



☎ +44 7989 401397

✉ info@olsensoft.com

## Angular 2 Development

(4 days)

### Course overview

This course takes a detailed look at Angular 2. We show how to create Rich Internet Applications incorporating powerful Angular features such as data binding, service dependency injection, routing, and REST / Web Sockets integration. Note that we also have a course that covers Angular 4, which is the latest version of the Angular library.

### What you'll learn

- Angular architectural concepts
- Creating and testing Angular apps
- Templates
- Displaying data and accepting user input
- Dependency injection
- Pipes
- Directives
- Routing and navigation
- Lifecycle hooks
- Calling REST services
- Web sockets
- Angular CLI

### Prerequisites

- Familiarity with HTML and JavaScript programming
- Experience in AngularJS 1.x is beneficial but not required

### Course details

- **Angular 2 Essentials:** Angular 2 vs Angular 1; Angular 2 architecture components; Development tools state-of-play
- **TypeScript and ECMAScript 6 Essentials:** Overview of TypeScript and ECMAScript 6; Transpilation; Using the TypeScript Playground; Configuring and using the TypeScript transpiler
- **Understanding TypeScript Syntax:** Variables and types; Functions; Classes; Generics; Inheritance; Interfaces
- **Creating a Simple Angular 2 Application:** Structure of an Angular 2 application; Importing Angular libraries; Angular bundles; Angular components; Templates; Binding expressions; Bootstrapping Angular

- **Data Binding:** A closer look at data binding; Two-way data binding; Using built-in pipes; Defining custom pipes; Applying pipes programmatically; Built-in structural directives
- **Configuration:** A closer look at SystemJS; Loading multiple modules; Configuring SystemJS; Using a package manager
- **Tooling:** Overview of Node Package Manager, Bower, Grunt, and Gulp; Using Bower to manage client packages; Defining static content; Gulp tasks
- **Component Hierarchies:** Structuring applications; Defining component inputs; Defining component outputs; The mediator pattern
- **Dependency Injection:** Overview of dependency injection in Angular; Simple dependency injection; Using the provide() function; A hierarchy of injectors; Global provision; Injection into services; Values and factories
- **Angular 2 Routing:** Getting started with routing; The role of routing in Single Page Applications; Route parameters; Lazy loading
- **Calling Rest Web Services:** Overview of Rest services; Calling Rest services from Angular 2; Using the conditional navigation operator
- **Web Sockets:** Web Sockets concepts; Calling Web Sockets from JavaScript; Using RxJs Observables with Web Sockets
- **Testing Angular Code:** Setting the scene for unit testing; Getting started with Jasmine; Using Jasmine to test Angular code
- **Implementing Custom Directives:** Attribute directives; Specifying parameters and events; Structural directives; Transclusion
- **Angular CLI:** Overview of Angular CLI; Creating a template application; Generating components, services, and other artefacts; Bundling and tree-shaking