

Name: Nguyen Khanh Toan

Student ID: 104180605

## Lab 5.1-6.2

### Lab 5.1:

```
1 <!DOCTYPE html>
2 <html lang = "en">
3 <head>
4     <meta charset = "utf-8" />
5     <meta name = "descriptio    content =
  "Data Visualisatio" />
6 n" <meta name = "keyword        content = "HTML,CSS,D    />
7     <meta name = "author"      content = "Toan Nguye    />
8                                     n"
9     <title> Task 5.1 Drawing with dat </title>
10         a
11     <script src = "https://d3js.org/d3.v7.min.j    ></script
12 </head>                                s"                >
13
14 <body>
15     <h1> The D3 Journey Start Here </h1>
16     <button id="update">Update</button>
17     <br>
18     <script src="scripts.j    ></script
19                                     s"                >
20     <br>
21     <bf>
22     <footer style = "color:grey">
  COS30045 Data Visualisatio <br>
23 n    Toan Nguye </footer>
24 n</body>
25
26 </html>
```

Fig 1: 5.1 HTML code

```

1
2 var w = 500;
3 var h = 100;
4 // padding = 3
5 5
6
7 var dataset = [14,5,26,23,9,12,28,22,16,21,25];
8
9
10 //xScale and yScale are for scaling the value of dataset to match wit
11 h
12 //the xScale in width for categorical values
13 var xScale = d3.scaleBand()
14   .domain(d3.range(dataset.length))
15   .rangeRound ([0,w])
16   .paddingInne (0.05);
17   r
18 //the yScale in height for numerical value
19 var yScale = d3.scaleLinear()
20   .domain([0,d3.max(dataset, function (d){
21     return d;
22   })))
23   .rangeRound ([ h , 0]);
24   d
25 // Selecting the body to draw the char
26 var svg1 = d3.select("body")
27   .append("svg")
28   .attr("height",h)
29   .attr("width", w);
30
31 // Draw the bar chart with xScale width and yScale heigh
32 svg1.selectAll("rect")
33   .data(dataset)
34   .enter()
35   .append("rect")
36   .attr("x", function(d,i) {
37     return xScale(i);
38   })
39   .attr("y", function(d,i) {
40     return yScale(d);
41   })
42   .attr("width",xScale.bandwidth())
43   .attr("height", function(d,i){
44     return h-yScale(d);
45   })
46   .attr("fill", function(d){
47     return "rgb(0,0, " + Math.round(d*10) + ")";
48   });
49
50 // Set the random value for update butto
51 d3.select("#update")
52   .on("click", function(){
53     alert("Button clicke ");
54     d"
55     var numValues = dataset.length;
56     var maxValues = 25;
57     dataset = [];
58
59     for (var i = 0 ; i < numValues; i++) {
60       var newNumber = Math.floor(Math.random() * maxValues);
61       dataset.push(newNumber);
62     }
63
64 // Update the bars with new dat
65 avg1.selectAll("rect")
66   .data(dataset)
67   .attr("y", function(d,i) {
68     return yScale(d);
69   })
70   .attr("height", function(d,i){
71     return h-yScale(d);
72   })
73   .attr("fill", function(d){
74     return "rgb(0,0, " + Math.round(d*10) + ")";
75   });
76 });
77
78

```

Fig 2: 5.2 JS code

## Lab 5.2:

```
1 <!DOCTYPE html>
2 <html lang = "en">
3 <head>
4   <meta charset = "utf-8" />
5   <meta name = "descriptio   content = "Data Visualisatio   />
6   <meta name = "keyword   content = "HTML,CSS,D   />
7   <meta name = "author"   content = "Toan Nguye   />
8   n"
9   <title> Task 5.2 Drawing with dat </title>
10  a
11  <script src = "https://d3js.org/d3.v7.min.j   ></script
12 </head>   s"   >
13
14 <body>
15   <h1> The D3 Journey Start Here </h1>
16   <button id="update">Update</button>
17   <select id="transitionTyp   >
18     <option value="easeLinea   >Linear</option>
19     <option value="easeExp">Ease Exp</option>
20     <option value="easeElasti   >Elastic</option>
21     <option value="easeBounc   >Bounce</option>
22   </select>   e"
23   <br>
24   <script src="scripts.j   ></script
25   s"   >
26   <br>
27   <bf>
28   <footer style = "color:grey"> COS30045 Data Visualisatio <br>
29   Toan Nguye </footer>   n
30 </body>
31
32 </html>
```

Fig 3: 5.2 HTML code

```

1
2   var w = 500;
3   var h = 100;
4   // padding = 3
5   5
6
7   var dataset = [14,5,26,23,9,12,28,22,16,21,25];
8
9
10  //xScale and yScale are for scaling the value of dataset to match wit
11  h
12  //the xScale in width for categorical values
13  var xScale = d3.scaleBand()
14    .domain(d3.range(dataset.length))
15    .rangeRound ([0,w])
16    .paddingInne (0.05);
17
18  //the yScale in height for numerical value
19  var yScale = d3.scaleLinear()
20    .domain([0,d3.max(dataset, function (d){
21      return d;
22    })])
23    .rangeRound ([ h , 0]);
24
25  // Selecting the body to draw the char
26  var svg1 = d3.select("body")
27    .append("svg")
28    .attr("height",h)
29    .attr("width", w);
30
31  // Draw the bar chart with xScale width and yScale height
32  svg1.selectAll("rect")
33    .data(dataset)
34    .enter()
35    .append("rect")
36    .attr("x", function(d,i) {
37      return xScale(i);
38    })
39    .attr("y", function(d,i) {
40      return yScale(d);
41    })
42    .attr("width",xScale.bandwidth())
43    .attr("height", function(d,i){
44      return h-yScale(d);
45    })
46    .attr("fill", function(d){
47      return "rgb(0,0, " + Math.round(d*10) + ")";
48    });
49
50  // Set the random value for update button
51  d3.select("#update")
52    .on("click", function(){
53      alert("Button clicked");
54      d
55      var transitionTyp = document.getElementById('transitionTyp').value;
56      var numValues = dataset.length;
57      var maxValues = 25;
58
59      dataset = [];
60
61      for (var i = 0 ; i < numValues; i++) {
62        var newNumber = Math.floor(Math.random() * maxValues);
63        dataset.push(newNumber);
64      }
65
66      // Update the bars with new data
67      svg1.selectAll("rect")
68        .data(dataset)
69        .transition()
70          .duration(1000)
71          .delay(function(d,i){
72            return i/dataset.length*100;
73            // return i*10
74            0
75          })
76        .ease(d3.easeCubicInOut)
77        // .ease(d3.easeCircleIn
78        // .ease(d3.easeCircleOut
79        // .ease(d3.easeElasticOut
80        // .ease(d3.easeBounce
81        // .ease(d3.easeLinear
82        )
83        .attr("y", function(d,i) {
84          return yScale(d);
85        })
86        .attr("height", function(d,i){
87          return h-yScale(d);
88        })
89        .attr("fill", function(d){
90          return "rgb(0,0, " + Math.round(d*10) + ")";
91        })
92        .ease(d3[transitionTyp]);
93      });
94
95
96

```

Fig 4: 5.2 JS code

## Lab 5.3:

```
1 <!DOCTYPE html>
2 <html lang = "en">
3 <head>
4   <meta charset = "utf-8" />
5   <meta name = "descriptio" content = "Data Visualisatio" />
6   <meta name = "keyword" content = "HTML,CSS,D" />
7   <meta name = "author" content = "Toan Nguye" />
8   <title> Task 5.3 Drawing with dat </title>
9   <script src = "https://d3js.org/d3.v7.min.js" /></script>
10 </head>
11 <body>
12   <h1> The D3 Journey Start Here </h1>
13   <button id="update">Update</button>
14   <select id="transitionType">
15     <option value="easeLinear">Linear</option>
16     <option value="easeExp">Ease Exp</option>
17     <option value="easeElastic">Elastic</option>
18     <option value="easeBounce">Bounce</option>
19   </select>
20   <button id="add">Add</button>
21   <button id="remove">Remove</button>
22   <br>
23   <script src="scripts.js" /></script>
24   <br>
25   <bf>
26   <footer style = "color:grey"> COS30045 Data Visualisatio <br>
27   Toan Nguye </footer>
28 </body>
29 </html>
```

Fig 5: 5.3 HTML code

```

1
2 var w = 500;
3 var h = 100;
4 // padding = 3
5
6
7 var dataset = [14,5,26,23,9,12,28,22,16,21,25];
8
9
10 //xScale and yScale are for scaling the value of dataset to match wit
11 h
12 //the xScale in width for categorical values
13 var xScale = d3.scaleBand()
14   .domain(d3.range(dataset.length))
15   .rangeRound([0,w])
16   .paddingInner(0.05);
17
18 //the yScale in height for numerical value
19 var yScale = d3.scaleLinear()
20   .domain([0,d3.max(dataset, function(d){
21     return d;
22   })])
23   .rangeRound([0,h]);
24
25 // Selecting the body to draw the char
26 var svg1 = d3.select("body")
27   .append("svg")
28   .attr("height",h)
29   .attr("width",w);
30
31
32 // Draw the bar chart with xScale width and yScale heigh
33 svg1.selectAll("rect")
34   .data(dataset)
35   .enter()
36   .append("rect")
37   .attr("x", function(d,i) {
38     return xScale(i);
39   })
40   .attr("y", function(d,i) {
41     return yScale(d);
42   })
43   .attr("width", xScale.bandwidth())
44   .attr("height", function(d,i){
45     return h - yScale(d);
46   })
47   .attr("fill", function(d){
48     return "rgb(0,0, " + Math.round(d*10) + ")";
49   });
50
51 // Set the random value for update button
52 d3.select("#update")
53   .on("click", function(){
54     alert("Button clicked");
55     var transitionType = document.getElementById("transitionType").value;
56     var numValues = dataset.length;
57     var maxValues = 25;
58     dataset = [];
59     for (var i = 0; i < numValues; i++) {
60       var randomNumber = Math.floor(Math.random() * maxValues);
61       dataset.push(randomNumber);
62     }
63
64     // Update the bars with new dat
65     svg1.selectAll("rect")
66       .data(dataset)
67       .transition()
68         .duration(1000)
69         .delay(function(d,i){
70           return i/dataset.length*100;
71         })
72         .attr("fill", function(d){
73           return "rgb(0,0, " + Math.round(d*10) + ")";
74         })
75         .attr("y", function(d,i) {
76           return yScale(d);
77         })
78         .attr("height", function(d,i){
79           return h - yScale(d);
80         })
81         .attr("fill", function(d){
82           return "rgb(0,0, " + Math.round(d*10) + ")";
83         })
84         .ease(d3.transitionType);
85
86     d3.select("#add")
87       .on("click", function(){
88         // alert("Button Add clicked");
89         var maxValues = 25;
90         var randomNumber = Math.floor(Math.random() * maxValues);
91         dataset.push(randomNumber);
92         xScale.domain(d3.range(dataset.length));
93
94         var bars = svg1.selectAll("rect")
95           .data(dataset);
96
97         bars.enter()
98           .append("rect")
99           .attr("x", w)
100           .attr("y", function(d,i){
101             return h - yScale(d);
102           })
103           .attr("fill", function(d){
104             return "rgb(0,0, " + Math.round(d*10) + ")";
105           })
106           .merge(bars) // Merge the enter selection with the existing bar
107           .transition()
108             .duration(500)
109             .attr("x", function(d,i){
110               return xScale(i);
111             })
112             .attr("y", function(d,i){
113               return yScale(d);
114             })
115             .attr("width", xScale.bandwidth())
116             .attr("height", function(d,i){
117               return h - yScale(d);
118             })
119             .attr("fill", function(d){
120               return "rgb(0,0, " + Math.round(d*10) + ")";
121             })
122           .ease(d3.transitionType);
123
124     d3.select("#remove")
125       .on("click", function(){
126         // alert("Button Remove clicked");
127         d3.selectAll("rect")
128           .transition()
129             .duration(500)
130             .attr("x",w)
131             .remove();
132
133         var bars = svg1.selectAll("rect")
134           .data(dataset);
135         bars.exit()
136           .transition()
137             .duration(500)
138             .attr("x",w)
139             .remove();
140
141         bars.transition()
142           .duration(500)
143           .attr("x",function(d,i){
144             return xScale(i);
145           })
146           .attr("width", xScale.bandwidth())
147
148
149
150

```

Fig 6: 5.3 JS code

## Lab 6.1:

```
1 <!DOCTYPE html>
2 <html lang = "en">
3 <head>
4   <meta charset = "utf-8" />
5   <meta name = "descriptio content = "Data Visualisatio />
6   <meta name = "keyword content = "HTML,CSS,D />
7   <meta name = "author content = "Toan Nguye />
8     n"
9   <title> Task 6.1 Drawing with dat </title>
10   a
11   <script src = "https://d3js.org/d3.v7.min.js" ></script>
12 </head>
13
14 <body>
15   <h1> The D3 Journey Start Here </h1>
16   <button id="update">Update</button>
17   <select id="transitionType" >
18     <option value="easeLinear">Linear</option>
19     <option value="easeExp">Ease Exp</option>
20     <option value="easeElastic">Elastic</option>
21     <option value="easeBounce">Bounce</option>
22   </select>
23   <button id="add">Add</button>
24   <button id="remove">Remove</button>
25   <br>
26   <script src="scripts.js" ></script>
27     s"
28
29   <br>
30   <bf>
31   <footer style = "color:grey"> COS30045 Data Visualisatio <br>
32   Toan Nguye </footer> n
33 </body>
34
35 </html>
```

Fig 7: 6.1 HTML code

[illegible]



Fig 8: 6.1 JS code

## Lab 6.2:

```
1 <!DOCTYPE html>
2 <html lang = "en">
3 <head>
4   <meta charset = "utf-8" />
5   <meta name = "descriptio" content = "Data Visualisatio" />
6   <meta name = "keyword" content = "HTML,CSS,D" />
7   <meta name = "author" content = "Toan Nguye" />
8   <title> Task 6.1 Drawing with dat </title>
9   <script src = "https://d3js.org/d3.v7.min.js" />
10 </head>
11 <body>
12   <h1> The D3 Journey Start Here </h1>
13   <button id="update">Update</button>
14   <select id="transitionType">
15     <option value="easeLinear">Linear</option>
16     <option value="easeExp">Ease Exp</option>
17     <option value="easeElastic">Elastic</option>
18     <option value="easeBounce">Bounce</option>
19   </select>
20   <button id="add">Add</button>
21   <button id="remove">Remove</button>
22   <button id="sortAsc">Sort ASC</button>
23   <button id="sortDsc">Sort DSC</button>
24   <br>
25   <script src="scripts.js" />
26   <br>
27   <div>
28     <div>
29       <div>
30         <div>
31           <div>
32             <div>
33               <div>
34                 <div>
35                   <div>
36                     <div>
37                       <div>
```

Fig 9: 6.2 HTML code

```

1
2 var w = 500;
3 var h = 100;
4 // padding = 3
5
6 var dataset = [14,5,26,23,9,12,28,22,16,21,25];
7
8
9
10 //xScale and yScale are for scaling the value of dataset to match wit
11 h
12 //the xScale in width for categorical values
13 var xScale = d3.scaleBand()
14   .domain(d3.range(dataset.length))
15   .rangeRound ([0,w])
16   .paddingInner (0.05);
17
18 //the yScale in height for numerical value
19 var yScale = d3.scaleLinear()
20   .domain([0,d3.max(dataset, function (d){
21     return d;
22   })])
23   .rangeRound ([ h , 0]);
24
25 // Selecting the body to draw the char
26 var svg1 = d3.select("body")
27   .append("svg")
28   .attr("height",h)
29   .attr("width", w);
30
31
32 // Draw the bar chart with xScale width and yScale heigh
33 t
34 svg1.selectAll("rect")
35   .data(dataset)
36   .enter()
37   .append("rect")
38   .attr("x", function(d,i) {
39     return xScale(i);
40   })
41   .attr("y", function(d,i) {
42     return yScale(d);
43   })
44   .attr("width",xScale.bandwidth())
45   .attr("height", function(d,i){
46     return h-yScale(d);
47   })
48   .attr("fill", function(d){
49     return "rgb(0,0, " + Math.round(d*10) + ")";
50   })
51
52 .on("mouseover", function(event, d){
53   d3.select(this)
54     .transition ()
55     .duration(200)
56     .attr("fill", "orange")
57   var xPos = parseFloat (d3.select(this).attr("x")) + xScale.bandwidth()/3 ;
58   var yPos = parseFloat (d3.select(this).attr("y")) + 15;
59   t
60   svg1.append("text")
61     .attr("id", "tooltip")
62     .attr("x", xPos)
63     .attr("y", yPos)
64     .text(d);
65
66
67
68 }) .on("mouseout", function(){
69   d3.select(this)
70     .transition ()
71     .duration(250)
72     .attr("fill", function(d){
73       return "rgb(0,0, " + Math.round(d*10) + ")";
74     })
75   d3.select("#tooltip").remove();
76   p
77 }) ;
78
79
80 // Set the random value for update butto
81 d3.select("#update")
82 .on("click", function(){
83   // alert("Button clicke
84   d0;transitionTyp = document.getElementById ('transitionTyp ').value;
85   e
86   var numValues = dataset.length;
87   var maxValues = 25;
88   dataset = [];
89   for (var i = 0 ; i < numValues; i++) {
90     var newNumber = Math.floor(Math.random() * maxValues);
91     dataset.push(newNumber);
92   }
93
94   // Update the bars with new dat
95   svg1.selectAll("rect")
96     .data(dataset)
97     .transition ()
98     .duration(400)
99     .delay(function(d,i){
100       return i/dataset.length*100;
101       // return i*10
102     })
103   .attr("y", function(d,i) {
104     return yScale(d);
105   })
106   .attr("height", function(d,i){
107     return h-yScale(d);
108   })
109   .attr("fill", function(d){
110     return "rgb(0,0, " + Math.round(d*10) + ")";
111   })
112   .ease(d3[transitionTyp ]);
113   });
114
115

```

Fig 10: 6.2 JS code part 1

```

1
2      d3.select("#add")
3      on("click", function(){
4          // alert("Button Add clicke
5          d");
6          var maxValues = 25;
7          var newNumber = Math.floor(Math.random() * maxValues);
8          dataset.push(newNumber);
9          xScale.domain(d3.range(dataset.length));
10
11          var bars = svg1.selectAll('rect')
12          data(dataset);
13
14          bars.enter()
15          append("rect")
16          .attr("x", w) //Start of transitio
17          .attr("y", function(d,i){
18              return h-yScale(d);
19          })
20      })
21      merge(bars) // Merge the enter selection with the existing bar
22      transitio ()$
23      duration(500)
24      .attr("x", function(d,i){
25          return xScale(i);
26      })
27      .attr("y", function(d,i){
28          return yScale(d);
29      })
30      .attr("width", xScale.bandwidth())
31      .attr("height", function(d){
32          return h - yScale(d);
33      })
34      .attr("fill", function(d){
35          return "rgb(0,0, " + Math.round(d*10) + ")";
36      })
37
38      // Re-Select the bars to allow mouse hoverin
39      bars = svg1.selectAll('rect')
40      data(dataset)
41      on("mouseover", function(event,d){
42          d3.select(this)
43          transitio ()
44          duration(20)
45          .attr("fill", "orange")
46          var xPos = parseFloat (d3.select(this).attr("x")) + xScale.bandwidth()/3 ;
47          var yPos = parseFloat (d3.select(this).attr("y")) + 15;
48          //
49          svg1.append("text")
50          .attr("id", "tooltip")
51          .attr("x", xPos)
52          .attr("y", yPos)
53          .text(d);
54      })
55      on("mouseou ,function(){
56          d3.select(this)
57          transitio ()
58          duration(250)
59          .attr("fill", function(d){
60              return "rgb(0,0, " + Math.round(d*10) + ")";
61          })
62          transitio ()
63          delay(3000)
64          duration(500)
65          d3.select("#tooliti ").remove();
66          p"
67      });
68      console.log(dataset);
69      d3.select("#remove")
70      on("click", function(){
71          // alert("Button Remove clicke
72          d3.shift();
73          xScale.domain(d3.range(dataset.length))
74          var bars = svg1.selectAll('rect')
75          data(dataset);
76          console.log(dataset);
77          bars.exit()
78          transitio ()
79          duration(500)
80          .attr("x",w)
81          .remove();
82          bars .transitio ()
83          duration(500)
84          .attr("x",function(d,i){
85              return xScale(i)
86          })
87          .attr("width", xScale.bandwidth());
88      });
89      d3.select("#sortAs ")
90      on("click", function(){
91          var sortBars = function(){
92              svg1.selectAll('rect')
93              sort(function(a,b){
94                  return d3.ascending(a,b);
95              })
96          }
97          transitio ()
98          duration(500)
99          .attr("x", function(d,i){
100              return xScale(i);
101          })
102      })
103      sortBars();
104
105
106      d3.select("#sortDe ")
107      on("click", function(){
108          var sortBars = function(){
109              svg1.selectAll('rect')
110              sort(function(a,b){
111                  return d3.descendin (a,b);
112              })
113          }
114          transitio ()
115          duration(500)
116          .attr("x", function(d,i){
117              return xScale(i);
118          })
119      })
120      sortBars();
121
122      });
123
124
125

```

Fig 11: 6.2 JS code part 2

