

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 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 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 dat
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.easeCubicInOu
77        t) .ease(d3.EaseCircleI
78        t) .ease(d3.easeCircleOu
79        t) .ease(d3.easeElasticOu
80        t) .ease(d3.easeBounc
81        t) .ease(d3.easeLinea
82        r)
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   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  })})
24    .rangeRound([h, 0]);
25  d
26  // Selecting the body to draw the char
27  var svg1 = d3.select("body")
28    .append("svg")
29    .attr("height",h)
30    .attr("width", w);
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 butto
52  d3.select("#update")
53    .on("click", function(){
54      var transitionType = document.getElementById("transitionType").value;
55      e
56      d
57      e
58      var numValues = dataset.length;
59      var maxValues = 25;
60      dataset = [];
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 dat
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        })
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
87  d3.select("#add")
88    .on("click", function(){
89      // alert("Button Add clicke
90      d;
91      var maxValues = 25;
92      var newNumber = Math.floor(Math.random() * maxValues);
93      dataset.push(newNumber);
94      xScale.domain(d3.range(dataset.length));
95
96      var bars = svg1.selectAll("rect")
97        .data(dataset);
98
99      bars.enter()
100        .append("rect")
101        .attr("x", w)
102        .attr("y", function(d,i){
103          return h-yScale(d);
104        })
105    });
106
107    merge(bars) // Merge the enter selection with the existing bar
108    .transition()
109    .duration(500)
110    .attr("x", function(d,i){
111      return xScale(i);
112    })
113    .attr("y", function(d,i){
114      return yScale(d);
115    })
116    .attr("width", xScale.bandwidth())
117    .attr("height", function(d){
118      return h - yScale(d);
119    })
120    .attr("fill", function(d){
121      return "rgb(0,0, " + Math.round(d*10) + ")";
122    });
123  });
124  });
125
126  d3.select("#remove")
127    .on("click", function(){
128      // alert("Button Remove clicke
129      d3.selectAll("rect")
130        .xScale.domain(d3.range(dataset.length))
131        .var bars = svg1.selectAll("rect")
132        .data(dataset);
133        bars.exit()
134        .transition()
135        .duration(500)
136        .attr("x",w)
137        .remove();
138
139        bars.transition()
140        .duration(500)
141        .attr("x",function(d,i){
142          return xScale(i)
143        })
144        .attr("width", xScale.bandwidth())
145    });
146  });
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

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206

```

```

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206

```



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   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 s"
14 >
15 <body>
16   <h1> The D3 Journey Start Here </h1>
17   <button id="update">Update</button>
18   <select id="transitionType" >
19     <option value="easeLinear" >Linear</option>
20     <option value="easeExp">Ease Exp</option>
21     <option value="easeElastic" >Elastic</option>
22     <option value="easeBounce" >Bounce</option>
23   </select>
24   e"
25   <button id="add">Add</button>
26   <button id="remove">Remove</button>
27   <button id="sortAsc">Sort ASC</button>
28   <button id="sortDsc">Sort DSC</button>
29   <br>
30   <script src="scripts.js" ></script>
31   s"
32   >
33   <br>
34   <bf>
35   <footer style = "color:grey"> COS30045 Data Visualisatio <br>
36   Toan Nguye </footer>
37   n
38 </body>
39 </html>

```

Fig 9: 6.2 HTML code

```

1  var s = 100;
2  var s = 100;
3  // padding = 5
4  //
5  //
6  var dataset = [14,10,11,11,10,11,11,11];
7
8
9  //scale and yscale are for scaling the value of dataset to width w
10
11  //the scale is width for categorical values
12  var scale = d3.scale.ordinal()
13    .domain(d3.range(dataset.length))
14    .range(d3.range(0,w));
15  //padding for numerical values
16  //padding for numerical values
17  var yscale = d3.scale.linear()
18    .domain([0,d3.max(dataset, function(d){
19      return d;
20    })])
21    .range([0,h]);
22
23  // Draw the bar chart with scale width and yscale height
24  //
25  //
26  // Selecting the body to draw the chart
27  var svg = d3.select("body")
28    .append("svg")
29    .attr("width", w)
30    .attr("height", h);
31
32  // Draw the bar chart with scale width and yscale height
33  //
34  //
35  //
36  //
37  //
38  //
39  //
40  //
41  //
42  //
43  //
44  //
45  //
46  //
47  //
48  //
49  //
50  //
51  //
52  //
53  //
54  //
55  //
56  //
57  //
58  //
59  //
60  //
61  //
62  //
63  //
64  //
65  //
66  //
67  //
68  //
69  //
70  //
71  //
72  //
73  //
74  //
75  //
76  //
77  //
78  //
79  //
80  //
81  //
82  //
83  //
84  //
85  //
86  //
87  //
88  //
89  //
90  //
91  //
92  //
93  //
94  //
95  //
96  //
97  //
98  //
99  //
100 //
101 //
102 //
103 //
104 //
105 //
106 //
107 //
108 //
109 //
110 //
111 //
112 //
113 //
114 //
115 //
116 //
117 //
118 //
119 //
120 //
121 //
122 //
123 //
124 //
125 //
126 //
127 //
128 //
129 //
130 //
131 //
132 //
133 //
134 //
135 //
136 //
137 //
138 //
139 //
140 //
141 //
142 //
143 //
144 //
145 //
146 //
147 //
148 //
149 //
150 //
151 //
152 //
153 //
154 //
155 //
156 //
157 //
158 //
159 //
160 //
161 //
162 //
163 //
164 //
165 //
166 //
167 //
168 //
169 //
170 //
171 //
172 //
173 //
174 //
175 //
176 //
177 //
178 //
179 //
180 //
181 //
182 //
183 //
184 //
185 //
186 //
187 //
188 //
189 //
190 //
191 //
192 //
193 //
194 //
195 //
196 //
197 //
198 //
199 //
200 //
201 //
202 //
203 //
204 //
205 //
206 //
207 //
208 //
209 //
210 //
211 //
212 //
213 //
214 //
215 //
216 //
217 //
218 //
219 //
220 //
221 //
222 //
223 //
224 //
225 //
226 //
227 //
228 //
229 //
230 //
231 //
232 //
233 //
234 //
235 //
236 //
237 //
238 //
239 //
240 //
241 //
242 //
243 //
244 //
245 //
246 //
247 //
248 //
249 //
250 //
251 //
252 //
253 //
254 //
255 //
256 //
257 //
258 //
259 //
260 //
261 //
262 //
263 //
264 //
265 //
266 //
267 //
268 //
269 //
270 //
271 //
272 //
273 //
274 //
275 //
276 //
277 //
278 //
279 //
280 //
281 //
282 //
283 //
284 //
285 //
286 //
287 //
288 //
289 //
290 //
291 //
292 //
293 //
294 //
295 //
296 //
297 //
298 //
299 //
300 //
301 //
302 //
303 //
304 //
305 //
306 //
307 //
308 //
309 //
310 //
311 //
312 //
313 //
314 //
315 //
316 //
317 //
318 //
319 //
320 //
321 //
322 //
323 //
324 //
325 //
326 //
327 //
328 //
329 //
330 //
331 //
332 //
333 //
334 //
335 //
336 //
337 //
338 //
339 //
340 //
341 //
342 //
343 //
344 //
345 //
346 //
347 //
348 //
349 //
350 //
351 //
352 //
353 //
354 //
355 //
356 //
357 //
358 //
359 //
360 //
361 //
362 //
363 //
364 //
365 //
366 //
367 //
368 //
369 //
370 //
371 //
372 //
373 //
374 //
375 //
376 //
377 //
378 //
379 //
380 //
381 //
382 //
383 //
384 //
385 //
386 //
387 //
388 //
389 //
390 //
391 //
392 //
393 //
394 //
395 //
396 //
397 //
398 //
399 //
400 //
401 //
402 //
403 //
404 //
405 //
406 //
407 //
408 //
409 //
410 //
411 //
412 //
413 //
414 //
415 //
416 //
417 //
418 //
419 //
420 //
421 //
422 //
423 //
424 //
425 //
426 //
427 //
428 //
429 //
430 //
431 //
432 //
433 //
434 //
435 //
436 //
437 //
438 //
439 //
440 //
441 //
442 //
443 //
444 //
445 //
446 //
447 //
448 //
449 //
450 //
451 //
452 //
453 //
454 //
455 //
456 //
457 //
458 //
459 //
460 //
461 //
462 //
463 //
464 //
465 //
466 //
467 //
468 //
469 //
470 //
471 //
472 //
473 //
474 //
475 //
476 //
477 //
478 //
479 //
480 //
481 //
482 //
483 //
484 //
485 //
486 //
487 //
488 //
489 //
490 //
491 //
492 //
493 //
494 //
495 //
496 //
497 //
498 //
499 //
500 //
501 //
502 //
503 //
504 //
505 //
506 //
507 //
508 //
509 //
510 //
511 //
512 //
513 //
514 //
515 //
516 //
517 //
518 //
519 //
520 //
521 //
522 //
523 //
524 //
525 //
526 //
527 //
528 //
529 //
530 //
531 //
532 //
533 //
534 //
535 //
536 //
537 //
538 //
539 //
540 //
541 //
542 //
543 //
544 //
545 //
546 //
547 //
548 //
549 //
550 //
551 //
552 //
553 //
554 //
555 //
556 //
557 //
558 //
559 //
560 //
561 //
562 //
563 //
564 //
565 //
566 //
567 //
568 //
569 //
570 //
571 //
572 //
573 //
574 //
575 //
576 //
577 //
578 //
579 //
580 //
581 //
582 //
583 //
584 //
585 //
586 //
587 //
588 //
589 //
590 //
591 //
592 //
593 //
594 //
595 //
596 //
597 //
598 //
599 //
600 //
601 //
602 //
603 //
604 //
605 //
606 //
607 //
608 //
609 //
610 //
611 //
612 //
613 //
614 //
615 //
616 //
617 //
618 //
619 //
620 //
621 //
622 //
623 //
624 //
625 //
626 //
627 //
628 //
629 //
630 //
631 //
632 //
633 //
634 //
635 //
636 //
637 //
638 //
639 //
640 //
641 //
642 //
643 //
644 //
645 //
646 //
647 //
648 //
649 //
650 //
651 //
652 //
653 //
654 //
655 //
656 //
657 //
658 //
659 //
660 //
661 //
662 //
663 //
664 //
665 //
666 //
667 //
668 //
669 //
670 //
671 //
672 //
673 //
674 //
675 //
676 //
677 //
678 //
679 //
680 //
681 //
682 //
683 //
684 //
685 //
686 //
687 //
688 //
689 //
690 //
691 //
692 //
693 //
694 //
695 //
696 //
697 //
698 //
699 //
700 //
701 //
702 //
703 //
704 //
705 //
706 //
707 //
708 //
709 //
710 //
711 //
712 //
713 //
714 //
715 //
716 //
717 //
718 //
719 //
720 //
721 //
722 //
723 //
724 //
725 //
726 //
727 //
728 //
729 //
730 //
731 //
732 //
733 //
734 //
735 //
736 //
737 //
738 //
739 //
740 //
741 //
742 //
743 //
744 //
745 //
746 //
747 //
748 //
749 //
750 //
751 //
752 //
753 //
754 //
755 //
756 //
757 //
758 //
759 //
760 //
761 //
762 //
763 //
764 //
765 //
766 //
767 //
768 //
769 //
770 //
771 //
772 //
773 //
774 //
775 //
776 //
777 //
778 //
779 //
780 //
781 //
782 //
783 //
784 //
785 //
786 //
787 //
788 //
789 //
790 //
791 //
792 //
793 //
794 //
795 //
796 //
797 //
798 //
799 //
800 //
801 //
802 //
803 //
804 //
805 //
806 //
807 //
808 //
809 //
810 //
811 //
812 //
813 //
814 //
815 //
816 //
817 //
818 //
819 //
820 //
821 //
822 //
823 //
824 //
825 //
826 //
827 //
828 //
829 //
830 //
831 //
832 //
833 //
834 //
835 //
836 //
837 //
838 //
839 //
840 //
841 //
842 //
843 //
844 //
845 //
846 //
847 //
848 //
849 //
850 //
851 //
852 //
853 //
854 //
855 //
856 //
857 //
858 //
859 //
860 //
861 //
862 //
863 //
864 //
865 //
866 //
867 //
868 //
869 //
870 //
871 //
872 //
873 //
874 //
875 //
876 //
877 //
878 //
879 //
880 //
881 //
882 //
883 //
884 //
885 //
886 //
887 //
888 //
889 //
890 //
891 //
892 //
893 //
894 //
895 //
896 //
897 //
898 //
899 //
900 //
901 //
902 //
903 //
904 //
905 //
906 //
907 //
908 //
909 //
910 //
911 //
912 //
913 //
914 //
915 //
916 //
917 //
918 //
919 //
920 //
921 //
922 //
923 //
924 //
925 //
926 //
927 //
928 //
929 //
930 //
931 //
932 //
933 //
934 //
935 //
936 //
937 //
938 //
939 //
940 //
941 //
942 //
943 //
944 //
945 //
946 //
947 //
948 //
949 //
950 //
951 //
952 //
953 //
954 //
955 //
956 //
957 //
958 //
959 //
960 //
961 //
962 //
963 //
964 //
965 //
966 //
967 //
968 //
969 //
970 //
971 //
972 //
973 //
974 //
975 //
976 //
977 //
978 //
979 //
980 //
981 //
982 //
983 //
984 //
985 //
986 //
987 //
988 //
989 //
990 //
991 //
992 //
993 //
994 //
995 //
996 //
997 //
998 //
999 //
1000 //

```

Fig 10: 6.2 JS code