

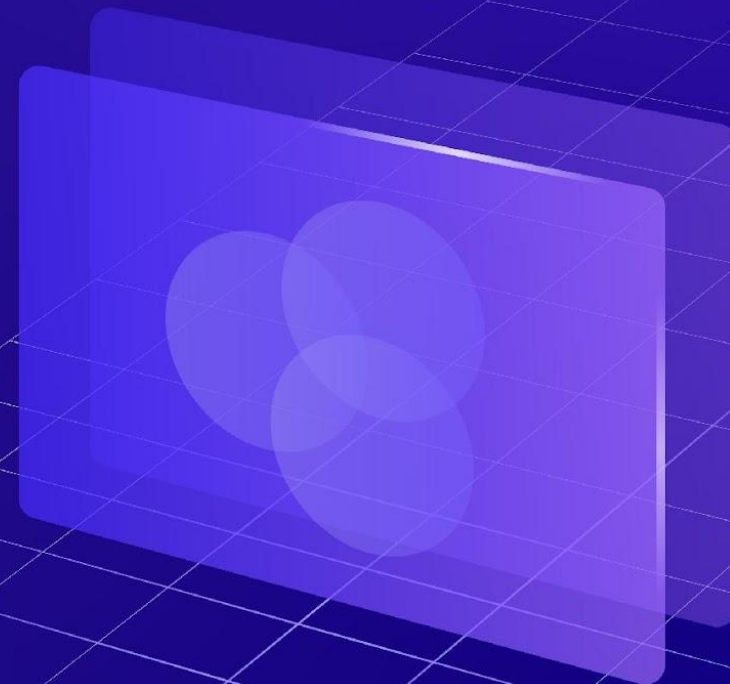


# 資料可視化

## HW02 - 直方圖及長條圖

許加宜

國立臺北科技大學資訊工程系





# 01

## 目錄



# 目錄

- Histogram vs Bar chart
- 繪製 Histogram 和 Bar chart
- 評分標準
- 作業說明
- 繳交資訊



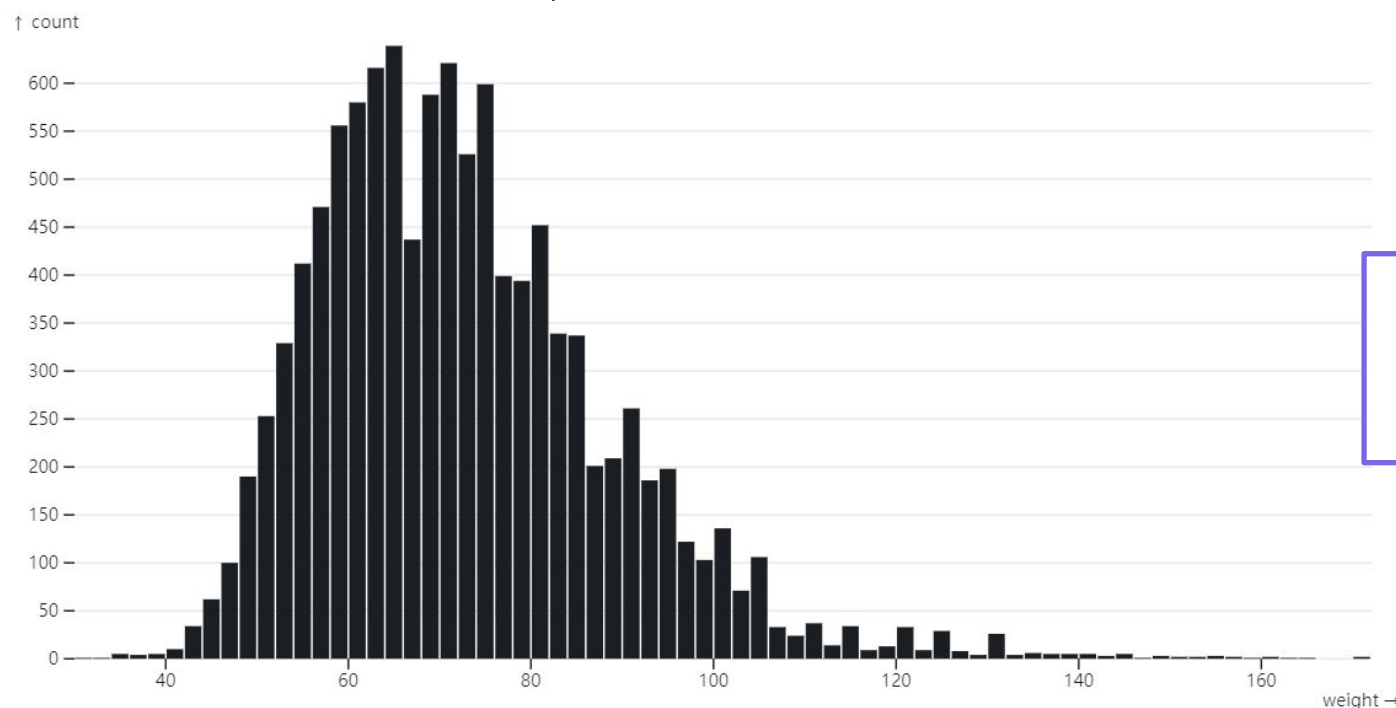
# 02

## Histogram vs Bar chart



# Observable Histogram 介紹

- 橫軸將連續的資料(例如體重、時間), 分成離散的區間, 縱軸為各組的統計
- 直方圖的組距之間, 有一定的排列順序

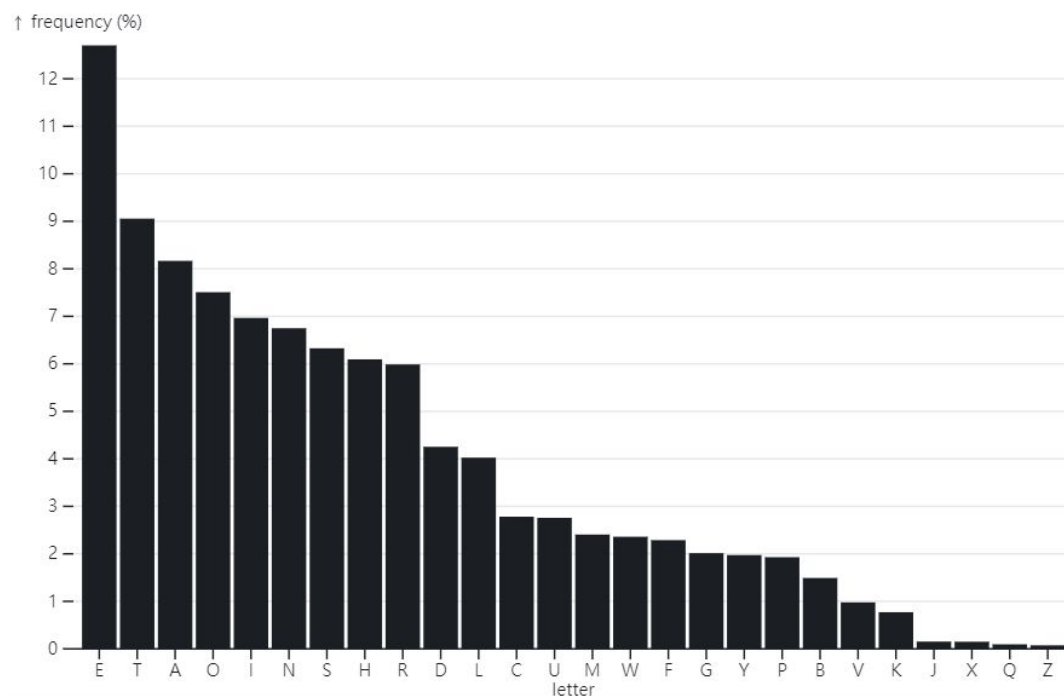


olympians dataset 中各個體重的人數  
(x軸:體重(連續數字), Y軸:人數)



# Observable Bar chart 介紹

- 橫軸為類別，縱軸為各組的次數
- 長條圖的組距之間，並未有一定的排列順序



alphabet dataset 中各個字母的百分比  
(x軸:不同字母(無順序), Y軸:百分比)



03

# 繪製 Histogram 和 Bar chart





# Observable教學

🌟 New feature alert: Accelerate your data visualization workflow with AI Assist [Learn more →](#)

🔍 Search [Sign in](#) [Sign up](#)

## Push the limits of data visualization

With 200,000+ community examples, Observable is the fastest way to build custom data visualizations, apps, and dashboards to uncover deeper insights.

[Try it for free](#)

1. 登入 Observable





# Observable教學

Product Pricing Solutions Explore Learn Community Search

New



Free

資料可視化   
@vis2023f-joyce

Follow

Home

Profile

Notebooks

Collections

Suggestions

Trash

Settings

## 2. 進入Notebooks頁面

## Home

0 Notebooks 0 Likes 0 Forks

### Quick guides

These 5-minute explorations show you how to build useful notebooks using real data.



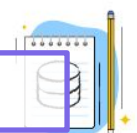
Learn how to write and run JavaScript on Observable



Write a tip calculator to dive deeper into JavaScript



Explore a dataset using functional programming



Follow clues in a database to solve a murder mystery



Create interactive charts from simple tabular data



Visualize time series data with smoothed line charts

### Recent notebooks



# Observable教學



Free

資料可視化  
@vis2023f-joyce

Follow

Home

Profile

Notebooks

Collections

Suggestions

Trash

Settings

## Notebooks

Access All

0 notebooks

No notebooks found.

New

3. 創建新的Notebook



# Observable教學

New notebook

Workspace

許加宜's Worksp...

資料可視化

Visibility

4. 選擇Public

Public

Team Pro

Only you Pro

Edit in private with Pro

Add to collection

Add a collection

5. 選擇Blank

Basic Templates

Blank

Start from scratch with an empty notebook.

Input + Chart

Connect a slider to a chart.

Search + Table

Connect a freeform search input to a table.

SQL + Chart

Connect a SQLite database to a chart.

Boilerplates

6. Create

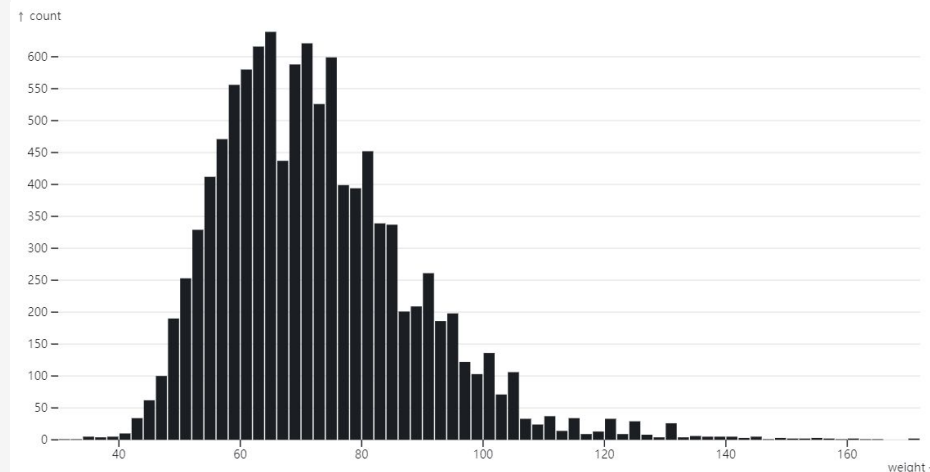
Create notebook



# 繪製Histogram

範例 1: 將olympians 資料集繪製成直方圖 ( X軸: 體重, Y軸: 人數)

```
Plot.plot({  
  width: 800, // 設定整個圖表的寬度=800  
  y: {grid: true, label: "count"}, // 呈現Y軸數值的網格，並將Y軸的標籤指定為"count"  
  
  marks: [ // marks為用來定義視覺元素  
    // 使用 Plot.rectY 函數來繪製柱狀圖  
    // 並使用 Plot.binX 函數將數據按照 "weight" 欄位進行分組，並計算每個組別中的數量  
    Plot.rectY(olympians, Plot.binX({y: "count"}, {x: "weight"})),  
    // 創建了Y軸的網格線  
    // interval 設定網格線之間的間隔  
    // stroke 設定網格線的顏色  
    // strokeOpacity 設定網格線的透明度 (0: 完全透明, 1: 完全不透明)  
    Plot.gridY({ interval: 1, stroke: "white", strokeOpacity: 0 })),  
  ]  
})
```



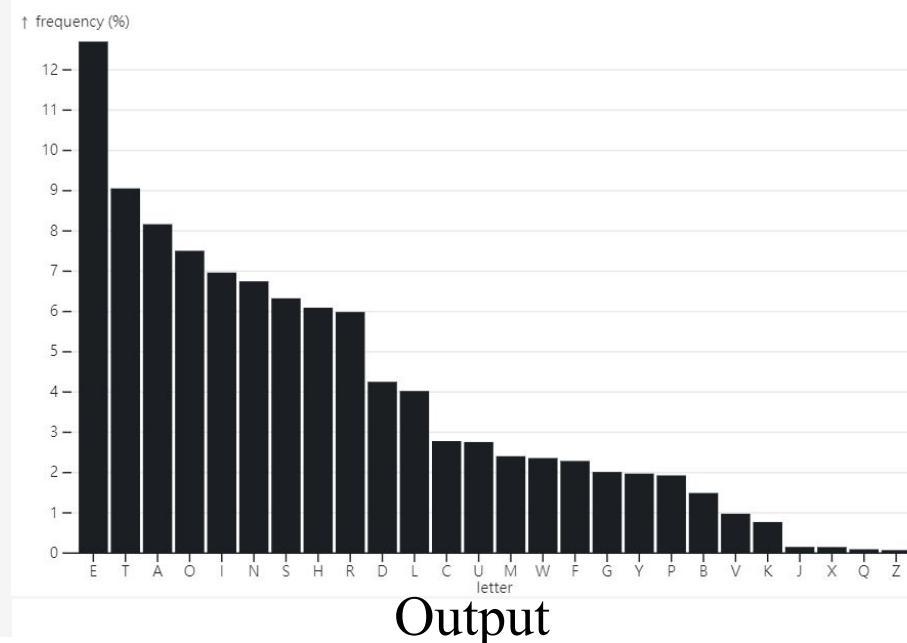
Output



# 繪製Bar chart

範例 2: 將 alphabet 資料集繪製成長條圖 (X軸: 字母, Y軸: 百分比)

```
Plot.plot({  
  y: {  
    grid: true, //呈現Y軸數值的網格  
    percent: true //將Y軸的值設為百分比  
  },  
  marks: [  
    Plot.ruleY([0]), //在Y為0的地方(X軸)畫一條線  
    //X軸按照 "letter" 欄位進行分組, Y軸標示各個字母的"frequency"值  
    Plot.barY(alphabet,  
      {x: "letter", y: "frequency",  
       sort: {x: "y", reverse: true}} //依照百分比進行由多至少的排序  
    )  
  ]  
})
```





# 04

## 評分標準





# 評分標準

- Simple baseline (4pt)
  - 以 histogram 呈現每個年份出生的人數 (2pt)
  - 可調整margin, fill color, tip (2pt)
- Medium baseline (4pt)
  - 以 bar chart 呈現每個年份出生的人數 (2pt)
  - 可調整margin, fill color, tip (2pt)
- Strong baseline (2pt)
  - 以 bar chart 呈現每個星座的人數 (1pt)
  - 以 histogram 呈現每個星座的人數 (1pt)



# 05

## 作業說明





The screenshot shows the DataCamp interface. At the top, there's a header with '資料可視化' and a 'Follow' button. Below that, a 'Public' badge and the user 'By 許加宜' are visible. A '+ Add tags' button is also present. The main content area is titled 'Data'. On the left, a dropdown menu is open, showing options like 'Unpin', 'Add comment', 'Copy import', 'Select', 'Duplicate', 'Copy link', 'Embed', 'Download CSV', 'Download JSON', and 'Delete'. The 'Download JSON' option is highlighted with a purple box. To the right of the dropdown, a code editor shows a snippet of JSON data. Below the code editor, there are buttons for 'Create table' and 'Create chart'. A purple box also highlights the text '2. Download JSON' in the bottom right corner of the image.



# 讀JSON檔

## 2. 將檔案加入自己的Notebook

The screenshot shows a Jupyter Notebook interface. The main area displays a notebook titled "HW2 Simple baseline (4pt)" with a "Public" status and a "Follow" button. The notebook is created by "許加宜" and was last edited on "Oct 14". The notebook content shows a code cell with a "+" button and a dropdown menu with options: "JavaScript", "Text", "SQL", "Data", and "Chart".

On the right side, the "Files" sidebar is visible. It contains a "File attachments" section with a "+" button. A blue box highlights the "+" button with the text "3.". Another blue box highlights the "File attachments" section with the text "4. File attachments, 將剛剛下載的 data.json 加入".



# 讀JSON檔

## 2. 將檔案加入自己的Notebook

資料可視化

Follow

Public

By 加宜 許加宜

Edited Oct 14

+ Add tags

HW2 Simple baseline (4pt)

+

JavaScript

Text

SQL

Data

Chart

Share...

Files

File attachments (13 KB)

Unused

data.json

Unused • 13 KB • seconds ago

Cloud file

Only available in private notebooks

5. Insert into Notebook+



# 讀JSON檔

3. 即可在此Notebook使用 data 這個 Object Array。

## HW2 Simple baseline (4pt)

```
data = ▼ Array(84) [  
  0: ▶ Object {LivingPlaceFirst: "臺北市", Constellation: 11, LivingPlaceLast: "中山區", Year: "2001", Gender: "男", Email:  
  1: ▶ Object {LivingPlaceFirst: "新北市", Constellation: 2, LivingPlaceLast: "中和區", Year: "1999", Gender: "男", Email:  
  2: ▶ Object {LivingPlaceFirst: "新北市", Constellation: 7, LivingPlaceLast: "汐止區", Year: "2002", Gender: "男", Email:  
  3: ▶ Object {LivingPlaceFirst: "新北市", Constellation: 11, LivingPlaceLast: "樹林區", Year: "1999", Gender: "女", Email:  
  4: ▶ Object {LivingPlaceFirst: "新北市", Constellation: 8, LivingPlaceLast: "中和區", Year: "1998", Gender: "男", Email:  
  5: ▶ Object {LivingPlaceFirst: "臺北市", Constellation: 11, LivingPlaceLast: "中正區", Year: "1999", Gender: "男", Email:  
  6: ▶ Object {LivingPlaceFirst: "高雄市", Constellation: 9, LivingPlaceLast: "鼓山區", Year: "1998", Gender: "女", Email:  
  7: ▶ Object {LivingPlaceFirst: "新北市", Constellation: 4, LivingPlaceLast: "土城區", Year: "2000", Gender: "男", Email:  
  8: ▶ Object {LivingPlaceFirst: "臺北市", Constellation: 9, LivingPlaceLast: "大安區", Year: "2000", Gender: "女", Email:  
  9: ▶ Object {LivingPlaceFirst: "新北市", Constellation: 9, LivingPlaceLast: "新莊區", Year: "2001", Gender: "男", Email:  
 10: ▶ Object {LivingPlaceFirst: "新北市", Constellation: 0, LivingPlaceLast: "中和區", Year: "1996", Gender: "男", Email:  
 11: ▶ Object {LivingPlaceFirst: "新竹縣", Constellation: 8, LivingPlaceLast: "新豐鄉", Year: "2000", Gender: "女", Email:  
 12: ▶ Object {LivingPlaceFirst: "臺中市", Constellation: 10, LivingPlaceLast: "北屯區", Year: "2001", Gender: "女", Email:  
 13: ▶ Object {LivingPlaceFirst: "桃園市", Constellation: 10, LivingPlaceLast: "桃園區", Year: "2001", Gender: "女", Email:  
 14: ▶ Object {LivingPlaceFirst: "桃園市", Constellation: 5, LivingPlaceLast: "八德區", Year: "2000", Gender: "男", Email:  
 15: ▶ Object {LivingPlaceFirst: "新北市", Constellation: 9, LivingPlaceLast: "永和區", Year: "2000", Gender: "男", Email:  
 16: ▶ Object {LivingPlaceFirst: "臺北市", Constellation: 1, LivingPlaceLast: "松山區", Year: "2000", Gender: "男", Email:  
 17: ▶ Object {LivingPlaceFirst: "新北市", Constellation: 4, LivingPlaceLast: "新莊區", Year: "2002", Gender: "女", Email:  
 18: ▶ Object {LivingPlaceFirst: "桃園市", Constellation: 5, LivingPlaceLast: "中壢區", Year: "2001", Gender: "男", Email:  
 19: ▶ Object {LivingPlaceFirst: "臺北市", Constellation: 6, LivingPlaceLast: "中正區", Year: "2001", Gender: "男", Email:
```



# 作業注意事項

此次作業的3個baseline請開三個不同Notebook, 並分別命名為:

- **HW2 Simple baseline (4pt)**
- **HW2 Medium baseline (4pt)**
- **HW2 Strong baseline (2pt)**

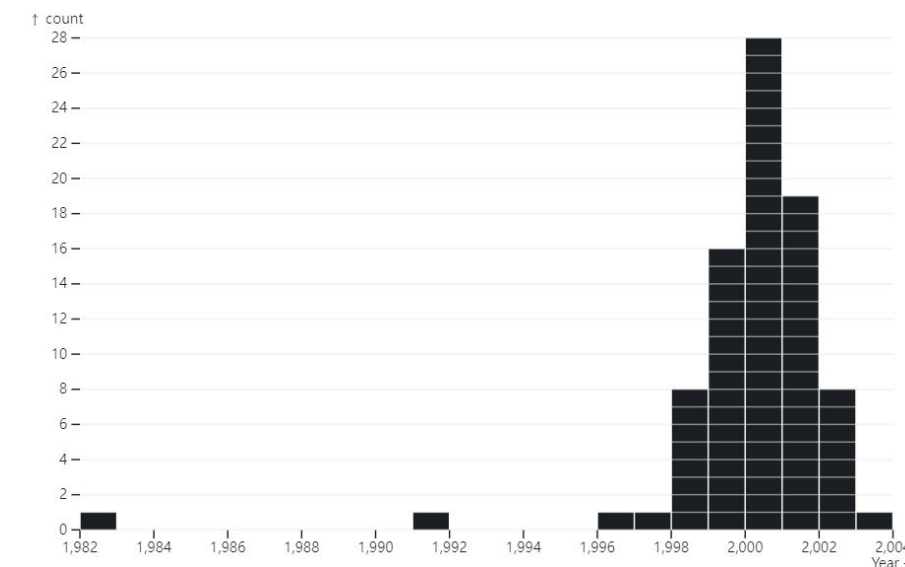
請注意每個Notebook都要分別加入 data.json



# Simple baseline (4pt)

- 以histogram呈現每個年份出生的人數 (2pt)

```
Plot.plot({  
  y: {grid: true, label: "count"},  
  marks: [  
    Plot.rectY(data, Plot.binX({y:"count"}, { x:"Year", interval: 1 })),  
    Plot.gridY({ interval: 1, stroke: "white", strokeOpacity: 0.5 })  
  ]  
})
```





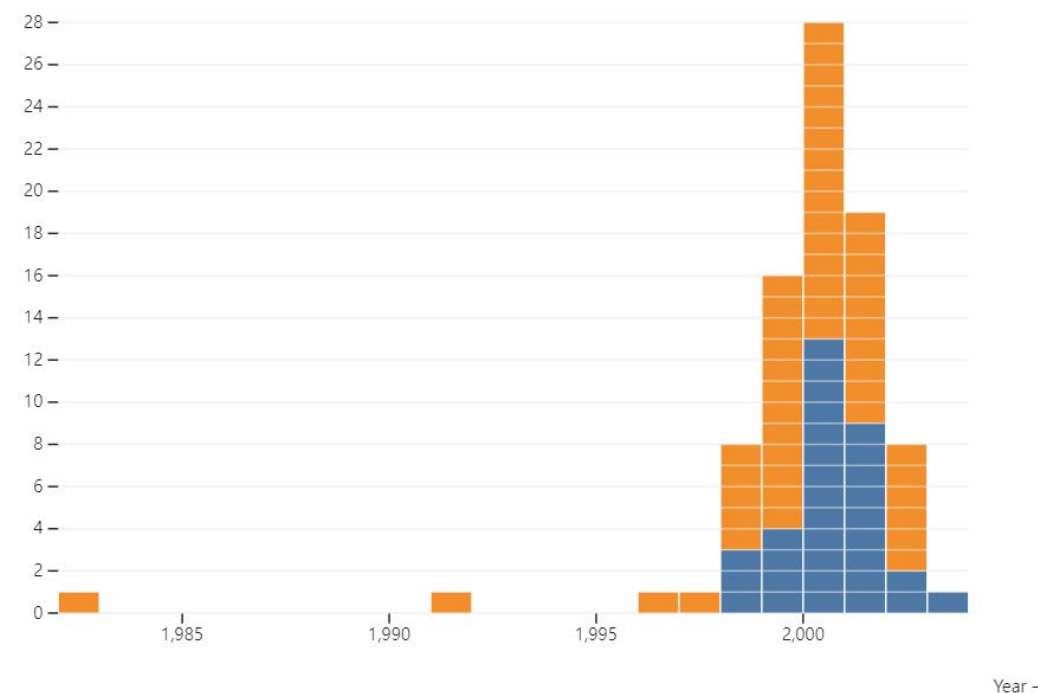
# Simple baseline (4pt)

- 可調整margin, fill color, tip (2pt)

- margin: 圖表區域周圍的空間
- fill color: 填充矩形的顏色
- tip: 矩形額外顯示的資訊

marginTop	<input type="text" value="50"/>	<input type="range"/>
marginRight	<input type="text" value="50"/>	<input type="range"/>
marginBottom	<input type="text" value="50"/>	<input type="range"/>
marginLeft	<input type="text" value="50"/>	<input type="range"/>

↑ count





# Simple baseline (4pt)

- 可調整margin, fill color, tip (2pt)

```
viewof plot1 = Inputs.form({  
  mt: Inputs.range([0, 100], {label: "marginTop", step: 1}),  
  mr: Inputs.range([0, 100], {label: "marginRight", step: 1}),  
  mb: Inputs.range([0, 100], {label: "marginBottom", step: 1}),  
  ml: Inputs.range([0, 100], {label: "marginLeft", step: 1}),  
})
```

marginTop	<input type="text" value="50"/>	<input type="range" value="50"/>
marginRight	<input type="text" value="50"/>	<input type="range" value="50"/>
marginBottom	<input type="text" value="50"/>	<input type="range" value="50"/>
marginLeft	<input type="text" value="50"/>	<input type="range" value="50"/>



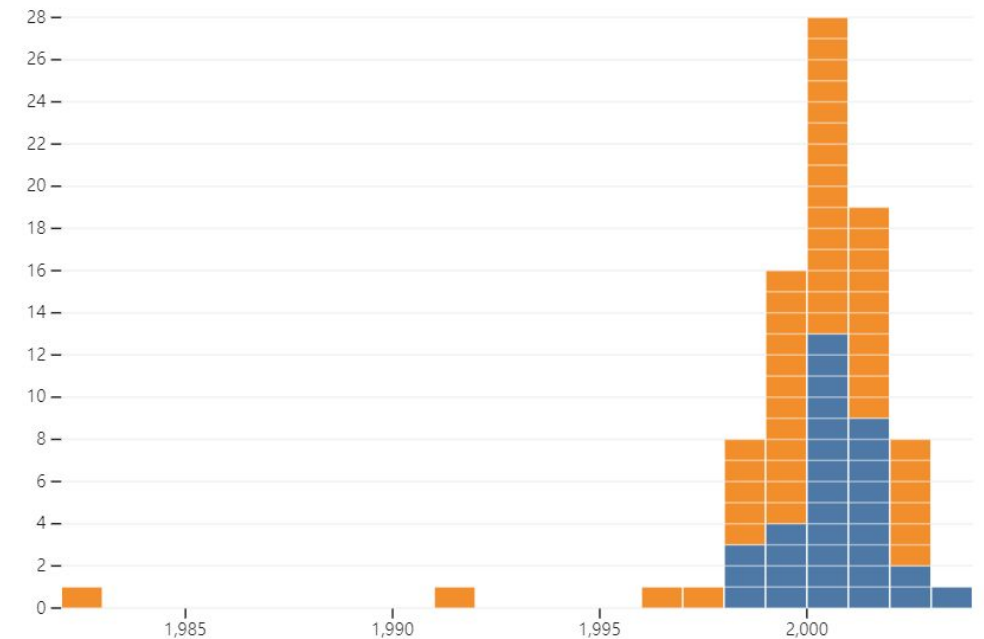
# Simple baseline (4pt)

- 可調整margin, fill color, tip (2pt)

```
Plot.plot({  
  marginTop: plot1.mt,  
  marginRight: plot1.mr,  
  marginBottom: plot1.mb,  
  marginLeft: plot1.ml,  
  y: {grid: true, label: "count"},  
  marks: [  
    Plot.rectY(data, Plot.binX({y:"count"}, { x:"Year", interval:1, fill:"Gender", tip: true })),  
    Plot.gridY({ interval: 1, stroke: "white", strokeOpacity: 0.5 })  
  ]  
})
```

marginTop 50  
marginRight 50  
marginBottom 50  
marginLeft 50

↑ count



Year →



# Medium baseline (4pt)

## • 資料處理

```
yCounts = [];  
//第一個cell, 建立空的Array, 存放要呈現在 bar chart的資料
```

```
years = data.map(item => item.Year);  
//第二個cell, 建立一個Map存放所有出生年份, 以便找到最早的  
出生年和最晚出生年
```

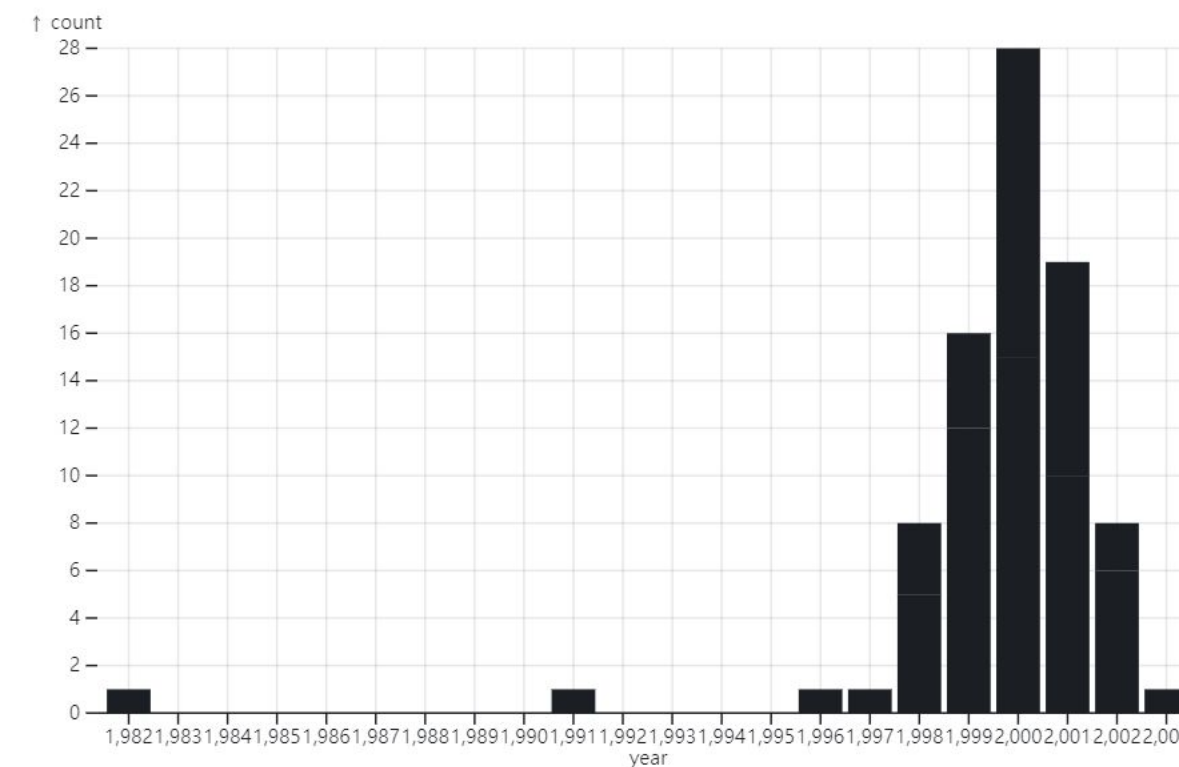
```
//第三個cell  
{  
  yCounts.length = 0; //將yCounts清空  
  var minYear = Math.min(...years); //最早出生年  
  var maxYear = Math.max(...years); //最晚出生年  
  for (var y=minYear; y<=maxYear; y++) {  
    //所有年份都建立兩個 Object, 一個存放男性資料, 一個存放女性資料  
    yCounts.push({year:y, gender:"male", count:0});  
    //Object包含: 1. 出生年, 2. 男性, 3. 人數(設為0)  
    yCounts.push({year:y, gender:"female", count:0});  
    //Object包含: 1. 出生年, 2. 女性, 3. 人數(設為0)  
  }  
  data.forEach (x=> {  
    var i = (x.Year-minYear)*2 + (x.Gender=="男" ? 0 : 1);  
    yCounts[i].count++;  
    //讀取data array, 加總每個年份出生的人  
  })  
  return yCounts  
}
```



# Medium baseline (4pt)

- 以 bar chart 呈現每個年份出生的人數 (2pt)

```
Plot.plot({  
  
  grid: true,  
  y: {label: "count"},  
  
  marks: [  
    Plot.ruleY([0]),  
    Plot.barY(yCounts, {x: "year", y: "count"}),  
  ]  
});
```





# Medium baseline (4pt)

- 可調整margin, fill color, tip (2pt)

```
viewof plot2 = Inputs.form({  
  mt: Inputs.range([0, 100], {label: "marginTop", step: 1}),  
  mr: Inputs.range([0, 100], {label: "marginRight", step: 1}),  
  mb: Inputs.range([0, 100], {label: "marginBottom", step: 1}),  
  ml: Inputs.range([0, 100], {label: "marginLeft", step: 1}),  
})
```

marginTop	<input type="text" value="50"/>	<input type="range" value="50"/>
marginRight	<input type="text" value="50"/>	<input type="range" value="50"/>
marginBottom	<input type="text" value="50"/>	<input type="range" value="50"/>
marginLeft	<input type="text" value="50"/>	<input type="range" value="50"/>



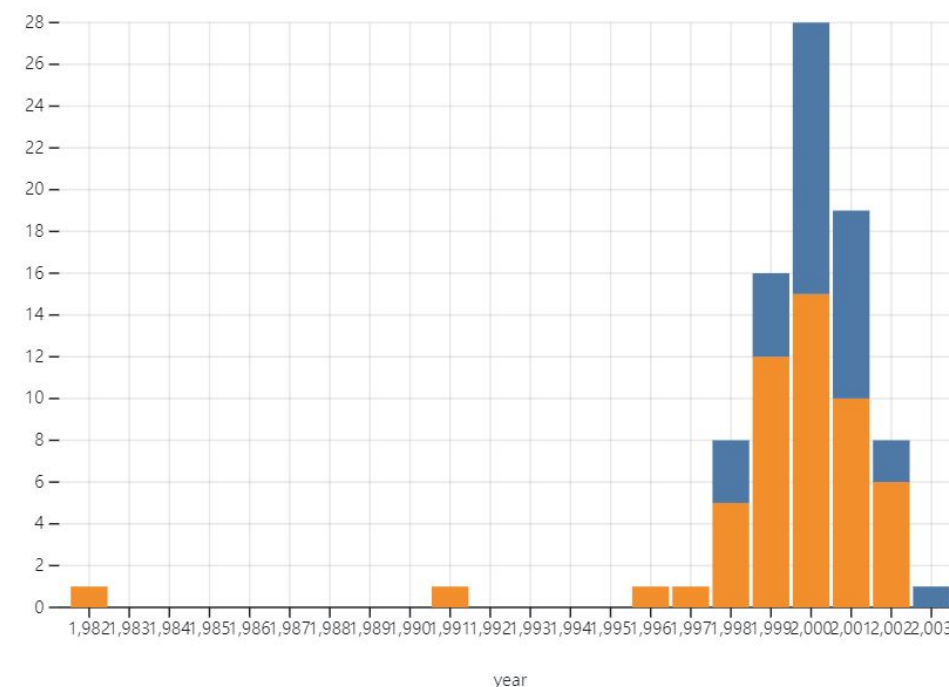
# Medium baseline (4pt)

- 可調整margin, fill color, tip (2pt)

```
Plot.plot({  
  marginTop: plot2.mt,  
  marginRight: plot2.mr,  
  marginBottom: plot2.mb,  
  marginLeft: plot2.ml,  
  
  grid: true,  
  y: {label: "count"},  
  marks: [  
    Plot.ruleY([0]),  
    Plot.barY(yCounts, {x: "year", y: "count", tip: true, fill:"gender"}),  
  ]  
});
```

marginTop	<input type="text" value="50"/>	<input type="range" value="50"/>
marginRight	<input type="text" value="50"/>	<input type="range" value="50"/>
marginBottom	<input type="text" value="50"/>	<input type="range" value="50"/>
marginLeft	<input type="text" value="50"/>	<input type="range" value="50"/>

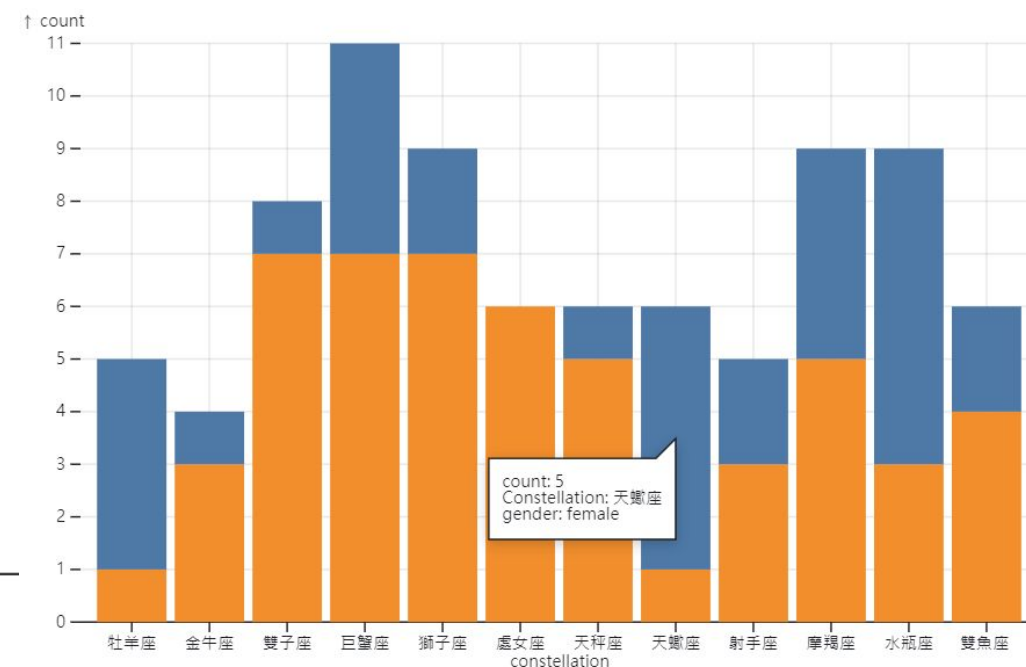
↑ count





# Strong baseline -1 (1pt)

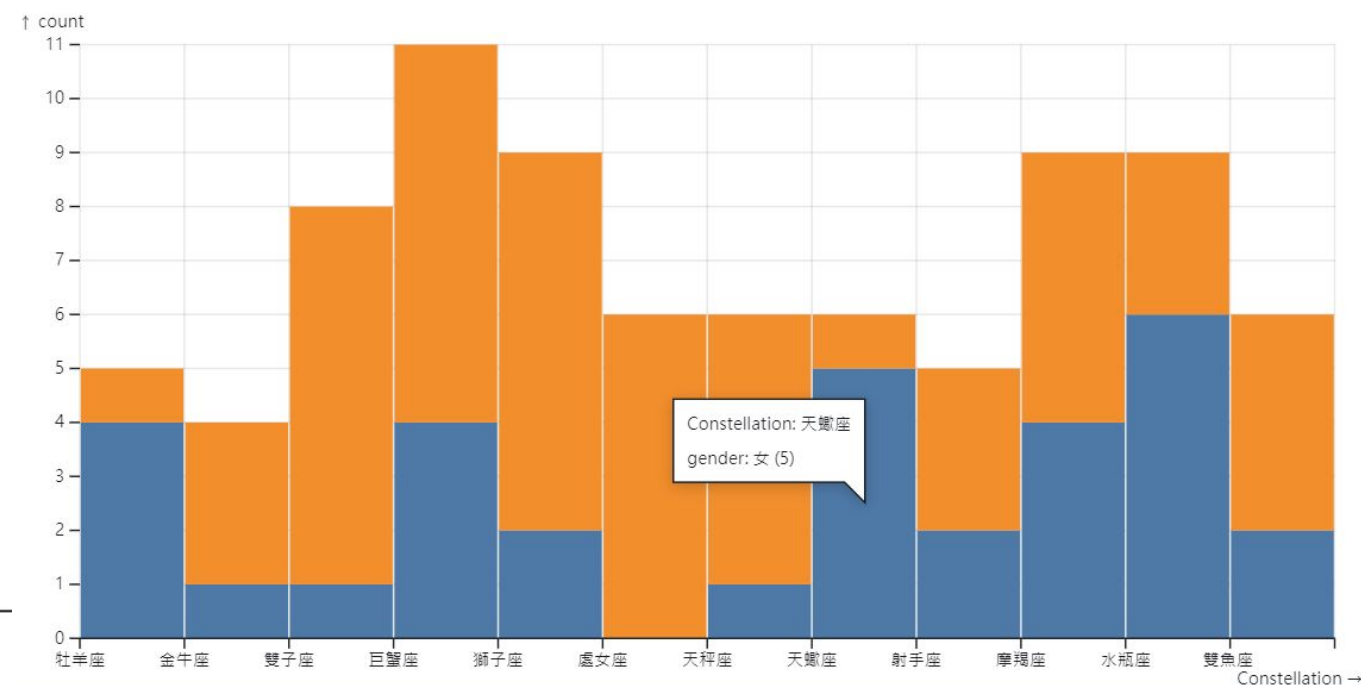
- 以 bar chart 呈現每個星座的人數, 用顏色表示性別
- X軸為各個星座的名稱(須為中文), Y軸為人數
- 必須包含sign, 格式如範例圖





# Strong baseline -2 (1pt)

- 以 histogram 呈現每個星座的人數, 用顏色表示性別
- X軸為各個星座的名稱(須為中文), Y軸為人數
- 必須包含sign, 格式如範例圖





# Strong baseline

星座和對應的數字：

牡羊座	金牛座	雙子座	巨蟹座	獅子座	處女座	天秤座	天蠍座	射手座	摩羯座	水瓶座	雙魚座
0	1	2	3	4	5	6	7	8	9	10	11





# 06

## 繳交資訊

# Export Observable code

1. 在匯出檔案前請確認所有cell都有run過、圖表皆有呈現。

資料可視化Follow

PublicBy 許加宜Edited Oct 14

+ Add tags

## HW2 Simple baseline (4pt)

data = ▶ Array(84) [Object, Object, Object, Object, Object, Object, Object, Object, Object, Object]

```
data = FileAttachment("data.json").json()
```

↑ count  
28 —  
26 —  
24 —  
22 —  
20 —  
18 —

Share...Fork...LikeAdd to collections...Pause live editingViewExportDelete...

Embed cellsDownload code

1. 點擊 ...

2. 選擇 Export

3. Download code



# Export Observable code

2. 將下載下來的檔案解壓縮，刪除裡面的files資料夾。並修改最主要的js檔案。




(名字是亂碼的那個，例如 ce913880495a6f30@76.js)，並將URL路徑改成 **../data.json**

```
43
44 export default function define(runtime, observer) {
45   const main = runtime.module();
46   function toString() { return this.url; }
47   const fileAttachments = new Map([
48     ["data.json", {url: new URL("../data.json", import.meta.url), mimeType: "application/json", toString}]
49   ]);
50   main.builtin("FileAttachment", runtime.fileAttachments(name => fileAttachments.get(name)));
```




# Github



## 3. 將data.json放到 src 檔案夾底下

  main vis2023f / hw02 / src / 

t Add file ...

Directory successfully deleted.

 **JoyceHsu0** Delete hw02/src/simple directory 3ae2474 · now History

Name	Last commit message	Last commit date
 ..		
 data.json	Add files via upload	15 minutes ago



# Github

4. 在 src 檔案夾底下建立simple/medium/strong 檔案夾，並分別將修改過後的檔案放入。

vis2023f / hw02 / src / simple / 在simple檔案夾底下  t Add file ...

JoyceHsu0 Add files via upload ✓ ff6d240 · 1 hour ago History

Name	Last commit message	Last commit date
..		
README.md	Add files via upload	1 hour ago
ce913880495a6f30@76.js	Add files via upload	1 hour ago
index.html	Add files via upload	1 hour ago
index.js	Add files via upload	1 hour ago
inspector.css	Add files via upload	1 hour ago
package.json	Add files via upload	1 hour ago
runtime.js	Add files via upload	1 hour ago



# Github

## 5. 更改 hw02/index.html 178~185行程式碼為：

```
<!-- ----->
<div class="row hw12">
  <div class="col-md-12 twenty">
    Simple baseline 程式碼, 請放在 https://github.com/
    <span style="color:red;">你的帳號</span>
    /vis2023f/hw
    <span style="color:red;">02</span>
    /src/simple 檔案夾。
    <br>

    <hr>
    Medium baseline 程式碼, 請放在 https://github.com/
    <span style="color:red;">你的帳號</span>
    /vis2023f/hw
    <span style="color:red;">02</span>
    /src/medium 檔案夾。
    <br>

    <hr>
    Strong baseline 程式碼, 請放在 https://github.com/
    <span style="color:red;">你的帳號</span>
    /vis2023f/hw
    <span style="color:red;">02</span>
    /src/strong 檔案夾。
    <br>
```

```
<!-- ----->
```

# Export notebook

資料可視化Follow

PublicBy 許加宜 Edited Oct 14

+ Add tags

## HW2 Simple baseline (4pt)

data = ▶ Array(84) [Object, Object, Object, Object, Object, Object, Object, Object, Object, Object]

```
data = FileAttachment("data.json").json()
```

↑ count  
28 —  
26 —  
24 —  
22 —  
20 —  
18 —  
16 —

Share...Fork...LikeAdd to collections...Pause live editingViewExportSettingsEmbed cellsDownload codeDelete...

1. 點擊 ...

2. 選擇 Export

3. Embed cells



# Export notebook

## Embed

Select cells to embed in any web page. [Learn more.](#)

☒ Entire notebook

1. 選擇 Entire notebook

iframe

2. 確認是Iframe

```
<iframe width="100%" height="500" frameborder="0" src="https://observablehq.com/embed/ce913880495a3f30?ce=">
```

3. Copy

☐ Copy URL only ☒ Copy

## Preview

### HW2 Simple baseline (4pt)

data = ▶ Array(84) [Object, Object, Object, Object]

marginTop: 50

資料可視化 HW2 Simple baseline (4pt)

Observable





# Iframe

在hw02/index.html中原本為影片的地方程式碼如下：

```
<!-- ----->
<div class="row hw12">
  <div class="col-md-12 twenty">
    <video id="video1" style="width:100%;max-width:100%;"
controls="">
      <source src="https://www.w3schools.com/html/mov_bbb.mp4"
type="video/mp4">
    </video>
  </div>
</div>

<!-- ----->
```



# Iframe

將其整段更改為以下程式碼，並將剛剛 Embed notebook 時copy的程式碼貼上(如下方紅字處)並新增下方紫字的程式碼，用來將Iframe背景顏色設為白色。

```
<!-- ----->
<div class="row hw12">
  <div class="col-md-12 twenty">
    <h4>Simple baseline - 以 histogram 呈現每個年份出生的人數，可調整margin, fill color, tip</h4>
  </div>
</div>

<div class="row hw12">
  <div class="col-md-12 twenty">
    <iframe width="100%" height="500" frameborder="0" style="background-color: #FFFFFF"
src="...."></iframe>
  </div>
</div>

<!-- ----->
<div class="row hw12">
  <div class="col-md-12 twenty">
    <h4>Medium baseline - 以 bar chart 呈現每個年份出生的人數，可調整margin, fill color, tip</h4>
  </div>
</div>
```



# Iframe-圖表沒有跑出來的問題





# Iframe-圖表沒有跑出來的問題

1. 設定

2. 隱私權和安全性

3. 清除瀏覽資料

3. 清除快取  
4. 重新整理網頁



# 評分表

## 更改 hw02/index.html 中程式碼

總分	完成後打勾	配分	分項描述
10	<input type="checkbox"/> 完成	2	Simple baseline-1 - 以histogram呈現每個年份出生的人數
	<input type="checkbox"/> 完成	1	Simple baseline-2 - 可調整margin, fill color, tip
	<input type="checkbox"/> 完成	2	Medium baseline-1 - 以bar chart呈現每個年份出生的人數
	<input type="checkbox"/> 完成	1	Medium baseline-2 - 可調整margin, fill color, tip
	<input type="checkbox"/> 完成	2	Strong baseline-1 - 以bar chart呈現每個星座的人數
	<input type="checkbox"/> 完成	2	Strong baseline-2 - 以histogram呈現每個星座的人數

```
<table>
  <tr>
    <th>總分</th>
    <th>完成後打勾</th>
    <th>配分</th>
    <th>分項描述</th>
  </tr>

  <tr>
    <td rowspan="6" id="myTotal"></td>
    <td><input type="checkbox" class="flipswitch" id="myCheckbox1"
checked="checked"></td>
    <td id='m1'>2</td>
    <td>Simple baseline-1 - 以histogram呈現每個年份出生的人數</td>
  </tr>

  <tr>
    <td><input type="checkbox" class="flipswitch" id="myCheckbox2"
checked="checked"></td>
    <td id='m2'>2</td>
    <td>Simple baseline-2 - 可調整margin, fill color, tip</td>
  </tr>

  <tr>
    <td><input type="checkbox" class="flipswitch" id="myCheckbox3"
checked="checked"></td>
    <td id='m3'>2</td>
    <td>Medium baseline-1 - 以bar chart呈現每個年份出生的人數</td>
  </tr>

  <tr>
    <td><input type="checkbox" class="flipswitch" id="myCheckbox4"
checked="checked"></td>
    <td id='m4'>2</td>
    <td>Medium baseline-2 - 可調整margin, fill color, tip</td>
  </tr>

  <tr>
    <td><input type="checkbox" class="flipswitch" id="myCheckbox5"
checked="checked"></td>
    <td id='m5'>2</td>
    <td>Strong baseline-1 - 以bar chart呈現每個星座的人數</td>
  </tr>

  <tr>
    <td><input type="checkbox" class="flipswitch" id="myCheckbox6"
checked="checked"></td>
    <td id='m6'>2</td>
    <td>Strong baseline-2 - 以histogram呈現每個星座的人數</td>
  </tr>
</table>
```

# 評分表

更改 hw02/index.html 中程式碼

```
function update(){  
  
    var score = 0;  
  
    if (d3.select("#myCheckbox1").property("checked")){  
        score += parseInt(d3.select('#m1').html());  
    }  
  
    if (d3.select("#myCheckbox2").property("checked")){  
        score += parseInt(d3.select('#m2').html());  
    }  
  
    if (d3.select("#myCheckbox3").property("checked")){  
        score += parseInt(d3.select('#m3').html());  
    }  
  
    if (d3.select("#myCheckbox4").property("checked")){  
        score += parseInt(d3.select('#m4').html());  
    }  
  
    if (d3.select("#myCheckbox5").property("checked")){  
        score += parseInt(d3.select('#m5').html());  
    }  
  
    if (d3.select("#myCheckbox6").property("checked")){  
        score += parseInt(d3.select('#m6').html());  
    }  
  
    d3.select("#myTotal").html(score);  
}
```





<https://joycehsu0.github.io/vis2023f/hw02/index.html>

## 作業 02

Simple baseline 程式碼，請放在 [https://github.com/ 你的帳號 /vis2023f/hw 02 /src/simple](https://github.com/你的帳號/vis2023f/hw02/src/simple) 檔案夾。

Medium baseline 程式碼，請放在 [https://github.com/ 你的帳號 /vis2023f/hw 02 /src/medium](https://github.com/你的帳號/vis2023f/hw02/src/medium) 檔案夾。

Strong baseline 程式碼，請放在 [https://github.com/ 你的帳號 /vis2023f/hw 02 /src/strong](https://github.com/你的帳號/vis2023f/hw02/src/strong) 檔案夾。

前往你的 Demo 網頁

<https://joycehsu0.github.io/vis2023f/hw02/src/simple/>

<https://joycehsu0.github.io/vis2023f/hw02/src/medium/>

<https://joycehsu0.github.io/vis2023f/hw02/src/strong/>

心得

這次的作業遇到的困難：1. 更改路由標題的值 2. 將一開始讀到的資料集轉換成能用 react 呈現的 Object Array 3. 更改



# Regulations

- **You should finish your homework on your own.**
- **Do not share your codes with any living creatures.**
- **Your HW will get 0 pt if you violate any of the above rules.**
- **Professor & TAs preserve the rights to change the rules & grades.**





# 07

## 助教聯絡資訊



# 助教聯絡資訊

- TA Email
  - 許加宜 [t112598016@ntut.org.tw](mailto:t112598016@ntut.org.tw)
  - Email 標題請按照此格式(X為作業編號) : [vis2023f-hwX-學號]