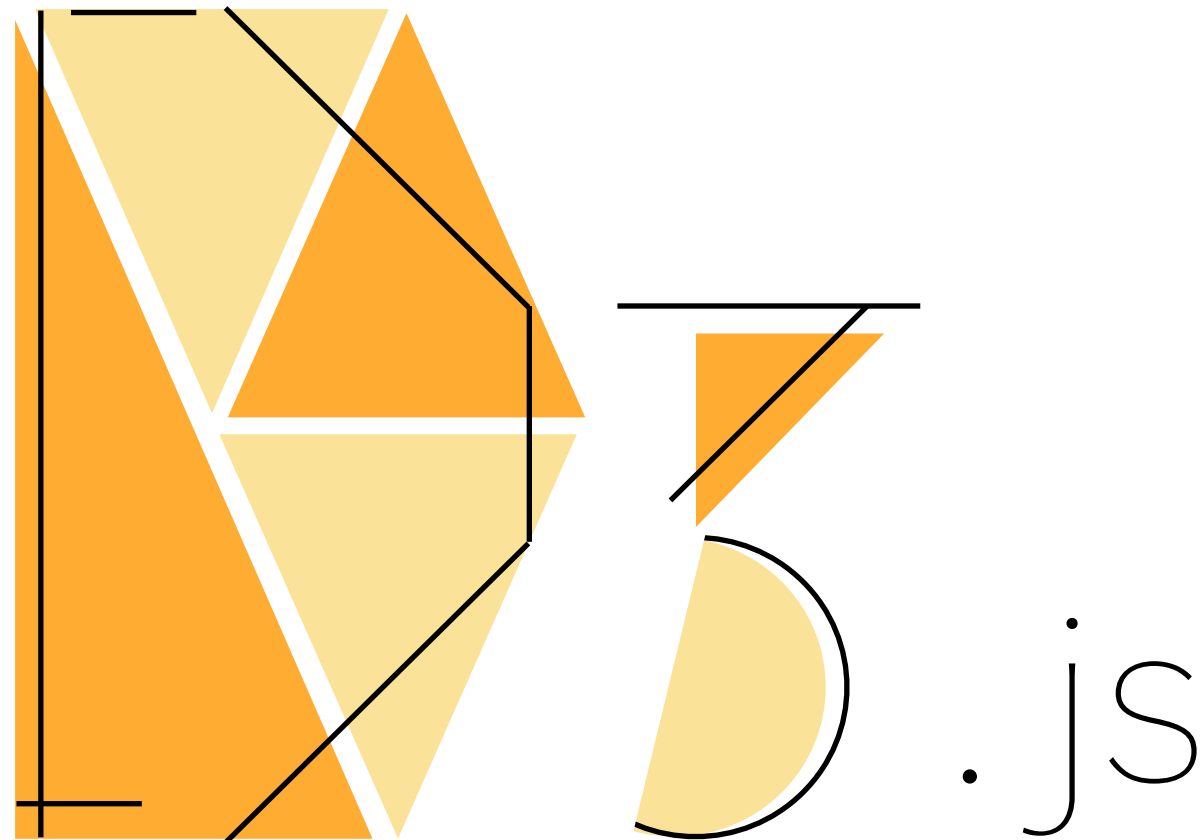


Date:12/18

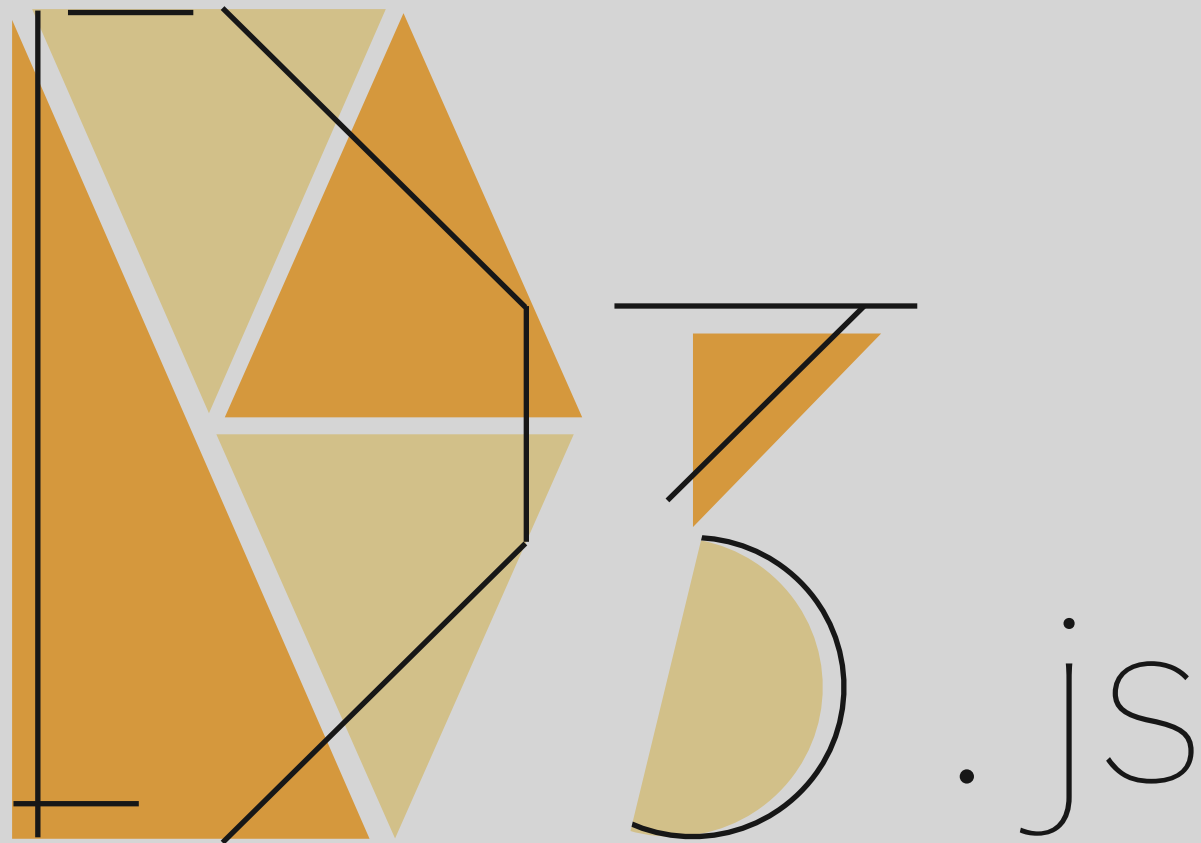
Presented by Willy/Shou



Outline

Before we start...

Basic INFO.



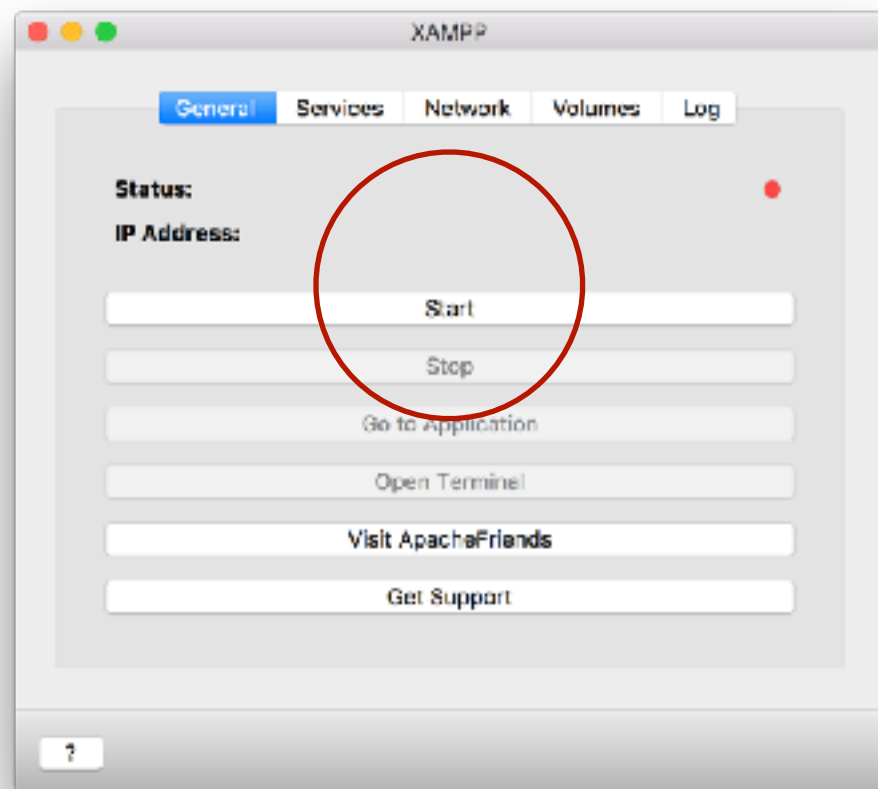
D3.js入門

實例解析

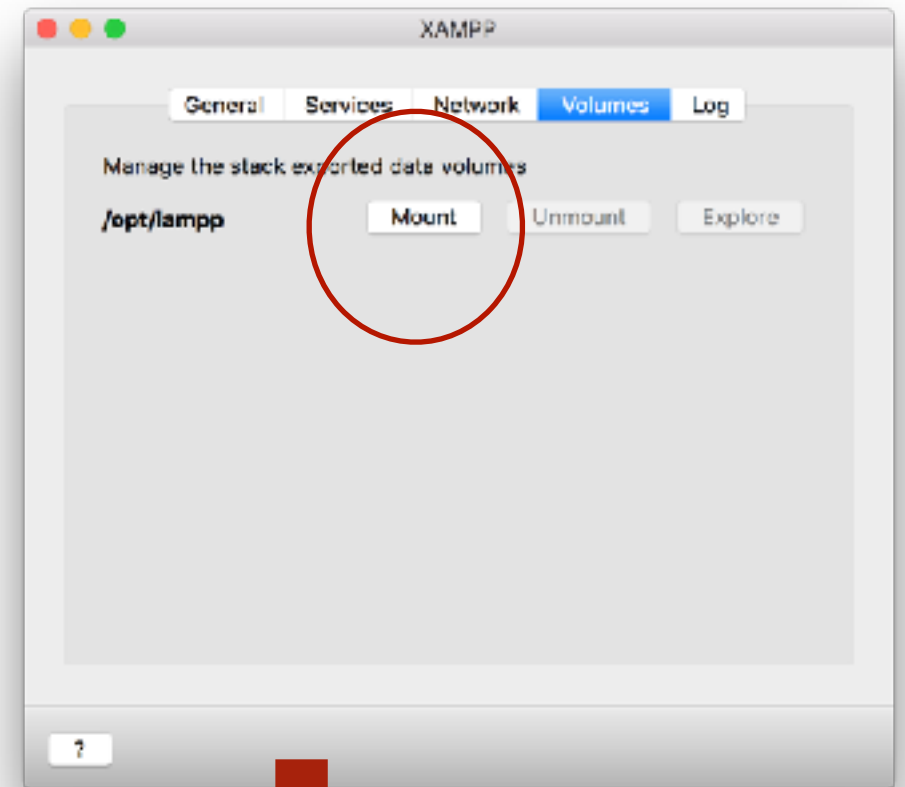
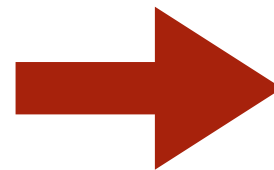
Before we start...

1. 安裝 Sublime
2. 安裝 XAMPP
3. 下載 D3.js 函式庫

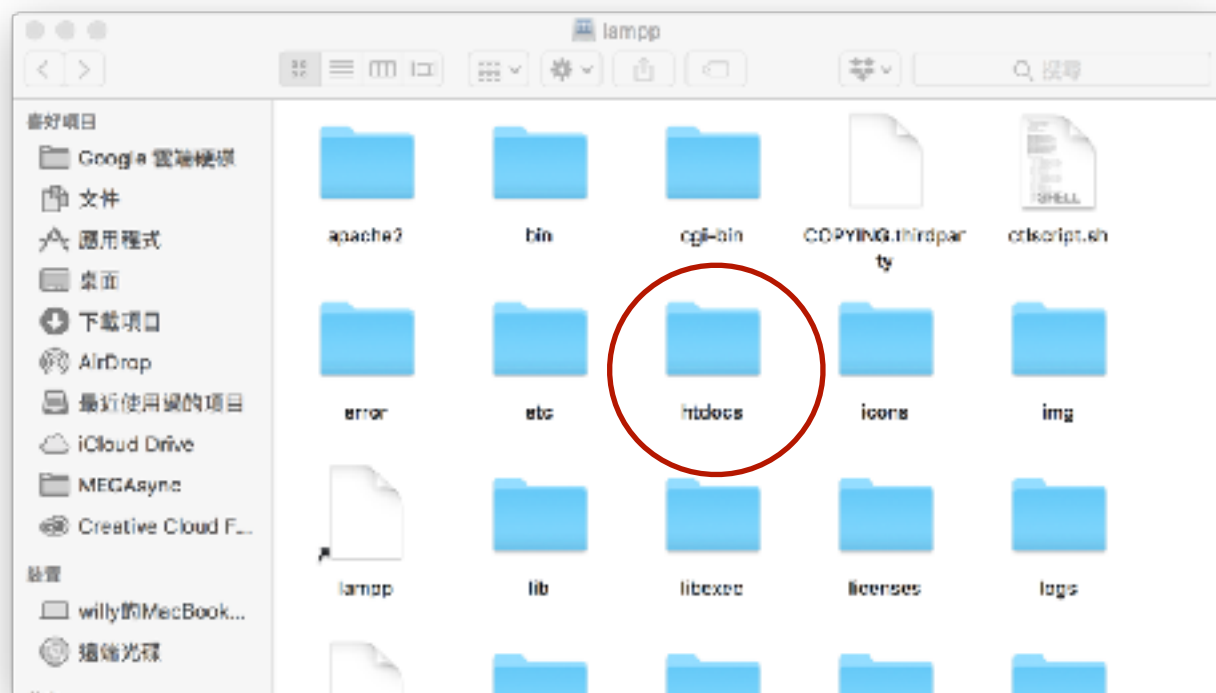




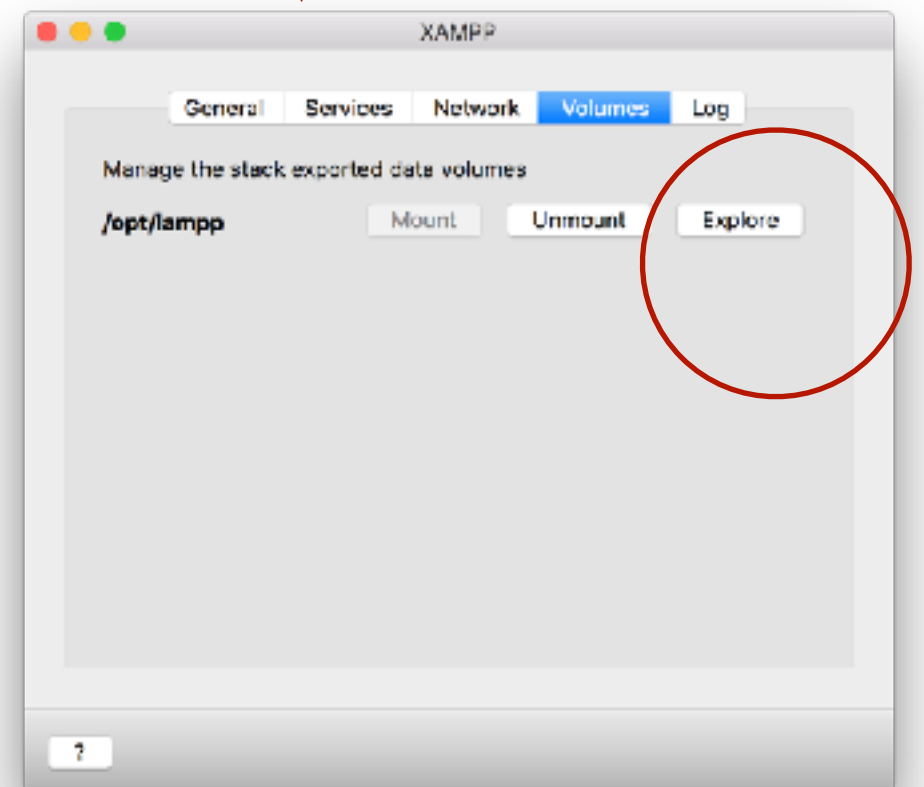
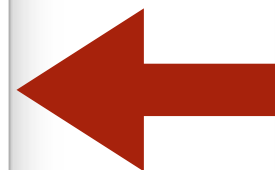
Click start



Click mount



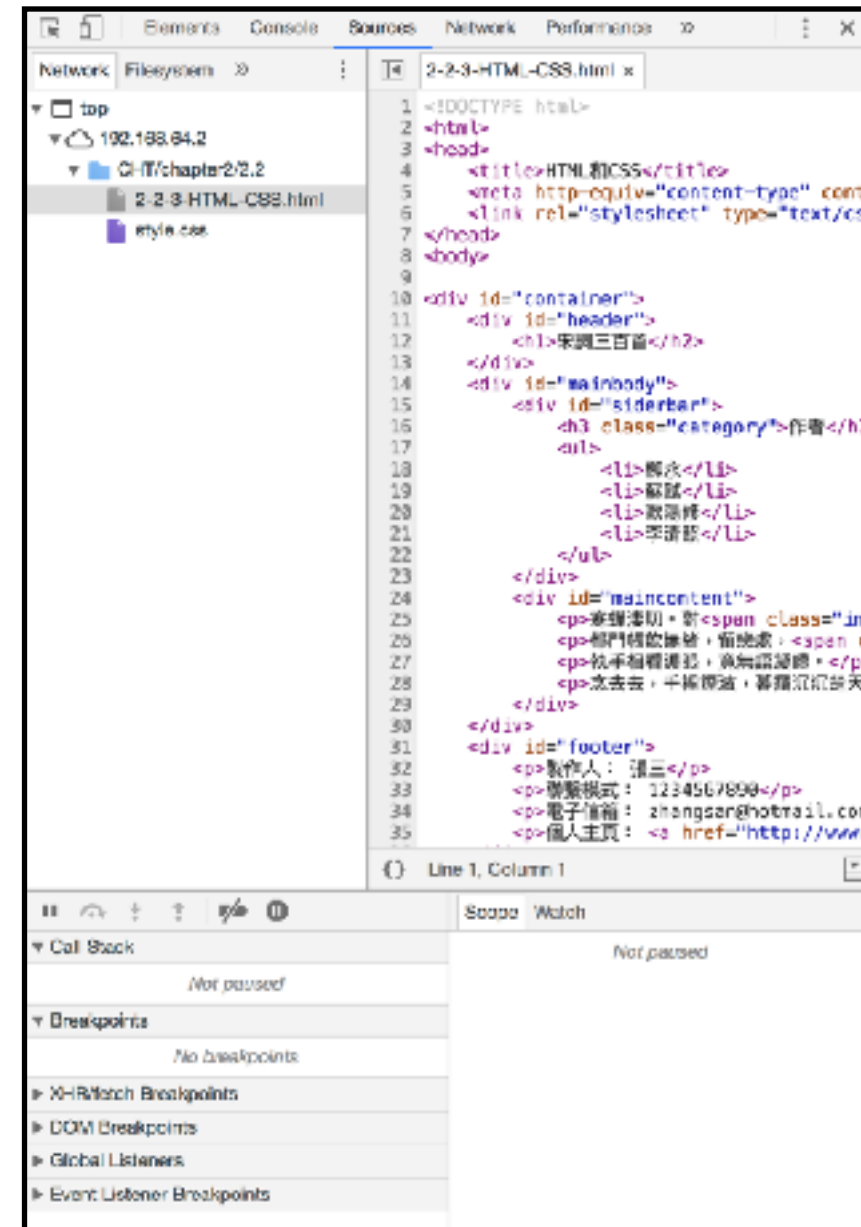
Find 'htdocs'



Click explore

Before we start...

4. 準備好你的Chrome
5. 記住192.168.64.2
6. 善用"開發人員工具"



Basic INFO.

HTML&CSS

- HTML(Hyper Text Markup Language/超文字標記語言)

一種標記語言，每個元素都有一個標籤。每組標籤代表一個內容：

標題：h1、h2...h6 段落：p 定義文件的分區：div

超連結：a 圖片：img 註解：<!-- --> or //

- CSS選擇器(Cascading Style Sheet/層疊樣式表)

為了給HTML定義樣式

.類別選擇器

對應html中的class屬性

#ID選擇器

對應html中的id屬性

程式碼範例

HTML

```
< > 2-2-3-HTML-CSS.html x style.css x
1 #container{
2   margin: 10px auto;
3   width: 700px;
4   font-family: Kaiti;
5 }
6 #header{
7 }
8
9 #mainbody{
10  margin: 0px 0px 0px 0px;
11  height: 200px;
12 }
13
14 #sidebar{
15  width: 25%;
16  height: 100%;
17  float: left;
18  clear: left;
19  border: 1px solid gray;
20  border-radius: 5px;
21 }
22
23 #maincontent{
24  width: 73%;
25  height: 100%;
26  float: right;
27  clear: right;
28  text-align: center;
29  border: 1px solid gray;
30  border-radius: 5px;
31  font-size: 20px;
32 }
33
34 #footer{
35  border-top: thin solid gray;
36 }
37
```

CSS

```
< > 2-1-3-HTML-CSS.html x style.css x
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>HTML和CSS</title>
5   <meta http-equiv="content-type" content="text/html; charset=big5"/>
6   <link rel="stylesheet" type="text/css" href="style.css" />
7 </head>
8 <body>
9
10  <div id="container">
11    <div id="header">
12      <h1>宋詞三百首</h1>
13    </div>
14    <div id="mainbody">
15      <div id="sidebar">
16        <h3 class="category">作者</h3>
17        <ul>
18          <li>楊永</li>
19          <li>蘇軾</li>
20          <li>歐陽修</li>
21          <li>李清照</li>
22        </ul>
23      </div>
24      <div id="maincontent">
25        <p>寒蟬凄切，對<span class="important">長亭</span>晚，腸斷初歌。</p>
26        <p>都門轉欲無，離愁處，<span class="important">蘭舟</span>催發。</p>
27        <p>執手相看淚眼，竟無語凝噎。</p>
28        <p>念去去，千里煙波，暮雲初盡楚天闊。</p>
29      </div>
30    </div>
31    <div id="footer">
32      <p>製作人：張三</p>
33      <p>聯繫模式：1234567890</p>
34      <p>電子信箱：zhangsan@hotmail.com </p>
35      <p>個人主頁：<a href="http://www.zhangsan.com/">www.zhangsan.com</a></p>
36    </div>
37
```


JS(JavaScript)

- 在html檔案中透過<script></script>標籤來使用JS
- 處理網頁上的所有動作行為，並能同時支援不同瀏覽器

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="utf-8">
5   <script src="http://d3js.org/d3.v4.min.js"></script>
6   <title></title>
7 </head>
8 <body>
9   <!-- Hello World -->
10  <!-- <p>Cat</p>
11  <p>Dog</p>
12  <p>Dolphin</p> -->
13  <!-- <p></p> -->
14  <p></p>
15  <p></p>
16  <script>
17    透過原生的JS將p的內容更改為hello world
18    var paragraphs = document.getElementsByTagName('p');
19    for (var i = 0; i < paragraphs.length; i++) {
20      var paragraph = paragraphs.item(i);
21      paragraph.innerHTML = "Hello World";
22    }
23
24    透過d3.js將兩段落更改為hello world
25    d3.select("body") //選擇body標籤
26    .selectAll("p") //選擇p標籤
27    .text("Hello world"); //將文字更改為hello world
28  </script>
29 </body>
30 </html>
```



D3.js(Data Driven Document)

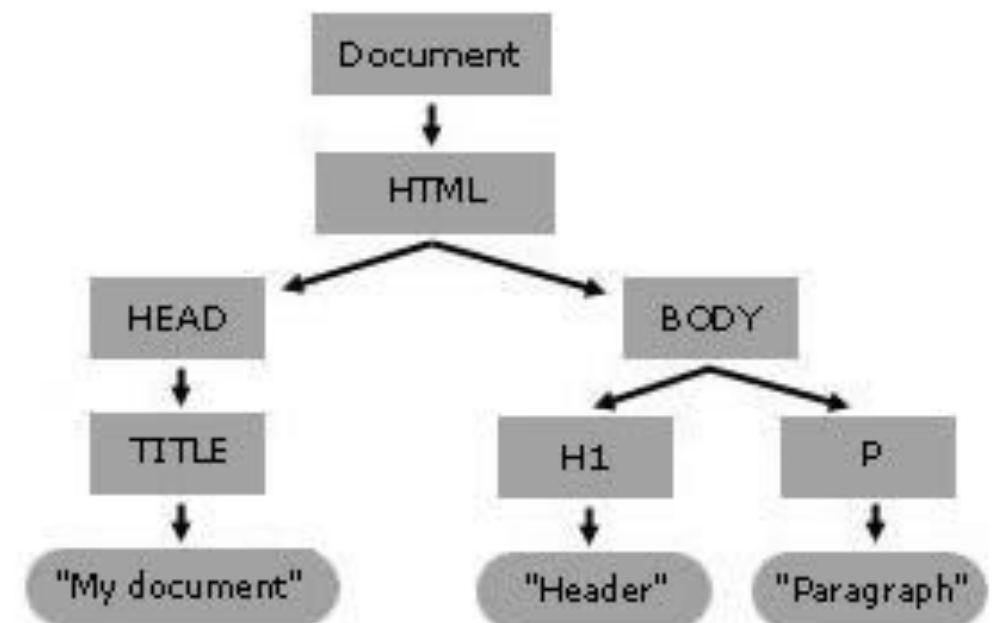
- 是一個前端的JavaScript函數庫
 - 包含一整組操縱畫圖的輔助工具，還有資料操作模型
 - 的基礎是在資料與物件的結合
- 優勢：
 1. 計算跟繪圖相互獨立
 2. 鏈式語法
 3. 強大圖形運算能力
 4. 同時支援SVG跟Canvas
 5. 資料與元素綁定的特性



DOM(Document Object Model文件物件模型)

- 當開發者需要動態修改網頁元素，需要用到！
- 提供了JS對瀏覽器存取的API介面
- 核心物件：document
- DOM將整個HTML視為許多的節點
- 提供許多API方法存取、新增、刪除、修改節點

```
//傳回id為myid的元素  
document.getElementById("myid");  
//傳回所有標籤為p的元素  
document.getElementsByTagName("p");  
//傳回類別為myclass的元素  
document.getElementsByClassName("myclass");
```



SVG(Scalable Vector Graphics/可縮放向量圖形)

點陣圖 & 向量圖

點陣圖

- 由一系列像素組成且像素具有自己的顏色(RGB值)，以一定的排列組成影像
- 放大後產生馬賽克
- 色彩表現能力較豐富
- 佔用空間較大

向量圖

- 由一系列數學元素定義而成，EX：點、線段、圓、曲線
- 與解析度無關，縮放後不會失真
- 基於數學圖形，圖形和色彩較簡單
- 佔用空間較小

Canvas(純量圖、點陣圖)

- 是一個圖形的容器，必須使用指令稿（JS）來繪圖
- 結果圖可以直接存為png或jpg，但他放大的話圖會失真
- 在html檔案裡是一個標籤

```
<canvas id ="mycanvas" width="960" height="600"></canvas>
```

————→ 尺寸由<canvas>的width和height兩個屬性決定

D3.js入門

選擇元素

Select/SelectAll

- Select:傳回比對選擇器的第一個元素
- SelectAll:選擇符合選擇器的所有元素

```
//select傳回比對選擇器的第一個元素
d3.select("body");//選擇body元素
d3.select("#important");//選擇id為important的元素
d3.select(".content");//選擇類別為content的第一個元素

//selectAll傳回比對選擇器的所有元素
d3.selectAll("p");//選擇所有的p元素
d3.selectAll(",content");//選擇類別為content的所有元素
d3.selectAll("ul li");//選擇 ul 中所有的li元素

//select跟selectAll的參數 除了CSS選擇器也可以是已經被dom
api選擇的元素
var important = document.getElementById("important");
d3.select(important);

//選擇body中所有的p元素
d3.select("body").selectAll("p");
```

檢視狀態

Empty/Node/Size

```
<p>paragraph 1</p>  
<p>paragraph 2</p>  
<p>paragraph 3</p>
```

- `Selection.empty()`:選擇集為空，傳回`true`;反之傳回`false`
- `Selection.node()`:傳回第一個不可是空的元素，若選擇集是空則傳回`null`
- `Selection.size()`:傳回選擇集中的元素個數

```
var paragraphs = d3.selectAll("p");  
//empty:選擇集為空，傳回true;反之false  
console.log(paragraphs.empty()); //false  
//node:傳回第一個不可為空元素，若選擇集為空，傳回null  
console.log(paragraphs.node()); //<p> paragraph 1</P>  
//size傳回選擇集的元素個數  
console.log(paragraphs.size()); //3
```


設定 & 取得屬性

Attr/Classed/Style/Property/Text/Html

- Attr:設定或取得選擇集的屬性
- Classed:設定或取得選擇的css類別
- Style:設定或取得選擇集的樣式
- Property:有部分屬性不能用attr設定或取得，就用它
- Text:設定或取得選擇集文字內容，
文字內容相當於DOM的innerText，不包含元素內部標籤
- Html:設定或取得選擇集文字內容，
文字內容相當於DOM的innerHTML，包含元素內部標籤

增加、插入 & 刪除

Append/Insert/Remove

- `selection.append(name)`: 在選擇集的尾端增加一個元素，`name`為元素名稱
- `selection.insert(name[,before])`: 在選擇集中的指定元素之前插入一個元素，`name`是被插入的元素名稱，`before`是css選擇器名稱
- `selection.remove()`: 刪除選擇集中的元素

```
//選擇body元素
var body = d3.select("body");

//在body中所有元素的尾端處增加一個p元素 內容為train
body.append("p").text("Train");//car plane ship train

//在body中id為plane的元素前增加一個p元素 內容為Bike
body.insert("p","#plane").text("Bike");//car bike plane
ship train

//選擇id為plane的元素
var plane = d3.select("#plane");
//刪除id為plane的元素
plane.remove();//car bike ship train
```

資料綁定

Data/Datum

- Datum:一個資料綁定到選擇集的所有元素上，每個元素擁有相同的資料

```
var p = d3.select("body").selectAll("p");  
//綁定數值7到選擇集上  
p.datum(7);  
//在主舞台輸出選擇集  
console.log(p);  
  
//沒有參數的情形  
var p = d3.select("body").selectAll("p");  
p.datum(7);  
console.log(p.datum()); //在主舞台輸出被綁定的資料  
  
//用被綁定的字串，取代段落原來的字串  
var p = d3.select("body").selectAll("p");  
p.datum("Thunder")  
  .text(function(d,i){ //綁定字串thunder到選擇集上  
    return d + " " + i; //取代內容  
  });  
  
//能使被綁定的資料傳遞給子元素  
p.datum("Thunder")  
  .append("span") //在每一個被選擇元素後增加元素span  
  .text(function(d,i){ //綁定字串thunder到選擇集上  
    return d + " " + i; //取代內容  
  });
```

```
  __data__: 7  
  ► __proto__: HTMLParagraphElement  
  ► 1: p  
  ► 2: p  
  ► parentNode: body  
    length: 3  
  ► __proto__: Array(0)  
    length: 1  
  ► __proto__: Array(0)
```

- Data:能將陣列各項分別綁定到選擇集的各元素上，並且能指定綁定的規則

```
var dataset =[3,6,9];  
  
//選擇body中的p元素  
var p = d3.select("body").selectAll("p");  
  
//綁定資料到選擇集  
var update = p.data(dataset);  
  
//輸出綁定的結果  
console.log(update);
```

資料綁定

Enter/update/exit

- Enter:代表沒有足夠的元素，因此處理方法是增加元素
- Update:修改attr屬性
- Exit:刪除多餘的元素

```
//以下是若在不知道新資料長度情況下更新圖形
var dataset = [10,20,30];
var p = d3.select("body").selectAll("p");
//綁定資料後分別回傳update enter exit部分
var update = p.data(dataset);

var enter = update.enter();

var exit = update.exit();

//1.update部分處理方法
update.text(function(d){return d;});

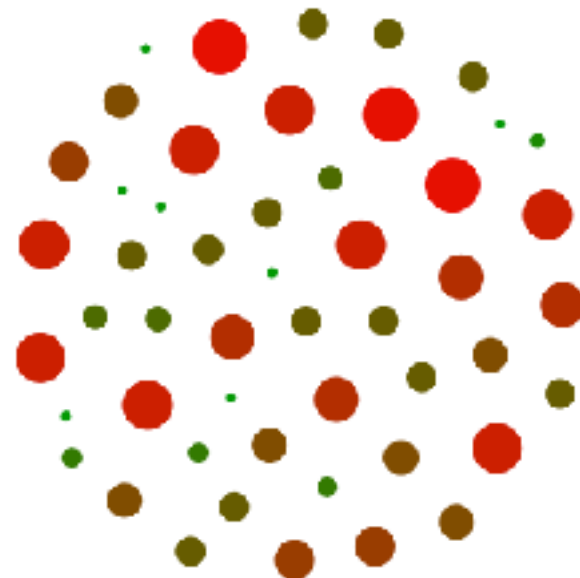
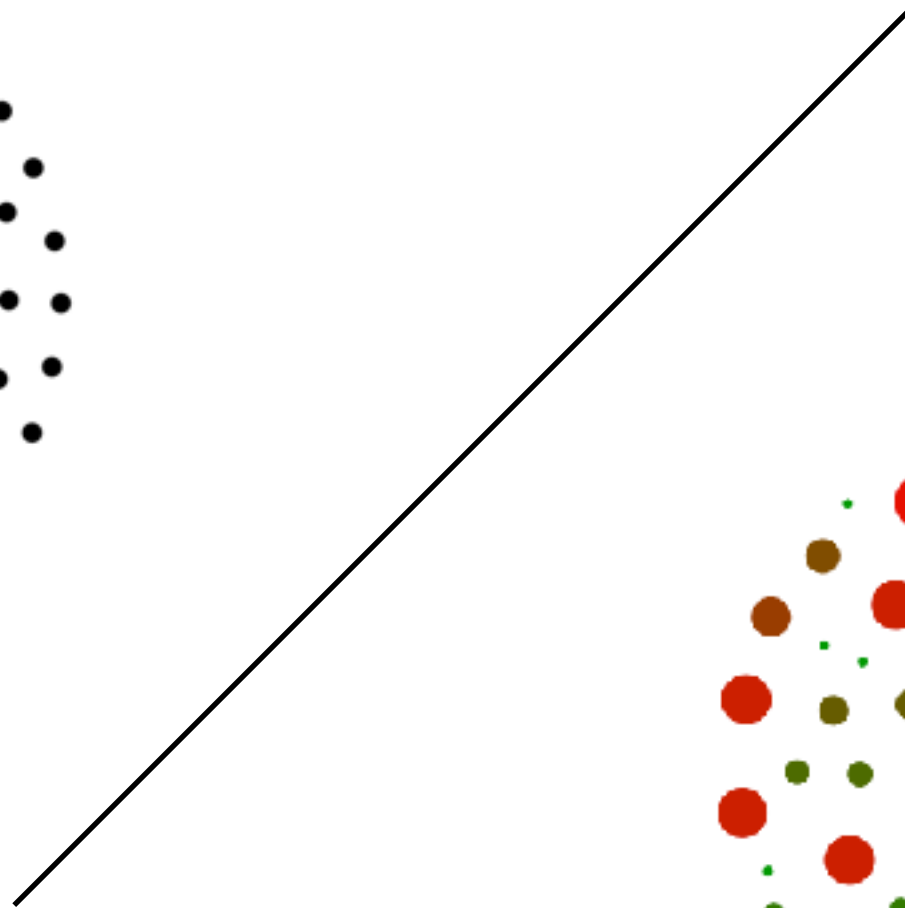
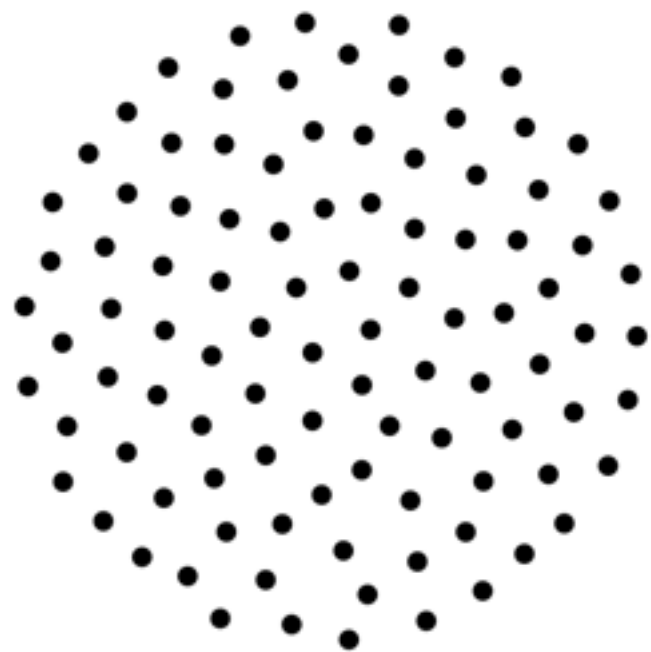
//2.enter部分處理方法
enter.append("p")
    .text(function(d) {return d;});

//3.exit部分的處理方法
exit.remove();
```

D3.js實作

Force Layout

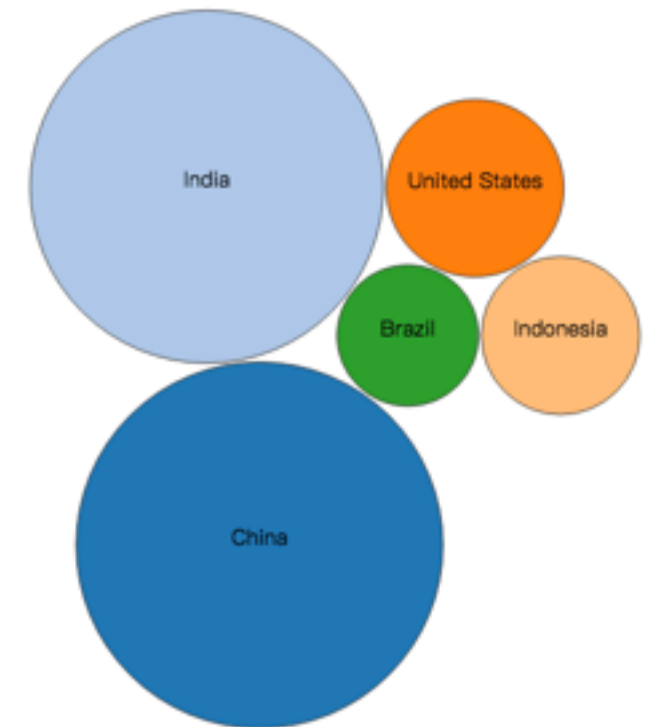
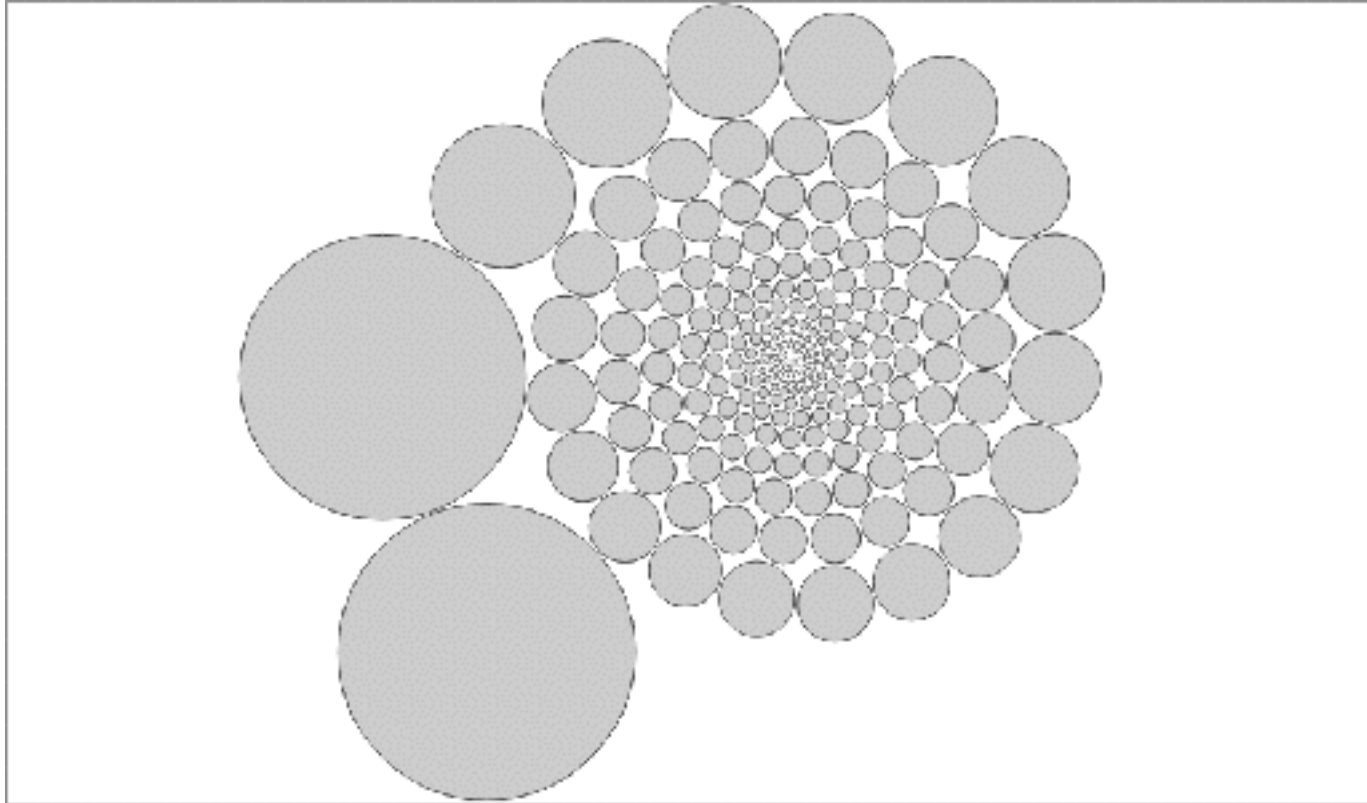
- d3.layout排版函式
- 利用引力與斥力計算物體位置的版面配置
- 適合利用來呈現網路圖等需要動態配置顯示位置的圖表類型



→more details

Pack Layout(泡泡圖)

- d3.layout排版函式
- 利用階層性的結構把資料分類，並在排版上使用資料數據中「大小」來代表各個元素的圓圈大小



→more details

Something FUN

中央政府總預算視覺化

公眾人物關係視覺化

Date:12/18

Presented by Willy/Shou

D3.js