

**Introduction to Angular**

**Course Number:** ANG-170  
**Duration:** 3 days

**Overview**

[Angular](https://angular.io/) allows developers to easily build dynamic, responsive single-page web applications that dynamically rewrite portions of the current page rather than having to generate a new page in response to every request. Accelebrate's Introduction to Angular training teaches attendees how to build applications using ES6, TypeScript, and modern front-end tools, including npm and Webpack.

**Note:** This course is taught in the current version of Angular at the time of teaching.

**Prerequisites**

Students must have object-oriented programming experience. Some experience with JavaScript is helpful; the new language features of JavaScript and TypeScript are covered/reviewed in class.

**Materials**

All Angular training attendees receive comprehensive courseware.

**Software Needed on Each Student PC**

* Google Chrome
* Other modern browsers as desired
* IDE/development environment of your choice
* Other free software and lab files that Accelebrate would specify

**Objectives**

* Understand how single-page web application architectures are different than traditional web application architectures
* Use new JavaScript (ES6) language features including Classes, Modules, and Arrow Functions
* Use new TypeScript language features including Types, Decorators, Interfaces, and Generics
* Learn Angular coding and architecture best practices including project layout and using container and presentation components
* Understand and use Angular model-driven forms, observables, dependency injection, and routing
* Communicate with a backend server using Angular’s HttpClient to load and save data
* Configure the router and navigate between components

**Outline**

* Introduction
* TypeScript and ECMAScript 6 (ES6) Fundamentals
  + TypeScript Installation, Configuration & Compilation
  + Type Annotations
  + Classes
  + Scoping using let, var, and const Keywords
  + Arrow Functions
  + ES Modules
  + Decorators
  + Template Literals
  + Spread Syntax and Rest Parameters
  + Destructuring
* Angular Overview
  + Benefits of Building using Angular
  + Understanding Angular Versions
  + Single-page Web Application Architectures vs. Traditional Server-side Web Application Architectures
  + Angular Style Guide
  + Angular Architecture
  + Angular Compared to Other JavaScript Libraries and Frameworks (React, VueJS, etc…)
  + Your First Angular Application
* Components
  + Understanding Components
  + Component Properties & Methods
  + Templates: Inline, Multi-line, and External with Component-relative Paths
* Angular Modules (NgModule)
  + Angular Modules vs. ES Modules
  + Organizing your code into Feature Modules
* Project Set-Up (Using the Angular CLI)
  + Angular CLI Features
  + Creating a New Project (with new CLI Prompts)
  + Generating Code
  + Customizing the Angular CLI
* Data Binding
  + Interpolation
  + Property binding
  + Event binding
  + Two-way data binding
* Directives
  + Structural: ngFor, ngIf, ngSwitch
  + Attribute: ngClass, ngStyle
* Pipes
  + Built-in Pipes: Using, Passing Parameters, Chaining
* Advanced Components
  + Component Communication using @Input, @Output
  + Component Architecture
  + Component Styles
  + Component Lifecycle Hooks
  + Evaluating UI Component Frameworks & Libraries
* Services & Dependency Injection
  + Using a service to access data
  + Using a service to encapsulate business logic
  + Understanding the scope of services
* Dependency Injection
  + Understanding Dependency Injection
  + Angular’s Dependency Injection System
  + Registering
  + Injecting
* Model-driven Forms (Reactive Forms)
  + Importing the ReactiveFormsModule
  + FormControl, FormGroup, and AbstractControl
  + Binding DOM Elements to FormGroups and FormControls
  + Validation Rules, Messages, and Styles
  + Refactoring Reactive Forms for Reuse
  + Custom Validators
* Communicating with the Server using the HttpClient Service
  + Deciding between Promises or Observables (RxJS)
  + Making an HTTP GET Request
  + Sending data to the server using Http POST and PUT Requests
  + Issuing an HTTP DELETE Request
  + Intercepting Requests and Responses
* Router
  + Importing the RouterModule
  + Configuring Routes
  + Displaying Components using a RouterOutlet
  + Navigating declaratively with RouterLink
  + Navigating with code using the Router
  + Accessing parameters using ActivatedRoute
* Deploying an Angular Application to Production
  + Building an application using the Angular CLI
  + Differential loading: creating a modern build (ES2015) and a legacy build (ES5)
  + Deploying to a web server
* Upgrading to the latest version of Angular from earlier versions
  + 2.x and above
  + Update Guide
  + Deprecation Guide
  + Patching Dependencies: npm audit fix
  + Looking for AngularJS to Angular upgrades? See our
* Roadmap to the future Angular
  + Getting Started with Standalone Components
  + Understanding Signals
  + Built-in control flow
  + Angular Beta Documentation (Angular.dev)
* Conclusion