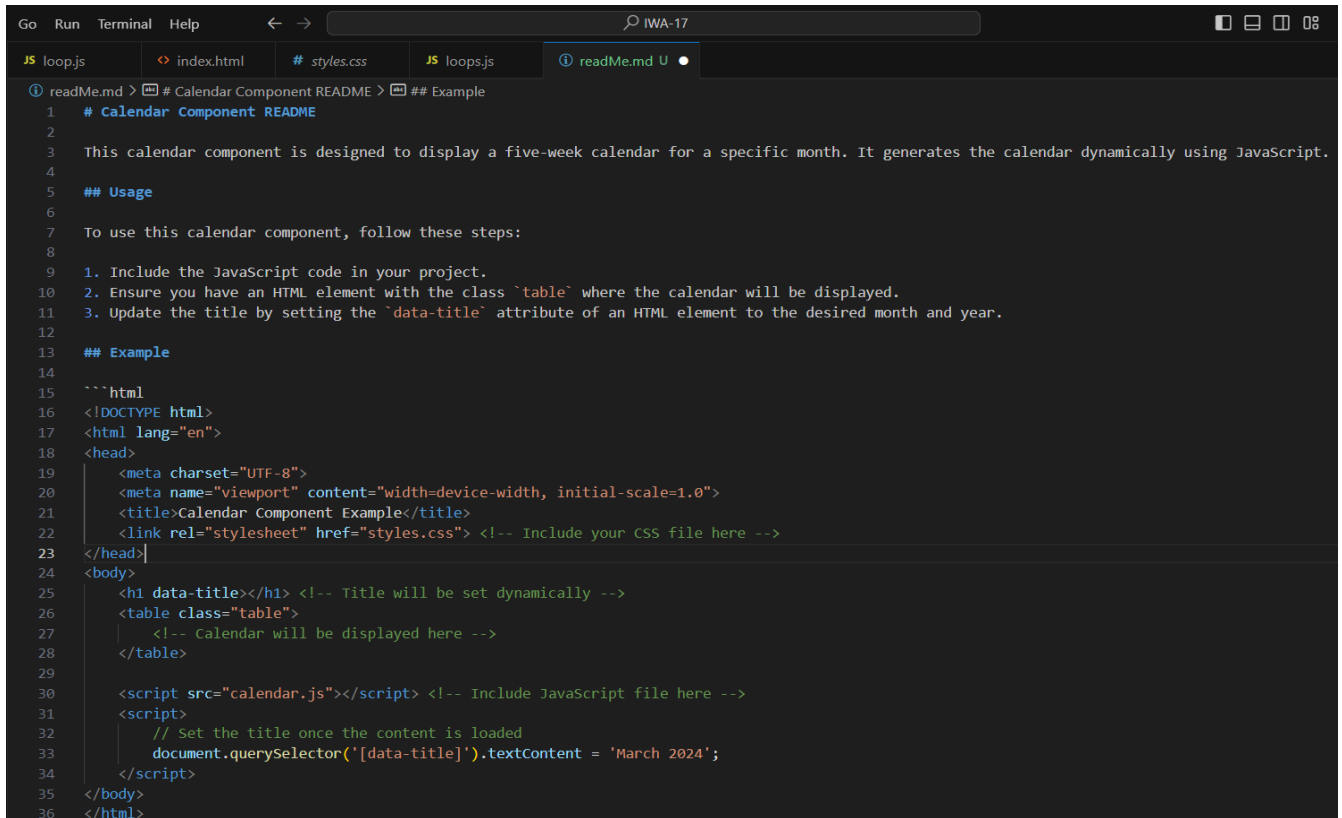


## DWA\_03.4 Knowledge Check\_DWA3.1

1. Please show how you applied a Markdown File to a piece of your code.



```
Go Run Terminal Help IWA-17
JS loop.js index.html styles.css JS loops.js readMe.md U
① readMe.md > # Calendar Component README > ## Example
1 # Calendar Component README
2
3 This calendar component is designed to display a five-week calendar for a specific month. It generates the calendar dynamically using JavaScript.
4
5 ## Usage
6
7 To use this calendar component, follow these steps:
8
9 1. Include the JavaScript code in your project.
10 2. Ensure you have an HTML element with the class `table` where the calendar will be displayed.
11 3. Update the title by setting the `data-title` attribute of an HTML element to the desired month and year.
12
13 ## Example
14
15 ```html
16 <!DOCTYPE html>
17 <html lang="en">
18 <head>
19   <meta charset="UTF-8">
20   <meta name="viewport" content="width=device-width, initial-scale=1.0">
21   <title>Calendar Component Example</title>
22   <link rel="stylesheet" href="styles.css"> <!-- Include your CSS file here -->
23 </head>
24 <body>
25   <h1 data-title></h1> <!-- Title will be set dynamically -->
26   <table class="table">
27     <!-- Calendar will be displayed here -->
28   </table>
29
30   <script src="calendar.js"></script> <!-- Include JavaScript file here -->
31   <script>
32     // Set the title once the content is loaded
33     document.querySelector('[data-title]').textContent = 'March 2024';
34   </script>
35 </body>
36 </html>
```

The provided JavaScript code does the following:

- Gets the current date and determines the last date of the current month.
- Creates a five-week calendar with each week represented as a row in an HTML table.
- Populates each cell with the corresponding day number.
- Highlights the current day.
- Sets the title of the calendar based on the provided month and year.

2. Please show how you applied JSDoc Comments to a piece of your code.

```

JS pract.js > ...
1  /**
2   * Adds two numbers.
3   *
4   * @param {number} a - The first number.
5   * @param {number} b - The second number.
6   * @returns {number} - The sum of a and b.
7   */
8  function addNumbers(a, b) {
9      return a + b;
10 };
11
12

```

---

3. Please show how you applied the `@ts-check` annotation to a piece of your code.

```

12
13 // @ts-check
14
15 /**
16  * Adds two numbers.
17  *
18  * @param {number} a - The first number.
19  * @param {number} b - The second number.
20  * @returns {number} The sum of a and b.
21  */
22 function addNumbers(a,
23     return a + b;
24 }
25 const c = addNumbers(3, "hello");
26 //

```

Argument of type 'string' is not assignable to parameter of type 'number'. ts(2345)

[View Problem \(Alt+F8\)](#) [Quick Fix... \(Ctrl+.\)](#)

---

4. As a BONUS, please show how you applied any other concept covered in the 'Documentation' module

```
27
28  /**
29   * @typedef {Object} User
30   * @property {string} firstName - The user's first name.
31   * @property {string} lastName - The user's last name.
32   * @property {number} age - The user's age.
33   * @property {string} email - The user's email address.
34   */
```

---