

Exercise01

Input

```
<> exercise1.html > html > head > style > body
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4  <meta charset="UTF-8">
5  <meta name="viewport" content="width=device-width, initial-scale=1.0">
6  <title>Number Counter App</title>
7  <style>
8      body {
9          border: 5px solid black;
10         padding-top: 130px;
11         text-align: center;
12         height: 500px;
13     }
14     .counter {
15         font-size: 30px;
16         text-align: center;
17         margin-bottom: 20px;
18         color: rgb(1, 13, 1);
19     }
20     }
21     button {
22
23         padding: 10px 20px;
24         font-size: 18px;
25         cursor: pointer;
26     }
27     }
28
29     .btn2{
30         background-color: rgb(221, 214, 214);
31         color: rgb(1, 4, 4);
32         border-radius: 8px;
33         text-align: center;
34     }
35     .btn1{
36         background-color: black;
37         color: azure;
```

```
37     color: azure;
38     border-radius: 8px;
39     text-align: center;
40 }
41 #counterValue{
42     font-size: 170px;
43     color: darkgreen;
44 }
45 }
46 </style>
47 </head>
48 <body>
49
50 <div class="counter" id="counterDisplay">
51     <h1>Counter</h1>
52     <span id="counterValue">550</span>
53 </div>
54 <button class="btn1" onclick="incrementCounter()">Increase</button>
55 <button class="btn2" onclick="resetCounter()">Reset</button>
56 <button class="btn2" onclick="decrementCounter()">Decrease</button>
57
58
59 <script>
60
61     class Counter {
62         constructor() {
63             this.value = 0;
64             this.updateDisplay();
65         }
66
67         increment() {
68             this.value++;
69             this.updateDisplay();
```

```

69         this.updateDisplay();
70     }
71
72     decrement() {
73         if (this.value > 0) {
74             this.value--;
75             this.updateDisplay();
76         }
77     }
78
79     reset() {
80         this.value = 0;
81         this.updateDisplay();
82     }
83
84     updateDisplay() {
85         document.getElementById('counterValue').innerText = this.value;
86     }
87 }
88
89
90 const counter = new Counter();
91
92 function incrementCounter() {
93     counter.increment();
94 }
95
96 function decrementCounter() {
97     counter.decrement();
98 }
99
100

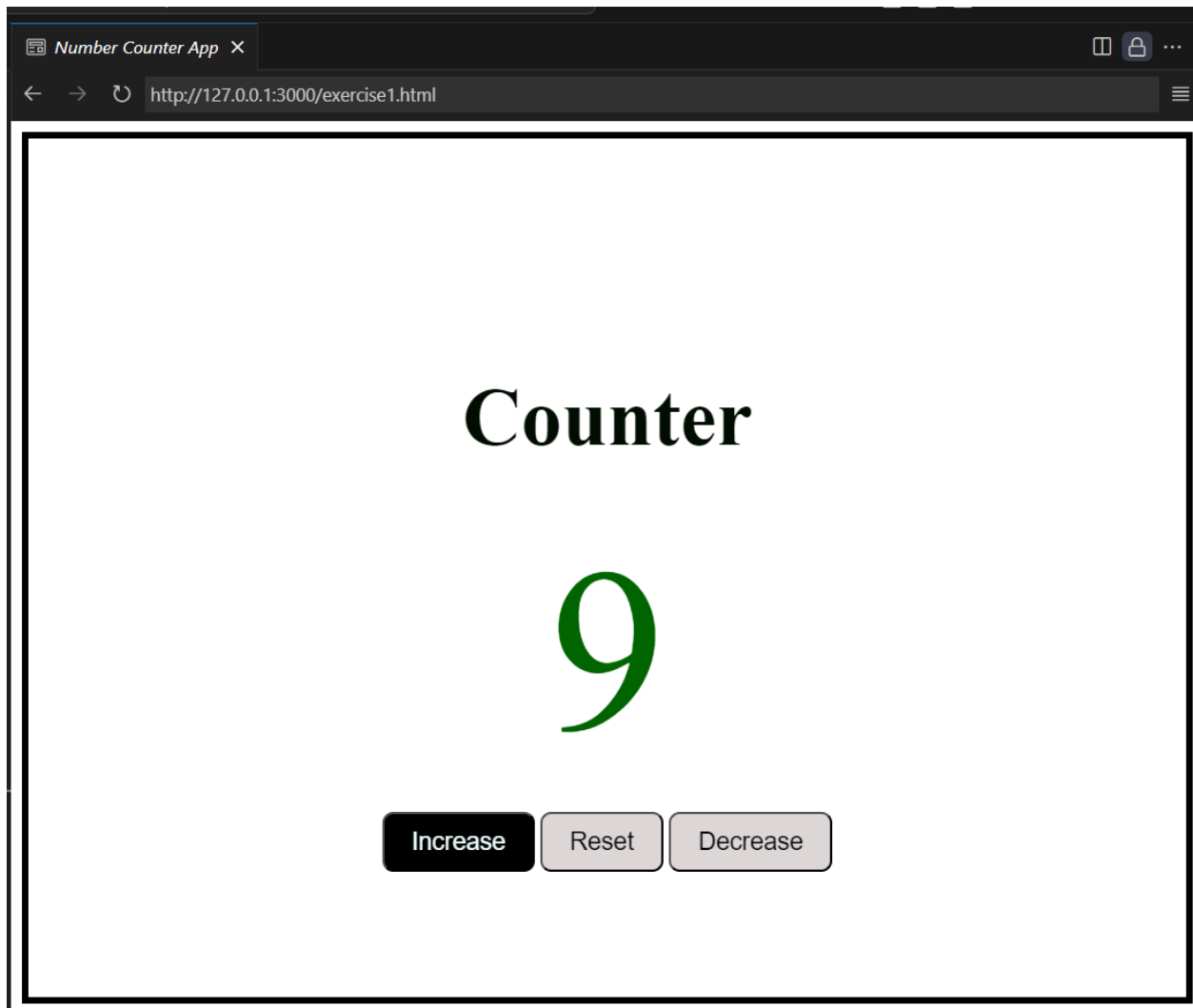
```

```

99     }
100
101     function resetCounter() {
102         counter.reset();
103     }
104 </script>
105
106 </body>
107 </html>
108

```

OUTPUT



Exercise02

Input

```
<> exercise2.html > html > head > style > .btn1
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Color Random Generate</title>
7    <style>
8      .btn1 {
9        height: 50px;
10       width: 150px;
11       border: 2px whitesmoke solid;
12       border-radius: 10px;
13       margin-top: 20px;
14       margin-bottom: 20px;
15       margin-left: 15px;
16       font-size: 20px;
17     }
18   }
19   .btn2 {
20     height: 50px;
21     width: 120px;
22     border: 2px whitesmoke solid;;
23     border-radius: 10px;
24     margin-top: 20px;
25     margin-bottom: 20px;
26     margin-left: 9px;
27     font-size: 20px;
28   }
29   }
30   }
31   .record {
32     background-color: aliceblue;
33     font-size: 20px;
34     color: black bold;
35     width: 390px;
36     height: 250px;
37     border-radius: 5px;
```

```

38     overflow-y: auto;
39     padding: 10px;
40     text-align: left;
41 }
42 .heading {
43     font-size: 20px;
44     background-color: #rgb(234, 225, 225);
45     width: 449px;
46     height: 35px;
47     margin-top: 20px;
48     padding-top: 20px;
49 }
50 #colorBox {
51     background-color: #rgb(210, 194, 226);
52     width: 390px;
53     height: 150px;
54     border-radius: 5px;
55     margin-top: 10px;
56     font-size: larger;
57     display: flex;
58     justify-content: center;
59     align-items: center;
60 }
61 </style>
62 </head>
63 <body>
64     <div align="center" style="background-color: #white; width: 500px; height: 800px; border: 5px solid #black;">
65         <div align="start" class="heading">Color Random Generate</div>
66         <div style="background-color: #pink; width: 449px; height: 700px; border: 1px solid #white;">
67             <div id="colorBox">
68                 <h1>Background Color:</h1>
69             </div>

```

```

70         <button id="generateButton" class="btn1">Generate</button>
71         <button id="resetButton" class="btn2">Reset</button>
72         <div id="record" class="record">Record History</div>
73     </div>
74 </div>
75 <script>
76     class ColorGenerator {
77     constructor() {
78         this.colorBox = document.getElementById('colorBox');
79         this.generateButton = document.getElementById('generateButton');
80         this.generateButton.addEventListener('click', this.generate.bind(this));
81         this.resetButton = document.getElementById('resetButton');
82         this.resetButton.addEventListener('click', this.reset.bind(this));
83         this.record = document.getElementById('record');
84         this.serial = 1;
85     }
86
87     generate() {
88         let randomColor = this.getRandomColor();
89         this.colorBox.innerHTML = `<h1>Background Color: ${randomColor}</h1>`;
90         this.colorBox.style.backgroundColor = randomColor;
91         this.addToRecord(randomColor);
92     }
93
94     getRandomColor() {
95         let letters = '0123456789ABCDEF';
96         let color = '#';
97         for (let i = 0; i < 6; i++) {
98             color += letters[Math.floor(Math.random() * 16)];
99         }
100         return color;
101     }
102

```

```
101     }
102
103     addToRecord(color) {
104         let newItem = document.createElement('div');
105         newItem.textContent = `Color ${this.serial++}: ${color}`;
106         newItem.style.marginBottom = '5px';
107         this.record.appendChild(newItem);
108     }
109
110     reset() {
111         this.colorBox.innerHTML = `<h1>Background Color:</h1>`;
112         this.colorBox.style.backgroundColor = '';
113         this.record.innerHTML = 'Record';
114         this.serial = 1;
115     }
116 }
117
118 const colorGenerator = new ColorGenerator();
119 </script>
120 </body>
121 </html>
122
```

OUTPUT

Color Random Generate

Background Color:

Generate

Reset

Record History

Color Random Generate

Background Color:
#E24FB7

Generate

Reset

Record History

Color 1: #2D2234

Color 2: #806DFF

Color 3: #83C831

Color 4: #9DE3D0

Color 5: #07CDF1

Color 6: #FF40E3

Color 7: #A956C7

Color 8: #7B627E

Color 9: #D65312

