COS30045: Data Visualisation – Lab 3.1 – 4.1

Nguyen Khanh Toan – 104180605

Task 3.1

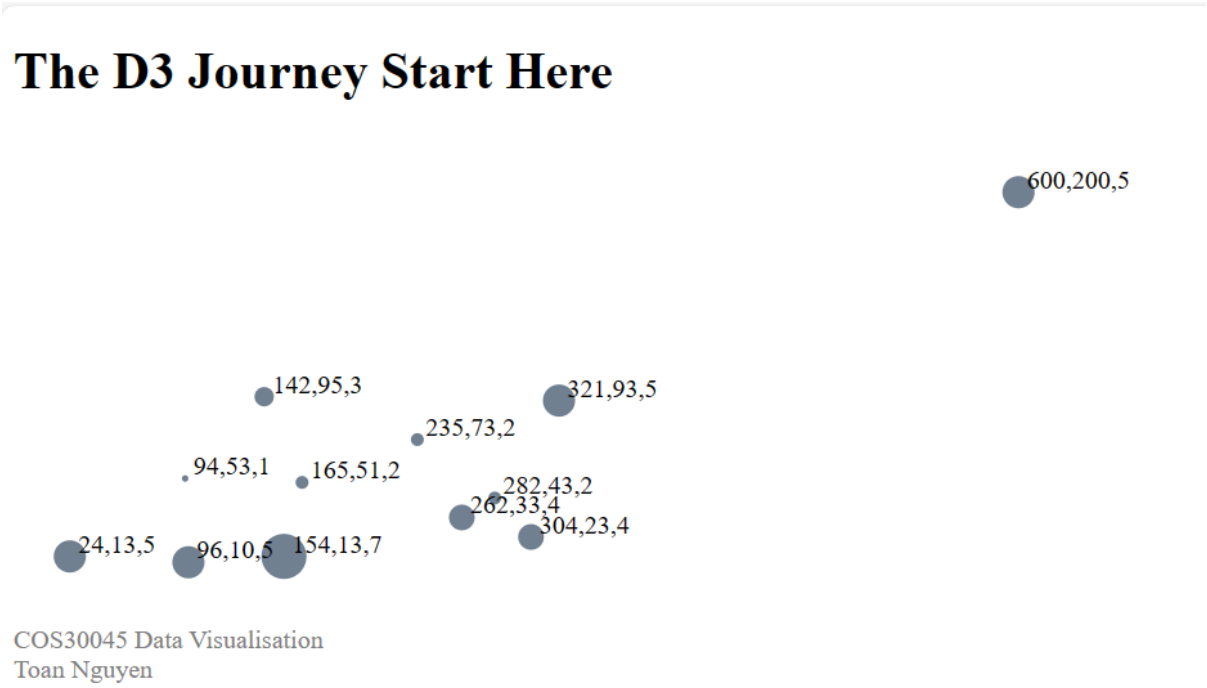
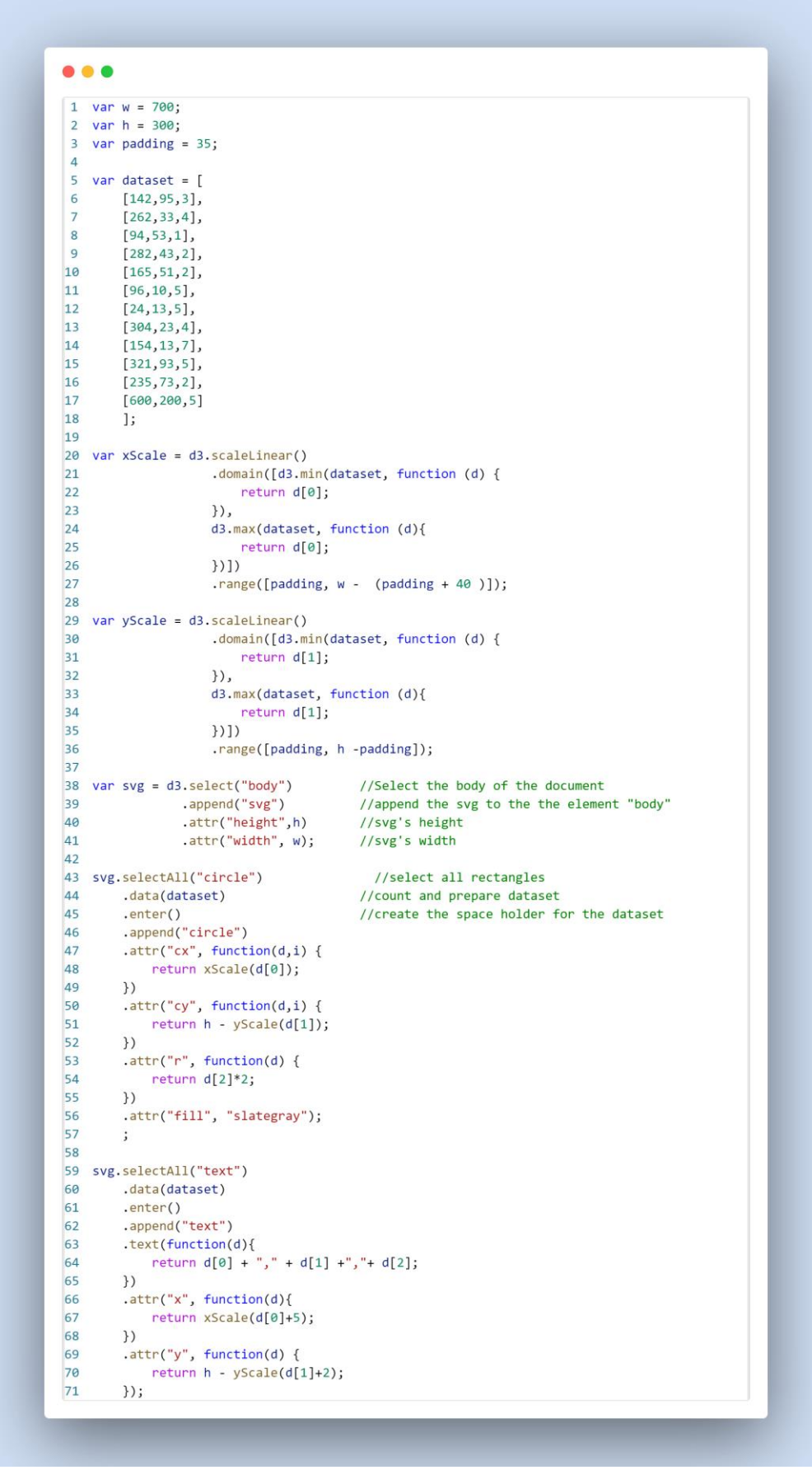


Fig1. Output

```
<!DOCTYPE html>
<html lang = "en">
<head>
    <meta charset = "utf-8" />
    <meta name = "description"  content = "Data Visualisation"/>
    <meta name = "keywords"     content = "HTML,CSS,D3" />
    <meta name = "author"       content = "Toan Nguyen" />

    <title> Task 3.1 Drawing with data</title>

    <script src = "https://d3js.org/d3.v7.min.js" ></script>

</head>

<body>
    <h1> The D3 Journey Start Here </h1>
    <script src="scripts.js"></script>
    <br>
    <bf>
    <footer style = "color:grey"> COS30045 Data Visualisation<br>
    Toan Nguyen</footer>
</body>

</html>
```

Fig2. 3.1 HTML code



```
var w = 700;
var h = 300;
var padding = 35;

var dataset = [
    [142,95,3],
    [262,33,4],
    [94,53,1],
    [282,43,2],
    [165,51,2],
    [96,10,5],
    [24,13,5],
    [304,23,4],
    [154,13,7],
    [321,93,5],
    [235,73,2],
    [600,200,5]
    ];

var xScale = d3.scaleLinear()
                .domain([d3.min(dataset, function (d) {
                    return d[0];
                }),
                d3.max(dataset, function (d){
                    return d[0];
                })])
                .range([padding, w -  (padding + 40 )]);

var yScale = d3.scaleLinear()
                .domain([d3.min(dataset, function (d) {
                    return d[1];
                }),
                d3.max(dataset, function (d){
                    return d[1];
                })])
                .range([padding, h -padding]);

var svg = d3.select("body")         //Select the body of the document
            .append("svg")          //append the svg to the the element "body"
            .attr("height",h)       //svg's height
            .attr("width", w);      //svg's width

svg.selectAll("circle")               //select all rectangles
    .data(dataset)                  //count and prepare dataset
    .enter()                        //create the space holder for the dataset
    .append("circle")
    .attr("cx", function(d,i) {
        return xScale(d[0]);
    })
    .attr("cy", function(d,i) {
        return h - yScale(d[1]);
    })
    .attr("r", function(d) {
        return d[2]*2;
    })
    .attr("fill", "slategray");
    ;

svg.selectAll("text")
    .data(dataset)
    .enter()
    .append("text")
    .text(function(d){
        return d[0] + "," + d[1] +","+ d[2];
    })
    .attr("x", function(d){
        return xScale(d[0]+5);
    })
    .attr("y", function(d) {
        return h - yScale(d[1]+2);
    });
```

Fig 3: Task 3.1 JS code

Task 3.2

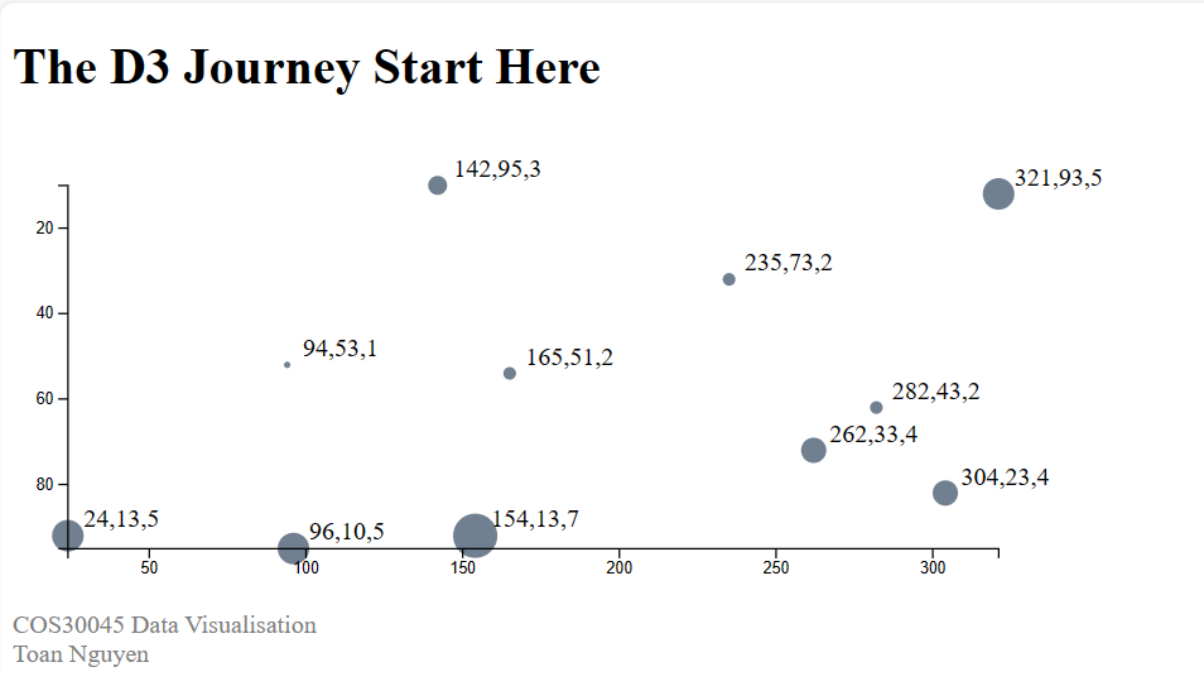
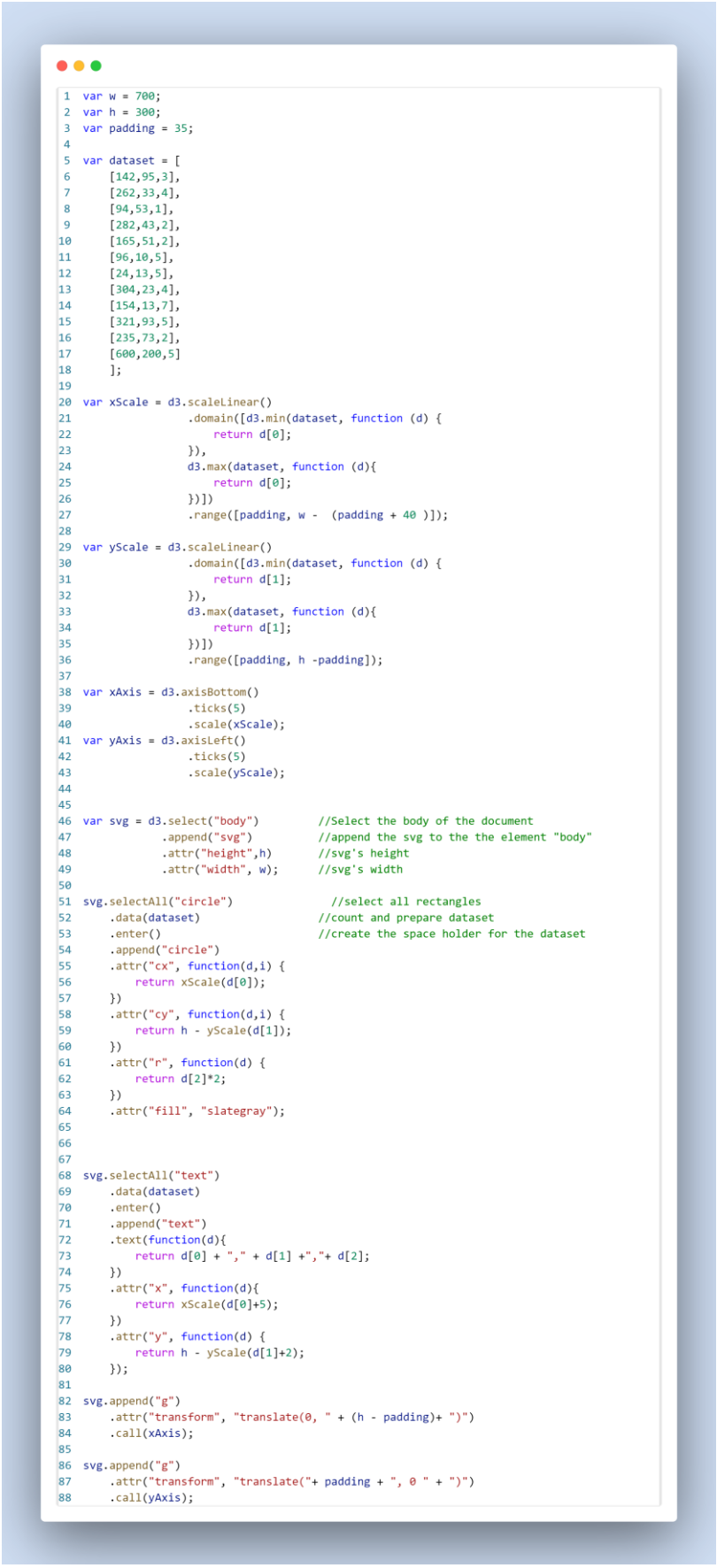


Fig 4: Task 3.2 Output

```html
<!DOCTYPE html>
<html lang = "en">
<head>
    <meta charset = "utf-8" />
    <meta name = "description"  content = "Data Visualisation"/>
    <meta name = "keywords"     content = "HTML,CSS,D3" />
    <meta name = "author"       content = "Toan Nguyen" />

    <title> Task 3.2 Drawing with data</title>

    <script src = "https://d3js.org/d3.v7.min.js" ></script>

</head>

<body>
    <h1> The D3 Journey Start Here </h1>

    <script src="scripts.js"></script>

    <br>
    <bf>
    <footer style = "color:grey"> COS30045 Data Visualisation<br>
    Toan Nguyen</footer>
</body>

</html>
```

Fig 5: Task 3.2 HTML code



```
var w = 700;
var h = 300;
var padding = 35;

var dataset = [
    [142,95,3],
    [262,33,4],
    [94,53,1],
    [282,43,2],
    [165,51,2],
    [96,10,5],
    [24,13,5],
    [304,23,4],
    [154,13,7],
    [321,93,5],
    [235,73,2],
    [600,200,5]
    ];

var xScale = d3.scaleLinear()
                .domain([d3.min(dataset, function (d) {
                    return d[0];
                }),
                d3.max(dataset, function (d){
                    return d[0];
                })])
                .range([padding, w -  (padding + 40 )]);

var yScale = d3.scaleLinear()
                .domain([d3.min(dataset, function (d) {
                    return d[1];
                }),
                d3.max(dataset, function (d){
                    return d[1];
                })])
                .range([padding, h -padding]);

var xAxis = d3.axisBottom()
                .ticks(5)
                .scale(xScale);
var yAxis = d3.axisLeft()
                .ticks(5)
                .scale(yScale);


var svg = d3.select("body")         //Select the body of the document
            .append("svg")          //append the svg to the the element "body"
            .attr("height",h)       //svg's height
            .attr("width", w);      //svg's width

svg.selectAll("circle")               //select all rectangles
    .data(dataset)                  //count and prepare dataset
    .enter()                        //create the space holder for the dataset
    .append("circle")
    .attr("cx", function(d,i) {
        return xScale(d[0]);
    })
    .attr("cy", function(d,i) {
        return h - yScale(d[1]);
    })
    .attr("r", function(d) {
        return d[2]*2;
    })
    .attr("fill", "slategray");



svg.selectAll("text")
    .data(dataset)
    .enter()
    .append("text")
    .text(function(d){
        return d[0] + "," + d[1] +","+ d[2];
    })
    .attr("x", function(d){
        return xScale(d[0]+5);
    })
    .attr("y", function(d) {
        return h - yScale(d[1]+2);
    });

svg.append("g")
    .attr("transform", "translate(0, " + (h - padding)+ ")")
    .call(xAxis);

svg.append("g")
    .attr("transform", "translate("+ padding + ", 0 " + ")")
    .call(yAxis);
```

Fig 6: Task 3.2 JS code