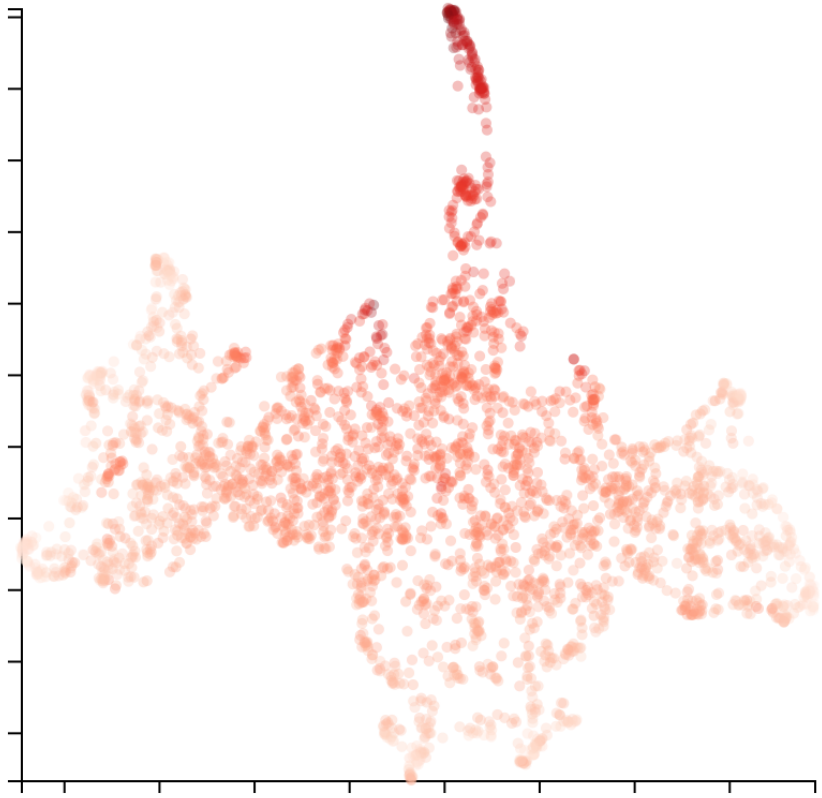
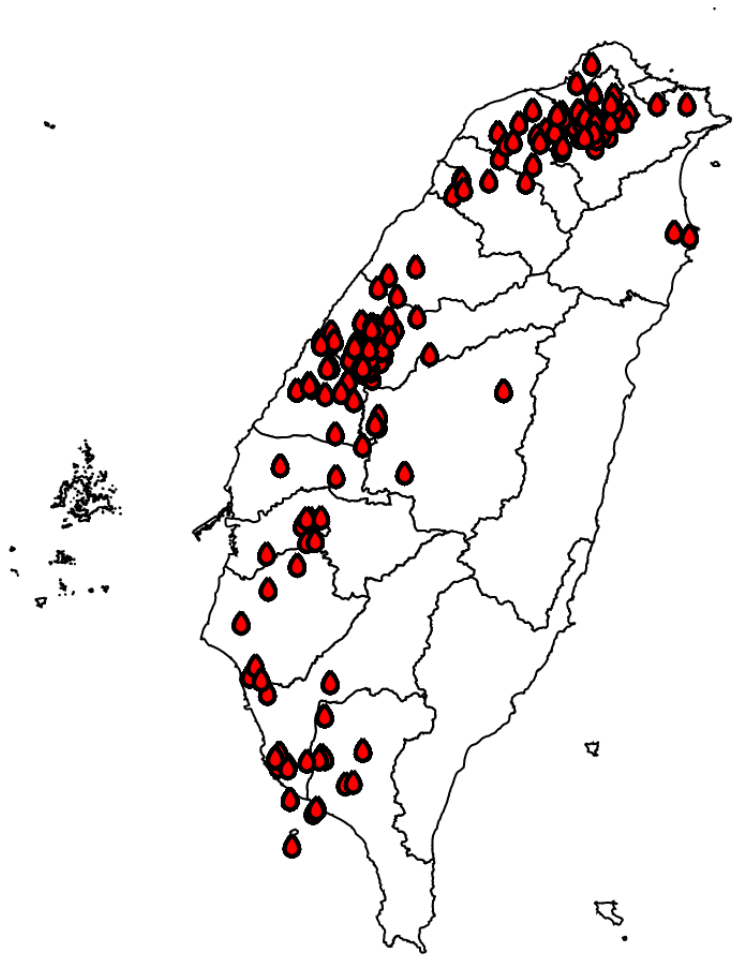
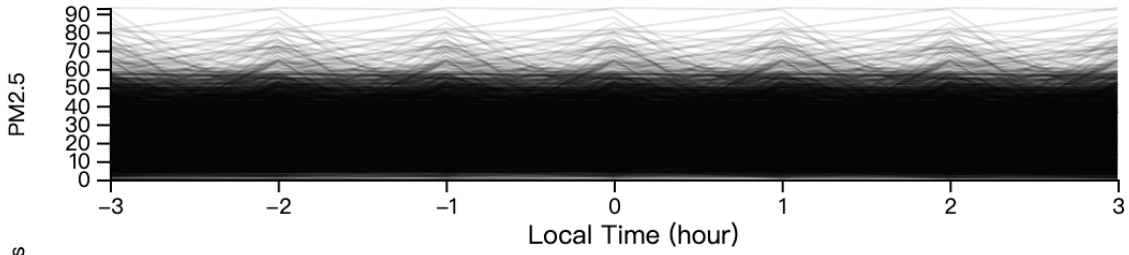
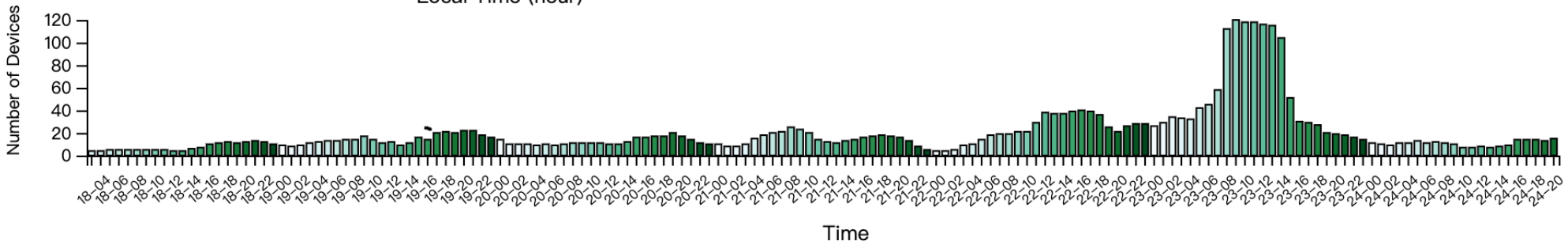
D3: Static Visualization

Homework2






PM2.5
90
80
70
60
50
40
30
20
10
0
−3
−2
−1
0
1
2
3
Local Time (hour)


Number of Devices
120
100
80
60
40
20
0
Time

Dataset

- highPollution.csv
  - I have done a process of data filtering and data transformation from airBoxData.csv(python HW1) and stored the result in highPollution.csv
  - In highPollution.csv, each record is a consecutive 7 hours PM2.5 values. In addition, a record is kept in the file only if at least one of the PM2.5 value is greater than 35. Therefore, I call it ”highPollution.csv”
  - Data attributes
    - siteID: device ID
    - year, month, day, hour, weekday: time stamp of a record (median from the 7 hours)
    - value: PM2.5 values
    - umapX, umapY: the result of projection ‘value’ vectors to 2D space
- taiwan.json: map of Taiwan

Deadline?

- What do we have to know (D3) to make this homework done?
  - 01_BeforeD3
  - 02_Selection
  - 03_DataBinding
  - 04_Scale_Axis
  - 05_BasicShape
  - 06_Map
- You will have 2 weeks from now on to do this homework

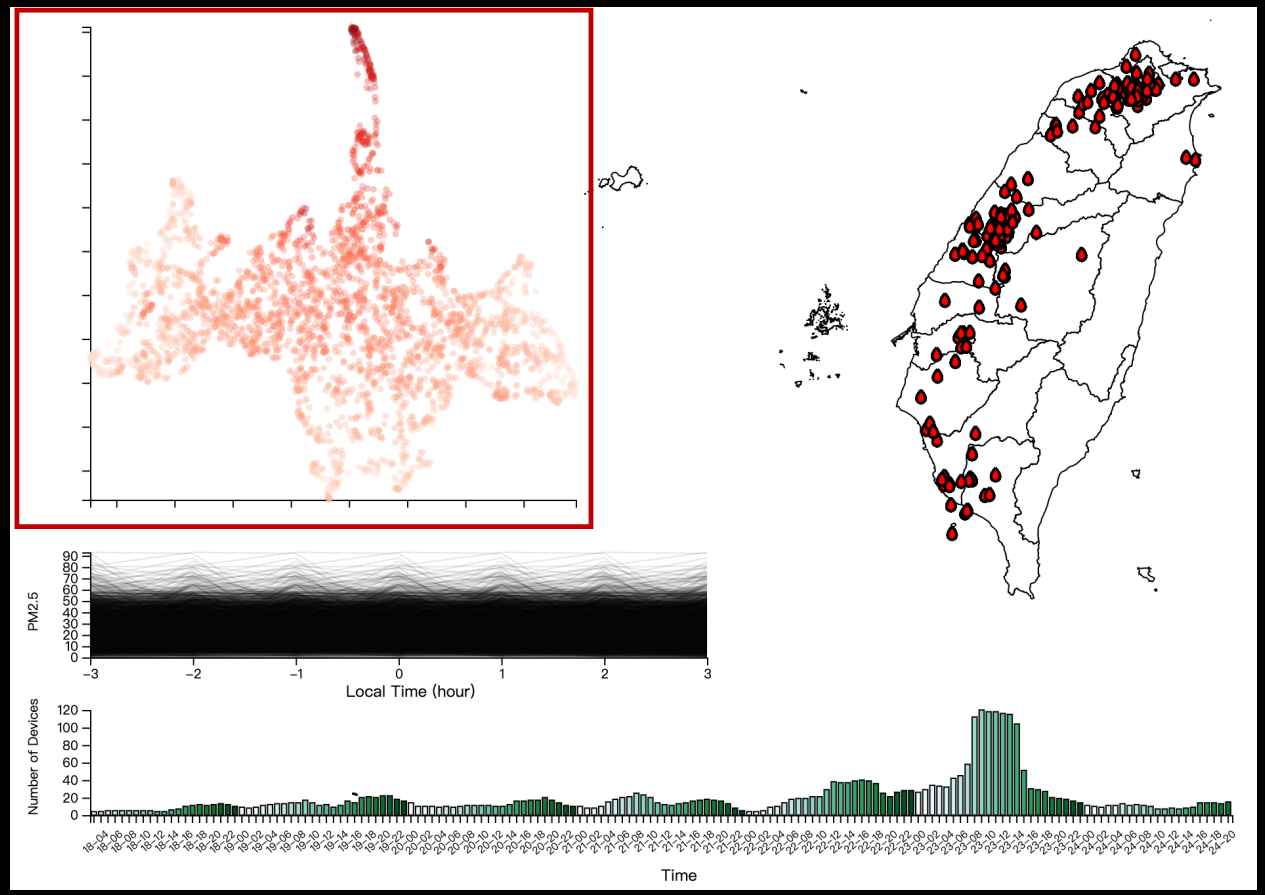
Practice1 : Javascript Data Process

- Step 1: convert the data attributes to “Number” or the data type you want in the very beginning after you load the data
  - For example: after you load highPollution.csv, Javascript considers most of the attributes as “string” data type, and that is not convenient for the following data visualization.

- The data attribute, “value”, is an array and stored as a string. You may want to transform them to multiple “floats”
  - E.g. [14.6, 11.6, 11.1, 17.0, 23.3, 30.55555556, 35.5]

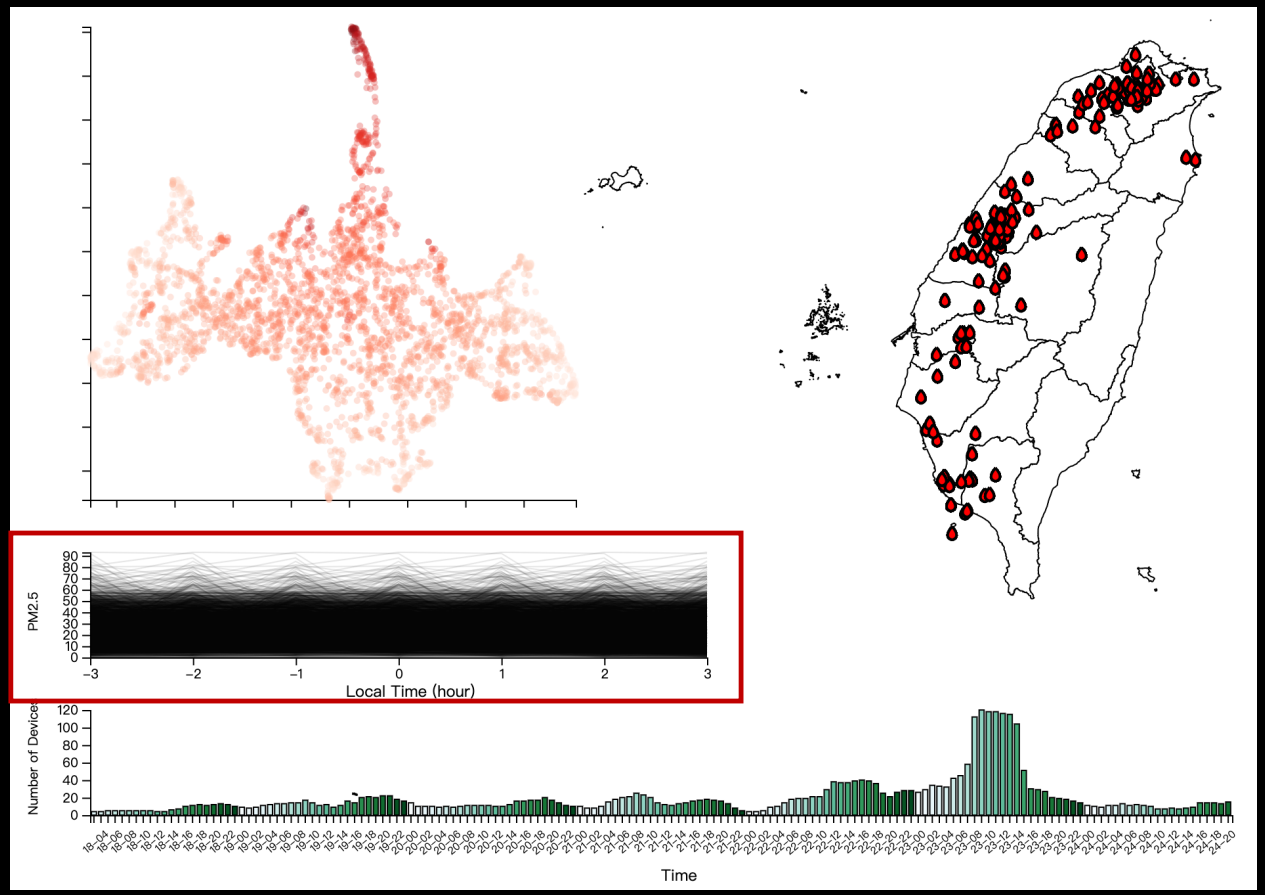
Practice 2: Scatterplot

- Visual channel “X position” encodes data attribute “umapX”
- Visual channel “Y position” encodes data attribute “umapY”
- The color of circles encodes the average of data attribute “value”
  - Use this color map, “d3.interpolateReds”
- Plot x and y axes without tick text
- No x and y axis labels



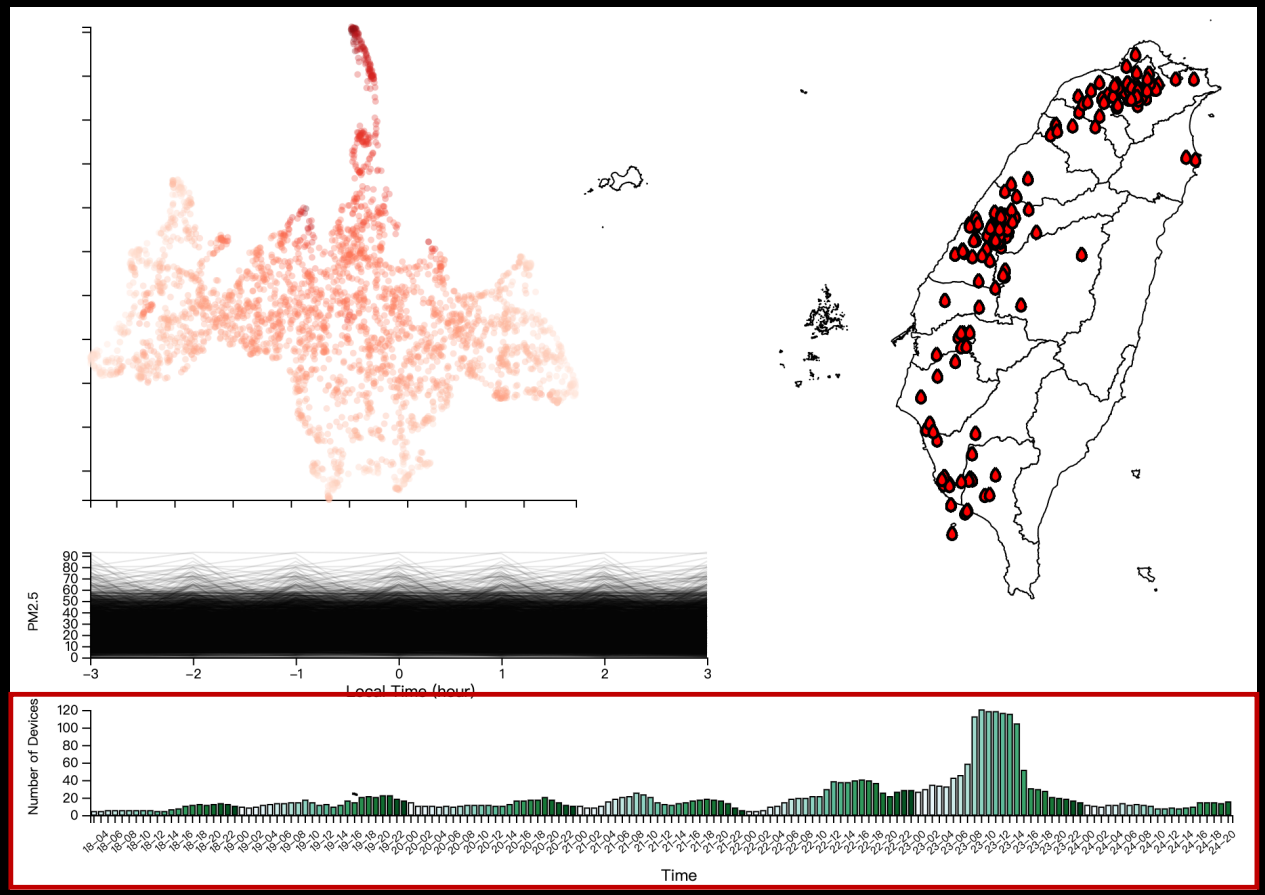
Practice 3: Line Chart

- Each line represents the data attribute ‘value’ of a data item
  - connect 7 values to be one line
  - Set ‘opacity’ of each line to 0.1
- Remember to draw x, y axes, label and ticks (as what I show in the figure)
  - y axis range should set to the min and max values of the data attribute ’value’



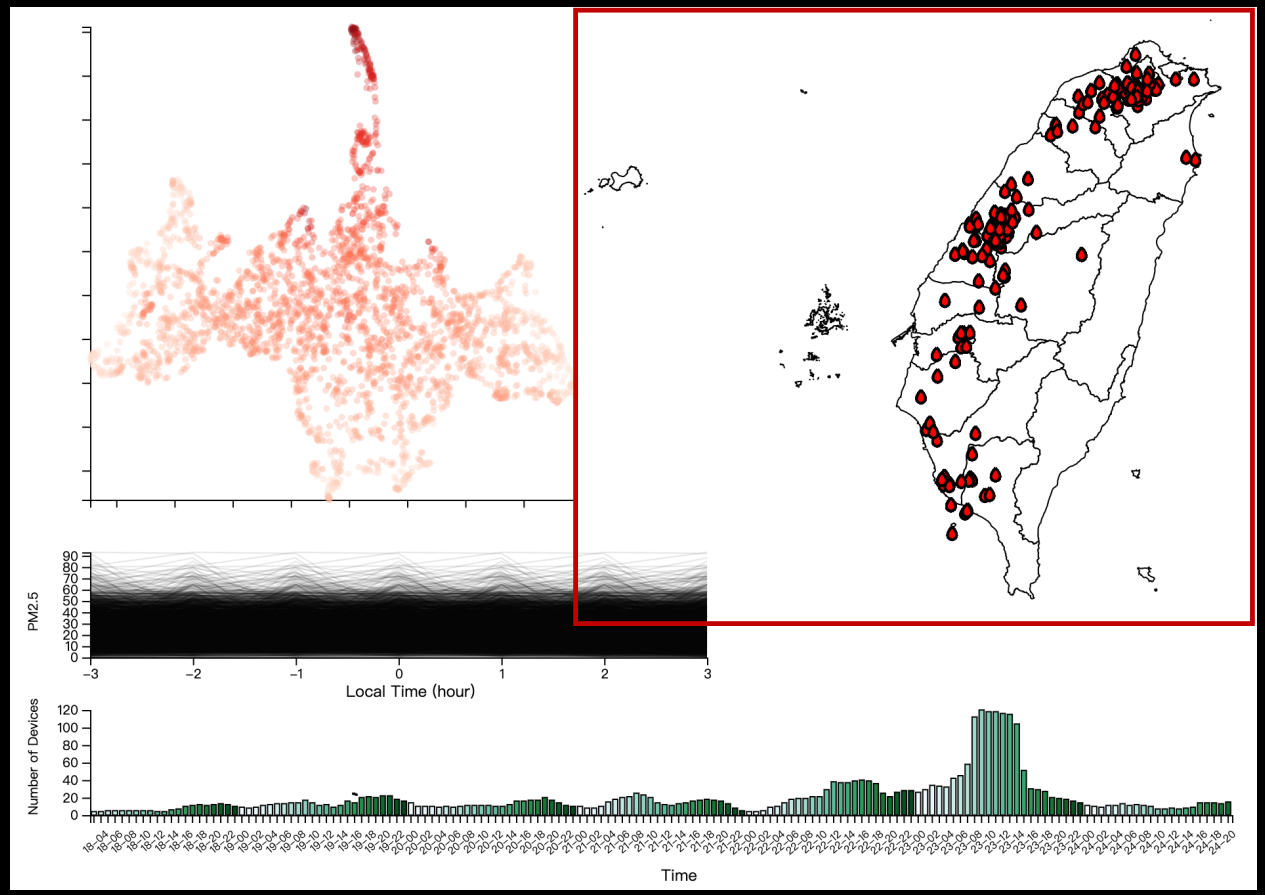
Practice 4: Bar Chart

- Visualize the number of records (number of devices) of each hour
  - You should calculate the number records for each hour first
- X axis is “day and hour”
  - Make sure the x axis tick texts do not overlap with each other
    - You can draw the x axis tick texts every two hours
    - Rotate the texts
- Y axis is the number of records
- The color of bar encodes “hour”
  - Use this color map, d3.interpolateBuGn
- Remember to add x and y axis labels (as what I show in the figure)



Practice 5: Map and Customized Mark

- Visualize locations of records on the map (taiwan.json)
- Each record is shown by a “customized mark” on the map
  - A customized mark is created by a ”<path>”
  - You can check this tutorial https://observablehq.com/@philippkoytek/d3-advanced-drawing-custom-shapes and this website https://mavo.io/demos/svgpath/ to create your own mark
- Because multiple records could be from the same device, you will draw multiple marks at exactly same location. That is ok.



Note

- You can refer to this setup to manage the plots
  - From the top to bottom, they are the layout for the scatterplot, line chart, bar chart and map.
  - If you do not know what is this for, check Ex04-16 in our D3 lecture.
  - If you have your own way to manage your plots, you do not have to follow this suggestion.
- I strongly encourage you to well structure your code because your D3 Homework 3 will an extension of this homework. In addition, I may ask you to move you plots around (or resize these plots) in next D3 homework.

```
let scatterLeft = 0, scatterTop = 0;
let scatterTotalWidth = 500, scatterTotalHeight = 400;
let scatterMargin = {top: 10, right: 30, bottom: 30, left: 100},
    scatterWidth = scatterTotalWidth - scatterMargin.left - scatterMargin.right,
    scatterHeight = scatterTotalHeight - scatterMargin.top - scatterMargin.bottom;

let lineLeft = 0, lineTop = 400; //400 actually is scatterTotalHeight
let lineTotalWidth = 600, lineTotalHeight = 100;
let lineMargin = {top: 10, right: 30, bottom: 10, left: 100},
    lineWidth = lineTotalWidth - lineMargin.left - lineMargin.right,
    lineHeight = lineTotalHeight - lineMargin.top - lineMargin.bottom;

let barLeft = 0, barTop = 500; //500 actually is scatterTotalHeight + lineTotalHeight
let barTotalWidth = 1000, barTotalHeight = 150;
let barMargin = {top: 30, right: 30, bottom: 40, left: 100},
    barWidth = barTotalWidth - barMargin.left - barMargin.right,
    barHeight = barTotalHeight - barMargin.top - barMargin.bottom;

let mapLeft = 500-200, mapTop = 0-150;
let mapTotalWidth = 1000, mapTotalHeight = 800;
let mapMargin = {top: 10, right: 10, bottom: 10, left: 10},
    mapWidth = mapTotalWidth - mapMargin.left - mapMargin.right,
    mapHeight = mapTotalHeight - mapMargin.top - mapMargin.bottom;
```

Seek for Help?

- Instructor: Ko-Chih Wang (kcwang@ntnu.edu.tw)

- Our TA Team:
  - 志融 (60947081S@gapps.ntnu.edu.tw)
  - 柏丞 (60947015S@gapps.ntnu.edu.tw)
  - 旭廷 (60947072S@gapps.ntnu.edu.tw)
  - 建霖 (60947092S@gapps.ntnu.edu.tw)
  - lab location: 應用科學大樓109