

JavaScript Boot Camp

Course Summary

Description

The JavaScript Boot Camp provides a comprehensive exploration of JavaScript, empowering participants with practical skills to develop dynamic and interactive web applications. This course serves as an essential foundation for aspiring front-end and full-stack developers, equipping them with in-depth knowledge of JavaScript syntax, object-oriented programming, and asynchronous programming. With hands-on exercises and real-world examples, this boot camp enables learners to build functional and efficient JavaScript code following best practices.

Objectives

The JavaScript Boot Camp offers a comprehensive curriculum that covers fundamental JavaScript concepts, advanced topics, and best practices. Participants will gain proficiency in writing clean and efficient JavaScript code to create dynamic and responsive user interfaces. The course includes in-depth coverage of variables, functions, objects, arrays, and control structures. Moreover, learners will delve into asynchronous programming, handling events, and working with the Document Object Model (DOM). Through a combination of theoretical learning and hands-on projects, this boot camp equips participants with the essential skills to build modern web applications and enhance existing projects.

Topics

- Introduction to JavaScript
- Setting up a JavaScript Project
- Setting up your Development Environment
- JavaScript Libraries
- JavaScript Framework
- Where is JavaScript written?
- Comments, Logging, and Documentation
- Syntax, Statements, and Expressions
- Variables and Constants
- JavaScript Primitive Types
- JavaScript Reference Types
- JavaScript Objects and the DOM
- Functions
- Events and Event Handling
- Function Signatures and Function Parameters
- Function Types
- JavaScript “this”
- Conditionals
- Arrays and Loops
- Asynchronous Programming
- Prototype
- Classes
- Getting Data from the Server

Audience

This course is designed for web developers who want to enhance their JavaScript skills and create robust web applications. It caters to both beginners seeking to learn JavaScript from scratch and experienced developers looking to deepen their understanding of advanced JavaScript concepts.

Prerequisites

Participants should have basic knowledge of HTML, CSS, and web development concepts. Prior programming experience is helpful but not required.

Duration

Five days

JavaScript Boot Camp

Course Outline

- I. *Introduction to JavaScript*
 - A. What is JavaScript?
 - B. Who is responsible for JavaScript?
 - C. Why should I know JavaScript?
 - D. JavaScript Pros, Cons, and Alternatives
 - E. When was JavaScript created?
 - F. How is JavaScript/ECMAScript maintained?
 - G. JavaScript as a multi-paradigm language
- II. *Setting up a JavaScript Project*
 - A. Setting Up a Project Directory
 - B. Packages and the package.json file
 - C. Package Managers
 - D. Version Control Systems
- III. *Setting up your Development Environment*
 - A. Code Formatters
 - B. Using WebStorm/VSCoDe
 - C. Configuring your IDE
 - D. Installing helpful IDE extensions
- IV. *JavaScript Libraries*
 - A. What is a JavaScript library?
 - B. Benefits of using a JavaScript library
 - C. Finding open-source JavaScript libraries
 - D. Vetting a JavaScript library
 - E. JavaScript mobile libraries
- V. *JavaScript Frameworks*
 - A. What is a JavaScript framework?
 - B. How Frameworks Help the Developer
- VI. *Where is JavaScript Written?*
 - A. Including External JavaScript files in a web page: the HTML <script> tag
 - B. JavaScript Modules
- VII. *Comments, Logging, and Documentation*
 - A. JavaScript comments
 - B. The use of comments in this course
 - C. Comments for Documentation
 - D. Comments vs. Logging
 - E. Logging Libraries and Frameworks
- VIII. *Syntax, Statements, and Expressions*
 - A. Statements, Variables, Expressions, and Syntax
 - B. What is a JavaScript statement?
 - C. Basic JavaScript Terminology
 - D. JavaScript Types
 - E. JavaScript Automatic Semicolon Insertion
 - F. Syntax Best Practices
 - G. What are JavaScript expressions?
- H. JavaScript Objects
 - I. The Document Object Model (DOM)
- IX. *Variables and Constants*
 - A. Variables vs. Constants
 - B. Using var, let, and const
 - C. Variable Scope
- X. *JavaScript Primitive Types*
 - A. Primitive types
 - B. Reference types
- XI. *JavaScript Reference Types*
 - A. What is an Object?
 - B. Reference Types vs. Primitive Types
- XII. *JavaScript Objects and the DOM*
 - A. Summary review of JavaScript Objects Properties and Methods
 - B. What is a property?
 - C. What is a method?
 - D. JavaScript built-in Objects
 - E. Calculating the difference between two dates
 - F. The Document Object Model
 - G. Getting Attribute Nodes
 - H. Working with Attributes, creating and removing Nodes
 - I. JavaScript Destructuring
- XIII. *Functions*
 - A. Function Invocation
 - B. Function Declarations vs. Function Expressions
 - C. Function Hoisting, classes, and the temporal dead zone
- XIV. *Events and Event Handling*
 - A. What are events?
 - B. How are events handled?
 - C. The event object
 - a. Event properties
 - b. Using events and event handlers
 - c. Event bubbling
 - d. Understanding event propagation
 - e. Custom events
 - i. The detail property
 - f. Creating a custom event

JavaScript Boot Camp

Course Outline (cont)

XV. Function Signatures and Function Parameters

- A. Function parameters
- B. Function signatures and overloading functions
- C. Default parameters
- D. Rest parameters
- E. Functions as arguments

XVI. Function Types

- A. Immediately Invoked Function Expressions (IIFE)
- B. ECMA6 Arrow Functions
- C. Arrow functions and scope

XVII. JavaScript "this"

- A. Function invocation in strict mode

XVIII. Conditionals

- A. Conditionals
- B. If statement syntax

XIX. Arrays and Loops

- A. Arrays
- B. Associative Arrays
- C. Sets vs. Arrays

XX. Asynchronous Programming

- A. Handling asynchronicity
- B. Introduction to promises
 - a. What is a promise?
 - b. How do promises work?
- C. Understanding promises
- D. Using Fetch to get data from the server

XXI. Prototype

- A. What is the prototype property?
- B. Creating your own objects
- C. The prototype property illustrated

XXII. Classes

- A. What is a class?
- B. Creating a class
- C. Using a class
- D. Getters and Setters

XXIII. Getting Data from the Server

- A. The XMLHttpRequest Object
- B. Using XHR to get data from the server
- C. Understanding the progress event
- D. Displaying data from the server