Visualization Lab Assignment 1 Report

- **Dataset**

https://www.kaggle.com/chaitanyahivlekar/fifa-19-player-dataset
FIFA 19 player dataset has been chosen as the dataset for this assignment. It describes the various information regarding accuracies and ratings of FIFA 19 players all over the world. Through this dataset we can gain insight regarding the various characteristics of the players such as how many aged between 32-35 played in 2019 and their different accuracies and so on.

Note : Used python server to host the project (python -m http.server).

The following variables are chosen to visualize:

Numerical Variables :

1. **Heading -** rating of a particular player in terms of heading.
2. **Short Passing -** rating of a particular player in terms of short pass.
3. **Age** - age of a particular player in terms of years.
4. **Height** - height of a particular player in terms of feet.
5. **Weight -** weight of a particular player in terms of lbs.
6. **Dribbling** - rating of a particular player in terms of dribbling.
7. **Sprint Speed -** rating of a particular player in terms of speed.
8. **Jumping -** rating of a particular player in terms of jumping.
9. **Vision** - rating of a particular player in terms of vision.
10. **Finishing -** rating of a particular player in terms of finishing a game.

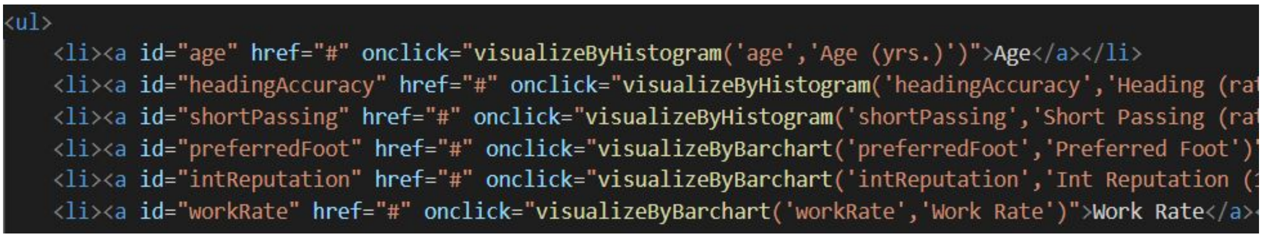
Categorical Variables:

1. **Preferred Foot -** the preferred foot for a particular player (Left or Right)
2. **International Reputation -** reputation of a player (1 being Lowest)
3. **Work Rate** - practise rate of a player (say Low/Low or Low/High etc)
4. **Weak Foot** - player having weak foot (1 being the weakest)
5. **Skill Move** - skill level of a player in terms of move (1 being Lowest)

- **Code**

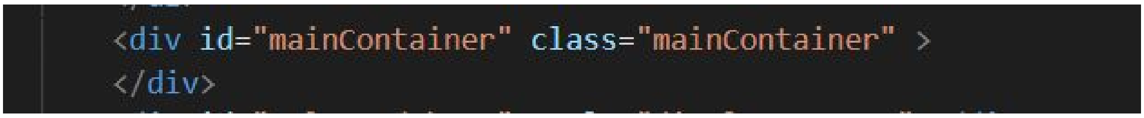
This assignment consists of 6 files :

1. **Index.html :**
   Index.html consists of a menu of all the variables needed to visualize in terms of bar chart, histogram as well as a scatter plot button to draw the scatter plot.



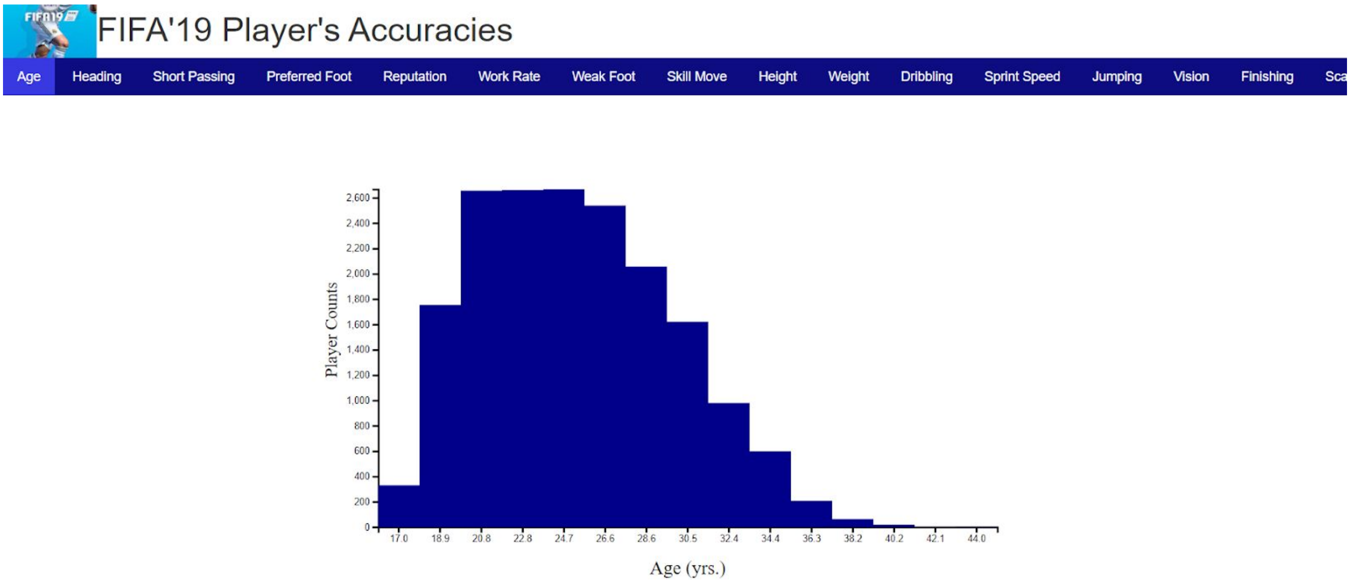
```
<ul>
    <li><a id="age" href="#" onclick="visualizeByHistogram('age','Age (yrs.)')">Age</a></li>
    <li><a id="headingAccuracy" href="#" onclick="visualizeByHistogram('headingAccuracy','Heading (ra
    <li><a id="shortPassing" href="#" onclick="visualizeByHistogram('shortPassing','Short Passing (ra
    <li><a id="preferredFoot" href="#" onclick="visualizeByBarchart('preferredFoot','Preferred Foot')
    <li><a id="intReputation" href="#" onclick="visualizeByBarchart('intReputation','Int Reputation (
    <li><a id="workRate" href="#" onclick="visualizeByBarchart('workRate','Work Rate')">Work Rate</a>
```

It also includes the main container where the svg gets appended and drawn.



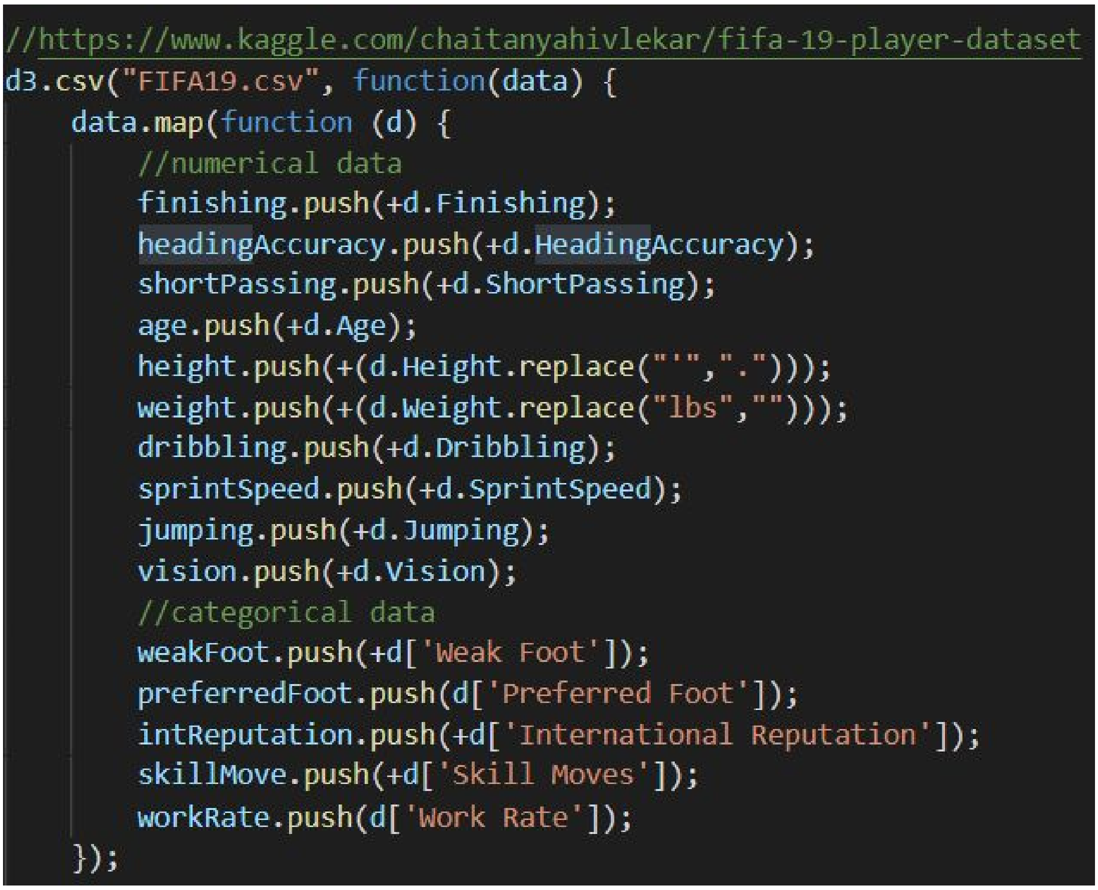
```
<div id="mainContainer" class="mainContainer" >
</div>
```

Index.html shows the histogram of Age vs. Player Count by default.



2. **main.js**

   main.js is mainly responsible for reading the various variables from the csv file and storing it in arrays which can be used for visualization.

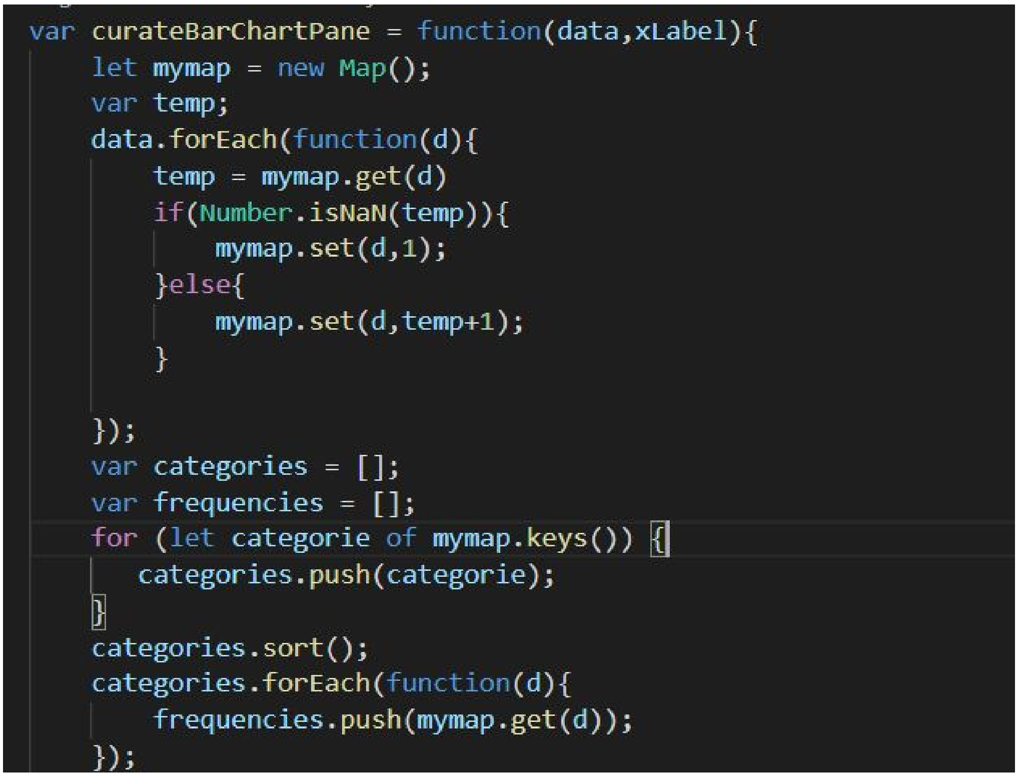


```
//https://www.kaggle.com/chaitanyahivlekar/fifa-19-player-dataset
d3.csv("FIFA19.csv", function(data) {
    data.map(function (d) {
        //numerical data
        finishing.push(+d.Finishing);
        headingAccuracy.push(+d.HeadingAccuracy);
        shortPassing.push(+d.ShortPassing);
        age.push(+d.Age);
        height.push(+(d.Height.replace("'",".")));
        weight.push(+(d.Weight.replace("lbs","")));
        dribbling.push(+d.Dribbling);
        sprintSpeed.push(+d.SprintSpeed);
        jumping.push(+d.Jumping);
        vision.push(+d.Vision);
        //categorical data
        weakFoot.push(+d['Weak Foot']);
        preferredFoot.push(d['Preferred Foot']);
        intReputation.push(+d['International Reputation']);
        skillMove.push(+d['Skill Moves']);
        workRate.push(d['Work Rate']);
    });
```

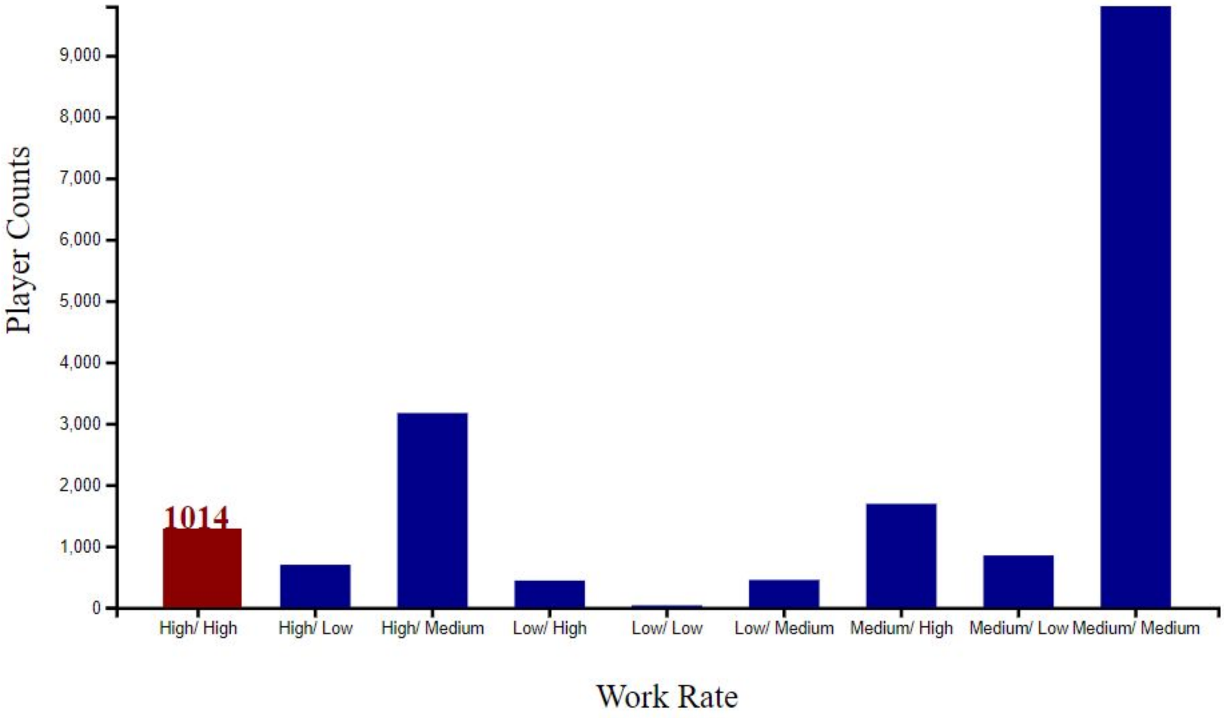
main.js also call the function for drawing the bar chart or histogram or scatter which is defined in their respective js file.

3. barchart.js

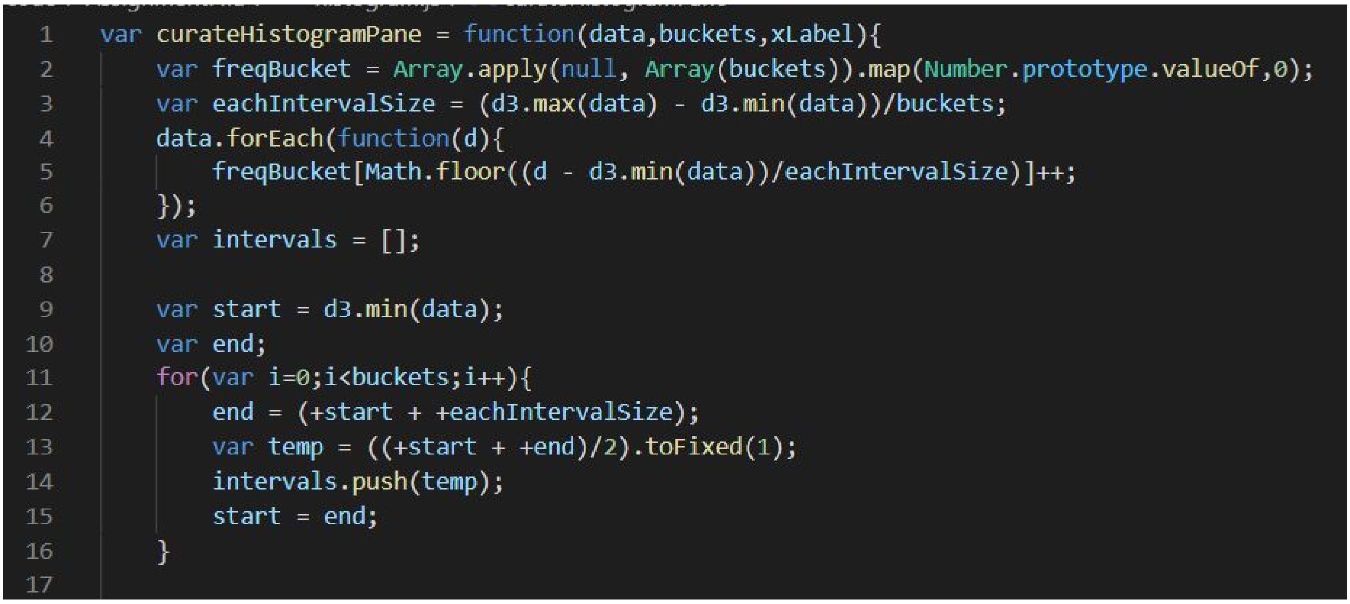
barchart.js is responsible for drawing the barchart. It defines a function **curateBarChartPane** which takes the data and label for the x axis and creates a map to get the respective count of individual elements and ultimately two arrays of elements and its count.



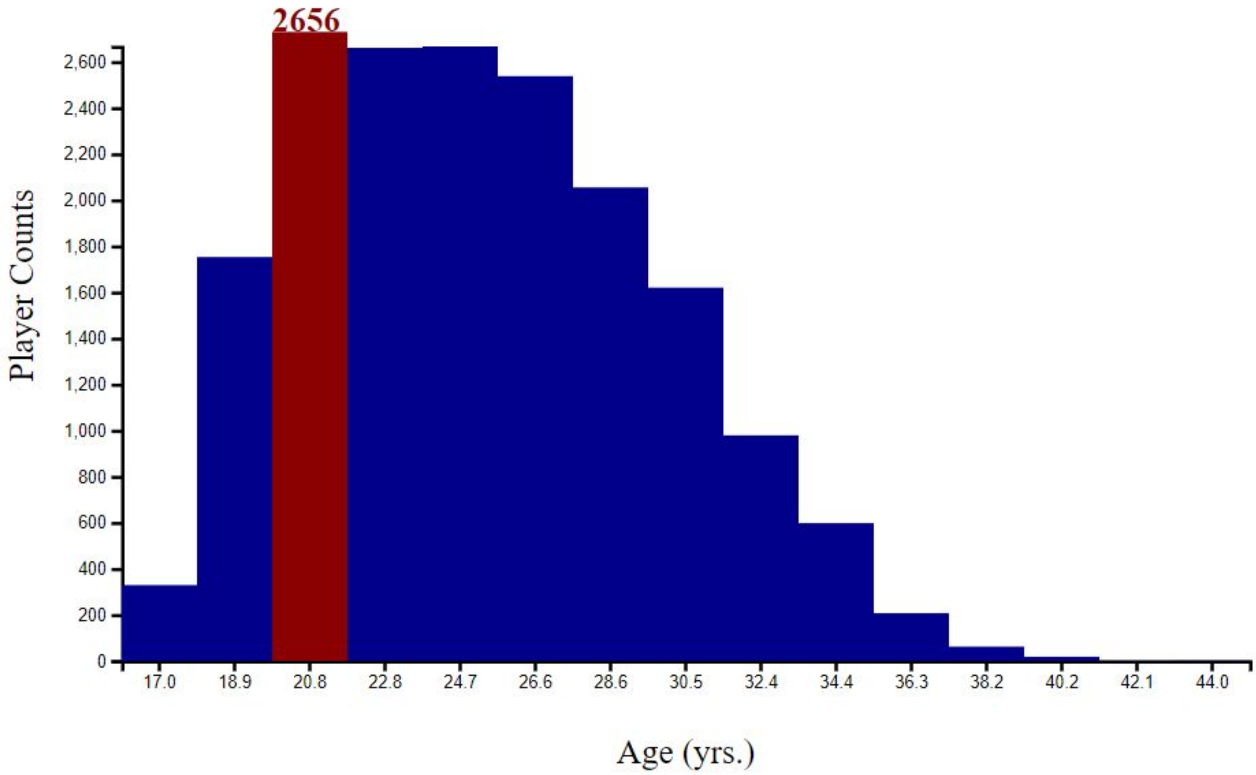
```
var curateBarChartPane = function(data,xLabel){
    let mymap = new Map();
    var temp;
    data.forEach(function(d){
        temp = mymap.get(d)
        if(Number.isNaN(temp)){
            mymap.set(d,1);
        }else{
            mymap.set(d,temp+1);
        }

    });
    var categories = [];
    var frequencies = [];
    for (let categorie of mymap.keys()) {
      categories.push(categorie);
    }
    categories.sort();
    categories.forEach(function(d){
        frequencies.push(mymap.get(d));
    });
```



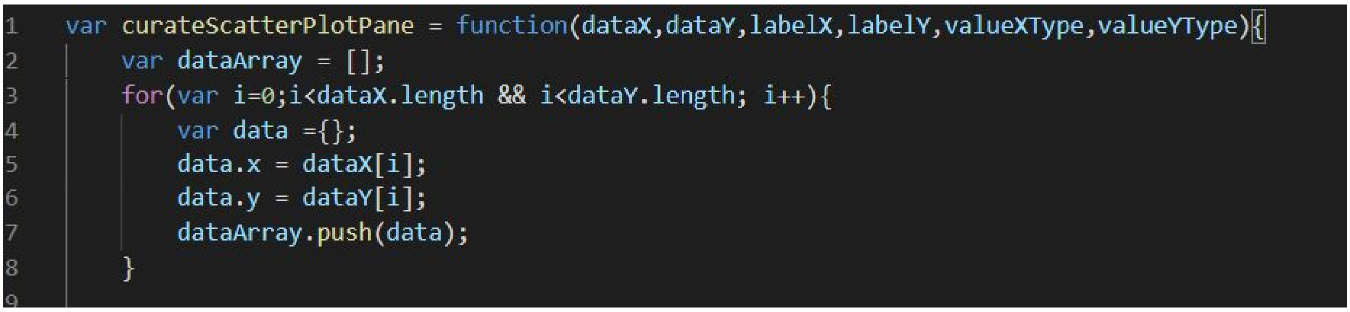
4. **histogram.js**
   histogram.js is mainly responsible for drawing the histogram. Number of buckets is passed to the function and it calculates the frequency in each bucket by dividing with the each interval size. Ultimately the histogram is drawn w.r.t calculated frequency in each bucket.



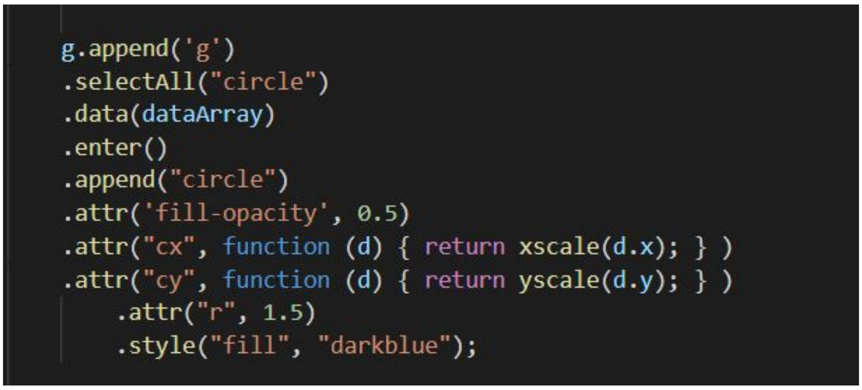
```
var curateHistogramPane = function(data,buckets,xLabel){
    var freqBucket = Array.apply(null, Array(buckets)).map(Number.prototype.valueOf,0);
    var eachIntervalSize = (d3.max(data) - d3.min(data))/buckets;
    data.forEach(function(d){
        freqBucket[Math.floor((d - d3.min(data))/eachIntervalSize)]++;
    });
    var intervals = [];

    var start = d3.min(data);
    var end;
    for(var i=0;i<buckets;i++){
        end = (+start + +eachIntervalSize);
        var temp = ((+start + +end)/2).toFixed(1);
        intervals.push(temp);
        start = end;
    }

```



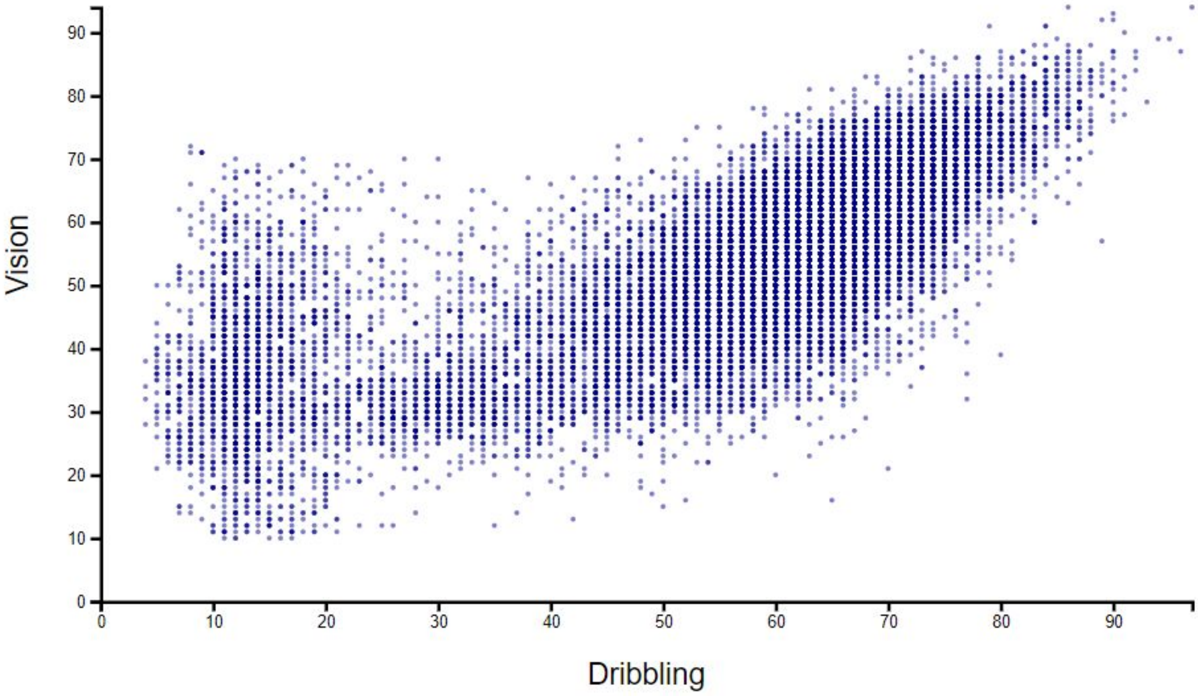
5. **scatterplot.js**
   scatterplot.js is mainly responsible for drawing the scatter plot between two variables. After the default scatter plot is shown, the user can select any variable and can choose the axis of the selected variable. It iterates over two selected variables and creates a dataArray having respective data points. With this dataArray scatter plot is drawn.



```
var curateScatterPlotPane = function(dataX,dataY,labelX,labelY,valueXType,valueYType){
    var dataArray = [];
    for(var i=0;i<dataX.length && i<dataY.length; i++){
        var data ={};
        data.x = dataX[i];
        data.y = dataY[i];
        dataArray.push(data);
    }
```



```
g.append('g')
.selectAll("circle")
.data(dataArray)
.enter()
.append("circle")
.attr('fill-opacity', 0.5)
.attr("cx", function (d) { return xscale(d.x); } )
.attr("cy", function (d) { return yscale(d.y); } )
    .attr("r", 1.5)
    .style("fill", "darkblue");
```



6. **fifa.css**
   fifa.css is responsible for styling the menu and positioning of the remaining elements.