Intermediate Angular Programming (2 Days)

OVERVIEW

This Intermediate Angular Programming training covers more advanced topics of this next generation of the Angular framework. This Angular training course is intended for those that already have a basic understanding of the Angular fundamentals and now want to cover more advanced features the Angular framework offers.

COURSE BENEFITS

- Learn to create Advanced HTTP clients.
- Learn to consume WebSocket data.
- Learn to use the featured of Advanced Routing.
- Learn to test and debug Angular applications.

PREREQUISITE COURSE

Courses that can help you meet these prerequisites:

- Introduction to Angular Programming Training

COURSE OUTLINE

Advanced HTTP Client

Request Options
Returning an HttpResponse Object
Setting Request Headers
Creating New Observables
Creating a Simple Observable
The Observable Constructor Method
Observable Operators
The map and filter Operators
The flatMap() Operator
The tap() Operator
The zip() Combinator
Caching HTTP Response
Making Sequential HTTP Calls
Making Parallel Calls
Customizing Error Object with catchError()
Error in Pipeline
Error Recovery

Angular Modules

Anatomy of a Module Class
@NgModule Properties
Feature Modules
Example Module Structure
Create a Domain Module
Create a Routed/Routing Module Pair
Create a Service Module
Creating Common Modules
Using One Module From Another

Advanced Routing

- Routing Enabled Feature Module
- Using the Feature Module
- Lazy Loading the Feature Module
- Creating Links for the Feature Module Components
- More About Lazy Loading
- Preloading Modules
- routerLinkActive binding
- Default Route
- Wildcard Route Path
- redirectTo
- Child Routes
- Defining Child Routes
- for Child Routes
- Links for Child Routes
- Navigation Guards
- Creating Guard Implementations
- Using Guards in a Route

Unit Testing Angular Applications

- Unit Testing Angular Artifacts
- Testing Tools
- Typical Testing Steps
- Test Results
- Jasmine Test Suites
- Jasmine Specs (Unit Tests)
- Expectations (Assertions)
- Matchers
- Examples of Using Matchers
- Using the not Property
- Setup and Teardown in Unit Test Suites
- Example of beforeEach and afterEach Functions
- Angular Test Module
- Example Angular Test Module
- Testing a Service
- Injecting a Service Instance
- Test a Synchronous Method
- Test an Asynchronous Method
- Using Mock HTTP Client
- Supplying Canned Response
- Testing a Component
- Component Test Module
- Creating a Component Instance
- The ComponentFixture Class
- Basic Component Tests
- The DebugElement Class
- Simulating User Interaction

Debugging

- Overview of Angular Debugging
- Viewing TypeScript Code in Debugger
- Using the debugger Keyword
- Debug Logging
- What is Angular DevTools?
- Using Angular DevTools
- Angular DevTools - Component Structure
- Angular DevTools - Change Detection Execution
- Catching Syntax Errors