CSS Variables Explanation

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <title>Scoped CSS Variables and JS</title>
  <link rel="icon" href="https://fav.farm/" />

  <style>
    :root {
      --base: #ffc600;
      --spacing: 10px;
      --blur: 10px;
    }

    img {
      padding: var(--spacing);
      background-color: var(--base);
      filter: blur(var(--blur));
      max-width: -webkit-fill-available;
      max-height: 50vh;
      margin-top: 3%;
    }

    h2 {
      font-size: inherit;
      margin: auto auto 3%;
    }

    button {
      min-width: fit-content;
    }

    .hl {
      color: var(--base);
    }

    /*
          misc styles, nothing to do with CSS variables
        */

    body {
      text-align: center;
      background: #193549;
      color: white;
```

```
      font-family: 'helvetica neue', sans-serif;
      font-weight: 100;
      font-size: 50px;
    }

    .controls {
      margin-bottom: 3%;
      font-size: initial;
      display: flex;
      align-items: center;
      flex-direction: row;
      justify-content: center;
    }

    input {
      width: 100px;
      margin: 0 2% 0 1%;
    }

    label {
      font-size: initial;
      min-width: fit-content;
    }
  </style>
</head>

<body>
  <h2>Update CSS Variables with <span class='hl'>JS</span></h2>

  <div class="controls">
    <label for="spacing">Spacing:</label>
    <input id="spacing" type="range" name="spacing" min="10" max="200"
value="10" data-sizing="px">

    <label for="blur">Blur:</label>
    <input id="blur" type="range" name="blur" min="0" max="25"
value="10" data-sizing="px">

    <label for="base">Base Color</label>
    <input id="base" type="color" name="base" value="#ffc600">
  </div>

  <img src="https://source.unsplash.com/7bwQXzbF6KE/800x500">

  <script>
    const inputs = document.querySelectorAll('.controls input');

    function handleUpdate() {
      const suffix = this.dataset.sizing || '';
```

```
      document.documentElement.style.setProperty(`--${this.name}`,
this.value + suffix);
    }

    inputs.forEach(input => input.addEventListener('change',
handleUpdate));
    inputs.forEach(input => input.addEventListener('mousemove',
handleUpdate));

  </script>
```

Below is a brief explanation of how the CSS Variables are build:

1. Firstly, the provided code is an HTML document with embedded CSS and JavaScript. The HTML structure consists of a <head> section, where the document's metadata is defined, and a <body> section that contains the visible content.

2. Moving on to the CSS code, it defines various styles for the elements on the page. There are CSS variables defined using the -- prefix, such as --base, --spacing, and --blur, which hold specific values like colors and measurements. These variables can be dynamically updated using JavaScript later.

3. The CSS rules also define styles for the <img> element, setting padding, background color, and blur effect using the CSS variables. The <h2> heading and element with the class hl have their own styles, and there are some additional styles for other elements like <body>, .controls, and <input>.

4. Now the JavaScript code starts by selecting all elements with the class controls and input using the document.querySelectorAll() method. These elements are stored in the inputs variable.

5. Next, there's a function called handleUpdate() defined. This function is responsible for updating the CSS variables based on the input values. It retrieves the data-sizing attribute of the input (if available) and appends it to the value. Then, it sets the corresponding CSS variable value using document.documentElement.style.setProperty().

6. After defining the function, two event listeners are attached to each input element. The first one listens for the 'change' event, which triggers when the input value is changed. The second listener is for the 'mousemove' event, which triggers when the mouse pointer is moved while holding down the mouse button on the input. Both event listeners call the handleUpdate() function.

What I have learned

From this challenge, I have learned how to update CSS variables dynamically using JavaScript based on user input. By examining the provided code, I discovered that by adding event listeners to the input elements, I was able to respond to user interactions such as changes and mouse movements.