Assignment 4: GIT(hub) Viz

Student Name	Seow Wei Jie	Chen Rui Wen
Matriculation Number	A0139515A	A0138431L

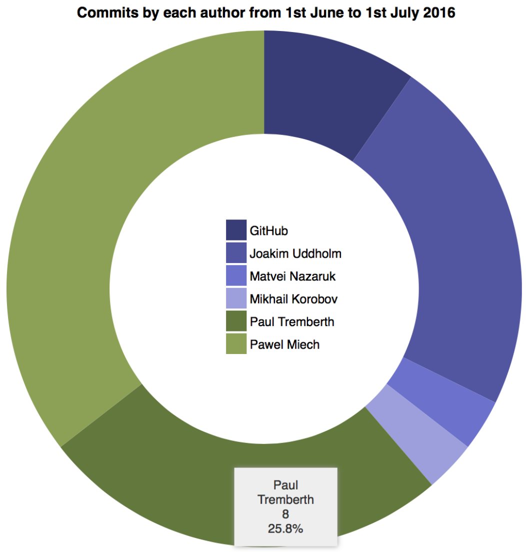
1. Introduction

In this assignment, D3 visualization tool will be used to display data retrieved from a github repository in a meaningful and informative manner. The use of D3 allows us to combine large chunk of data and present it with the use of appealing visuals. Hence, this highlights important information that is usually kept from the viewer if the data is not presented in this way. All in all, the most valuable use of D3 is to present data in a simple but effective manner. The code for each objective can be found in this link: https://github.com/weijieseow/CS3219A4-D3

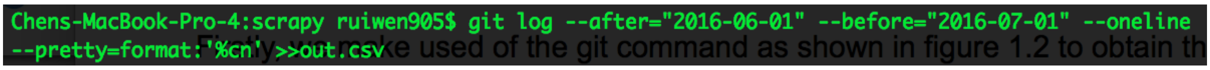
2. Visualizations - Purpose & Method

Objective	Done by:	Visualization
1	Rui Wen	Pie Chart
2	Rui Wen & Wei Jie	Grouped and Stacked Bar Chart
3	Wei Jie	Grouped Bar Chart

Objective 1 (Pie Chart)



The objective of this visualisation as shown above is to depict the commits by each author as a percentage of the total commits during the period from 1st June 2016 to 1st July 2016 for the given scrapy git repository.



```
Chens-MacBook-Pro-4:scrapy ruiwen905$ git log --after="2016-06-01" --before="2016-07-01" --oneline
--pretty=format:'%cn' >>out.csv
```

Firstly, we make used of the git command as shown above to obtain the list of authors who have committed from 1st June 2016 to 1st July 2016. Next, open “out.csv” using Microsoft Excel and utilise Excel functions such as sort and =SUM() to sort the authors’ names in alphabetical order and count the number duplicates for each distinct name.

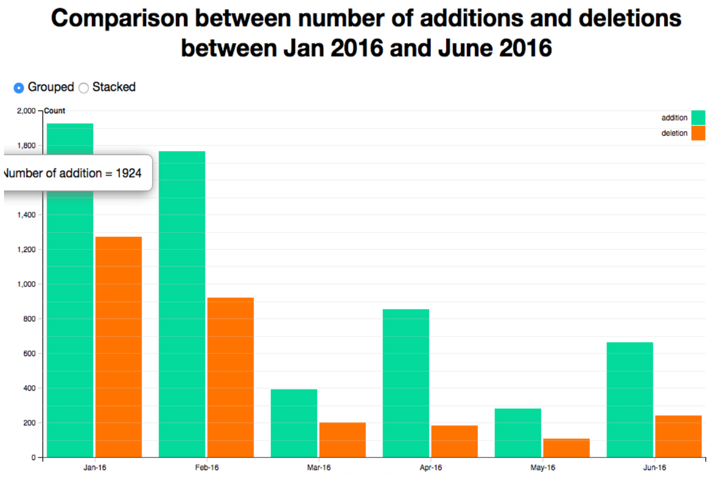
The last step would be to integrate our data into the code for a pie chart. We have chose to use a pie chart for this visualisation as a pie chart is useful in displaying data that are classified into nominal categories. Author's names are nominal data as they are descriptive and not quantitative information.

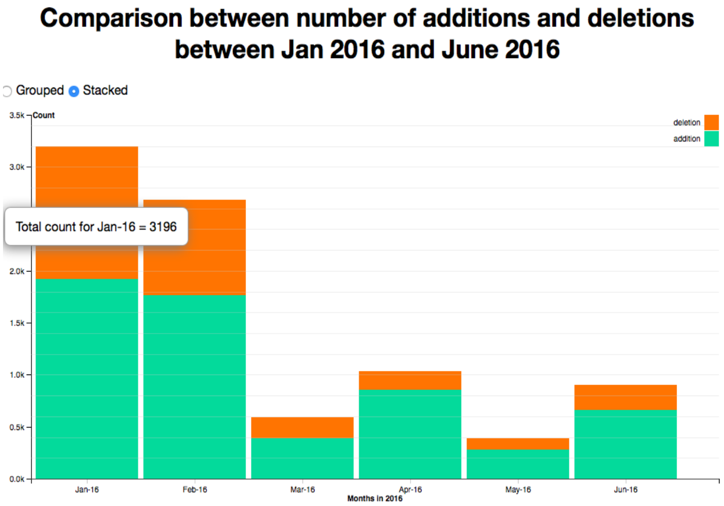
Pie charts are generally used to show percentage data as the arc length of each slice is proportional to the quantity it represents thus pie chart was chosen as the objective of this visualisation was to show the commits by each author as a percentage of the total commits.

As we analyse our results, we realise that we only have 6 distinct author's names to be displayed. Pie charts are good for displaying data for around 6 categories or fewer. When there are more categories it is difficult for the eye to distinguish between the relative sizes of the different sectors and so the chart becomes difficult to interpret.

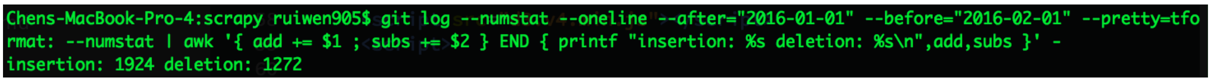
The tooltip provides additional information. When you hover your mouse over one of the sections, a tooltip with the author's name, the total number of commits and the percentage of that author's commits against the total number of commits within that period of time specified.

Objective 2 (Grouped and Stacked Bar Chart)





The objective of this visualisation as shown above is to depict the sums of additions by all authors and sums of deletions by all authors, for each month from Jan 2016 to June 2016 for the given scrapy git repository.



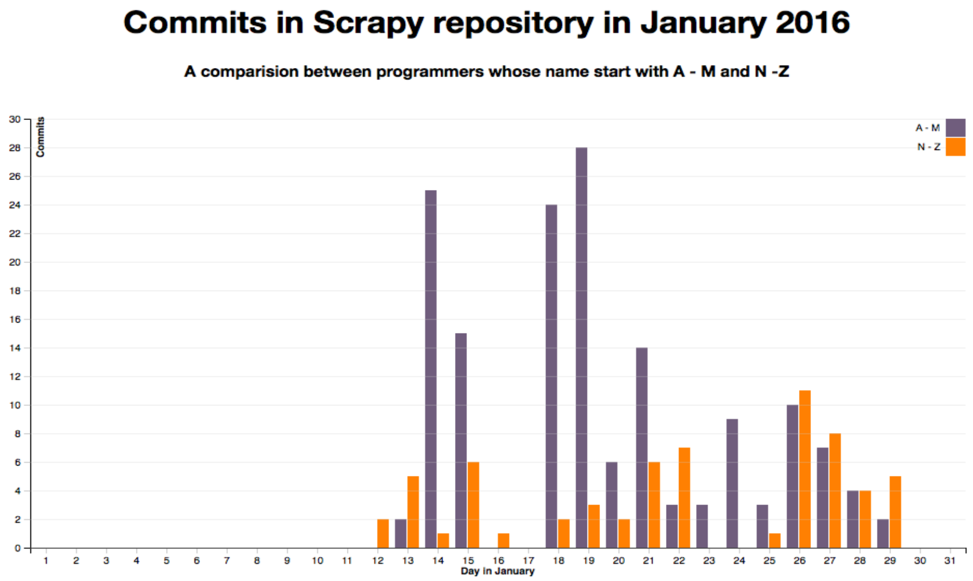
```
Chens-MacBook-Pro-4:scrapy ruiwen905$ git log --numstat --oneline --after="2016-01-01" --before="2016-02-01" --pretty=tfo
rmat: --numstat | awk '{ add += $1 ; subs += $2 } END { printf "insertion: %s deletion: %s\n",add,subs }' -
insertion: 1924 deletion: 1272
```

Firstly, we make used of the git command as shown above to obtain the sums of additions by all authors and sums of deletions by all authors for January 2016. Next, open a new .csv file in Microsoft Excel and manual input the number of insertion and deletion lines for each of the 6 months.

The last step would be to integrate our data into the code for bar chart. The grouped bar chart was adapted from https://bl.ocks.org/mbostock/3887051 and the stacked bar chart was adapted from https://bl.ocks.org/mbostock/3886208. We have chose to use bar chart for this visualisation as bar chart is useful in comparing the different series of data for each individual columns in the x-axis. In this case, a **grouped bar chart** make it easier for viewers to compare the total number of insertion lines against the total number of deletion lines in each month. Viewers can also toggle the visualisation into a **stacked bar chart** using the radio button at the top left corner, to view the proportional of either the sum of insertion or deletion against the total number of changed line for each month.

This toggling is achieved by this line of code **“d3.selectAll("svg > g > *").remove();”** which removes the current selected elements. Then after that we populate the data of the new graph with the layout that we want. Instead of stacked bar chart and grouped bar chart, interested developers can also explore it with the use of other charts using this method.

Objective 3 (Grouped Bar Chart)



In this objective, the goal is the present commits by two groups of programmers (names starting with A-M and N-Z) in the January 2106. Also, this information have to be displayed in such a way that viewer can immediately see the daily difference between each groups. As such, the most suitable chart to use will be a grouped bar chart to display commits of each group side by side for daily comparison. This grouped bar chart was adapted from https://bl.ocks.org/mbostock/3887051 and I have included tooltip and y-axis lines to increase the ease of reading of information.

First, we will need to retrieve the information from the git repository using git log command. Here, I made used of a script (https://gist.github.com/textarcana/1306223) to convert information from git log to a JSON object that be used for filtering of information. The use of the script is as follows: 1) Navigate to the repository 2) run the script and direct the output to a json file 3) done.

Next, I wrote a node.js program that reads the JSON object created and retrieve the commits from two groups of programmers in January 2016. Then, I convert this information into a CSV file that is used to populate the grouped bar chart in D3. More information of the node.js program can be found here: https://github.com/weijieseow/CS3219A4-D3/blob/master/obj3/handleJson.js

The last step will to be modifying the grouped bar chart source code to match the CSV file and adding additional visualization to make information display better. We have chosen to use a grouped bar chart for this visualisation as a grouped bar chart is useful in comparing the different series of data for each individual columns in the x-axis. In this case, a grouped bar chart makes it easier for viewers to compare the total number of programmer’s names starting with A-M commits against the total number of programmer’s names starting with N-Z commits in each day. The tooltip provides additional information. When you hover your mouse over one of the sections, a tooltip with the name of that series (A-M or N-Z) and its respective total number commits will be shown.