Document Generated: 11/14/2026

Learning Style: Virtual Classroom

Technology: JavaScript

Difficulty: Beginner

Course Duration: 3 Days

Introduction to JavaScript (TT4110)



About this course:

Introduction to JavaScript / Modern JavaScript Essentials, is a three day hands-on course that equips you with essential JavaScript skills to build interactive, responsive web applications. Beginning with core programming concepts, you'll learn to write clean, efficient code using variables, operators, and flow control, gaining a strong understanding of how JavaScript structures dynamic functionality in modern web projects. Through 50% hands-on labs, you'll apply your knowledge in real coding scenarios, building confidence as you write, test, and refine JavaScript code.

As you advance, you'll sharpen your debugging skills, using browser tools to set breakpoints, manage performance, and handle exceptions effectively. This course empowers you to work with complex functions, closures, and arrays, showing you

how to manipulate data, control scope, and create more efficient, reusable code. You'll gain the ability to identify and fix issues on the fly, improve your program's efficiency, and use advanced JavaScript features to structure code that is both powerful and maintainable.

By the end of the course, you'll be equipped with practical skills in Object-Oriented JavaScript and DOM manipulation, enabling you to craft responsive interfaces that update seamlessly. You'll also dive into the latest ES6 enhancements, TypeScript for strong data typing, and secure coding practices, preparing you to build scalable applications with confidence. Additionally, you'll explore HTML5 JavaScript APIs, such as geolocation and canvas, to create interactive features that enhance user engagement. Whether you're a budding web developer or looking to refresh your skills, this course delivers the practical JavaScript expertise needed to succeed in today's digital landscape.

Course Objectives:

Working within in an engaging, hands-on learning environment, guided by our expert team, attendees will learn to:

- Master Core JavaScript Syntax and Flow Control - Develop proficiency in writing efficient JavaScript code by understanding variables, operators, loops, and conditional statements, setting a foundation for creating dynamic web functionality.
- Debug and Optimize JavaScript Code - Gain hands-on experience with browser debugging tools to identify, troubleshoot, and improve code performance, ensuring applications run smoothly and efficiently.
- Utilize Advanced Functions and Closures - Learn to create and work with complex functions, closures, and anonymous functions to enhance code reusability and manage scope effectively within larger applications.
- Build Interactive Web Pages Using DOM Manipulation - Acquire the skills to access, modify, and dynamically update the Document Object Model (DOM) to create responsive, interactive user interfaces.
- Implement ES6+ Features and TypeScript for Scalable Code - Apply modern JavaScript features, such as ES6 syntax, modules, and TypeScript, to write clean, maintainable, and scalable code suited to modern web development needs.

Audience:

- This course is designed for beginner to intermediate web developers, software engineers, and IT professionals who want to build a strong foundation in JavaScript for web development. Ideal for roles such as front-end developers, UX/UI designers, and junior full-stack developers, it's suited to anyone looking to enhance their skills in building dynamic and interactive

web applications.

Prerequisite:

- This is an introductory level course for experienced software developers seeking to enhance and extend their core web development skillset leveraging JavaScript. Attendees should have practical experience developing basic software applications. This course provides an excellent foundation for continued learning to gain in-demand skills in in-demand skills and technologies such as NodeJS, Angular, React, Redux and more. This course can also be tailored for less experienced or non-developers as needed. Please inquire for details.

Course Outline:

JavaScript Basics

- JavaScript Defined
- ECMAScript
- Variables and Operators
- Flow Control and Conditionals

Debugging Tools

- Using the strict mode and setting breakpoints
- Browser debugging tools
- Monitoring resource usage and performance
- Emulating devices
- Exceptions in JavaScript

JavaScript Functions

- Functions in JavaScript
- Anonymous Functions
- Invoking Functions
- Immediately Invoked Function Expression (IIFE)
- Function Constructor
- Function Scope and Closures

JavaScript Arrays, Math and Date

- JavaScript Arrays and Singleton Objects
- Array Methods
- Understanding References vs. Copies of Arrays
- JavaScript Literal Syntax for Data Structures
- Looping Through Data Structures
- Working with the Number, String, Math, and Date Built-in Objects

JavaScript Event Handling and the DOM

- Events and Event Handling
- HTML Document Object Model
- Accessing the DOM
- Dynamically Working with the DOM
- DOM Challenges

Object-Oriented JavaScript

- JavaScript “Objects” and “Classes”
- Constructors
- Extending Classes with Prototype
- Reusable, Flexible Classes
- ES6 Class Keyword
- ES6 Extending a Class
- ES6 Static Methods
- Getters and Setters

ES6: The Next Step

- ES6 as Syntactic Sugar
- Lexically Scoped Variables and Constants
- Default Parameters
- Rest Parameters
- Spread Operator
- Arrow Syntax for Functions
- Modules: Import and Export
- Destructuring
- ES6 Data Structures: Map, Set, WeakMap, WeakSet
- ES6 Array Methods
- ES6 Array Loops
- ES6 Template Strings
- ES6 Internationalization and Localization

TypeScript

- Strong Data Typing with TypeScript
- Compiling TypeScript to JavaScript
- TypeScript Datatypes
- TypeScript Classes and Interfaces

Introduction to JSON and AJAX

- JSON: JavaScript Object Notation
- AJAX
- XMLHttpRequest Object
- Fetch API
- The Callback

JavaScript Best Practices

- Code Conventions for the Lifecycle
- Resolving Shared Resources
- Constructs to Use
- Constructs to Avoid
- Memory Considerations

JavaScript Scheduling, Execution, and Security

- Timers and Scheduling
- Execution Contexts
- Same Origin Policy and JavaScript
- Security Considerations

HTML5 JavaScript API

- Working with Web Storage API
- Overview of Canvas API
- Drag and Drop API
- Cross-Domain Messaging
- Offline with Application Cache
- Geolocation: What, Why, and How

Credly Badge:



Display your Completion Badge And Get The Recognition You Deserve.

Add a completion and readiness badge to your Linkedin profile, Facebook page, or Twitter account to validate your professional and technical expertise. With badges issued and validated by Credly, you can:

- Let anyone verify your completion and achievement by clicking on the badge
- Display your hard work and validate your expertise
- Display each badge's details about specific skills you developed.

Badges are issued by QuickStart and verified through Credly.

Find Out More or See List Of Badges