

Lab 6_1.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 6.1 D3 Interactivity - Mouse Over Effects</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 6.1 D3 Interactivity - Mouse Over Effects</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     // Mouseover effect
61     .on("mouseover", function(event, d) {
62         d3.select(this)
63             .transition()
64             .duration(200)
65             .attr("fill", "orange");
66         // SVG tooltip
67         svg.append("text")
68             .attr("class", "svg-tooltip")
69             .attr("x", +d3.select(this).attr("x") + xScale.bandwidth()/2)
70             .attr("y", +d3.select(this).attr("y") - 8)
71             .attr("text-anchor", "middle")
72             .attr("font-size", "14px")
73             .attr("font-weight", "bold")
74             .attr("fill", "black")
75             .text(d);
76     })
77     .on("mouseout", function(event, d) {
78         d3.select(this)
79             .transition()
80             .duration(200)
81             .attr("fill", "steelblue");
82         svg.selectAll(".svg-tooltip").remove();
83     });
84
85     // Draw initial labels
86     svg.selectAll("text")
87         .data(dataset)
88         .enter()
89         .append("text")
90         .text(function(d) {
91             return d; // This displays the number on each bar
92         })
93         .attr("x", function(d, i) {
94             return xScale(i) + xScale.bandwidth() / 2;
95         })
96         .attr("y", function(d) {
```

```
97         return h - yScale(d) + 14;
98     })
99     .attr("text-anchor", "middle")
100     .attr("fill", "white")
101     .attr("font-size", "12px");
102
103 // Add bar function
104 function addBar() {
105     var newValue = Math.floor(Math.random() * maxValue);
106     dataset.push(newValue);
107     xScale.domain(d3.range(dataset.length));
108     yScale.domain([0, d3.max(dataset)]);
109     var delayStep = duration / dataset.length;
110
111     // Bars
112     var bars = svg.selectAll("rect")
113         .data(dataset, function(d, i) { return i; });
114
115     // ENTER new bar
116     var barsEnter = bars.enter()
117         .append("rect")
118         .attr("x", w) // start off right edge
119         .attr("y", function(d) { return h - yScale(d); })
120         .attr("width", xScale.bandwidth())
121         .attr("height", function(d) { return yScale(d); })
122         .attr("fill", "steelblue");
123     // Mouseover effect for new bars
124     .on("mouseover", function(event, d) {
125         d3.select(this)
126             .transition()
127             .duration(200)
128             .attr("fill", "orange");
129         svg.append("text")
130             .attr("class", "svg-tooltip")
131             .attr("x", +d3.select(this).attr("x") + xScale.bandwidth()/2)
132             .attr("y", +d3.select(this).attr("y") - 8)
133             .attr("text-anchor", "middle")
134             .attr("font-size", "14px")
135             .attr("font-weight", "bold")
136             .attr("fill", "black")
137             .text(d);
138     })
139     .on("mouseout", function(event, d) {
140         d3.select(this)
141             .transition()
142             .duration(200)
143             .attr("fill", "steelblue");
144         svg.selectAll(".svg-tooltip").remove();
145     });
146 }
```

```
147 // MERGE and transition all bars
148 bars.merge(barsEnter)
149   .transition()
150   .duration(duration)
151   .delay(function(d, i) { return i * delayStep; })
152   .attr("x", function(d, i) { return xScale(i); })
153   .attr("y", function(d) { return h - yScale(d); })
154   .attr("width", xScale.bandwidth())
155   .attr("height", function(d) { return yScale(d); });
156
157 // Labels
158 var labels = svg.selectAll("text")
159   .data(dataset, function(d, i) { return i; });
160
161 var labelsEnter = labels.enter()
162   .append("text")
163   .text(function(d) { return d; })
164   .attr("x", w + xScale.bandwidth() / 2)
165   .attr("y", function(d) { return h - yScale(d) + 14; })
166   .attr("text-anchor", "middle")
167   .attr("fill", "white")
168   .attr("font-size", "12px");
169
170 labels.merge(labelsEnter)
171   .transition()
172   .duration(duration)
173   .delay(function(d, i) { return i * delayStep; })
174   .text(function(d) { return d; })
175   .attr("x", function(d, i) { return xScale(i) + xScale.bandwidth() / 2; })
176   .attr("y", function(d) { return h - yScale(d) + 14; });
177 }
178
179 // Remove bar function
180 function removeBar() {
181   if (dataset.length === 0) return;
182   dataset.shift();
183   xScale.domain(d3.range(dataset.length));
184   yScale.domain([0, d3.max(dataset)]);
185   var delayStep = duration / (dataset.length || 1);
186
187   // Bars
188   var bars = svg.selectAll("rect")
189     .data(dataset, function(d, i) { return i; });
190
191   // EXIT first bar
192   bars.exit()
193     .transition()
194     .duration(duration)
195     .attr("x", w)
196     .remove();
```

```
197
198 // UPDATE remaining bars
199 bars.transition()
200   .duration(duration)
201   .delay(function(d, i) { return i * delayStep; })
202   .attr("x", function(d, i) { return xScale(i); })
203   .attr("y", function(d) { return h - yScale(d); })
204   .attr("width", xScale.bandwidth())
205   .attr("height", function(d) { return yScale(d); });
206
207 // Re-apply mouseover/mouseout to updated bars
208 bars.on("mouseover", function(event, d) {
209   d3.select(this)
210     .transition()
211     .duration(200)
212     .attr("fill", "orange");
213   svg.append("text")
214     .attr("class", "svg-tooltip")
215     .attr("x", +d3.select(this).attr("x") + xScale.bandwidth()/2)
216     .attr("y", +d3.select(this).attr("y") - 8)
217     .attr("text-anchor", "middle")
218     .attr("font-size", "14px")
219     .attr("font-weight", "bold")
220     .attr("fill", "black")
221     .text(d);
222 })
223 .on("mouseout", function(event, d) {
224   d3.select(this)
225     .transition()
226     .duration(200)
227     .attr("fill", "steelblue");
228   svg.selectAll(".svg-tooltip").remove();
229 });
230
231 // Labels
232 var labels = svg.selectAll("text")
233   .data(dataset, function(d, i) { return i; });
234
235 labels.exit()
236   .transition()
237   .duration(duration)
238   .attr("x", w)
239   .remove();
240
241 labels.transition()
242   .duration(duration)
243   .delay(function(d, i) { return i * delayStep; })
244   .text(function(d) { return d; })
245   .attr("x", function(d, i) { return xScale(i) + xScale.bandwidth() / 2; })
246   .attr("y", function(d) { return h - yScale(d) + 14; });
```

```
247     }
248
249     // Button event listeners
250     d3.select("#add").on("click", addBar);
251     d3.select("#remove").on("click", removeBar);
252 </script>
253
254 <br>
255
256 <footer style="color: grey">COS30045 Data Visualization<br>
257     Dai Vy
258 </footer>
259 </body>
260 </html>
```