

JavaScript Mastery

Beginner to Advanced with Real Projects

Suman Biswas

Amazebazaar Publications
<https://amazebazaar.com>

JavaScript Mastery — Beginner to Advanced with Real Projects

First Edition, 2026

© 2026 Suman Biswas. All Rights Reserved.

No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means — electronic, mechanical, photocopying, recording, or otherwise — without the prior written permission of the author, except for brief quotations used in reviews and articles.

Published by: AmazeBazaar Publications

Website: <https://amazebazaar.com>

ISBN: 978-X-XXXX-XXXX-X (*placeholder*)

Official Download: This book is distributed only through <https://amazebazaar.com>. If you received it from anywhere else, please download the latest, official edition from the website.

Disclaimer: Every effort has been made to make this book accurate. However, the information is provided without warranty. The author and publisher are not liable for any loss or damage caused, directly or indirectly, by the instructions or code in this book. Technologies change quickly; always check official documentation for the latest details.

Trademarks: JavaScript is a trademark of Oracle Corporation. Visual Studio Code, Google Chrome, Node.js, and all other product names mentioned are trademarks of their respective owners and are used only for identification.

Table of Contents

Preface

Who This Book Is For
How to Use This Book

System Requirements

Installation Guide

Installing Visual Studio Code
Installing Google Chrome

Installing Node.js
Opening Chrome Developer Tools

Part 1 — JavaScript Fundamentals

Chapter 1: What Is JavaScript?

Learning Objectives
1.1 A Simple Definition
1.2 A Short History of JavaScript
1.3 What Is ECMAScript?
1.4 The JavaScript Engine
1.5 Where Does JavaScript Run?
1.6 Your First Look at JavaScript Code
1.7 Advantages of JavaScript

1.8 Disadvantages of JavaScript
1.9 Tips, Best Practices and Common Mistakes
1.10 Summary
1.11 Practice Exercises
1.12 Quiz
1.13 Interview Questions
1.14 Answers

Chapter 2: Setting Up Your Environment

Learning Objectives
2.1 The Three Ways to Run JavaScript
2.2 Setting Up VS Code
2.3 Running JavaScript with Node.js
2.4 Running JavaScript in the Browser
2.5 Live Server
2.6 Mastering the Console

2.7 Comments and Statements
2.8 Tips, Best Practices and Common Mistakes
2.9 Summary
2.10 Exercises
2.11 Quiz
2.12 Interview Questions
2.13 Answers

Chapter 3: Variables

Learning Objectives
3.1 What Is a Variable?
3.2 **let** — for values that change
3.3 **const** — for values that should not be reassigned
3.4 **var** — the old way
3.5 Hoisting and the Temporal Dead Zone
3.6 Comparison Table
3.7 Naming Rules (must follow)

3.8 Naming Conventions (should follow)
3.9 Coding Standards
3.10 Real-world Example: Shopping Cart Total
3.11 Common Mistakes
3.12 Summary
3.13 Exercises
3.14 Quiz
3.15 Interview Questions
3.16 Answers

Chapter 4: Data Types

Learning Objectives
4.1 Overview
4.2 Strings
4.3 Numbers
4.4 Booleans
4.5 Undefined and Null
4.6 BigInt
4.7 Symbol
4.8 Objects (Introduction)

4.9 Arrays (Introduction)
4.10 Primitive vs Reference Types
4.11 The **typeof** Operator
4.12 Best Practices and Common Mistakes
4.13 Summary
4.14 Exercises
4.15 Quiz
4.16 Interview Questions
4.17 Answers

Chapter 5: Operators

Learning Objectives
5.1 What Is an Operator?
5.2 Arithmetic Operators
5.3 Assignment Operators
5.4 Comparison Operators
5.5 Logical Operators
5.6 Nullish Coalescing **??**
5.7 Optional Chaining **?.**
5.8 Ternary Operator
5.9 Bitwise Operators

5.10 Spread Operator **...**
5.11 Rest Parameter **...**
5.12 Other Operators
5.13 Operator Precedence
5.14 Common Mistakes
5.15 Summary
5.16 Exercises
5.17 Quiz
5.18 Interview Questions
5.19 Answers

Chapter 6: Type Conversion

Learning Objectives

- 6.1 Why Conversion Matters
- 6.2 Explicit Conversion
- 6.3 Implicit Conversion (Coercion)
- 6.4 Real-world Example: Adding Two Inputs
- 6.5 Conversion Cheat Sheet

6.6 Best Practices and Common Mistakes

- 6.7 Summary
- 6.8 Exercises
- 6.9 Quiz
- 6.10 Interview Questions
- 6.11 Answers

Chapter 7: Control Flow

Learning Objectives

- 7.1 What Is Control Flow?
- 7.2 The `if` Statement
- 7.3 `if...else`
- 7.4 `else if` Chains
- 7.5 The `switch` Statement
- 7.6 Loops

7.7 `break` and `continue`

- 7.8 Real-world Example: Login Attempts
- 7.9 Tips, Best Practices and Common Mistakes
- 7.10 Summary
- 7.11 Exercises
- 7.12 Quiz
- 7.13 Interview Questions
- 7.14 Answers

Chapter 8: Functions

Learning Objectives

- 8.1 What Is a Function?
- 8.2 Function Declaration
- 8.3 Function Expression
- 8.4 Arrow Functions
- 8.5 Parameters and Arguments
- 8.6 Return Values
- 8.7 Scope Inside Functions
- 8.8 Callback Functions
- 8.9 Higher-Order Functions

8.10 Pure Functions

- 8.11 Recursion
- 8.12 Closures
- 8.13 IIFE — Immediately Invoked Function Expression
- 8.14 Tips, Best Practices and Common Mistakes
- 8.15 Summary
- 8.16 Exercises
- 8.17 Quiz
- 8.18 Interview Questions
- 8.19 Answers

Chapter 9: Arrays

Learning Objectives

- 9.1 What Is an Array?
- 9.2 Basic Operations (CRUD)
- 9.3 Mutating vs Non-mutating Methods
- 9.4 `forEach` — do something with each element
- 9.5 `map` — transform every element
- 9.6 `filter` — keep elements that pass a test
- 9.7 `reduce` — combine everything into one value
- 9.8 `find`, `findIndex`, `findLast`, `findLastIndex`
- 9.9 `some` and `every`
- 9.10 `sort`

9.11 `splice` vs `slice`

- 9.12 `flat` and `flatMap`
- 9.13 Chaining Methods
- 9.14 Complete Array Method Reference
- 9.15 Destructuring Arrays
- 9.16 Best Practices and Common Mistakes
- 9.17 Summary
- 9.18 Exercises
- 9.19 Quiz
- 9.20 Interview Questions
- 9.21 Answers

Chapter 10: Objects

Learning Objectives

- 10.1 What Is an Object?
- 10.2 CRUD Operations
- 10.3 Nested Objects
- 10.4 Looping Over Objects
- 10.5 Object Destructuring
- 10.6 Spread with Objects
- 10.7 Methods and `this`
- 10.8 Getters and Setters

10.9 Object APIs

- 10.10 Comparing and Copying Objects
- 10.11 A First Look at Prototypes
- 10.12 Real-world Example: A Product Catalogue
- 10.13 Best Practices and Common Mistakes
- 10.14 Summary
- 10.15 Exercises
- 10.16 Quiz
- 10.17 Interview Questions
- 10.18 Answers

Part 2 — Intermediate JavaScript

Chapter 11: Scope, Closures, Execution Context and Memory

Learning Objectives

- 11.1 Scope
- 11.2 Closures — A Deeper Look
- 11.3 Execution Context
- 11.4 The Call Stack
- 11.5 Memory: Stack and Heap
- 11.6 Garbage Collection
- 11.7 Memory Leaks

11.8 Strict Mode

- 11.9 Best Practices and Common Mistakes
- 11.10 Summary
- 11.11 Exercises
- 11.12 Quiz
- 11.13 Interview Questions
- 11.14 Answers

Chapter 12: The DOM (Document Object Model)

Learning Objectives

- 12.1 What Is the DOM?
- 12.2 Selecting Elements
- 12.3 Changing Content
- 12.4 Attributes
- 12.5 Classes and Styles
- 12.6 Creating and Inserting Elements
- 12.7 Removing and Cloning
- 12.8 Traversing the DOM

12.9 Dimensions and Position

- 12.10 Real-world Example: Product Cards
- 12.11 Best Practices and Common Mistakes
- 12.12 Summary
- 12.13 Exercises
- 12.14 Quiz
- 12.15 Interview Questions
- 12.16 Answers

Chapter 13: Events

Learning Objectives

- 13.1 What Is an Event?
- 13.2 Common Event Types
- 13.3 The Event Object
- 13.4 Event Propagation
- 13.5 Event Delegation
- 13.6 Preventing Default Behaviour
- 13.7 Listener Options and Removal
- 13.8 Page Lifecycle Events

13.9 Debounce and Throttle

- 13.10 Custom Events
- 13.11 Real-world Example: Keyboard-Accessible Modal
- 13.12 Best Practices and Common Mistakes
- 13.13 Summary
- 13.14 Exercises
- 13.15 Quiz
- 13.16 Interview Questions
- 13.17 Answers

Chapter 14: Forms and Validation

Learning Objectives

- 14.1 Reading Form Controls
- 14.2 Handling Submit
- 14.3 Built-in HTML Validation
- 14.4 Custom JavaScript Validation
- 14.5 A Quick Regular Expression Primer
- 14.6 Live Feedback: Password Strength

14.7 File Inputs

- 14.8 Best Practices and Common Mistakes
- 14.9 Summary
- 14.10 Exercises
- 14.11 Quiz
- 14.12 Interview Questions
- 14.13 Answers

Chapter 15: Timers, Web Storage and Cookies

Learning Objectives

- 15.1 `setTimeout` — run once, later
- 15.2 `setInterval` — run repeatedly
- 15.3 `requestAnimationFrame` — smooth animations
- 15.4 Web Storage: `localStorage` and `sessionStorage`
- 15.5 Cookies
- 15.6 Choosing Storage

15.7 Real-world Example: Persistent Theme Toggle

- 15.8 Best Practices and Common Mistakes
- 15.9 Summary
- 15.10 Exercises
- 15.11 Quiz
- 15.12 Interview Questions
- 15.13 Answers

Chapter 16: Modules — Import, Export and Dynamic Import

Learning Objectives

- 16.1 Why Modules?
- 16.2 Named Exports
- 16.3 Default Exports
- 16.4 Re-exporting (Barrel Files)
- 16.5 Modules in the Browser
- 16.6 Modules in Node.js
- 16.7 Dynamic Import

16.8 Live Bindings and Circular Imports

- 16.9 `import.meta`
- 16.10 Best Practices and Common Mistakes
- 16.11 Summary
- 16.12 Exercises
- 16.13 Quiz
- 16.14 Interview Questions
- 16.15 Answers

Chapter 17: Error Handling and Debugging

Learning Objectives

- 17.1 Types of Errors
- 17.2 `try...catch...finally`
- 17.3 Throwing Errors
- 17.4 Custom Error Classes
- 17.5 Errors in Asynchronous Code
- 17.6 Debugging

17.7 Real-world Example: Safe Data Loader

- 17.8 Best Practices and Common Mistakes
- 17.9 Summary
- 17.10 Exercises
- 17.11 Quiz
- 17.12 Interview Questions
- 17.13 Answers

Chapter 18: Promises and Async/Await

Learning Objectives

- 18.1 Synchronous vs Asynchronous
- 18.2 Callbacks and Callback Hell
- 18.3 What Is a Promise?
- 18.4 Promise Chaining
- 18.5 Promise Combinators
- 18.6 `async` and `await`

18.7 Sequential vs Parallel

- 18.8 Timeouts and Cancellation
- 18.9 Promisifying Callback APIs
- 18.10 Real-world Example: Loading a Dashboard
- 18.11 Best Practices and Common Mistakes
- 18.12 Summary
- 18.13 Exercises

18.14 Quiz
18.15 Interview Questions

Chapter 19: JSON, the Fetch API and REST APIs

Learning Objectives
19.1 JSON
19.2 How the Web Talks: HTTP
19.3 What Is a REST API?
19.4 The Fetch API
19.5 A Reusable API Client
19.6 UI States: Loading, Error, Empty, Success

Chapter 20: Building a CRUD Application

Learning Objectives
20.1 What We Will Build
20.2 Project Setup
20.3 The HTML
20.4 The API Layer
20.5 The UI Layer
20.6 The Controller (main.js)
20.7 Minimal Styling

Chapter 21: Modern ES6+ Features

Learning Objectives
21.1 ES6+ at a Glance
21.2 Template Literals
21.3 Enhanced Object Literals
21.4 Destructuring Patterns (Advanced)
21.5 Optional Chaining, Nullish Coalescing, Logical Assignment
21.6 Numbers, Strings and Arrays — Small but Useful
21.7 Dates and the `Intl` API

Chapter 22: Classes and Inheritance

Learning Objectives
22.1 Why Classes?
22.2 Defining a Class
22.3 Class Fields, Getters and Setters
22.4 Private Members (`#`)
22.5 Static Members
22.6 Inheritance with `extends` and `super`
22.7 Classes Under the Hood: Prototypes
22.8 `this` in Classes

Chapter 23: Iterators and Generators

Learning Objectives
23.1 The Iteration Protocol
23.2 Making an Object Iterable
23.3 Generators
23.4 Infinite and Lazy Sequences
23.5 `yield*` — Delegation
23.6 Two-way Communication: `next(value)`, `return`, `throw`
23.7 Practical Uses

Chapter 24: Map, Set, WeakMap and WeakSet

Learning Objectives
24.1 `Map`
24.2 `Set`
24.3 `WeakMap`
24.4 `WeakSet`
24.5 `WeakRef` and `FinalizationRegistry` (Advanced)

Chapter 25: Proxy and Reflect

Learning Objectives
25.1 What Is a Proxy?
25.2 The Traps
25.3 What Is `Reflect`?

18.16 Answers

19.7 CORS in One Paragraph
19.8 Best Practices and Common Mistakes
19.9 Summary
19.10 Exercises
19.11 Quiz
19.12 Interview Questions
19.13 Answers

20.8 What You Practised
20.9 Best Practices and Common Mistakes
20.10 Summary
20.11 Exercises
20.12 Quiz
20.13 Interview Questions
20.14 Answers

21.8 ES2025 Highlights
21.9 Best Practices and Common Mistakes
21.10 Summary
21.11 Exercises
21.12 Quiz
21.13 Interview Questions
21.14 Answers

22.9 The Four Pillars of OOP
22.10 Composition over Inheritance
22.11 Real-world Example: Shopping Cart Class
22.12 Best Practices and Common Mistakes
22.13 Summary
22.14 Exercises
22.15 Quiz
22.16 Interview Questions
22.17 Answers

23.8 Async Iterators and `for await...of`
23.9 Iterator Helpers (ES2025)
23.10 Best Practices and Common Mistakes
23.11 Summary
23.12 Exercises
23.13 Quiz
23.14 Interview Questions
23.15 Answers

24.6 Real-world Example: LRU Cache with Map
24.7 Best Practices and Common Mistakes
24.8 Summary
24.9 Exercises
24.10 Quiz
24.11 Interview Questions
24.12 Answers

25.4 Practical Proxies
25.5 Reactive State (How Vue Works, Simplified)
25.6 Revocable Proxies
25.7 Limitations and Performance

25.8 Best Practices and Common Mistakes
 25.9 Summary
 25.10 Exercises

25.11 Quiz
 25.12 Interview Questions
 25.13 Answers

Part 3 — Advanced JavaScript

Chapter 26: The Event Loop, Microtasks and Macrotasks

Learning Objectives
 26.1 The Big Picture
 26.2 The Event Loop Algorithm
 26.3 Predicting Output
 26.4 `async/await` and the Event Loop
 26.5 Blocking the Main Thread
 26.6 Web Workers — Real Parallelism

26.7 Node.js Event Loop (Brief)
 26.8 Best Practices and Common Mistakes
 26.9 Summary
 26.10 Exercises
 26.11 Quiz
 26.12 Interview Questions
 26.13 Answers

Chapter 27: Performance and Optimization

Learning Objectives
 27.1 Measure First
 27.2 Core Web Vitals
 27.3 Loading Performance
 27.4 Runtime Performance: The DOM
 27.5 Runtime Performance: JavaScript
 27.6 Memory Optimisation

27.7 Performance Checklist
 27.8 Summary
 27.9 Exercises
 27.10 Quiz
 27.11 Interview Questions
 27.12 Answers

Chapter 28: Functional Programming

Learning Objectives
 28.1 What Is Functional Programming?
 28.2 Core Principles
 28.3 Immutability in Practice
 28.4 Function Composition
 28.5 Currying and Partial Application
 28.6 Higher-Order Utility Functions
 28.7 A Functional Pipeline: Sales Report
 28.8 Recursion Instead of Loops

28.9 Handling “Nothing”: Option-style Helpers
 28.10 When Not to Go “Full FP”
 28.11 Tips, Best Practices and Common Mistakes
 28.12 Summary
 28.13 Exercises
 28.14 Quiz
 28.15 Interview Questions
 28.16 Answers

Chapter 29: Design Patterns, SOLID and Clean Code

Learning Objectives
 29.1 What Is a Design Pattern?
 29.2 Creational Patterns
 29.3 Structural Patterns
 29.4 Behavioural Patterns
 29.5 SOLID Principles
 29.6 Clean Code

29.7 Tips, Best Practices and Common Mistakes
 29.8 Summary
 29.9 Exercises
 29.10 Quiz
 29.11 Interview Questions
 29.12 Answers

Chapter 30: Testing with Jest and Vitest

Learning Objectives
 30.1 Why Test?
 30.2 Setting Up
 30.3 Your First Test
 30.4 Matchers
 30.5 Setup and Teardown
 30.6 Testing Asynchronous Code
 30.7 Mocks, Spies and Fake Timers
 30.8 Testing DOM Code

30.9 Code Coverage
 30.10 Test-Driven Development (TDD)
 30.11 Best Practices and Common Mistakes
 30.12 Summary
 30.13 Exercises
 30.14 Quiz
 30.15 Interview Questions
 30.16 Answers

Chapter 31: Introduction to TypeScript

Learning Objectives
 31.1 What Is TypeScript?
 31.2 Setup
 31.3 Basic Types
 31.4 Object Types: `type` and `interface`
 31.5 Functions
 31.6 Type Narrowing
 31.7 Classes
 31.8 Generics
 31.9 Utility Types

31.10 Enums and `as const`
 31.11 Typing the DOM
 31.12 Adopting TypeScript Gradually
 31.13 Best Practices and Common Mistakes
 31.14 Summary
 31.15 Exercises
 31.16 Quiz
 31.17 Interview Questions
 31.18 Answers

Chapter 32: Build Tools — npm, Vite, Webpack and Babel

Learning Objectives
32.1 Why Build Tools?
32.2 npm Essentials
32.3 Vite
32.4 Webpack
32.5 Babel
32.6 Linting and Formatting

32.7 Tips, Best Practices and Common Mistakes
32.8 Summary
32.9 Exercises
32.10 Quiz
32.11 Interview Questions
32.12 Answers

Chapter 33: Security — Authentication, JWT, OAuth, Cookies, XSS, CSRF and CORS

Learning Objectives
33.1 Authentication vs Authorisation
33.2 Password Basics (Server Side)
33.3 Session-Based Authentication (Cookies)
33.4 Token-Based Authentication and JWT
33.5 OAuth 2.0 and OpenID Connect
33.6 Secure Cookies
33.7 Cross-Site Scripting (XSS)
33.8 Cross-Site Request Forgery (CSRF)

33.9 CORS — Cross-Origin Resource Sharing
33.10 More Front-end Security Practices
33.11 Security Checklist
33.12 Tips, Best Practices and Common Mistakes
33.13 Summary
33.14 Exercises
33.15 Quiz
33.16 Interview Questions
33.17 Answers

Chapter 34: Deployment — GitHub Pages, Netlify and Vercel

Learning Objectives
34.1 What Is Deployment?
34.2 Pre-deployment Checklist
34.3 Git and GitHub Basics
34.4 GitHub Pages
34.5 Netlify
34.6 Vercel
34.7 Comparison

34.8 Custom Domains
34.9 After Deployment
34.10 Tips, Best Practices and Common Mistakes
34.11 Summary
34.12 Exercises
34.13 Quiz
34.14 Interview Questions
34.15 Answers

Part 4 — Real-World Projects

How to Work Through the Projects

Project 1: Calculator

Overview
Step 1 — HTML
Step 2 — CSS
Step 3 — JavaScript
How It Works

Enhancements
Exercises
Interview Questions
Answers

Project 2: To-do App

Overview
Step 1 — HTML
Step 2 — CSS
Step 3 — JavaScript
How It Works

Enhancements
Exercises
Interview Questions
Answers

Project 3: Weather App

Overview
Step 1 — HTML
Step 2 — CSS
Step 3 — API module
Step 4 — UI module
Step 5 — Main controller

How It Works
Enhancements
Exercises
Interview Questions
Answers

Project 4: Movie & TV Search App

Overview
Step 1 — HTML
Step 2 — CSS
Step 3 — JavaScript
How It Works

Enhancements
Exercises
Interview Questions
Answers

Project 5: Quiz App

Overview
Step 1 — Data
Step 2 — HTML
Step 3 — CSS
Step 4 — JavaScript

How It Works
Enhancements
Exercises
Interview Questions
Answers

Project 6: Expense Tracker

Overview

Step 1 — HTML

Step 2 — CSS

Step 3 — Store module

Step 4 — Chart module (Canvas doughnut)

Step 5 — Main controller

How It Works

Enhancements

Exercises

Interview Questions

Answers

Project 7: Notes App

Overview

Step 1 — HTML

Step 2 — CSS

Step 3 — IndexedDB wrapper

Step 4 — Safe mini-Markdown renderer

Step 5 — Main controller

How It Works

Enhancements

Exercises

Interview Questions

Answers

Project 8: Portfolio Website

Overview

Step 1 — HTML

Step 2 — Projects data

Step 3 — CSS

Step 4 — JavaScript

How It Works

Enhancements

Exercises

Interview Questions

Answers

Project 9: Blog (Single-Page App)

Overview

Step 1 — Content

Step 2 — HTML and CSS

Step 3 — Router

Step 4 — Views

Step 5 — Main

How It Works

Enhancements

Exercises

Interview Questions

Answers

Project 10: Real-Time Chat App

Overview

Step 1 — Server

Step 2 — Client HTML & CSS

Step 3 — Client JavaScript

How It Works

Enhancements

Exercises

Interview Questions

Answers

Project 11: E-commerce Front End

Overview

Step 1 — HTML

Step 2 — CSS

Step 3 — API module

Step 4 — Cart store (Observer pattern)

Step 5 — Rendering helpers

Step 6 — Main controller

How It Works

Enhancements

Exercises

Interview Questions

Answers

Project 12: Admin Dashboard

Overview

Step 1 — HTML

Step 2 — CSS

Step 3 — Data module

Step 4 — Reusable data table

Step 5 — Charts with Chart.js

Step 6 — Main

How It Works

Enhancements

Exercises

Interview Questions

Answers

Project 13: Complete Authentication System

Overview

Step 1 — Setup

Step 2 — User store

Step 3 — Middleware

Step 4 — Server

Step 5 — Front end

Security Review

Production Checklist

Exercises

Interview Questions

Answers

Project 14: REST API Integration — Build and Consume a Products API

Overview

Step 1 — API design

Step 2 — Data layer

Step 3 — Express app

Step 4 — A robust HTTP client

Step 5 — Inventory UI

Step 6 — API tests

How It Works

Enhancements
Exercises

Interview Questions
Answers

Part 5 — Interview Preparation

Chapter 35: 300 JavaScript Interview Questions and Answers

How to Use This Chapter

- A. Basics and Syntax (Q1–Q40)
- B. Types, Coercion and Operators (Q41–Q70)
- C. Functions, Scope, Closures and `this` (Q71–Q110)
- D. Objects, Prototypes and Classes (Q111–Q145)

- E. Arrays and Collections (Q146–Q170)
- F. Asynchronous JavaScript (Q171–Q205)
- G. DOM, Events and Browser APIs (Q206–Q240)
- H. ES6+ and Modules (Q241–Q260)
- I. Performance, Security, Testing and Tooling (Q261–Q285)
- J. Advanced and Scenario-Based (Q286–Q300)

Chapter 36: Coding Challenges

- Challenge 1 — Reverse a String
- Challenge 2 — Palindrome Check
- Challenge 3 — FizzBuzz
- Challenge 4 — Count Character Frequency
- Challenge 5 — First Non-Repeating Character
- Challenge 6 — Anagram Check
- Challenge 7 — Two Sum
- Challenge 8 — Find the Missing Number
- Challenge 9 — Chunk an Array
- Challenge 10 — Flatten Without `flat`
- Challenge 11 — Capitalise Each Word
- Challenge 12 — Longest Word
- Challenge 13 — Array Intersection
- Challenge 14 — Group Anagrams
- Challenge 15 — Valid Parentheses

- Challenge 16 — Maximum Subarray Sum (Kadane's Algorithm)
- Challenge 17 — Binary Search
- Challenge 18 — Fibonacci (Memoised and Iterative)
- Challenge 19 — Rotate an Array by k
- Challenge 20 — Deep Equality
- Challenge 21 — Implement `curry`
- Challenge 22 — Implement `memoize`
- Challenge 23 — Retry an Async Function with Back-off
- Challenge 24 — Run Promises in Sequence
- Challenge 25 — Get a Nested Value by Path
- Challenge 26 — Implement an LRU Cache
- Challenge 27 — Count Islands in a Grid
- Challenge 28 — Merge Overlapping Intervals
- Challenge 29 — Implement `Array.prototype.myReduce`
- Challenge 30 — Concurrency-Limited Task Runner

Chapter 37: Logic Problems

Chapter 38: Output Questions

Chapter 39: Multiple-Choice Questions (MCQs)

Chapter 40: Best Practices and Common Mistakes

- 40.1 Best Practices Checklist
- 40.2 The 30 Most Common JavaScript Mistakes

- 40.3 Interview Day Tips

Chapter 41: Practice Workbook — Extra Exercises with Solutions

- Part 1 Practice — Fundamentals
- Part 2 Practice — Intermediate

- Part 3 Practice — Advanced
- Mini-Projects for Extra Practice

Appendices

Appendix A: Glossary

Appendix B: JavaScript Cheat Sheet

Variables and Types
Strings
Numbers and Math
Operators
Control Flow
Functions
Arrays

Objects
Classes
Async
DOM and Events
Storage
Modules
Error Handling

Appendix C: Keyboard Shortcuts

VS Code

Chrome DevTools

Appendix D: Useful VS Code Extensions

Appendix E: Useful Websites and Practice Platforms

Documentation and Learning
Practice Platforms

Free APIs for Practice Projects

Appendix F: Career Roadmap

- F.1 Career Paths with JavaScript
- F.2 Front-End Roadmap

- F.3 Back-End Roadmap
- F.4 Learning Timeline (Suggested)

Enjoyed the free sample?

JavaScript Mastery

The full book has 502 pages. You have read 10.

Get the full book at

[**amazebazaar.com/shop**](https://amazebazaar.com/shop)