Exercise 1: JavaScript Debugging

Issue: The code is checking for "none" and trying to toggle it to "block", but it doesn't consider the initial value.

Debugged Code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Toggle Element</title>

</head>

<body>

<button onclick="toggleElement()">Toggle Element</button>

<div id="target" style="display: none;">This is the target element.</div>

<script>

function toggleElement() {

var element = document.getElementById("target");

if (element.style.display === "" || element.style.display === "none") {

element.style.display = "block";

} else {

element.style.display = "none";

}

}

</script>

</body>

</html>

Explaination: To fix this, you should use a different approach to toggle the visibility.

Exercise 2: CSS Troubleshooting

Issue: The issue in the provided CSS code is that the margin: auto; rule applied to the .container does not effectively center the container.

Debugged Code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Centered Container</title>

<style>

    div{

    margin: 0 auto;

    padding: 0;

    box-sizing: border-box;

}

.container {

margin-top: 50%;

margin-bottom: 50%;

width: 50%;

background-color: #f0f0f0;

padding: 20px;

text-align: center;

text

}

html,body {

            display: flex;

            justify-content: center;

            align-items: center;

            height: 100%;

            margin: 0;

        }

</style>

</head>

<body>

<div class="container">

<h1>Centered Container</h1>

<p>This container should be centered on the page.</p>

</div>

</body>

</html>

Explaination: Added display: flex;

justify-content: center;

align-items: center; in the body and html in css and text -align:center in .container

Exercise 3: Debugging JavaScript Functions

Issue: There is no issue with the provided code. It calculates the sum of the array correctly.

Debugged code:

function calculateSum(arr) {

    return arr.reduce((acc, currentValue) => acc + currentValue, 0);

}

const numbers = [1, 2, 3, 4, 5];

const result = calculateSum(numbers);

console.log(result);

Result: The code already works correctly and outputs 15 but I have made it more efficient by adding the .reduce function

Exercise 4: Debugging CSS Styling Issues

Issue: The boxes are not correctly aligned, and their text is misaligned.

Debugged Code:

<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

<meta name="viewport" content="width=device-width, initial-scale=1.0">

<title>Styling Debugging Exercise</title>

<style>

body {

display: flex;

justify-content: center;

align-items: center;

height: 100vh;

margin: 0;

}

.container {

display: flex;

}

.box {

width: 100px;

height: 100px;

background-color: #007bff;

color: #ffffff;

text-align: center;

line-height: 100px;

margin: 5px;

}

</style>

</head>

<body>

<div class="container">

<div class="box">Box 1</div>

<div class="box">Box 2</div>

<div class="box">Box 3</div>

</div>

</body>

</html>

Explaination: It centers the layout both horizontally and vertically on the page. It also adds a margin between the boxes to achieve the desired layout.