

AngularJS Training

[Request an offer for classroom delivery of this course](#)

[Online Registration and Course delivery](#)

Course duration

- 3 Days

Course Outline

The AngularJS framework has become a popular JavaScript framework for the development of "single page" Rich Internet Applications. The AngularJS framework augments applications with the "model-view-controller" pattern which makes applications easier to develop and test because there is a separation of responsibilities within the code. Although there are many benefits to using the AngularJS framework it is fairly different than the way "typical" web applications involving JavaScript have been designed. This AngularJS training class will provide an introduction to the benefits of AngularJS, so participants can start to develop responsive applications quickly using the framework.

- Create single page web applications using the MVC pattern of AngularJS.
- Understand the programming model provided by the AngularJS framework.
- Define Angular controllers and directives.
- Control Angular data bindings.
- Implement Responsive Web Applications with AngularJS.

- Some prior understanding of web development, HTML, Ajax, and JavaScript.

1. Advanced Objects and Functionality in JavaScript

1. Basic Objects
2. Constructor Function
3. More on the Constructor Function
4. Object Properties
5. Deleting a Property
6. The instanceof Operator
7. Object Properties
8. Constructor and Instance Objects
9. Constructor Level Properties

10. Namespace
 11. Functions Are First-Class Objects
 12. Closures
 13. Closure Examples
 14. Private Variables with Closures
 15. Immediately Invoked Function Expression (IIFE)
 16. Prototype
 17. Inheritance in JavaScript
 18. The Prototype Chain
 19. Traversing Prototype Property Hierarchy
 20. Prototype Chain
 21. Inheritance Using Prototype
 22. Extending Inherited Behavior
 23. Enhancing Constructors
 24. Improving Constructor Performance
 25. Inheritance with Object.create
 26. The hasOwnProperty Method
2. Introduction to AngularJS
 1. What Is AngularJS?
 2. Why AngularJS?
 3. Scope and Goal of AngularJS
 4. Using AngularJS
 5. A Very Simple AngularJS Application
 6. Building Blocks of an AngularJS Application
 7. Use of Model View Controller (MVC) Pattern
 8. A Simple MVC Application
 9. The View
 10. The Controller
 11. Data Binding
 12. Basics of Dependency Injection (DI)
 13. Other Client Side MVC Frameworks
3. AngularJS Module
 1. What Is a Module?
 2. Benefits of Having Modules
 3. Life Cycle of a Module
 4. The Configuration Phase
 5. The Run Phase
 6. Module Wide Data Using Value
 7. Module Wide Data Using Constant
 8. Module Dependency
 9. Using Multiple Modules in a Page
4. AngularJS Controllers
 1. Controller Main Responsibilities
 2. About Constructor and Factory Functions
 3. Defining a Controller
 4. Using the Controller
 5. Controller Constructor Function

6. More About Scope
7. Example Scope Hierarchy
8. Using Scope Hierarchy
9. Modifying Objects in Parent Scope
10. Modified Parent Scope in DOM
11. Handling Events
12. Another Example for Event Handling
13. Storing Model in Instance Property
5. AngularJS Expressions
 1. Expressions
 2. Operations Supported in Expressions
 3. AngularJS Expressions vs JavaScript Expressions
 4. AngularJS Expressions Are Safe to Use!
 5. What Is Missing in Expressions
 6. Considerations for Using src and href Attributes in Angular
 7. Examples of ng-src and ng-href Directives
6. Basic View Directives
 1. Introduction to AngularJS Directives
 2. Controlling Element Visibility
 3. Adding and Removing an Element
 4. Dynamically Changing Style Class
 5. The ng-class Directive
 6. Example Use of ng-class
 7. Setting Image Source
 8. Setting Hyperlink Dynamically
 9. Preventing Initial Flash
7. Advanced View Directives
 1. The ng-repeat Directive
 2. Example Use of ng-repeat
 3. Dynamically Adding Items
 4. Special Properties
 5. Example: Using the \$index Property
 6. Scope and Iteration
 7. Event Handling in Iterated Elements
 8. The ng-switch Directive
 9. Example Use of ng-switch
 10. Inserting External Template using ng-include
8. Working with Forms
 1. Forms and AngularJS
 2. Scope and Data Binding
 3. Role of a Form
 4. Using Input Text Box
 5. Using Radio Buttons
 6. Using Checkbox
 7. Using Checkbox - Advanced
 8. Using Select
 9. Using Select – Advanced

10. Disabling an Input
11. Reacting to Model Changes in a Declarative Way
12. Example of Using the ng-change Directive
9. Formatting Data with Filters in AngularJS
 1. What Are AngularJS Filters?
 2. The Filter Syntax
 3. Angular Filters
 4. More Angular Filters
 5. Using Filters in JavaScript
 6. Using Filters
 7. A More Complex Example
 8. The date Filter
 9. The date's format Parameter
 10. Examples of Using the date Filter
 11. The limitTo Filter
 12. Using limitTo Filter
 13. Filter Performance Considerations
10. AngularJS \$watch Scope Function
 1. The \$watch Function
 2. The \$watch Function Signature
 3. The \$watch Function Details
 4. Canceling the Watch Action
 5. Example of Using \$watch
 6. Things to be Aware Of
 7. More Things to Be Aware Of
 8. Performance Considerations
 9. Speeding Things Up
11. Communicating with Web Servers
 1. The \$http AngularJS Service
 2. The Promise Interface
 3. The \$http Service
 4. \$http Function Invocation
 5. Callback Parameters
 6. Request Configuration Properties
 7. Shortcut Methods
 8. Complete List of Shortcut Methods
 9. Using \$http.get()
 10. Working with JSON Response
 11. Using \$http.post()
 12. Combining \$http Post Request Data with Url Parameters
 13. The then() Method of the Promise Object
 14. The Response Object
 15. Making Parallel Web Service Calls
 16. Combining Multiple Promises into One
 17. Wait for the Combined Promise
 18. Setting Up HTTP Request Headers
 19. Caching Responses

20. Disabling Caching in IE9
21. Setting the Request Timeout
22. Unit Testing with ngMock
23. Writing Unit Tests
12. Custom Directives
 1. What Are Directives?
 2. Directive Usage Types
 3. Directive Naming Convention
 4. Defining a Custom Directive
 5. Using the Directive
 6. Scope of a Directive
 7. Isolating Scope
 8. Example Scope Isolation
 9. Using External Template File
 10. Manipulating a DOM Element
 11. The Link Function
 12. Event Handling from a Link Function
 13. Wrapping Other Elements
 14. Accepting a Callback Function
 15. Supplying Callback Function
13. AngularJS Services
 1. Introduction to Services
 2. Defining a Service
 3. The factory() Method Approach
 4. The service() Method Approach
 5. The provider() Method Approach
 6. Using a Service
 7. Configuring a Service using its Provider
14. Introduction to Single Page Applications (spa)
 1. What Is a Single Page Application (SPA)?
 2. Why Create SPA?
 3. Challenges to SPA
 4. Implementing SPA Using AngularJS
 5. Simple SPA Using Visibility Control
 6. Dynamic Templates Using ng-include
 7. SPA Using the \$route service
15. The Routing Service
 1. What does the \$route Service Do?
 2. When to use \$route Service?
 3. Understanding the \$location Service
 4. Setting up a Route Table
 5. Navigation Control
 6. Getting Route Parameters
 7. Bookmarking and history
 8. Adding Dependency to Routes
16. Advanced Forms
 1. Basic form validation

2. Validation based on input types
 3. Advanced form validation
 4. Monitoring validation status
 5. Styling elements based on validation status
 6. Showing specific error messages
 7. Deferring validation error messages
17. Advanced Promise Api
 1. Basics of the promise API
 2. \$http and Promise
 3. Waiting for multiple \$http Calls
 4. Caching a \$http Call in memory
 5. Caching a \$http Call in localStorage
18. Unit Test Using Jasmine
 1. Introduction to Node.js
 2. What Is Jasmine?
 3. Running Jasmine
 4. Jasmine Folder Structure
 5. Running Tests
 6. Example Explained
 7. End-to-End Testing with Protractor
 8. Writing E2E Test with Protractor

InterSource Geneva, a premier Information Technology training firm, offers over 400 different courses on server, database and programming technologies, as well as end-user classes for the most popular office, graphics and design applications. We serve clients in Switzerland (Geneva, Lausanne, Bern, Basel, Zurich) and throughout Europe (France, Germany, Austria, Finland, Sweden, England, Netherlands, Spain, etc.).

InterSource offers custom, private courses at client sites, standard public courses in our Geneva classroom, and online training via live Web conference. Training is offered in English and many other languages (Francais, Deutsch, Espanol, Italiano.)

For an overall view of our offerings, please visit us at www.intersource.ch.