Developing Web Applications with JavaScript, HTML5, and CSS LVC

JAVA

DURATION	MODULES	COURSE CODE
4 Days	**9 Lectures**	**—**

Course Overview

Develop front-end web application using JavaScript, HTML5 and CSS. Course starts with the coverage of web application fundamental concepts, JavaScript language syntax, construction user interfaces using HTML5 and CSS, building application logic using JavaScript, and eventually progresses to advanced topics such as using JavaScript to interact with server-side components, such as WebServices and WebSockets.

What You Will Learn

- I Developing Web Applications with JavaScript, HTML5, and CSS
- Course Goals I-2
- Audience I-3
- Course Outline I-4
- Course Practices I-5

Introduction to Web Applications

- Objectives
- Introduction to Web Application
- HTTP Requests Methods
- HTTP GET and HEAD Methods
- HTTP POST Method
- HTTP PUT and PATCH Methods
- HTTP DELETE Method
- HTTP Responses and Status Codes
- HTML Document Object Model (DOM)
- HTML Document Page Layout
- HTML Tables
- HTML Lists
- Add Style to HTML Elements
- Add JavaScript to HTML Document
- Associating Style Sheet with HTML Document

- Associating JavaScript with HTML Document Page
- HTML Forms
- Implementing Navigation in an HTML Document
- Summary
- Practice Overview

Introduction to JavaScript

- Objectives
- Agenda
- Introduction to JavaScript
- JavaScript Origins
- JavaScript Clones and Standards
- JavaScript Standards Split
- JavaScript Variants Proliferation
- JavaScript New Engine and Server-Side Expansion
- JavaScript Modern Evolution
- Agenda
- JavaScript Syntax
- JavaScript Variables
- JavaScript Data Types
- JavaScript Types Coercion
- Explicit Value Conversions
- JavaScript Objects and Functions
- Agenda
- JavaScript Operators
- Manipulate Numeric Values Using Arithmetic Operators
- Extract Portions of Text from a String
- Concatenate Strings
- Manipulate Text Values
- Compare Values
- Manipulate Date Values
- Agenda
- Manipulate Arrays
- Complex Arrays and Collections
- Array Object
- Agenda
- Implement Flow Control with if /else
- Ternary Operations
- Implement Flow Control with Switch
- Implement Iterations with Loops
- For Loop Variants
- Advanced Loops
- Summary
- Practice Overview

Functions, Classes, and Objects

- Objectives
- Classification of Functions, Classes, and Objects
- Define and Invoke a Function
- Arrow Functions
- Global Object
- this keyword
- Define and Access an Object
- Converting Objects to String
- Define and Use Accessors
- Define and Instantiate a Class
- Constructor Functions
- Define “Private" Properties
- Define and Access Static Context
- Dynamically Alter an Object
- Interrogate Object and Its Properties
- Retrieve and Populate Object Properties
- Summary
- Practice Overview

Extending Classes and Objects

- Objectives
- JavaScript Extensibility Mechanisms
- What Is a Prototype?
- Create Objects Based on a Specific Prototype
- Function Object
- Extending an Object vs Extending a Prototype
- Implement Inheritance with Prototypes
- Implement Inheritance with Classes
- Spreading Arrays and Objects
- Summary
- Practice Overview

JavaScript in a Web Browser Environment

- Objectives
- Retrieve DOM Elements
- Traverse Document Nodes
- Interact with Elements and their Attributes
- Interact with Selected Lists
- Interact with Check Boxes
- Interact with Radio Buttons
- Event Handling Overview
- Events and Listeners
- Dispatching Events

- Subscribing to Events
- Prevent Default Action
- React to User Input
- Validate User Input
- Assist User Input with Value Selectors
- Display and Input a Value within a Range
- Produce Image Map
- Other Input Types
- Button, Image, and Form Actions
- Write to HTML Document
- Add, Remove, and Modify HTML Document Content
- Summary
- Practice Overview

Cascading Style Sheets

- Objectives
- Defining Styles
- Element Selectors
- Attribute Selectors
- Use Pseudo-Class and Pseudo-Element Selectors
- Selecting Elements Based on Their State
- Selecting Elements Based on Their Properties
- Selecting Child Elements within a Parent
- Other Selectors
- Visual Properties
- Setting Sizes
- Setting Colors
- Selecting Fonts
- Positioning, Sizing and Orientation using Box Model
- Box Model Example
- Component Orientation
- Handling Overflow
- Control Display Layout
- Grid Display
- Flex Flow and Alignments
- Flex Display
- Altering Styles Based on Required Media
- Media Queries
- Dynamic Style Changes
- Summary
- Practice Overview

Advanced JavaScript

- Objectives
- Exception Handling

- Producing Exceptions
- Handling Exceptions
- Timeouts and Intervals
- Promises
- Execute an Array of Promises
- Async/Await
- Pass Functions as Arguments and Return Functions
- Modules
- Export Components
- Import Components
- Access Modules from HTML Document
- JavaScript Memory Organization and Code Execution
- JavaScript Concurrency Limitations
- Execute Concurrent Code with Workers
- Worker Life Cycle
- Worker Message Exchanges
- Summary
- Practice Overview

Exchanging Data with Servers

- Objectives
- What is JSON?
- REST Service Conventions and Resources
- Cross-Origin Resource Sharing (CORS)
- Dispatch Service Request Using AJAX
- Handle Service Response Using AJAX
- Dispatch Service Requests Using Modern JavaScript
- Handle Service Response Using Modern JavaScript
- Introduction to WebSockets
- WebSocket Life Cycle
- Send and Receive Messages Via the WebSocket
- Stateless Versus Stateful
- Manage Application State
- Retain State Using Parameters or Headers
- HTTP Cookies
- Set and Get HTTP Cookies in JavaScript
- Local and Session Web Storage
- Summary
- Practice 8

JavaScript Variants, APIs, Frameworks, and the Cloud Environment

- Objectives
- Agenda
- JavaScript Variants and Alternatives
- TypeScript Type Enforcement

- TypeScript Extra Types
- TypeScript Enumerations
- … And Many More TypeScript Features
- Agenda
- JavaScript Libraries and Frameworks
- Manage Library References
- Use jQuery Selectors and Functions
- Use jQueryUI Widgets
- Agenda
- Implement Server-Side JavaScript with Node.js
- Manage Node.js Projects
- Agenda
- JavaScript and Oracle Cloud Application Development
- Single-Page Application Design
- Visual Builder Studio Project Management
- Visual Builder Application Runtime
- Summary