Date:05/04

Presented by Willy/Shou

目錄

Outline

回顧

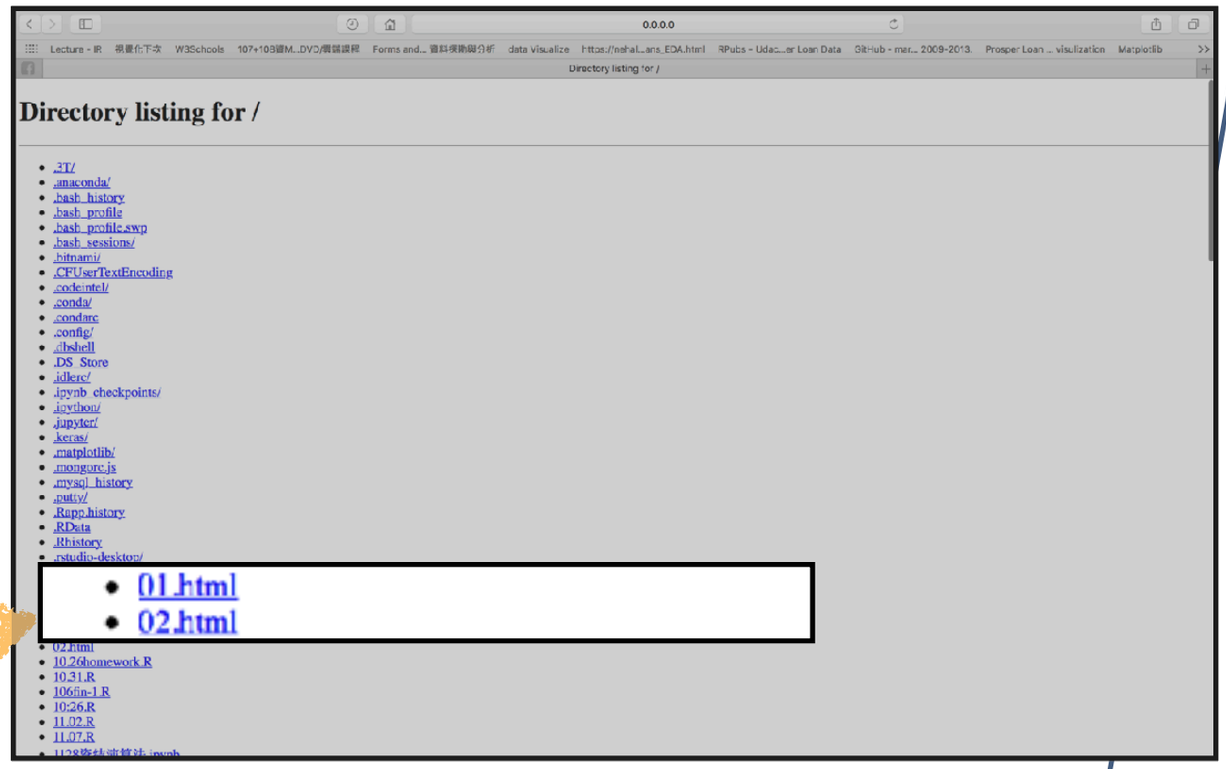
工具 / 準備事項

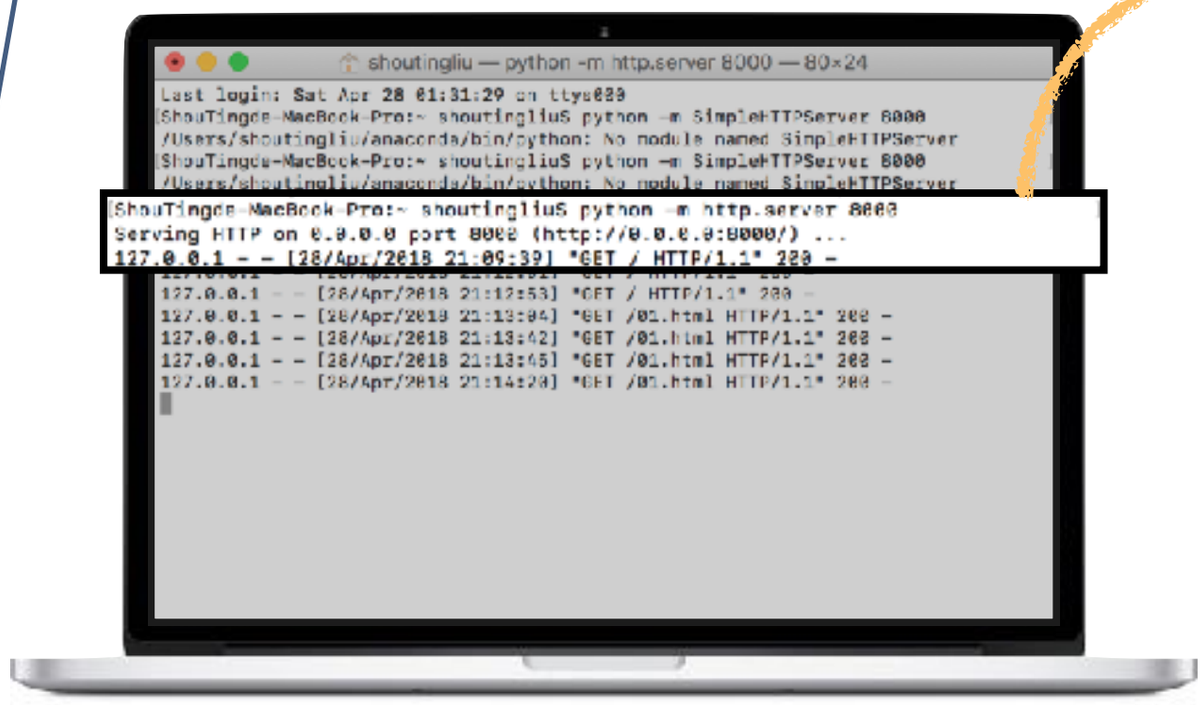
Sublime Text

Google Chrome

另一種方法

python -m http.server 8000





回顧

Scales

select( )傳回所選元素

append( )
新增元素

attr( )
設定屬性

```
// selection
d3.select('body')

// adding elements
var svg = d3.select('body').append('svg');

// setting attributes
svg.attr({"width":500,"height":500})

// appending data
// data
var data_values = [10,20,30,40,50,5,2,12,70,26]
// create rectangles
var bars = svg.selectAll("rect")
.data(data_values)
.enter()
.append("rect")
.attr("width","25px")
.attr("height", function(d){ return d; });
```

data( )綁定資料到元素上

enter( )沒有足夠的元素時增加元素

進階概念

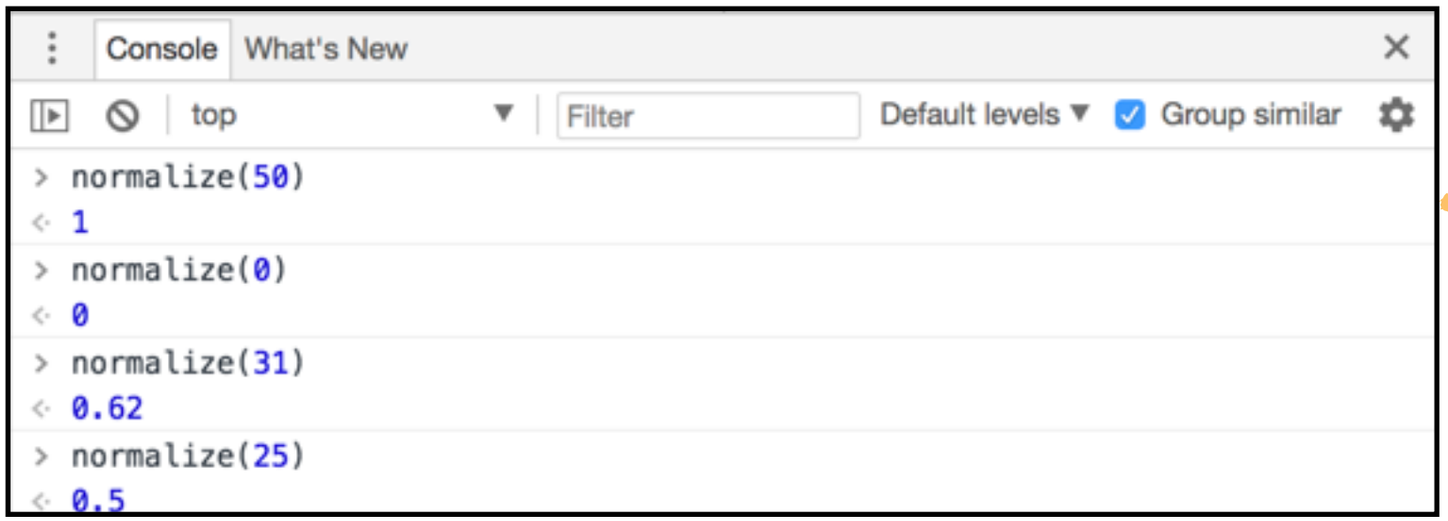
Scales

創一個線性標度

domain( )設定輸入範圍

range( )設定輸出範圍

```
    // Create a linear scale
var normalize = d3.scale.linear().domain([0,50]).range([0,1]
    );
```

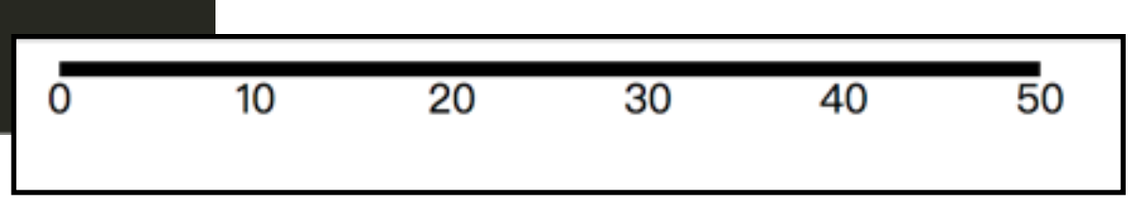


進階概念

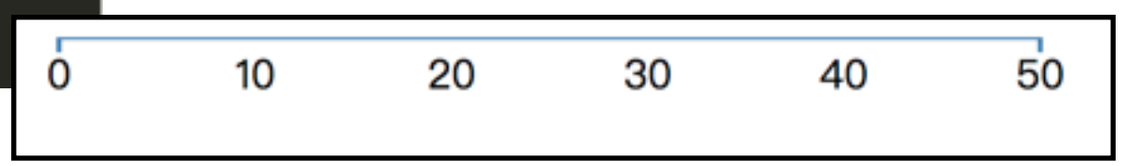
Axis

```
//create an svg element
var svgElement = d3.select("body")
.append("svg")
.attr({"width" : 500, "height" : 500});
//create a linear scale to map data to pixels, domain is [
0,50] and range is [10,400]
var xScale = d3.scale.linear().domain([0,50]).range([10,400]
    );
//create a axis based on the scale
var xAxis = d3.svg.axis().scale(xScale)
.ticks(5) //limit number of ticks to 5
.orient("bottom"); //horizontal axis with labels to  the
bottom
//create a new group to hold the axis
var x = svgElement.append("g")
.call(xAxis);
```

創建一個svg的軸線再套用尺度




```
<style type="text/css">
    path{
stroke: steelblue;
fill: none;
stroke-width: 2;
}
</style>
```



加入css的樣式

進階概念

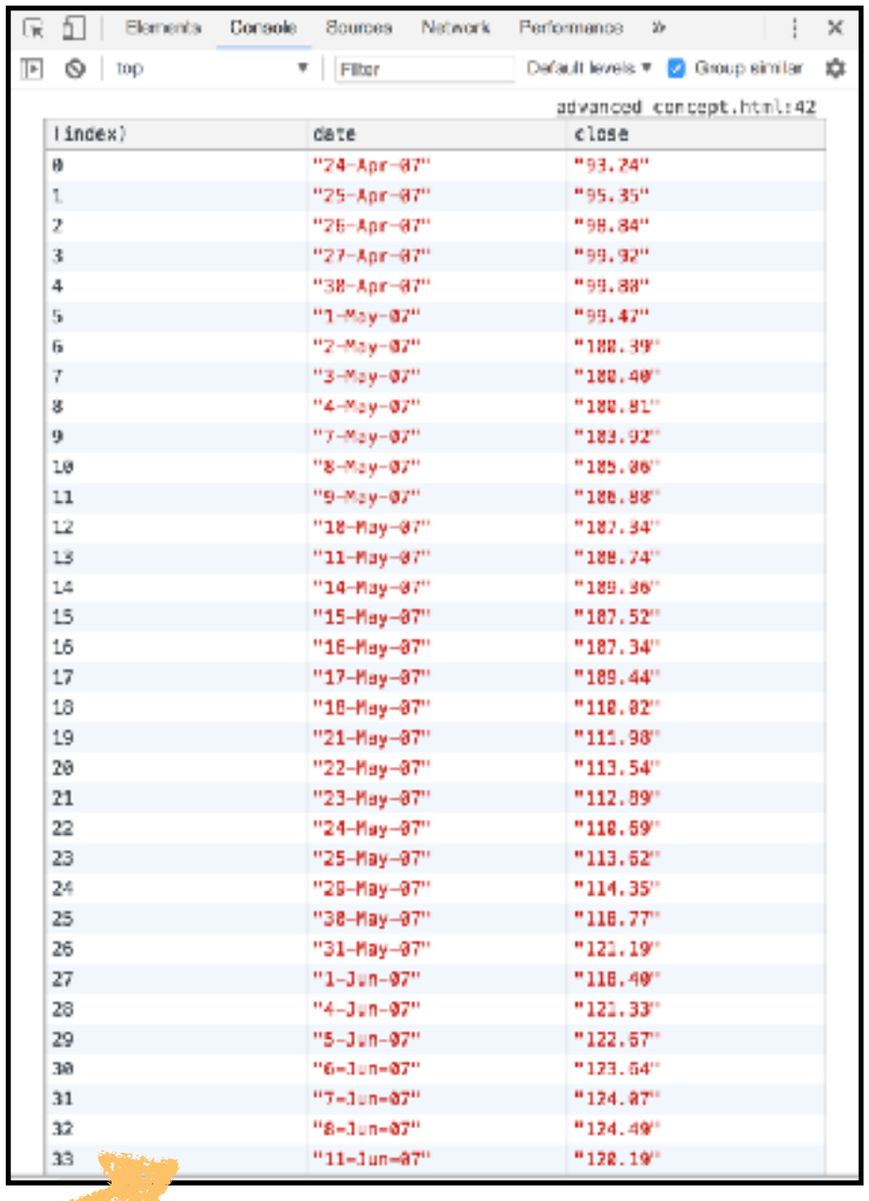
Loading External Data

載入資料

如果錯誤就在console的地方print error

反之則print data

```
d3.tsv("http://simplysanad.com/d3js/data.tsv", function(
    error, data){
if(error)
// If error, print the error message in the console
console.log(error);
else
// Print the data as a table in the console
console.table(data);
});
```



(index)	date	close
0	"24-Apr-07"	"93.24"
1	"25-Apr-07"	"95.35"
2	"26-Apr-07"	"98.84"
3	"27-Apr-07"	"99.92"
4	"30-Apr-07"	"99.80"
5	"1-May-07"	"99.47"
6	"2-May-07"	"100.39"
7	"3-May-07"	"100.40"
8	"4-May-07"	"100.81"
9	"7-May-07"	"103.92"
10	"8-May-07"	"105.06"
11	"9-May-07"	"106.88"
12	"10-May-07"	"107.34"
13	"11-May-07"	"108.74"
14	"14-May-07"	"109.36"
15	"15-May-07"	"107.52"
16	"16-May-07"	"107.34"
17	"17-May-07"	"109.44"
18	"18-May-07"	"110.02"
19	"21-May-07"	"111.98"
20	"22-May-07"	"113.54"
21	"23-May-07"	"112.89"
22	"24-May-07"	"110.69"
23	"25-May-07"	"113.62"
24	"28-May-07"	"114.35"
25	"30-May-07"	"118.77"
26	"31-May-07"	"121.19"
27	"1-Jun-07"	"118.40"
28	"4-Jun-07"	"121.33"
29	"5-Jun-07"	"122.67"
30	"6-Jun-07"	"123.64"
31	"7-Jun-07"	"124.07"
32	"8-Jun-07"	"124.49"
33	"11-Jun-07"	"120.19"

小試身手

Bar Chart

1.先建好一個基本架構

```
<!DOCTYPE html>
<html>
<head>

   <meta charset="utf-8">    <!-- 可以顯示中文 -->
   <title>好好玩的barchart</title> <!-- 網頁標題 -->
<script src="http://d3js.org/d3.v3.min.js"></script>
</head>

<body>
<script type="text/javascript">

</script>
</body>
</html>
```

2.建一個data array

```
var data_values = [5,10,30,8,45,24,16,55,60,45,32,18,11,
    3];
```

3.創一個寬500px高500px的svg

```
var svg = d3.select("body").append("svg").attr({"width":
    500,"height":500});//建一個svg的空間，設定長寬
```

小試身手

Bar Chart

4.定義bars、max、min

```
    var bars = svg.selectAll("rect").data(data_values).enter
        ().append("rect").attr("width","25px").attr("height"
        , function(d){ return d*5; });
var max = d3.max(data_values);
var min = d3.min(data_values);
```

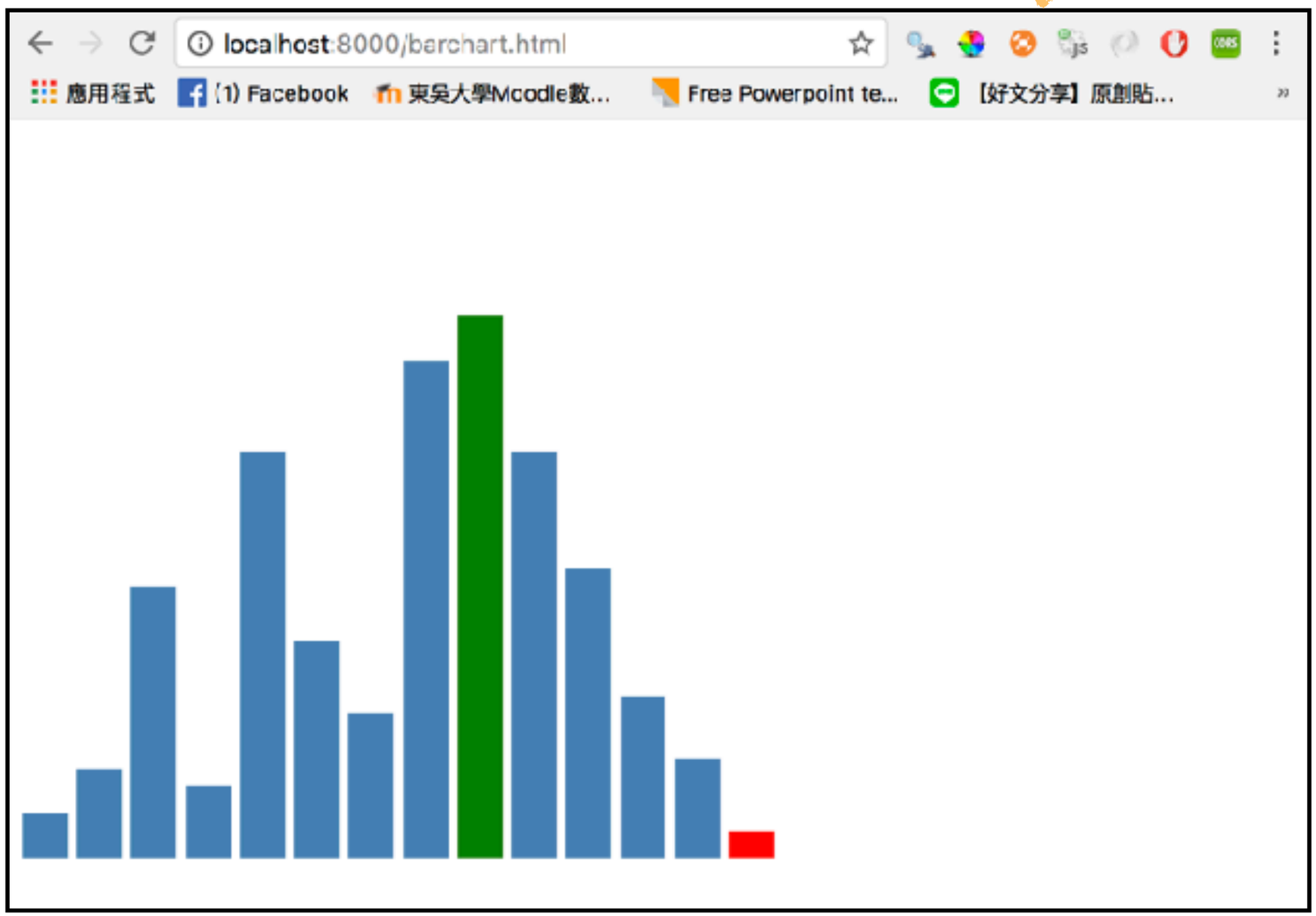
5.增加bars屬性

```
bars.attr("x", function(d,i){ return i*30; });
bars.attr("y", function(d){ return 500-d*5; });
bars.attr("fill", "steelblue");
bars.filter(function(d){ return d==max; }).attr("fill","green");
bars.filter(function(d){ return d==min; }).attr("fill","red");
```

小試身手

Bar Chart

跑出來的結果



基本圖型

Line Chart

```
//Set margins and sizes
var margin = {
top: 20,
bottom: 50,
right: 30,
left: 50
};

var width = 700 - margin.left - margin.right;
var height = 500 - margin.top - margin.bottom;
```

設邊際 再定義長寬

parse是轉換為原本的時間格式

ex:12/01/2014
->Mon Dec 01 2014 00:00:00 GMT+0800

```
//Create date parser(把原本的數值轉成格式化)
var ParseDate = d3.time.format("%d-%b-%y").parse;
//Create x and y scale(尺度) to scale inputs
//range是output
var xScale = d3.time.scale().range([0, width]);
var yScale = d3.scale.linear().range([height, 0]);
```

```
//Create x and y axes(座標軸)
var xAxis = d3.svg.axis().scale(xScale)
.orient("bottom")//向下靠
.ticks(5);
var yAxis = d3.svg.axis().scale(yScale)
.orient("left")//向左靠
.ticks(5);
```

x軸向下靠y軸向左靠
ticks()代表要幾個間隔

基本圖型

Line Chart

```
//Create a line generator
var valueline = d3.svg.line()
.x(function(d){
return xScale(d.date);
})

.y(function(d){
return yScale(d.close);
});
```

定義線
x尺度使用data那欄資料
y則是使用close那欄資料

```
//Create an SVG element and append it to the DOM
var svgElement = d3.select("body").append("svg")
.attr({"width": width+margin.left+margin.right, "height":
    height+margin.top+margin.bottom})
.append("g")
.attr("transform","translate("+margin.left+","+margin.top+"
    )");//g的位置從原點跑到(margin.left,margin.top)
```

加一個svg element
再定義屬性

```
//Read TSV file
d3.tsv("http://simplysanad.com/d3js/data.tsv", function(
    data){
//Parse Data into useable format
data.forEach(function(d){
d.date = ParseDate(d.date);
d.close = +d.close;//從串轉成數字
//the + sign converts string automagicall
});
```

匯入資料
“+”的符號是自動將字串轉為數字

基本圖型

Line Chart

```
//Set the domains(input的數值) of our scales
xScale.domain(d3.extent(data, function(d){ return d.date; }
    ));//ex:d3.extent([1,2,3,4,5])->[1,5]
yScale.domain([0, d3.max(data, function(d){ return d.close;
    })]);
```

先分別設x y的線性標度

```
//append the svg path
var path = svgElement.append("path")
.attr("d", valueline(data));

//Add X Axis
var x = svgElement.append("g")
.attr("transform", "translate(0,"+height+")")
.call(xAxis);

//Add Y Axis
var y = svgElement.append("g")
.call(yAxis);
```

先增加一個svg path
再加上x、y軸

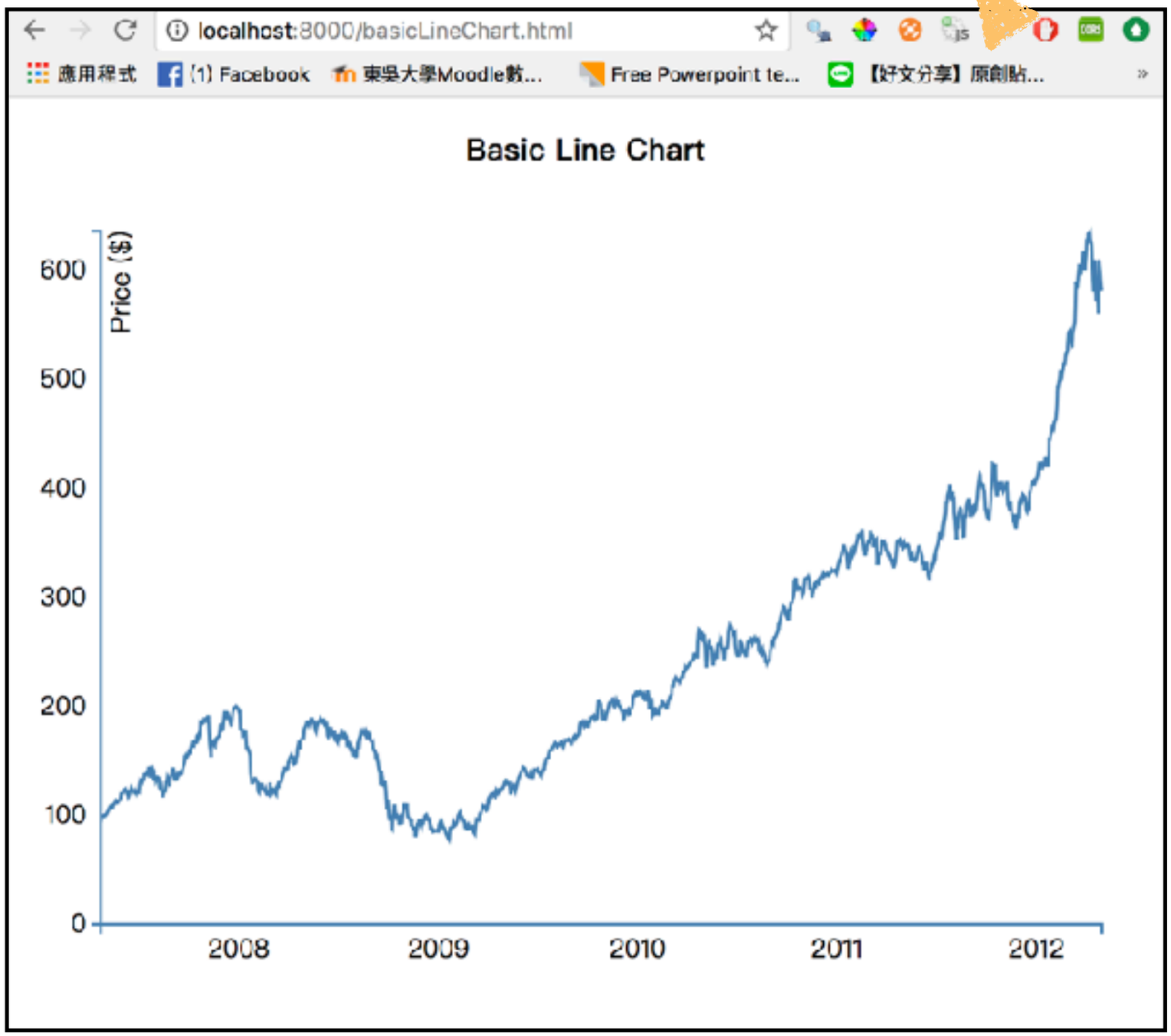
```
//Add label to y axis
y.append("text")
.attr("fill", "#000")
.attr("transform", "rotate(-90)")//旋轉90度
.attr("y", 6)
.attr("dy", "0.71em")
.attr("text-anchor", "end")
.text("Price ($)");
});
```

增加y軸的標籤屬性
其中rotate()代表旋轉度數

基本圖型

Line Chart

跑出來的結果



基本圖型

Area Chart

將原本的line改成area即可

```
//Create a line generator
var valueline = d3.svg.line()
.x(function(d){
return xScale(d.date);
})

.y(function(d){
return yScale(d.close);
});
```

```
//Create a area generator
var area = d3.svg.area()
.x(function(d){
return xScale(d.date);
})
.y1(function(d){
return yScale(d.close);
});
```

```
//append the svg path
var path = svgElement.append("path")
.attr("d", valueline(data));
```

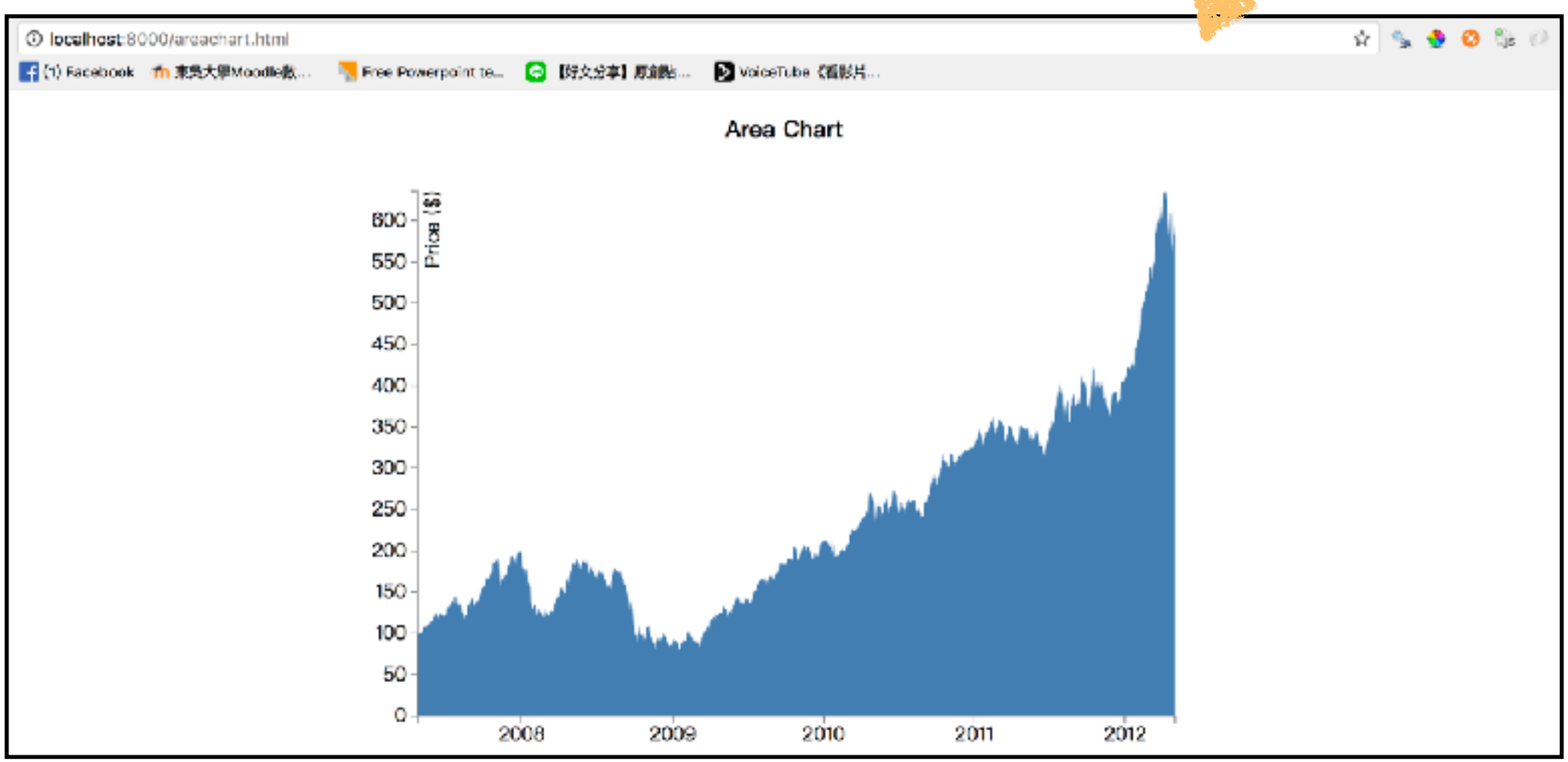
```
//append the svg path
var path = svgElement.append("path")
.attr("d", area(data))
.attr("fill", "steelblue");
```

```
//Set the domains(input的數值) of our scales
xScale.domain(d3.extent(data, function(d){ return d.date; }
    ));//ex:d3.extent([1,2,3,4,5])->[1,5]
yScale.domain([0, d3.max(data, function(d){ return d.close;
    })]);
area.y0(yScale(0));
```

基本圖型

Area Chart

跑出來的結果



小結

Conclusion

D3的大概脈絡

1. 基本的HTML框架跟CSS
2. 設定舞台
3. 針對特定的設定視覺化
4. 創建SVG
5. 載入外部資料
6. Mix!
7. Extras

範例研究

Case Study

step1：基本的HTML框架 跟 CSS

```
<!DOCTYPE html>
<html>
<head>
  <script src="d3.min.js"></script>
  <style type="text/css">
    .axis path,
    .axis line {
      fill: none;
      stroke: #000;
      shape-rendering: crispEdges;
    }
    .x.axis path{
      display: none;
    }
    body{
    font: 10px arial;
     text-align: center;
    }
  </style>
  <link rel="stylesheet" type="text/css" href="main.css">
</head>
<body>
  <h1>Animated Barchart</h1>
  <script type="text/javascript">
```

範例研究

Case Study

step2：設定舞台

```
//Set up margin and percentage formatter
    var margin = {top:20, right: 30, bottom: 30, left:40};
    var width = 800-margin.left-margin.right;
    var height = 400-margin.top-margin.bottom;

    //Creating a percentage formatter
    var formatPercent = d3.format("%.0");

     //Create x and y scale
    var yScale = d3.scale.linear().range([height,0]);
    var xScale = d3.scale.ordinal().rangeRoundBands([0,widt
      ],0.1,0.2);

    //Create category 10 scale for color
    var c10 = d3.scale.category10();
```

#1f77b4
#ff7f0e
#2ca02c
#d62728
#9467bd
#8c564b
#e377c2
#7f7f7f
#bcbd22
#17becf

```
 //Create SVG Element
var svg = d3.select("body")
        .append("svg")  .attr({"width":width+margin.left+
          margin.right,"height":height+margin.top+margin.
          bottom})
        .append("g")
        .attr("transform","translate("+margin.left+","+
          margin.right+")");
```

step4：創建SVG

範例研究

Case Study

step5：載入外部資料

```
//Load data from external tsv file
d3.tsv("http://simplysanad.com/d3js/words.tsv", function(data){
//Set domain of x and y scales based on loaded data
yScale.domain([0,d3.max(data, function(d){ return d.frequency; })]);
xScale.domain(data.map(function(d){ return d.letter; }));
//Create X and Y Axis based on scales and data
var xAxis = d3.svg.axis()
          .scale(xScale)
          .orient("bottom");
var yAxis = d3.svg.axis()
          .scale(yScale)
          .orient("left")
          .tickFormat(formatPercent);
```

範例研究

Case Study

step6：Mix!

```
//Create X and Y Axis based on scales and data
var xAxis = d3.svg.axis()
              .scale(xScale)
              .orient("bottom");
var yAxis = d3.svg.axis()
              .scale(yScale)
              .orient("left")
              .tickFormat(formatPercent);
//Add bars
var bars = svg.selectAll("rect")
                .data(data)
                .enter()
                .append("rect")
                .attr("class","bar")
                .attr("width", xScale.rangeBand())
                .attr("fill","steelblue")
                .attr("fill", function(d,i){
                  return c10(Math.random()*10*i);
                })
                .attr("y", function(d){
                  return yScale(d.frequency);
                })
                .attr("x", function(d){
                  return xScale(d.letter);
                })
               .attr("height", function(d){
                  return height-yScale(d.frequency);
               });

    //Add X Axis
    svg.append("g")
    .attr("transform","translate(0,"+height+")")
    .call(xAxis)
    .attr("class","x axis");
     //Add Y Axis
     svg.append("g")
       .attr("class", "y axis")
       .call(yAxis);
  });
</script>
</body>
</html>
```

範例研究

Case Study

跑出來的結果