

# Programming in Modern JavaScript

**Duration: 3 days (9 hours each day with working lunch)**

JavaScript is one of the most ubiquitous and powerful languages, yet it's one of the languages that's largely misunderstood and underutilized. In JavaScript we can create objects, but that's different from how its done in OO languages. We can use inheritance, but that's prototypal inheritance, which is far more powerful and superior to class based inheritance in a number of ways. While programmers have in general struggled to use JavaScript to the fullest extent, the language itself has evolved considerably. This course covers both the breadth and width of the essential parts of modern JavaScript. Learn about prototypal inheritance, arrow functions, concise fluent syntax to create classes and properties, metaprogramming, annotations and more.

The course has a good balance of interactive lectures and hands-on exercises. The attendees are expected to pair-up and work on the lab exercises. The instructor will assist the attendees as they work on the labs. The objective of the course is for the attendees to gain an in depth practical knowledge of the concepts so they can put them to immediate use on their real projects.

The course content will be customized to meet your teams' specific needs. Please review this detailed outline and suggest changes (additions, deletions, modifications) as you feel fit.

## Topics

### JavaScript Gotchas

- \* Optionals that really are not
- \* Dynamic and weak typing
- \* Semi-colon
- \* == vs ===
- \* Issues that arise from weak typing
- \* Array weirdness
- \* Exercises

### Variables, Functions, and Scopes

- \* variable scoping
- \* Using strict
- \* var, let, and const
- \* Ways to define functions
- \* Hoisting and issues
- \* Higher-Order Functions
- \* Lexical scoping
- \* call, apply, bind
- \* The problem of "this"
- \* Exercises

## Object and Classes

- \* Creating objects
- \* Creating classes
- \* Examining properties
- \* Ways to access properties
- \* Prototype
- \* Prototypal inheritance
- \* Properties and inheritance
- \* hasOwnProperty
- \* Exercises

## Asynchronous Programming

- \* Asynchrony
- \* Callbacks
- \* Issues with callbacks
- \* Promises
- \* Using Promises
- \* Creating Promises
- \* Exercises

## JavaScript Programming Idioms

- \* Creating Fluent Interfaces
- \* Cascade Method Pattern
- \* Execute Around Method Pattern
- \* Currying
- \* Memoization
- \* Functional Style
- \* Function Pipeline pattern
- \* Exercises

## Programming with Node

- \* Node.js
- \* Programming for the node
- \* Creating services
- \* Talking to services
- \* Exercises

## Automated Testing

- \* Nee for automated testing
- \* Tools and approaches
- \* Challenges with testing
- \* Creating automated tests
- \* Exercises

## ECMAScript 2015 and 2016 Features

- \* Variable Declarations
- \* Block scoping

- \* Looping
- \* Template Strings
- \* Enhanced Object literals
- \* Symbols
- \* Lazy Iterators
- \* Generators
- \* Destructuring
- \* New Collections
- \* Modules
- \* Exercises

### Arrow Functions

- \* Functions
- \* Concise syntax
- \* Lambda expressions
- \* Benefits of arrow functions
- \* Issues with arrow functions
- \* Change in semantics
- \* When to use arrow functions
- \* Exercises

### Fluent Class Syntax

- \* Creating classes
- \* Constructor
- \* Methods
- \* Static methods
- \* Properties, getters, and setters
- \* Inheritance
- \* super
- \* default pass through
- \* Exercises

### Metaprogramming in JavaScript

- \* Annotations and decorators
- \* Symbols and metaObjects
- \* Proxies
- \* Using Reflection
- \* Exercises

### JavaScript Code Quality

- \* Caring about code quality
- \* Design Principles
- \* Multiple tools to measure and improve quality of code
- \* Exercises

## About the Instructor

Dr. Venkat Subramaniam is an award-winning author, founder of Agile Developer, Inc., creator of agilelearner.com, and an instructional professor at the University of Houston.

He has trained and mentored thousands of software developers in the US, Canada, Europe, and Asia, and is a regularly-invited speaker at several international conferences. Venkat helps his clients effectively apply and succeed with sustainable agile practices on their software projects.

Venkat is a (co)author of multiple technical books, including the 2007 Jolt Productivity award winning book Practices of an Agile Developer. You can find a list of his books at agiledeveloper.com. You may read more about Venkat and Agile Developer, Inc. at <http://agiledeveloper.com>.

