TASK 5- Error finding

EXAMPLE 1- The task is to identify and fix the errors in the code:
CODE:

```
<!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:- This code doesn't have any critical bugs or errors

EXERCISE 2- JavaScript Debugging

Incorrect Code:
```html
<!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:
```html
<!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");

        // Check the current display state using getComputedStyle
      var currentDisplay = window.getComputedStyle(element).display;
        element.style.display = (currentDisplay === "none") ? "block" : "none";
     }
   </script>
</body>
</html>
```

Explanation: The error in this JavaScript code is that it doesn't handle the initial state of the element correctly. The element.style.display is checked initially, but it always starts as "none" because the element is hidden by default in the HTML. This means the code always sets the display to "block" on the first click, and then back to "none" on the second click, effectively showing the element only once.

EXAMPLE 3- CSS Troubleshooting

Incorrect 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%;                        //Instead of width: 50%, use max-width: 100%
background-color: #f0f0f0;
padding: 20px;
        //Add text-align: center; to center the content within it, including the container.
}
</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>
.container {
margin: auto;
max-width:100%;
background-color: #f0f0f0;
padding: 20px;
text-align: center;
}
</style>
</head>
<body>
<div class="container">
<h1>Centered Container</h1>
<p>This container should be centered on the page.</p>
</div>
</body>
</html>
```

Explanation:- This CSS code has error that while margin: auto horizontally centers the element,it doesn't account for the initial width of the element itself. Since the container has a fixed width of 50%, it only centers within that 50% space, not the entire page.

EXERCISE 4- Debugging JavaScript Functions

Code:
```
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); // Should output 15
```

Explanation:- There is no error to debug and it is already performing correctly and efficiently.

EXERCISE 5- Debugging CSS Styling Issues

Incorrect 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:
```html
<!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: 95%;
height: 100px;
background-color: #A367DC;
color: black;
text-align: center;
line-height: 100px;
margin: 10px;
}
</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 :- To achieve the desired layout, this code implements strategic adjustments to rectify misalignment, enhance visual clarity with revised colors, and introduce spacing for improved readability, effectively resolving overlapping issues.