ANGUALAR 8

Duration- 30 hours – 5 full days

Prerequisite - Prior knowledge of TypeScript.
A development machine with Node 8.9+ and NPM 5.5. 1+ installed. Node is required by the Angular CLI. You can head to the official website and grab the binaries for your system.

Course Objectives

- Develop single page Angular applications using Typescript
- Set up a complete Angular development environment
- Create Components, Directives, Services, Pipes, Forms and Custom Validators
- Handle advanced network data retrieval tasks using Observables
- Consume data from REST web services using the Angular HTTP Client
- Handle push-data connections using the WebSockets protocol
- Work with Angular Pipes to format data
- Use advanced Angular Component Router features
- Test and debug Angular applications using built in tools
- Work with Angular CLI

Lab Setup Details -
Hardware requirement:

CPU : I4/I5 , Hard Disk : 50GB , RAM :8 GB

Software Requirement:

- XAMPP https://www.apachefriends.org/download.html (DOWNLOAD VERSION 5.6.31)
- MSOFFICE 2010
- NOTEPAD++ https://notepad-plus-plus.org/download/v7.5.1.html (DOWNLOAD VER 7.5.1)
- SUBLIME TEXT3 https://www.sublimetext.com/3
- Visual Studio Code https://code.visualstudio.com/download
- PDF READER https://adobe-reader.en.softonic.com/download
- Browser Google Chrome OS Window 10

Course Outline

Day 1:
Angular Introduction
Introduction to Prerequisites- HTML5, CSS3, JAVA SCRIPT, JQUERY AND BOOTSTRAP
What is Angular?
AngularJS vs Angular 2 vs Angular 7
Angular Architecture
Angular Workflow
Introduction to Node.JS
What is Node.js?
Application of Node.js
Installing Node.js and NPM
How it Works

Why Node.js for the Angular 7
Using Node Package Manager (NPM)
Loading Module from node_modules Folders
Dependency Management Using NPM
Installing a Package
About Global Installation
Updating Packages
Uninstalling Packages
Angular Command Line Interface (CLI)
Using NG Commands

Project Setup and First App
Understanding Angular Project Structure
Editing the First App
What is Type Script?
Typescript as Programming Language for Angular
Trans compilation, Debugging Typescript code
A Basic Project Setup using Bootstrap for Styling
How an Angular App gets Loaded and Started?
Angular Bootstrapping
Component Based Approach (Angular)
Creating a New Component
A brief introduction on component architecture
Component Life Cycle
Understanding the Role of AppModule and Component Declaration
Working with Component Templates
Working with Component Styles
Fully Understanding the Component Selector
Splitting Apps into Components
Property & Event Binding Overview
Binding to Custom Properties
Assigning an Alias to Custom Properties
Binding to Custom Events
Assigning an Alias to Custom Events
Custom Property and Event Binding Summary
Understanding View Encapsulation
More on View Encapsulation
Projecting Content into Components with ng-content
Understanding the Component Lifecycle
Seeing Lifecycle Hooks in Action
Lifecycle Hooks and Template Access
Practicing Property & Event Binding and View Encapsulation
Using Libraries like JQuery, bootstrap.js, material, etc.

Day 2:
Data Binding
What is Databinding?
String Interpolation
Using PIPE in Angular
Using Slice in Angular
Property Binding
Property Binding vs String Interpolation
Event Binding
Passing and Using Data with Event Binding
Two-Way-Databinding

FormsModule is Required for Two-Way-Binding!
Adding Navigation with Event Binding and ngIf
Passing Recipe Data with Property Binding
Passing Data with Event and Property Binding
Building reusable components
@Input, @Output, ngContainer , ngContent

Understanding Directives
Using ngIf to Output Data Conditionally
Enhancing ngIf with an Else Condition
Styling Elements Dynamically with ngStyle
Applying CSS Classes Dynamically with ngClass
Outputting Lists with ngFor
Creating a Basic Attribute Directive
Binding to Directive Properties
Building a Structural Directive
Understanding ngSwitch

DAY 3:
ANGULAR ROUTING
Why do we need a Router?
Setting up and Loading Routes
Navigating with Router Links
Understanding Navigation Paths
Styling Active Router Links
Navigating Programmatically
Using Relative Paths in Programmatic Navigation
Passing Parameters to Routes
Fetching Route Parameters
Fetching Route Parameters Reactively
Configuring the Handling of Query Parameters
Passing Static Data to a Route
What is a Single Page Application (SPA)
SPA Workflow
Traditional Web Application Capabilities
Single Page Application Advantages
SPA and Traditional Web Sites
Implementing SPA's Using Angular 7
Build real-world single page applications (SPA) with Angular

Day 4:
HANDLING FORMS IN ANGULAR APPS
Template-Driven (TD) vs Reactive Approach
Creating the Form and Registering the Controls
TD: Submitting and Using the Form
TD: Understanding Form State
TD: Accessing the Form with @ViewChild
TD: Adding Validation to check User Input
TD: Using the Form State
TD: Outputting Validation Error Messages
TD: Set Default Values with ngModel Property Binding
TD: Using ngModel with Two-Way-Binding
TD: Grouping Form Controls
TD: Handling Radio Buttons
TD: Setting and Patching Form Values
TD: Using Form Data

TD: Resetting Forms
Introduction to the Reactive Approach
Reactive: Setup
Reactive: Creating a Form in Code
Reactive: Syncing HTML and Form
Reactive: Submitting the Form
Reactive: Adding Validation
Build forms with client-side validation
Reactive: Getting Access to Controls
Reactive: Grouping Controls
Reactive: Creating Custom Validators
Reactive: Using Error Codes
Reactive: Reacting to Status or Value Changes

Angular Material
What is Angular Material?
Various form elements and how to use them

Day 5:
Services & Dependency Injection
Why would you Need Services?
Creating a Service
Injecting the Service into Components
Creating a Data Service
Using Services for Cross-Component Communication
Setting up the Services
Consuming HTTP Services
CRUD Operations using Service
Get/Add/Update/Delete Data
Handling Errors
Map operator
Observables vs Promises
Data Access in Angular App
Using Observables & RxJs Library
Inter-component Communication

–**Debugging Angular Project**
Understanding Angular Error Messages
Troubleshoot common runtime errors
Using Chrome Debugger for the Angular Project
Using Augury to Dive into Angular Apps

-**Using DevOps Tools with Angular Project**
Using version control GIT
Creating Repository on GitHub
Integrating Angular Project with GIT
Migrating Angular Project from Local Machine to GitHub
Introduction Using Docker and Jenkins with Angular Project

DEPLOYING AN ANGULAR APP
Building Angular Project
Deployment Preparations and Important Steps
Deploying to AWS Production Server

CASE STUDY/LIVE PROJECT
POST EVALUATION