

# Angular Essentials

## Course Overview

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



This is a 3-day class

## Who Should Attend

This is an introductory level Angular course, designed for experienced web developers who wish to extend their skills in web development. Attendees should have an extensive working knowledge in developing basic web applications.

## Course Objectives

- 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

## Course Outline

### 1 Introducing Angular

- What is Angular?
- Central Features of the Angular Framework
- Why Angular?
- Building Blocks of an Angular Application
- Basic Architecture of an Angular Application
- Installing and Using Angular
- A Basic Angular Application
- Anatomy of a Basic Application
- The Main Component File
- The Application Module File
- The index.html File
- The Bootstrap File
- Running the Application
- Building the Application
- Summary



NH Computer Learning

<https://www.nhcomputerlearning.com>

866-702-3301 - [info@nhclc.com](mailto:info@nhclc.com)

# Angular Essentials

## 2 Development Setup of Angular

- Angular Development Lifecycle
- Angular is Modular
- What is Node.js?
- Installing Node.js and NPM
- Node Package Manager (NPM)
- Package Descriptor File - package.json
- Semantic Version Numbering
- Package Version Numbering Syntax
- Updating Packages
- Uninstalling Packages
- Installing Angular Packages
- Angular CLI
- Creating a New Project
- Angular
  - Development
  - Dependencies
  - TypeScript Typings
  - Testing Tools
  - Development Web Server
  - Configuring the Web Server
  - The Build System
  - Configuring the Build
- Summary

## 3 Introduction to TypeScript and ES6

- Programming Languages for Use with Angular
- TypeScript Syntax
- Programming Editors
- The Type System – Defining Variables
- The Type System – Defining Arrays
- Type in Functions
- Type Inference
- The Type System – Classes & Objects
- Class Constructors
- Class Constructors – Alternate Form
- Interfaces
- Working with ES6 Modules
- Visibility Control
- var, let and const - defined
- var, let and const - usage
- Arrow Functions
- Arrow Function Compact Syntax
- Arrow Function and Caller Context
- Template Strings
- Template Strings – Variables and Expressions
- Template Strings – Multiline
- Generics - Class
- Generics - Methods
- Generics - Restricting Types
- Generics - Restricting Types: Example
- TypeScript Transpilation
- Summary

# Angular Essentials

## 4 Components in Angular

- Chapter 4. Components in Angular
- What is a Component?
- An Example Component
- The Component Class
- Adding a Component to Its Module
- Creating a Component Using Angular CLI
- Developing a Simple Login Component
- Component Template
- Login Component: Add HTML
- The HTML Component Template
- The templateUrl property
- Login Component: Add CSS Styling
- Login Component: Hook Up Input Fields and Button
- Login Component: Fields & Button in the Component Class
- Component Decorator Properties
- Component Hierarchy
- The Application Root Component
- Using a Regular Component
- The Build System
- Component Lifecycle Hooks
- Using a Lifecycle Hook:
- OnInit
- Summary

## 5 Data and Event Binding

- Binding Syntax
- One-Way Output Binding
- Binding Displayed Output Values
- Setting Component Properties
- More About Setting Properties
- Setting DOM Element Properties
- Event Binding
- Binding Events Examples
- Firing Event from a Component
- @Output() Example - Child Component
- @Output() Example - Parent Component
- Two-Way Binding of Input Fields
- Input Binding Examples
- Two Way Binding in a Component
- Use Two Way Binding
- Breaking Down ngModel
- Summary

# Angular Essentials

## 6 Attribute Directives

- What are Attribute Directives
- Apply Styles by Changing CSS Classes
- Changing Classes – Example
- Applying Styles Directly
- Applying Styles Directly - Example
- Controlling Element Visibility
- Setting Image Source Dynamically
- Setting Hyperlink Source Dynamically
- Writing a Custom Attribute Directive
- Using a Custom Attribute Directive
- Supplying Input to a Directive
- Handling Event from a Custom Directive
- Summary

## 7 Structural Directives

- Structural Directives
- Adding and Removing Elements Dynamically
- If-Else Syntax of ngIf
- Looping Using ngFor
- ngFor - Basic Example
- Creating Tables with ngFor
- ngFor Local Variables
- Manipulating the Collection
- Example - Deleting an Item
- Swapping Elements with ngSwitch
- ngSwitch - Basic Syntax
- Summary

# Angular Essentials

## 8 Template Driven Forms

- Template Driven Forms
- Importing Forms Module
- A Basic Angular Form
- Binding Input Fields
- Accessing the NgForm Directive
- Binding the Form Submit Event
- The Submit Function
- Basic HTML5 Validation - "required" Attribute
- HTML5 vs. Angular Validation
- Angular Validators
- The NgModel Directive
- Controlling when validation is applied
- Displaying Form
- Validation State
- Displaying Field
- Validation State
- Displaying Validation State Using Classes
- Disabling Submit when Form is Invalid
- Submitting the Form
- Binding to Object Variables
- Binding to Object Variables - Code
- Additional Input Types
- Checkboxes
- Select (Drop Down) Fields
- Rendering Options for Select (Drop Down)
- Date fields
- Radio Buttons
- Summary

## 9 Reactive Forms

- Reactive Forms Overview
- The Building Blocks
- Import ReactiveFormsModule
- Construct a Form
- Design the Template
- FormControl Constructor
- Getting Form Values
- Setting Form Values
- The Synchronous Nature
- Subscribing to Input Changes
- Validation
- Built-In Validators
- Showing Validation Error
- Custom Validator
- Using a Custom Validator
- Sub FormGroup - Component Class
- Sub FormGroup - HTML Template
- Why Use Sub FormGroup
- Summary

# Angular Essentials

## 10 Angular Modules

- Why Angular Modules?
- But, We Already Had ES6 Module
- Angular Built-in Modules
- The Root Module
- Feature Modules
- Create Feature Module Using CLI
- The Module Class
- @NgModule Properties
- Using One Module From Another
- Importing BrowserModule or CommonModule
- Lazy-Loaded Modules
- How to Organize Modules?
- Third Party Modules
- Summary

## 11 Services and Dependency Injection

- What is a Service?
- Creating a Basic Service
- What is Dependency Injection?
- What Dependency Injection Looks Like
- Injectors
- Injector Hierarchy
- Register a Service with a Module Injector
- Registering a Service with the Root Injector
- Registering a Service with a Component's Injector
- Where to Register a Service?
- Dependency Injection in Other Artifacts
- Providing an Alternate Implementation
- Dependency Injection and @Host
- Dependency Injection and @Optional
- Summary

## 12 HTTP Client

- The Angular HTTP Client
- Using The HTTP Client - Overview
- Importing HttpClientModule
- Simple Example
- Service Using HttpClient
- ES6 Import Statements
- Making a GET Request
- What does an Observable Object do?
- Using the Service in a Component
- The PeopleService Client Component
- Error Handling
- Making a POST Request
- Making a PUT Request
- Making a DELETE Request
- Summary

# Angular Essentials

## 13 Pipes and Data Formatting

- What are Pipes?
- Built-In Pipes
- Using Pipes in HTML Template
- Chaining Pipes
- Using Pipes in Code
- Internationalized Pipes (i18n)
- Loading Locale Data
- Decimal Pipe
- Currency Pipe
- Custom Pipes
- Custom Pipe Example
- Using Custom Pipes
- Using a Pipe with ngFor
- A Filter Pipe
- A Sort Pipe
- Pipe Category: Pure and Impure
- Pure Pipe Example
- Impure Pipe Example
- Summary

## 14 Introduction to Single Page Applications

- What is a Single Page Application (SPA)
- Traditional Web Application
- SPA Workflow
- Single Page Application Advantages
- HTML5 History API
- SPA Challenges
- Implementing SPA's Using Angular
- Summary

## 15 The Angular Component Router

- The Component Router
- View Navigation
- The Angular Router API
- Creating a Router Enabled Application
- Hosting the Routed Component
- Navigation Using Links and Buttons
- Programmatic Navigation
- Passing Route Parameters
- Navigating with Route Parameters
- Obtaining the Route Parameter Values
- Retrieving the Route Parameter Synchronously
- Retrieving a Route Parameter Asynchronously
- Query Parameters
- Supplying Query Parameters
- Retrieving Query Parameters Asynchronously
- Problems with Manual URL entry and Bookmarking
- Summary

# Angular Essentials

## 16 Lab Exercises

- Lab 1. Introduction to Angular
- Lab 2. Introduction to TypeScript
- Lab 3. Introduction to Components
- Lab 4. Create the Subscription Form Component
- Lab 5. Understanding Data Binding
- Lab 6. One Way Data Binding in a Custom Component
- Lab 7. Using Basic Angular Directives
- Lab 8. Using Structural Directives
- Lab 9. Custom Attribute Directive
- Lab 10. Template Driven Form
- Lab 11. Validation of a Template Driven Form
- Lab 12. Reactive Forms
- Lab 13. Service and Dependency Injection
- Lab 14. HTTP Client Development
- Lab 15. Using Pipes
- Lab 16. Basic Single Page Application Using Router