

## Task 1:

```
<!DOCTYPE html>

<html lang="en">

<head>

<meta charset="UTF-8">

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

<title>Debugging Exercise</title>

<style>

body {

font-family: Arial, sans-serif;

background-color: #f4f4f4;

color: #333;

margin: 20px;

}

h1 {

color: #007bff;

}

p {

font-size: 16px;

margin-bottom: 20px;

}

.important-text {
```

```
font-weight: bold;
```

```
color: #d9534f;
```

```
}
```

```
</style>
```

```
</head>
```

```
<body>
```

```
<h1>Debugging Exercise</h1>
```

```
<p>This is a paragraph with some <span class="important-text">important  
text</span>.</p>
```

```
<p>Here's an unordered list:</p>
```

```
<ul>
```

```
<li>Item 1</li>
```

```
<li>Item 2</li>
```

```
<li>Item 3</li>
```

```
</ul>
```

```
<p>And here's an ordered list:</p>
```

```
<ol>
```

```
<li>First item</li>
```

```
<li>Second item</li>
```

```
<li>Third item</li>
```

```
</ol>
```

```
<p>This is a <a href="https://www.example.com">link to example.com</a>.</p>
```

```
<script>
```

```
console.log("Debugging exercise script");
```

```
</script>
```

```
</body>
```

```
</html>
```

### **Explanation:**

The code does not have any mistake or error.

### **Task 2:**

#### **Wrong 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");  
  
    element.style.display = (element.style.display === "none") ? "block" : "none";  
  
}  
  
</script>  
  
</body>  
  
</html>
```

### 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: none;">This is the target element.</div>  
  
    <script>  
        function toggleElement() {  
            var element = document.getElementById("target");  
            var displayStyle = window.getComputedStyle(element).display;  
  
            element.style.display = (displayStyle === "none") ? "block" : "none";  
        }  
    </script>  
</body>  
</html>
```

### Explanation:

The error in the provided code was that the JavaScript function `toggleElement()` didn't work as expected when the button was clicked. This was because the function checked the initial display state of the target element using `element.style.display`. However, since the initial display state was set using inline styles (`style="display: none;"`), this method didn't work as intended. To fix this, the code was updated to use `window.getComputedStyle()` to get the computed style of the element, including

styles applied via CSS. This allowed the function to correctly toggle the visibility of the target element when the button was clicked.

### Task 3:

#### Wrong 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: auto;

width: 50%;

background-color: #f0f0f0;

padding: 20px;

}

</style>

</head>

<body>

<div class="container">

<h1>Centered Container</h1>

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

```
</div>

</body>

</html>
```

### 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>
    body {
      margin: 0;
      padding: 0;
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
    }

    .container {
      width: 50%;
      background-color: #f0f0f0;
      padding: 20px;
    }
  </style>
</head>
<body>
  <div class="container">
    <h1>Centered Container</h1>
    <p>This container should be centered on the page.</p>
  </div>
</body>
</html>
```

### Explanation:

The error in the provided code was that the container wasn't displayed in the center of the page as expected. This happened because the CSS rule used to center the container (`margin: auto;`) didn't work properly in this context. To fix this issue, needed to adjust the CSS by using `display: flex;` and `justify-content: center;` to ensure the container was centered horizontally on the page. This allowed the container to display correctly in the center of the page.

#### Task 4:

```
function calculateSum(arr) {  
  let sum = 0;  
  for (let i = 0; i < arr.length; i++) {  
    sum += arr[i];  
  }  
  return sum;  
}  
  
const numbers = [1, 2, 3, 4, 5];  
const result = calculateSum(numbers);  
console.log(result);
```

#### Explanation:

The code does not have any mistake or error.

#### Task 4:

#### Wrong 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>  
  
.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;  
  
}  
  
</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>
```

### **Corrected 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>
```



```
.container {
  width: 50%;
  margin: 0 auto;
  background-color: #f0f0f0;
  padding: 20px;
  text-align: center;
}

.box {
  width: 100px;
  height: 100px;
  background-color: #007bff;
  color: #ffffff;
  display: inline-block;
  margin: 10px;
  line-height: 100px;
}
</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>
```

**Explanation:**

The error in the provided code was that the boxes were not displaying as intended. They were not centered horizontally within the container, and there was no spacing between them. This issue was likely due to missing CSS properties. To fix it, the person added `text-align: center;` to the container class to center the boxes horizontally, changed the `display` property of the box class to `inline-block` to ensure the boxes display inline, and added a `margin` to provide spacing between the boxes. These changes corrected the layout issue, ensuring that the boxes are now properly aligned and spaced within the container.

