**ANGULAR COURSE OUTLINE**

***Duration :*** 03 days

***Prerequisites :*** experienced JavaScript developers with a fairly advanced understanding of JavaScript, including prototypes and functions as first class citizens.

**Software/Hardware :**

* + Windows 7 or later
  + Node Installer
  + 4GB RAM
  + Latest Browser : Chrome, IE, Firefox
  + Any preferred IDE (Sublime, VS Code, Notepad++ etc.)
  + Internet Connection
  + Permission to download and install libraries on participants’ machine

***Course Outline :***

***Day 01***

1 - Introduction to Angular

* What is Angular?
* Central Features of the Angular Framework
* Why Angular?
* Scope and Goal of Angular
* Angular vs. AngularJS
* Installing and Using Angular
* Adding Angular and Dependencies to Your App
* Building Blocks of and Angular Application
* A Basic Angular Application

2 - Introduction to TypeScript and ES6

* Programming Languages for Use with Angular
* TypeScript Syntax
* The Type System
* Destructuring – Objects & Arrays
* Class Constructors
* Interfaces
* Parameter and Return Value Types
* Working with Modules
* TypeScript Transpilation
* Arrow Functions
* Template Strings

3 - Components in Angular

* What is a Component?
* An Example Component
* Component Starter
* Developing a Simple Login Component
* Component Decorator Properties
* Component Lifecycle Hooks
* Using a Lifecycle Hook: OnInit

4 - Data and Event Binding

* Binding Syntax
* One-Way Output Binding
* Binding Displayed Output Values
* Two-Way Binding of Input Fields
* Input Binding Examples
* Binding Events
* Setting Element Properties

5 - Structural Directives

* Structural Directives
* Adding and Removing Elements Dynamically
* Creating Tables/Lists with ngFor
* ngSwitch & ngIf/else

**Day 02 :**

6 - Attribute Directives and Property Bindings

* What are Directives
* Directive Types
* Apply Styles by Changing Classes
* Changing Classes - Example
* Applying Styles Directly
* Directives and Property Binding
* Controlling Element Visibility
* Setting Image Source Dynamically
* Setting Hyperlink Source Dynamically

7 - Template Driven Forms

* Template Driven Forms
* A Basic Angular Form
* Binding Input Fields
* Accessing the Form Object
* Binding the Form Submit Event
* Basic HTML5 Validation - "required" Attribute
* HTML5 vs. Angular Validation
* Angular Validations
* 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

8 - Pipes and Data Formatting

* What are Pipes?
* Formatting Changes in Angular
* Using a Built-in Pipe
* Using Pipes in HTML
* Chaining Pipes
* Decimal Pipe
* CurrencyPipe
* Custom Pipes
* Pure / Impure Pipes

9 - Service and Dependency Injection

* What is a Service?
* Creating a Basic Service
* What is Dependency Injection?
* Injecting Services
* Using a Service in a Component: Dedicated Instance
* Using onInit to Initialize Component Data
* Using a Shared Service Instance
* Lazy Loading of Module
* Multi-modules
* Exporting Modules
* Exporting Services

**Day 03 :**

10 - HTTP Client

* The Angular HTTP Client
* Using The HTTP Client - Overview
* Setting up the Root Component
* Service Using Http Client
* Importing Individual HTTP Providers into Services
* Service Imports
* The Observable object type
* Making a Basic HTTP GET Call
* Using the Service in a Component
* Importing Observable Methods
* Enhancing the Service with .map() and .catch()
* GET Request

11 - Introduction to Single Page Application

* What is a Single Page Application (SPA)
* SPA Workflow
* Traditional Web Application Capabilities
* Single Page Application Advantages
* Implementing SPA's Using Angular
* Simple SPA Using Visibility Control
* SPA Using Angular

12 - The Angular Component Router

* Routing and Navigation
* The Component Router
* Traditional Browser Navigation
* Component Router Terminology
* Setting up the Component Router
* Browser pushState and Routes
* Bootstrapping Routing
* App Routes
* Programmatic Navigation
* Creating Routes with Route Parameters
* Navigating with Route Parameters
* Using Route Parameter Values
* Retrieving the Route Parameter
* Query Parameters
* Guards at Gate / Authentication