# JavaScript 30 Project Report

## Table of Contents

1. Introduction

2. Objectives

3. Tools and Technologies

4. Project Summaries

1. JavaScript Drum Kit

2. JS and CSS Clock

3. CSS Variables

4. Array Cardio, Day 1

5. Flex Panels Image Gallery

6. Type Ahead

7. Array Cardio, Day 2

8. Fun with HTML5 Canvas

9. Hold Shift and Check Checkboxes

10. Key Sequence Detection

11. Slide in on Scroll

12. JavaScript References vs Copying

13. LocalStorage and Event Delegation

14. Mouse Move Shadow

15. Sort Without Articles

16. Adding Up Times with Reduce

17. Follow Along Link Highlighter

18. Speech Synthesis

19. Sticky Nav

20. Event Capture, Propagation, Bubbling, and Once

21. Stripe Follow Along Nav

22. Click and Drag

23. Video Speed Controller

24. Countdown Timer

25. Whack-a-Mole

5. Challenges Faced

6. Learnings and Insights

7. Conclusion

## 1. Introduction

JavaScript 30 is a series of 30 projects created by Wes Bos aimed at boosting JavaScript skills through hands-on practice. Each project focuses on different functionalities and concepts in JavaScript, making learning fun and engaging.

## 2. Objectives

- Sharpen my JavaScript skills.

- Understand and apply modern JavaScript features.

- Improve problem-solving and debugging abilities.

## 3. Tools and Technologies

- JavaScript (ES6+)

- HTML5

- CSS3

- Browser Developer Tools

- Git and GitHub

## 4. Project Summaries

### 1. JavaScript Drum Kit

Objective: Create a virtual drum kit that plays sounds on keypress.

Implementation: I used keydown events to play corresponding sounds and added visual feedback for key presses.

### 2. JS and CSS Clock

Objective: Create a functioning analog clock.

Implementation: I utilized CSS for styling and JavaScript to rotate the clock hands based on the current time.

### 3. CSS Variables

Objective: Manipulate CSS variables using JavaScript.

Implementation: I linked input elements to CSS variables and dynamically updated styles based on user input.

### 4. Array Cardio, Day 1

Objective: Practice array methods like `map`, `filter`, `reduce`, and `sort`.

Implementation: I performed various array manipulations and analyses using ES6 array methods.

### 5. Flex Panels Image Gallery

Objective: Create an expandable image gallery.

Implementation: I used CSS flexbox for layout and JavaScript to toggle classes on click events.

### 6. Type Ahead

Objective: Implement a live search feature.

Implementation: I fetched data and used `filter` and `regex` to display matching results as the user types.

### 7. Array Cardio, Day 2

Objective: Further practice with array methods.

Implementation: I worked on more complex array manipulations and challenges.

### 8. Fun with HTML5 Canvas

Objective: Draw on an HTML5 canvas.

Implementation: I utilized the Canvas API to create a drawing application with different brush styles.

### 9. Hold Shift and Check Checkboxes

Objective: Select multiple checkboxes using shift-click.

Implementation: I used event listeners and logic to handle shift-click for checkbox selection.

### 10. Key Sequence Detection

Objective: Detect specific key sequences.

Implementation: I used event listeners and array methods to detect and respond to key sequences.

### 11. Slide in on Scroll

Objective: Animate elements on scroll.

Implementation: I used the Intersection Observer API to trigger animations when elements come into view.

### 12. JavaScript References vs Copying

Objective: Understand references and copying in JavaScript.

Implementation: I demonstrated the differences between reference and value copying with various data types.

### 13. LocalStorage and Event Delegation

Objective: Implement a to-do list with localStorage.

Implementation: I used localStorage to persist data and event delegation for dynamic elements.

### 14. Mouse Move Shadow

Objective: Create a shadow effect that follows the mouse.

Implementation: I used mousemove events and CSS variables to dynamically adjust shadow properties.

### 15. Sort Without Articles

Objective: Sort a list of strings, ignoring leading articles.

Implementation: I implemented a custom sort function that strips articles before sorting.

### 16. Adding Up Times with Reduce

Objective: Sum up durations in an array.

Implementation: I used `reduce` to calculate total time from an array of time strings.

### 17. Follow Along Link Highlighter

Objective: Highlight links with a hover effect.

Implementation: I used JavaScript to create a dynamic underline effect that follows the cursor.

### 18. Speech Synthesis

Objective: Implement text-to-speech functionality.

Implementation: I used the SpeechSynthesis API to convert text input to speech.

### 19. Sticky Nav

Objective: Create a sticky navigation bar.

Implementation: I used JavaScript to add and remove classes based on scroll position.

### 20. Event Capture, Propagation, Bubbling, and Once

Objective: Understand event handling phases.

Implementation: I demonstrated different event phases and `addEventListener` options.

### 21. Stripe Follow Along Nav

Objective: Create an animated navigation menu.

Implementation: I used JavaScript to animate menu items and dropdowns.

### 22. Click and Drag

Objective: Implement click-and-drag functionality.

Implementation: I enabled dragging to scroll content using mouse events.

### 23. Video Speed Controller

Objective: Control video playback speed.

Implementation: I created a custom UI to adjust video speed dynamically.

### 24. Countdown Timer

Objective: Build a countdown timer.

Implementation: I used `setInterval` to update the countdown and displayed the remaining time.

### 25. Whack-a-Mole

Objective: Create a simple game.

Implementation: I used JavaScript to create a dynamic game where users click on appearing elements.

## 5. Challenges Faced

- Debugging Issues: Some projects required extensive debugging.

## 6. Learnings and Insights

- JavaScript Mastery: I gained a deep understanding of various JavaScript features.

- CSS Skills: I improved my ability to create dynamic and responsive layouts.

- Problem-Solving: I enhanced my skills in troubleshooting and debugging.

- Efficient Coding: I learned to write cleaner, more efficient code.

## 7. Conclusion

The JavaScript 30 assignment provided an invaluable hands-on experience, significantly enhancing my JavaScript skills. Each project offered unique challenges and learning opportunities, contributing to a comprehensive understanding of modern web development techniques.

---

This concludes my project report for the JavaScript 30 assignment, summarizing each project and the overall learning experience.