**Exercise 1**: JavaScript Debugging Problem Statement: You've been given a simple JavaScript code snippet that's intended to toggle the visibility of an element when a button is clicked

**Corrected 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: block;">This is the target element.</div>

<script>

function toggleElement() {

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

element.style.display = (element.style.display === "block") ? "none" : "block";

}

</script>

</body>

</html>

**Explanation :**

The issue in the provided JavaScript code is that the display property is initially set to "none," but in the toggleElement function, it checks for "none" and toggles it to "block" and vice versa. However, the initial value should be "block" if you want to toggle the visibility correctly. I've changed the initial display style to "block" to ensure that clicking the button will correctly toggle the visibility of the element.

**Exercise 2**: CSS Troubleshooting Problem Statement: You've been given an HTML and CSS code snippet that's supposed to create a centered, responsive container.

**Corrected 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>

.container {

margin-left: auto;

margin-right: auto;

width: 50%;

background-color: #f0f0f0;

padding: 20px;

display: block;

}

</style>

</head>

<body>

<div class="container">

<h1>Centered Container</h1>

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

</div>

</body>

</html>

**Explannation :**

The issue in the provided CSS code is that the "margin: auto;" property is not sufficient to center the container. To center the container horizontally, you should set the left and right margins to "auto," and you also need to specify a block formatting context by adding "display: block" to the container. By setting both left and right margins to "auto" and adding "display: block" to the container, you'll correctly center the container horizontally on the page.

**Exercise 3**: Debugging JavaScript Functions Objective: Identify and fix issues in JavaScript functions. This code snippet with a JavaScript function that performs a specific task, but contains bugs or inefficiencies.

**Explaination**

The JavaScript function you provided appears to be correct and efficient for calculating the sum of an array of numbers. There are no issues or bugs in this code.

The code will correctly calculate the sum of the numbers array, which contains [1, 2, 3, 4, 5], and the result will be logged to the console. It should work as expected.

**Exercise 4:** Debugging CSS Styling Issues Objective: Identify and fix CSS styling issues to achieve the desired layout. This code snippet with HTML and CSS code that creates a specific layout, but contains CSS issues like misalignment, overlapping elements, or incorrect colors.

**Corrected coded:** <!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>

.container {

width: 50%;

margin: 0 auto;

background-color: #f0f0f0;

padding: 20px;

}

.box {

width: 100px;

height: 100px;

background-color: #007bff;

color: #ffffff;

text-align: center;

line-height: 100px;

margin: 10px; /\* Add margin to separate the boxes \*/

}

</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>

**Explannation :**

I've added margin: 10px; to the .box class to separate the boxes and achieve the desired layout with proper spacing between them.