



To discuss this course and customizations:  
Call: 434-509-5680 or Email: [sales@cloudcontraptions.com](mailto:sales@cloudcontraptions.com)

## Advanced JavaScript

### Duration

5 days

### Description

This comprehensive course is designed for computer programming professionals seeking to deepen their understanding of JavaScript. The course covers a wide range of topics including JavaScript basics, scopes, functions, objects, classes, asynchronous JavaScript, modules, error handling, immutable programming, testing, debugging, performance, and NPM. Each topic is designed to provide a deep dive into the subject matter, with a strong focus on practical application and hands-on experience. The course is taught live and can be attended either online or in-person. Whether you're looking to enhance your JavaScript skills or learn new ones, this course is perfect for you.

### Objectives

- Understand the basics of JavaScript including variables, types, functions, objects, and arrays.
- Gain a deep understanding of JavaScript scopes including global, function, block, and lexical scopes, and hoisting.
- Master JavaScript functions including function expressions, arrow functions, closures, higher-order functions, and recursion.
- Learn about JavaScript objects including object literals, prototypes, classes, inheritance, and mixins.
- Get hands-on experience with JavaScript classes including class syntax, static methods, getters and setters, inheritance, mixins, class properties, and binding & "this".
- Dive into asynchronous JavaScript including event loop, single-threaded model, browser performance tool, callbacks, promises, async/await, generators, and observables (RxJS).
- Understand JavaScript modules including CommonJS, ES modules, and dynamic imports.
- Learn about error handling, immutable programming, testing, debugging, performance, and NPM.

### Prerequisites

All students must have JavaScript programming experience. Knowledge of Node.js and Web Browser APIs is helpful, but not required.



To discuss this course and customizations:  
Call: 434-509-5680 or Email: [sales@cloudcontraptions.com](mailto:sales@cloudcontraptions.com)

## Training Materials

All students receive comprehensive courseware covering all topics in the course. Courseware is distributed via GitHub in the form of documentation and extensive code samples. Students practice the topics covered through challenging hands-on lab exercises.

## Software Requirements

Students will need a free, personal GitHub account to access the courseware. No local student setup is required, this course is delivered with GitHub Codespaces. The free allotment of Codespaces hours from GitHub is sufficient to complete the course.

## Outline

- Introduction
- Basic JavaScript Review (as needed)
  - Variables and Types
  - Functions
  - Objects
  - Arrays
- Scope
  - Global Scope
  - Function Scope
  - Block Scope
  - Lexical Scope
  - Hoisting
- Functions
  - Function Expressions
  - Arrow Functions
  - Closures
  - Higher-Order Functions
  - Recursion
- Objects
  - Object Literals
  - Prototypes
  - Classes
  - Inheritance
  - Mixins



To discuss this course and customizations:  
Call: 434-509-5680 or Email: [sales@cloudcontraptions.com](mailto:sales@cloudcontraptions.com)

- Classes
  - Class Syntax
  - Static Methods
  - Getters and Setters
  - Inheritance
  - Mixins
  - Class Properties
  - Binding & “this”
- Asynchronous JavaScript
  - Event Loop
  - Single-Threaded Model
  - Browser Performance Tool
  - Callbacks
  - Promises
  - Async/Await
  - Generators
  - Observables (RxJS)
- Modules
  - CommonJS
  - ES Modules
  - Dynamic Imports
- Error Handling
  - try/catch
  - Error Objects
  - Custom Errors
  - Error Handling Patterns
- Immutable Programming
  - Immutability
  - Pure Functions
  - Side Effects
  - Referential Transparency
- Testing
  - Unit Testing
  - Integration Testing
  - Mocking



To discuss this course and customizations:  
Call: 434-509-5680 or Email: [sales@cloudcontraptions.com](mailto:sales@cloudcontraptions.com)

- Test Runners
- Code Coverage
- Jest Framework
- Debugging
  - Console
  - Breakpoints
  - Source Maps
  - Debugging Tools
- Performance
  - Profiling
  - Memory Leaks
  - Optimization
  - Web Workers
  - Service Workers
- NPM
  - Package Management
  - Semantic Versioning
  - Publishing Packages
  - NPM Scripts
  - Package Lock
  - Multiple Package Workspaces
- Conclusion