

Document Generated: 06/30/2024

Learning Style: Virtual Classroom

Provider: JavaScript

Difficulty: Beginner

Course Duration: 3 Days

Introduction to JavaScript (TT4110)



About this course:

Web Essentials: Modern JavaScript is a course of hands-on web advancement intended for software designers who need to comprehend what JavaScript is and the way to utilize it with the present frameworks and structures, expanding their insight and abilities in developing and architecting complex web interfaces. This course gives a decent blend of hypothesis and practical labs intended to take understudies from the basics of JavaScript and its related innovations through to the utilization of devices and libraries to facilitate the improvement of innovative web applications. Students will hit the ground running, spread over JavaScript to tasks at both an architectural and also a line by line level of coding. Understudies who go to this course will leave this course outfitted with the necessary abilities to execute,

design, and deploy vigorous, adaptable, and safe JavaScript-based applications to a primary level.

The normal pay of a JavaScript Developer is \$103,001 every year.

Course Objective:

Working within a hands-on, engaging learning condition, guided by our master group, participants will figure out how to:

- Comprehend what JavaScript is and the method it is utilized within the context of web applications
- Work with the various innovations that are the establishment of web applications.
- Comprehend and work with the major parts of JavaScript regarding web applications, devices, security, and systems.
- Figure out how to how to viably function with the most up to date progresses in JavaScript, for example, TypeScript and ES6.
- Create code using optimal and conventions develops for execution.
- Define HTML and review its history
- Look at XHTML and its relationship to HTML
- Identify HTML limitations and improvements
- JavaScript “Objects” and “Classes”
- Constructors and Prototypes
- Extending Classes with Prototype
- JavaScript Object Notation
- AJAX: Load Once and Change Data
- XML Request Object is at the Center
- Code Conventions for the Lifecycle
- Resolving Shared Resource

Audience:

This course is an introductory level web designer, intended for engineers that need to additionally broaden their abilities in web improvement.

Prerequisite:

Participants should have broad working information in developing essential software applications.

Course Outline:

Module 1: HTML

Lesson: HTML (Optional)

- Define HTML and review its history
- Look at XHTML and its relationship to HTML

- Identify HTML limitations and improvements
- Exercise: Working with HTML

Lesson: HTML5

- HTML5 Overview
- HTML5 Semantic Structure
- HTML5 Forms
- HTML5 Media Delivery
- Exercise: Writing a Valid HTML5 Document
- Exercise: Writing HTML5 Forms

Module 2: CSS

Lesson: CSS (Optional)

- Learn the basics of CSS
- Meaning of cascading in CSS
- Declaring CSS within your HTML page
- Creating styles in an external CSS file
- Control how to display and position HTML elements
- Overriding standard tag behavior
- Adding new classes
- Using custom classes in your page
- Exercise: Working with CSS

Lesson: CSS3 Overview

- What is new in CSS3
- The Advantages of CSS3
- Browser Support for CSS3
- Exercise: Working with CSS3

Module 3: JavaScript

Lesson: JavaScript Basics

- JavaScript Defined
- Variables and Operators
- Flow Control and Conditionals
- Exercise: JavaScript Basics

Lesson: Debugging Tools

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

- Exercise: JavaScript Debugging

Lesson: JavaScript Functions

- Functions in JavaScript
- Invoking Functions
- Function Constructor
- Function Scope and Closures

Lesson: JavaScript Arrays, Math and Date

- JavaScript Literals
- JavaScript Arrays
- Working with Numbers and Dates
- Exercise: JavaScript Built-in Objects

Lesson: JavaScript Event Handling and the DOM

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

Lesson: Object-Oriented JavaScript

- JavaScript “Objects” and “Classes”
- Constructors and Prototypes
- Prototypes
- Extending Classes with Prototype
- Reusable, Flexible Classes
- Exercise: Working with Classes

Module 4: Advanced JavaScript Topics

Lesson: ES6: The Next Step

- ES6 Defined
- ES6 Default and Rest Parameters
- IIFE and Closures
- ES6 Importing/Exporting Modules
- Arrays, ES6 Maps, and ES6 Sets
- Working Arrays with ES6
- Working Strings with ES6 Strings
- ES6 Constructs and Inheritance
- ES6: Getters and Setters
- Exercise: Working with ES6

Lesson: TypeScript

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

Lesson: Introduction to JSON and Ajax

- JavaScript Object Notation
- AJAX: Load Once and Change Data
- XMLHttpRequest is at the Center
- Calling the Service
- The Callback
- Exercise: Implementing JSON

Lesson: JavaScript Best Practices

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

Lesson: JavaScript Scheduling, Execution, and Security

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

Lesson: HTML5 JavaScript API

- Cross-Domain Messaging
- Working with Web Storage
- Offline with Application Cache
- Geolocation: What, Why, and How
- Exercise: Using the Web Storage API
- Exercise: Using the ApplicationCache API

Module 5: Working with XML (Optional)

Lesson: XML DOM Mechanics

- HTML vs XML DOM
- Building an XML DOM
- Integrating XML into HTML
- XML Response Processing
- Exercise: Loading XML Data into an HTML Page

Lesson: XSLT Applied

- XSLT in the Browser
- XSLT in Specific Browsers
- Sarissa Assistance
- XSLT Pluses and Minuses
- Exercise: XSLT Transformations with Sarissa

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](#)