

### Task 7.1



```
1 <!DOCTYPE html>
2 <html lang = "en">
3 <head>
4   <meta charset = "utf-8" />
5   <meta name = "description" content = "Data Visualisation"/>
6   <meta name = "keywords" content = "HTML,CSS,D3" />
7   <meta name = "author" content = "Toan Nguyen" />
8
9   <title> Task 7.1 Drawing with data</title>
10
11   <script src = "https://d3js.org/d3.v7.min.js" ></script>
12   <link rel="stylesheet" href="styles.css">
13 </head>
14
15 <body>
16   <h1> The D3 Journey Start Here </h1>
17
18   <script src="scripts.js"></script>
19   <div id = "chart"></div>
20
21   <br>
22   <bf>
23   <footer style = "color:grey"> COS30045 Data Visualisation<br>
24   Toan Nguyen</footer>
25 </body>
26
27 </html>
```

Fig 1: HTML code

```

1 var w = 600;
2 var h = 300;
3 var padding = 55;
4
5 function init(){
6     //Set the dataset for the chart with dat
7     var dataset
8     d3.csv("Unemployment_78-95.csv", function(d){
9         return {
10             date: new Date(+d.year, d.month-1),
11             number: +d.number
12         };
13     });
14     dataset = data;
15     lineChart(dataset);
16     console.table(dataset, ["date", "number"]);
17 };
18
19
20 function lineChart(dataset) {
21     xScale = d3.scaleTime() //Set the x scale for time serie
22     .domain([
23         d3.min(dataset, function(d) { return d.date; }),
24         d3.max(dataset, function(d) { return d.date; })
25     ])
26     .range([padding, w-padding]);
27
28     yScale = d3.scaleLinear() //Set the y scale for data valu
29     .domain([0, d3.max(dataset, function(d) { return d.number; })
30     ])
31     .range([h-padding, 0]);
32
33     var line = d3.line() //Assign value for lin
34     .x(function(d){return xScale(d.date);})
35     .y(function(d){return yScale(d.number);})
36
37     var svg = d3.select("#chart")
38     .append("svg")
39     .attr("width", w)
40     .attr("height", h);
41
42     svg.append("path")
43     .datum(dataset)
44     .attr("class", "line")
45     .attr("d", line);
46
47     var xAxis = d3.axisBottom() //Set tick and scale X-axis to match dat
48     .ticks(5)
49     .scale(xScale);
50
51     var yAxis = d3.axisLeft() //Set tick and scale Y-axis to match dat
52     .ticks(5)
53     .scale(yScale);
54
55     svg.append("g")
56     .attr("transform", "translate(0, +(h - padding) +)")
57     .call(xAxis);
58
59     svg.append("g")
60     .attr("transform", "translate(+padding, 0)")
61     .call(yAxis);
62
63     svg.append("line") //Set the baseline for half of the value
64     .attr("class", "line_half_million")
65     .attr("x1", padding)
66     .attr("y1", yScale(500000))
67     .attr("x2", w)
68     .attr("y2", yScale(500000));
69
70     svg.append("text")
71     .attr("class", "halfMillLabel")
72     .attr("x", padding+10)
73     .attr("y", yScale(500000) - 7)
74     .text("Half a million unemployed");
75 }
76
77 init();

```

Fig2: JS code for Line chart

```

1 var w = 600;
2 var h = 300;
3 var padding = 55;
4
5 function init(){
6     //Set the dataset for the chart with dat
7     var dataset
8     d3.csv("Unemployment_78-95.csv", function(d){
9         return {
10             date: new Date(+d.year, d.month-1),
11             number: +d.number
12         };
13     }).then(function(data){
14         dataset = data;
15         areaChart(dataset);
16         console.table(dataset, ["date", "number"]);
17     });
18 }
19
20 function areaChart(dataset) {
21     xScale = d3.scaleTime() //Set the x scale for time serie
22     .domain([
23         d3.min(dataset, function(d) { return d.date; }),
24         d3.max(dataset, function(d) { return d.date; })
25     ])
26     .range([padding, w-padding]);
27
28     yScale = d3.scaleLinear() //Set the y scale for data valu
29     .domain([0, d3.max(dataset, function(d) { return d.number; })
30     ])
31     .range([h-padding, 0]);
32
33     var area = d3.area() //Assign value for are
34     .x(function(d) { return xScale(d.date); })
35     .y0(function() { return yScale.range()[0]; })
36     .y1(function(d) { return yScale(d.number); });
37
38     var svg = d3.select("#chart")
39     .append("svg")
40     .attr("fill", "rgb(56,56,12)")
41     .attr("width", w)
42     .attr("height", h);
43
44     svg.append("path")
45     .datum(dataset)
46     .attr("class", "area")
47     .attr("d", area);
48
49     var xAxis = d3.axisBottom() //Set tick and scale X-axis to match dat
50     .ticks(5)
51     .scale(xScale);
52
53     var yAxis = d3.axisLeft() //Set tick and scale Y-axis to match dat
54     .ticks(5)
55     .scale(yScale);
56
57     svg.append("g")
58     .attr("transform", "translate(0, +(h - padding) +)")
59     .call(xAxis);
60
61     svg.append("g")
62     .attr("transform", "translate(+padding, 0)")
63     .call(yAxis);
64
65     svg.append("line") //Set the baseline for half of the value
66     .attr("class", "line_half_milio")
67     .attr("x1", padding)
68     .attr("y1", yScale(500000))
69     .attr("x2", w)
70     .attr("y2", yScale(500000))
71     .attr("fill", "rgb(255,0,0)");
72
73     svg.append("text")
74     .attr("class", "halfMillabe")
75     .attr("x", padding+10)
76     .attr("y", yScale(500000) - 7)
77     .attr("fill", "rgb(255,0,0)")
78     .text("Half a Million unemployed");
79 }
80
81 init();

```

Fig 3: JS code for Area chart

## Task 7.2



```
1 <!DOCTYPE html>
2 <html lang = "en">
3 <head>
4   <meta charset = "utf-8" />
5   <meta name = "description" content = "Data Visualisation"/>
6   <meta name = "keywords" content = "HTML,CSS,D3" />
7   <meta name = "author" content = "Toan Nguyen" />
8
9   <title> Task 7.2 Drawing with data</title>
10
11   <script src = "https://d3js.org/d3.v7.min.js" ></script>
12   <link rel="stylesheet" href="styles.css">
13 </head>
14
15 <body>
16   <h1> The D3 Journey Start Here </h1>
17
18   <script src="scripts.js"></script>
19
20   <br>
21   <bf>
22   <footer style = "color:grey"> COS30045 Data Visualisation<br>
23     Toan Nguyen</footer>
24 </body>
25
26 </html>
```

Fig 4: HTML code


```

1  var w = 300;
2  var h = 300;
3  var padding = 55;
4
5  var innerRadius = 0;
6  var outerRadius = w/2;
7
8  var dataset = [5,10,20,45,28,58,42]; //Set the value for pie char
9                                     t
10 var arc = d3.arc() //Set the radius for inner and outer circle
11     .innerRadius(innerRadius) s
12     .outerRadius(outerRadius);
13
14 var pie = d3.pie();
15
16 var svg = d3.select("body")
17     .append("svg")
18     .attr("width", w)
19     .attr("height", h);
20
21 var arcs = svg.selectAll("g.arc") //Set the curve part for pie char
22     .data(pie(dataset)) t
23     .enter()
24     .append("g")
25     .attr("class", "arc")
26     .attr("transform", "translate " + outerRadius+ ", " + outerRadius +");");
27     ("
28 var color = d3.scaleOrdinal(d3.schemeCategory1); //Set color for pie char
29     l           0           t
30 arcs.append("path") //Draw pie and its color
31     .attr("fill", function(d,i){ r
32         return color(i);
33     })
34     .attr("d", function(d,i){
35         return arc(d,i);
36     });
37
38 arcs.append("text") //Set value in text for each pie
39     .text(function(d){ e
40         return d.value;
41     })
42     .attr("transform",function(d){
43         return "translate " + arc.centroid(d) + " ";
44     }); ("
45
46
47
48

```

Fig 5: JS code

### Task 7.3



```
1 <!DOCTYPE html>
2 <html lang = "en">
3 <head>
4   <meta charset = "utf-8" />
5   <meta name = "description" content = "Data Visualisation"/>
6   <meta name = "keywords" content = "HTML,CSS,D3" />
7   <meta name = "author" content = "Toan Nguyen" />
8
9   <title> Task 7.3 Drawing with data</title>
10
11   <script src = "https://d3js.org/d3.v7.min.js" ></script>
12   <link rel="stylesheet" href="styles.css">
13 </head>
14
15 <body>
16   <h1> The D3 Journey Start Here </h1>
17
18   <script src="scripts.js"></script>
19
20   <br>
21   <bf>
22   <footer style = "color:grey"> COS30045 Data Visualisation<br>
23   Toan Nguyen</footer>
24 </body>
25
26 </html>
```

Fig 6: HTML code

```

1 var w = 300;
2 var h = 300;
3 var padding = 55;
4
5 //Dataset with stacked value
6 var dataset = [
7
8   { apples: 5, oranges: 10, grapes: 22 },
9
10  { apples: 4, oranges: 12, grapes: 28 },
11
12  { apples: 2, oranges: 19, grapes: 32 },
13
14  { apples: 7, oranges: 23, grapes: 35 },
15
16  { apples: 23, oranges: 17, grapes: 43 }
17
18 ];
19
20 //the xScale in width for categorical variable
21 var xScale = d3.scaleBand()
22   .domain(d3.range(dataset.length))
23   .rangeRound ([0,w])
24   .paddingInner (0.05);
25
26 //the yScale in height for numerical value
27 var yScale = d3.scaleLinear()
28   .domain([0, d3.max(dataset, function(d){
29     return d.apples + d.oranges + d.grapes;
30   }) ])
31   .range([ h , 0]);
32
33
34 var stack = d3.stack()           //Create stack for each variable
35   .keys(["apples", "oranges", "grapes"]);
36
37 var series = stack(dataset);      //Set value for stack
38
39 var svg = d3.select("body")
40   .append("svg")
41   .attr("width", w)
42   .attr("height", h);
43
44 var color = d3.scaleOrdinal (d3.schemeCategory1 ); //Set color
45
46 var groups = svg.selectAll("g")   //Group and fill color in each group
47   .data(series)
48   .enter()
49   .append("g")
50   .style("fill", function(d,i){
51     return color(i);
52   });
53
54 var rects = groups.selectAll("rect") //Draw rectangles in stacked group
55   .data(function(d){ return d;})
56   .enter()
57   .append("rect")
58   .attr("x", function(d,i){
59     return xScale(i)
60   })
61   .attr("y", function(d,i){
62     return yScale(d[1]);
63   })
64   .attr("height", function(d){
65     return yScale(d[0]) - yScale(d[1]);
66   })
67   .attr("width", xScale.bandwidth());
68
69
70

```

Fig 7: JS code

## Task 8.1



```
1 <!DOCTYPE html>
2 <html lang = "en">
3 <head>
4     <meta charset = "utf-8" />
5     <meta name = "description" content = "Data Visualisation"/>
6     <meta name = "keywords" content = "HTML,CSS,D3" />
7     <meta name = "author" content = "Toan Nguyen" />
8
9     <title> Task 8.1 Drawing with data</title>
10
11     <script src = "https://d3js.org/d3.v7.min.js" ></script>
12     <link rel="stylesheet" href="styles.css">
13 </head>
14
15 <body>
16     <h1> The D3 Journey Start Here </h1>
17
18     <script src="scripts.js"></script>
19
20     <br>
21     <bf>
22     <footer style = "color:grey"> COS30045 Data Visualisation<br>
23     Toan Nguyen</footer>
24 </body>
25
26 </html>
```

Fig 8: HTML code






```
1  var w = 600;
2  var h = 300;
3  var padding = 55;
4
5  //Specifying the projection
6  var projection = d3.geoMercator()
7                    .center([145,-36.5])
8                    .translate([w/2,h/2])
9                    .scale(2450);
10
11 var path = d3.geoPath()
12            .projection(projection);
13
14
15 var svg = d3.select("body")
16            .append("svg")
17            .attr("width", w)
18            .attr("height", h)
19            .attr("fill", "grey");
20
21 //Read and Set the GeoJSON file to draw path
22 d3.json("LGA_VIC.json").then(function(json){
23
24     svg.selectAll("path")
25         .data(json.features)
26         .enter()
27         .append("path")
28         .attr("d", path);
29 });
30
31
```

Fig 9: JS code

## Task 8.2



```
1 <!DOCTYPE html>
2 <html lang = "en">
3 <head>
4   <meta charset = "utf-8" />
5   <meta name = "description" content = "Data Visualisation"/>
6   <meta name = "keywords" content = "HTML,CSS,D3" />
7   <meta name = "author" content = "Toan Nguyen" />
8
9   <title> Task 8.2 Drawing with data</title>
10
11   <script src = "https://d3js.org/d3.v7.min.js" ></script>
12   <link rel="stylesheet" href="styles.css">
13 </head>
14
15 <body>
16   <h1> The D3 Journey Start Here </h1>
17
18   <script src="scripts.js"></script>
19
20   <br>
21   <b>
22     <footer style = "color:grey"> COS30045 Data Visualisation<br>
23     Toan Nguyen</footer>
24 </body>
25
26 </html>
```

Fig 10: HTML code

```

1 var w = 600;
2 var h = 300;
3 var padding = 55;
4
5 //Set a range color for each patia
6 var color = d3.scaleQuantiz ()
7     range(['#ffffcc', '#c2e699', '#78c679', '#31a354', '#006837'])
8
9 //Specifying the projection and Set the Geometry of the Ma
10 var projectio = d3.geoMercator()
11     .center([145,-36.5])
12     .translate([w/2,h/2])
13     .scale(2450);
14
15
16 var path = d3.geoPath()
17     .projectio (projectio );
18
19 var svg = d3.select("body")
20     .append("svg")
21     .attr("width", w)
22     .attr("height", h)
23     .attr("fill", "grey");
24
25 //Read the data form CSV fil
26 var data = d3.csv("VIC_LGA_unemployment.cs ").then(function(data) {
27     color.domain([v"
28         d3.min(data, function(d){ return d.unemploye ; })),
29         d3.max(data, function(d){ return d.dnemploye ; }])
30     ])
31     return data;
32 });
33
34 //Read and Set the GeoJSON file to draw pat
35 json = d3.json("LGA_VIC.jso ");
36 Promise.all([data, json]).then(function(values) {
37     var data = values[0];
38     var json = values[1];
39
40     for (var i = 0; i < data.length; i++) {
41         var dataState = data[i];
42         for (var j = 0; j < json.features.length; j++) {
43
44             var jsonState = json.features[j];
45             if (jsonState.proportie .LGA_name === dataState.LGA) {
46                 jsonState.proportie .value = +dataState.unemploye
47                 break;    s          d
48             }
49         }
50     }
51
52     //Draw the geometry and set its color properties coresponding to the dat
53     avg.selectAll("path")
54         .data(json.features)
55         .enter()
56         .append("path")
57         .attr("d", path)
58         .style("fill", function(d){
59             var value = d.proportie .value;
60             console.log("COLOR "+value);
61             if (value) {
62                 return color(value);
63             } else {
64                 return "#364";
65             }
66         });
67

```

Fig 11: JS code

