

Lab 5_3.html

```
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="description" content="Data Visualization"/>
6      <meta name="keywords" content="HTML, CSS, D3"/>
7      <meta name="author" content="Dai Vy"/>
8      <title>Task 5.3 D3 Adding and Removing Data</title>
9
10     <script src="https://d3js.org/d3.v7.min.js"></script>
11     <style>
12
13     </style>
14 </head>
15 <body>
16
17     <button id="add">Add</button>
18     <button id="remove">Remove</button>
19     <h1>LAB 5.3 D3 Adding and Removing Data</h1>
20
21     <script>
22         var w = 500;
23         var h = 100;
24         var maxValue = 25;
25         var dataset = [24, 10, 29, 19, 8, 15, 20, 12, 9, 6, 21, 28];
26         var duration = 2000;
27
28         // Use scaleBand() to create an ordinal scaleable x-axis based on the range of the
data set.
29         var xScale = d3.scaleBand()
30             .domain(d3.range(dataset.length))
31             .rangeRound([0, w])
32             .paddingInner(0.05);
33
34         // Use scaleLinear() to create a linear scaleable y-axis based on the range of the
data set.
35         var yScale = d3.scaleLinear()
36             .domain([0, d3.max(dataset)])
37             .range([0, h]);
38
39         var svg = d3.select("body")
40             .append("svg")
41             .attr("width", w)
42             .attr("height", h);
43
44         // Draw initial bars
45         svg.selectAll("rect")
46             .data(dataset)
```

```
47     .enter()
48     .append("rect")
49     .attr("x", function(d, i) {
50         return xScale(i);
51     })
52     .attr("y", function(d) {
53         return h - yScale(d);
54     })
55     .attr("width", xScale.bandwidth())
56     .attr("height", function(d) {
57         return yScale(d);
58     })
59     .attr("fill", "steelblue");
60
61 // Draw initial labels
62 svg.selectAll("text")
63     .data(dataset)
64     .enter()
65     .append("text")
66     .text(function(d) {
67         return d; // This displays the number on each bar
68     })
69     .attr("x", function(d, i) {
70         return xScale(i) + xScale.bandwidth() / 2;
71     })
72     .attr("y", function(d) {
73         return h - yScale(d) + 14;
74     })
75     .attr("text-anchor", "middle")
76     .attr("fill", "white")
77     .attr("font-size", "12px");
78
79 // Add bar function
80 function addBar() {
81     var newValue = Math.floor(Math.random() * maxValue);
82     dataset.push(newValue);
83     xScale.domain(d3.range(dataset.length));
84     yScale.domain([0, d3.max(dataset)]);
85     var delayStep = duration / dataset.length;
86
87     // Bars
88     var bars = svg.selectAll("rect")
89         .data(dataset, function(d, i) { return i; });
90
91     // ENTER new bar
92     var barsEnter = bars.enter()
93         .append("rect")
94         .attr("x", w) // start off right edge
95         .attr("y", function(d) { return h - yScale(d); })
96         .attr("width", xScale.bandwidth());
```

```
97     .attr("height", function(d) { return yScale(d); })
98     .attr("fill", "steelblue");
99
100 // MERGE and transition all bars
101 bars.merge(barsEnter)
102     .transition()
103     .duration(duration)
104     .delay(function(d, i) { return i * delayStep; })
105     .attr("x", function(d, i) { return xScale(i); })
106     .attr("y", function(d) { return h - yScale(d); })
107     .attr("width", xScale.bandwidth())
108     .attr("height", function(d) { return yScale(d); });
109
110 // Labels
111 var labels = svg.selectAll("text")
112     .data(dataset, function(d, i) { return i; });
113
114 var labelsEnter = labels.enter()
115     .append("text")
116     .text(function(d) { return d; })
117     .attr("x", w + xScale.bandwidth() / 2)
118     .attr("y", function(d) { return h - yScale(d) + 14; })
119     .attr("text-anchor", "middle")
120     .attr("fill", "white")
121     .attr("font-size", "12px");
122
123 labels.merge(labelsEnter)
124     .transition()
125     .duration(duration)
126     .delay(function(d, i) { return i * delayStep; })
127     .text(function(d) { return d; })
128     .attr("x", function(d, i) { return xScale(i) + xScale.bandwidth() / 2; })
129     .attr("y", function(d) { return h - yScale(d) + 14; });
130 }
131
132 // Remove bar function
133 function removeBar() {
134     if (dataset.length === 0) return;
135     dataset.shift();
136     xScale.domain(d3.range(dataset.length));
137     yScale.domain([0, d3.max(dataset)]);
138     var delayStep = duration / (dataset.length || 1);
139
140 // Bars
141 var bars = svg.selectAll("rect")
142     .data(dataset, function(d, i) { return i; });
143
144 // EXIT first bar
145 bars.exit()
146     .transition()
```

```
147         .duration(duration)
148         .attr("x", w)
149         .remove();
150
151     // UPDATE remaining bars
152     bars.transition()
153         .duration(duration)
154         .delay(function(d, i) { return i * delayStep; })
155         .attr("x", function(d, i) { return xScale(i); })
156         .attr("y", function(d) { return h - yScale(d); })
157         .attr("width", xScale.bandwidth())
158         .attr("height", function(d) { return yScale(d); });
159
160     // Labels
161     var labels = svg.selectAll("text")
162         .data(dataset, function(d, i) { return i; });
163
164     labels.exit()
165         .transition()
166         .duration(duration)
167         .attr("x", w)
168         .remove();
169
170     labels.transition()
171         .duration(duration)
172         .delay(function(d, i) { return i * delayStep; })
173         .text(function(d) { return d; })
174         .attr("x", function(d, i) { return xScale(i) + xScale.bandwidth() / 2; })
175         .attr("y", function(d) { return h - yScale(d) + 14; });
176 }
177
178 // Button event listeners
179 d3.select("#add").on("click", addBar);
180 d3.select("#remove").on("click", removeBar);
181 </script>
182
183 <br>
184
185 <footer style="color: grey">COS30045 Data Visualization<br>
186     Dai Vy
187 </footer>
188 </body>
189 </html>
```