



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 features in TypeScript: 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

Chapter 1 - Introduction to Angular

- > What is Angular?
- > Why using it?
- > What about Angular 1?
- > Why TypeScript?

LAB 0: Set-up

- > Introduction to Angular CLI

Chapter 2 - Building blocks of an Angular application

- > Modules and ngModule
- > Components, pipes, directives and services
- > main.ts
- > Importance of using angular CLI to keep project structure updated
- > Build modes and environments

Chapter 3: Components

- > What are components?
- > Example of Angular 2 component

LAB 1: Hello World

- > Template and expressions
- > Practicing with Angular CLI

LAB 2: Angular CLI

Chapter 4: Data bindings

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

LAB 3: Bindings

Chapter 5: Directives

- > What are directives for?
- > Different kinds of directives
- > ngFor

LAB 4: ngFor

- > ngIf

LAB 5: Hiding things

- > ngSwitch

LAB 6: ngSwitch

> How to create a custom directive?

LAB 7: Custom directive

Chapter 6: Pipes

> What are pipes for?

> Examples of common Angular 2 pipes

LAB 8: Date pipe

> How to create our own pipes?

LAB 9: Custom Pipe

Chapter 7: Services and dependency injection

> Difference between components and services

> Dependency injection in Angular 2

> Example of common service: Location

> How to create a service?

LAB 10: Services

Chapter 8: HTTP service and asynchronous data

> How to use the HTTP service?

> What is a promise?

> What is an observable?

LAB 11: Observables and Promises

LAB 12: Using the HTTP client

> Using the async pipe

LAB 13: Using the async pipe

Chapter 9: Component communication and lifecycle

> How do components share data?

LAB 14: Component communication

> Introduction to the component lifecycle

> Lifecycle hooks

Chapter 10: Component router

> What is the component router?

> Example of component router in action

> Child routes and route parameters

LAB 15: Component router

> Lazy-loading

> Guards for authentication and user rights

LAB 16: Authentication

Chapter 11: Forms

- > How to handle forms with Angular?
 - > Template driven forms and event driven forms
 - > Form validation and submission
- LAB 17: Template driven forms
LAB 18: Model driven forms

Chapter 12: Advanced tools

- > Angular Mobile toolkit for progressive web apps
 - > Angular Universal or how to pre-compile on the server
 - > Angular Augury: Chrome Debugger
- LAB 19: Angular Augury

Chapter 13: Testing Angular Applications

- > How to test Angular 2 applications?
 - > Karma and Jasmine
 - > End to end tests with Protractor
- LAB 20: Writing Unit tests
- > Angular Test Bed
- LAB 21: Writing Component Tests
LAB 22: Mocking and stubbing

Chapter 14: 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 learnt 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