



# ANGULAR

Nature of the Course: Theory + Practical

Total Hours per Day: 2 Hours

Course Duration: 3 weeks

## Course Summary

The Angular course at the Deerwalk Training Center (DTC) is designed for novices who wish to learn how to think about and produce meaningful Angular, as well as read Java Script codes written by others. The course also teaches how to translate a literary description of a problem (requirement) into a Java Script application or library. This is a foundational course for anyone with no prior programming expertise who wants to work as a professional Angular engineer in the future.

## Completion Criteria

After fulfilling all of the following criteria, the student will be deemed to have finished the Module:

- Has attended 90% of all classes held
- Has received an average grade of 80% on all assignments
- Has received an average of 60% in assessments
- The tutor believes the student has grasped all of the concepts and is ready to go on to the second module.

## Required Textbooks

- Node.js complete reference
- Node.js for .NET developers

## Prerequisites

- Basic knowledge about programming, bits/bytes, procedures, classes, computer architecture, etc. If you just have theoretical knowledge that is perfectly okay but you should have strong convictions on what programming is, and what you hope to achieve from this class.
- Willing and eager to spend at least 10-20 hours (varying from

student-to-student) per week outside of the training class to read/write codes in Angular (self-study and practice).

- There is no prior educational level requirement for this course.

## **Course Details**

### **Week I**

#### **Introduction To Angular**

- Introduction
- Component Based Model
- Setting Up
- Creating and running a project

#### **Components**

- Introducing the app component
- Creating your first component
- Anatomy of a component
- Binding data from component class
- Data binding and async
- Template Interpolation

### **Week II**

#### **Templates And Data Binding**

- Looping with ngFor
- Using ngIf
- Passing inputs to components
- Using ngOnInit Lifecycle hook
- Passing member variables to components
- Styling Angular components
- Handling click events
- Two-way data binding with ngModel

#### **Services**

- Creating and using multiple modules
- Creating a service
- Understanding dependency injection
- Service Injection Context
- Making REST calls with HttpClient

- Extending the sample app
- Building an Angular Project
- [Class Project: My GitHub Project Listing Web App]

## **Week III**

### **Routing**

- Introduction to Routing
- Creating an angular project with routing
- Creating routes
- Route redirects and wildcards
- Configuring child routes
- Showing navigation for routes
- Wrap Up
- [Class Project: CRUD Application using Angular and Firebase]

### **Labs**

Lab assignments will focus on the practice and mastery of contents covered in the lectures, and introduce critical and fundamental problem-solving techniques to the students.

### **Learning Outcomes**

- Create device-independent Angular applications
- Creating and validating Angular forms
- Adding routes to Angular applications
- Unit Testing Angular components