAB: ngFor
- > ngIf
- LAB: Hiding things
- > ngSwitch
- > ngClass and ngStyle
- > How to create a custom directive?
- LAB: Custom directive

Chapter 6: Pipes

- > What are pipes for?
- > Examples of common Angular pipes
- LAB: Currency pipe
- > Pure and Impure pipes
- > How to create our own pipes?
- LAB: Custom Pipe

Chapter 7: Project structure and build process

- > How Angular bootstraps an application
- > Importance of using angular CLI to keep the project structure updated
- > Angular builds and environments
- LAB: Angular builds

Chapter 8: Services and dependency injection

- > Difference between components and services
- > Dependency injection in Angular
- > Injector Tree
- > How to create a service?
- LAB: Services

Chapter 9: Reactive Programming with RxJs Observables

- > Introduction to RxJS and Reactive Programming
- > What is an observable?
- LAB: Observables
- > Writing our own Observable
- > RxJs operators
- > Examples of RxJs operators: max, merge, filter
- > Introduction to Subjects

Chapter 10: HTTP client

- > How to use the HTTP client?

- > Http Interceptor

LAB: Using the HTTP client

- > Async pipe

LAB: Using the async pipe

Chapter 11: Component router

- > What is the component router?

- > Example of component router in action

- > Child routes and route parameters

LAB: Component router

- > Guards for authentication and user rights

LAB: Authentication

- > Lazy-loading

Chapter 12: Forms

- > How to handle forms with Angular?

- > Template driven forms and reactive forms

- > Form validation and submission

LAB 19: Template driven forms - Form validation

- > Custom form validator

LAB 20: Reactive forms

Chapter 13: Component communication and lifecycle

- > How do components share data?

- > @Output() decorator

- > @ViewChild() decorator

LAB: Component communication

LAB: Using @Output()

- > Introduction to the component lifecycle

- > Lifecycle hooks

- > Change detection strategies

Chapter 14: Testing Angular Applications

- > How to test Angular applications?

- > Karma and Jasmine

LAB: Writing Unit tests

- > Angular Test Bed

- > Mocking and stubbing with Jasmine and Angular

LAB: Writing Component Tests

- > End to end tests with Protractor

Chapter 15: Angular Ecosystem: Other projects & tools

- > Progressive web apps
- > Compodoc
- > Custom Elements
- > Angular Material and CDK
- > CLI Schematics
- > Angular Universal or how to pre-render on the server
- > Angular Augury: Debugging Angular apps in Google Chrome

Chapter 16: Resources and best practices

- > How to use angular.io (cheatsheet, etc.)
- > Style guide and best practices
- > Official Angular blog

DAY 5: FINAL PROJECT

- > Students build an application from scratch and practice everything they learned during the training
- > Best way to get a lot of experience with Angular in very little time
- > Trainer acts as a mentor and answers all questions, provides guidance, and covers additional topics upon request
- > Usually the preferred part of the training for all students