```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Variables Example</title>
    <style>
        /* Define CSS variables in :root */
        :root {
            --red: #b00;
            --blue: #4679bd;
            --grey: #ddd;
        }

        /* Apply CSS variables to styles */
        .Bx1 {
            color: var(--red);
            background: var(--grey);
            border: 1px solid var(--red);
            padding: 10px;
            text-align: center;
        }
    </style>
</head>
<body>
    <div class="Bx1">
        This is a box with custom CSS variables for color and background.
    </div>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Variables Example</title>
    <style>
        /* Define CSS variables in :root */
        :root {
            --W200: 200px;
            --W10: 10px;
        }

        /* Apply CSS variables to styles */
        .Bx2 {
            width: var(--W200);
            height: var(--W200);
            margin: var(--W10);
            background-color: lightgray;
            text-align: center;
            line-height: var(--W200);
        }
    </style>
</head>
<body>
    <div class="Bx2">
        This is a box with custom CSS variables for width, height, and margin.
    </div>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Variables Example</title>
    <style>
        /* Define default CSS variables in :root */
        :root {
            --color: green;
        }

        /* Apply CSS variables to styles */
        .button {
            padding: .5rem;
            border: 1px solid var(--color);
            color: var(--color);
            text-decoration: none;
            display: inline-block;
            margin: 10px;
        }

        /* Update CSS variable on hover */
        .button:hover {
            --color: blue;
        }

        /* Additional styling for red button */
        .button_red {
            --color: red;
        }
    </style>
</head>
<body>
    <a class="button">Button Green</a>
    <a class="button button_red">Button Red</a>
    <a class="button">Button Hovered On</a>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Variable Names Examples</title>
    <style>
        /* Invalid variable names */
        :root {
            /* Cannot start with a number */
            /* --123color: blue; */

            /* Cannot start with a special character */
            /* --#color: red; */

            /* Use hyphen or underscore for separation */
            /* --bg_color: yellow; */

            /* Use hyphen or underscore for separation */
            /* --$width: 100px; */
        }

        /* Valid variable names */
        :root {
            --color: red;
            --bg-color: yellow;
            --width: 100px;
        }

        /* Usage of valid variables */
        .element {
            color: var(--color);
            background-color: var(--bg-color);
            width: var(--width);
        }
    </style>
</head>
<body>
```

```
    <div class="element">Element with Valid CSS Variables</div>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Variable Names Examples</title>
    <style>
        /* Define CSS variables with different names */
        :root {
            --pcolor: ;
            --Pcolor: ;
            --pColor: ;
        }

        /* Usage of CSS variables with different names */
        .element-1 {
            color: var(--pcolor);
        }

        .element-2 {
            color: var(--Pcolor);
        }

        .element-3 {
            color: var(--pColor);
        }
    </style>
</head>
<body>
    <div class="element-1">Element with CSS Variable --pcolor</div>
    <div class="element-2">Element with CSS Variable --Pcolor</div>
    <div class="element-3">Element with CSS Variable --pColor</div>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Variable Usage Examples</title>
    <style>
        /* Invalid - Cannot be empty */
        :root {
            --color: ;
        }

        /* Valid - Space is assigned */
        :root {
            --color: ; /* space is assigned */
        }

        /* Usage of CSS variable */
        .element {
            color: var(--color);
        }
    </style>
</head>
<body>
    <div class="element">Element with CSS Variable</div>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Variable Usage with Width</title>
    <style>
        /* Invalid - Missing unit in variable assignment */
        :root {
            --width: 10;
        }

        /* Invalid - Mixing unit with variable usage */
        .invalid-width {
            width: var(--width)px;
        }

        /* Valid - Proper unit in variable assignment */
        :root {
            --width: 10px;
        }

        /* Valid - Using variable directly without mixing units */
        .valid-width {
            width: var(--width);
        }

        /* Valid - Using calc() to convert variable to a different unit */
        .valid-width-calc {
            width: calc(1px * var(--width)); /* multiply by 1 unit to convert */
        }
    </style>
</head>
<body>
    <div class="invalid-width">Invalid Width (Missing Unit)</div>
    <div class="valid-width">Valid Width (Using Variable)</div>
    <div class="valid-width-calc">Valid Width (Using calc() to Convert)</div>
</body>
</html>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>CSS Variables and Media Queries</title>
    <style>
        :root {
            --width: 25%;
            --content: 'This is desktop';
        }

        @media only screen and (max-width: 767px) {
            :root {
                --width: 50%;
                --content: 'This is mobile';
            }
        }

        @media only screen and (max-width: 480px) {
            :root {
                --width: 100%;
            }
        }

        div {
            width: calc(var(--width) - 20px);
            height: 100px;
        }

        div::before {
            content: var(--content);
        }

        /* Other Styles */
        body {
            padding: 10px;
        }

        div {
            display: flex;
            align-items: center;
            justify-content: center;
            font-weight: bold;
            float: left;
            margin: 10px;
            border: 4px solid black;
            background: red;
        }
    </style>
</head>
<body>
    <div></div>
    <div></div>
    <div></div>
    <div></div>
</body>
</html>
```