## CSS Responsive

RWD(Responsive web desinged) intro

這個是響應式的網站設計，根據不同裝置做出不同顯示方式

主要是用下面的方式:

```

<meta name="viewport" content="width=device-width, initial-scale=1.0">

@media only screen and (max-width: 600px) {

/\* For tablets: \*/

//響應式的變化寫在裡面

}

@media only screen and (max-width: 768px) {

/\* For desktop: \*/

//響應式的變化寫在裡面

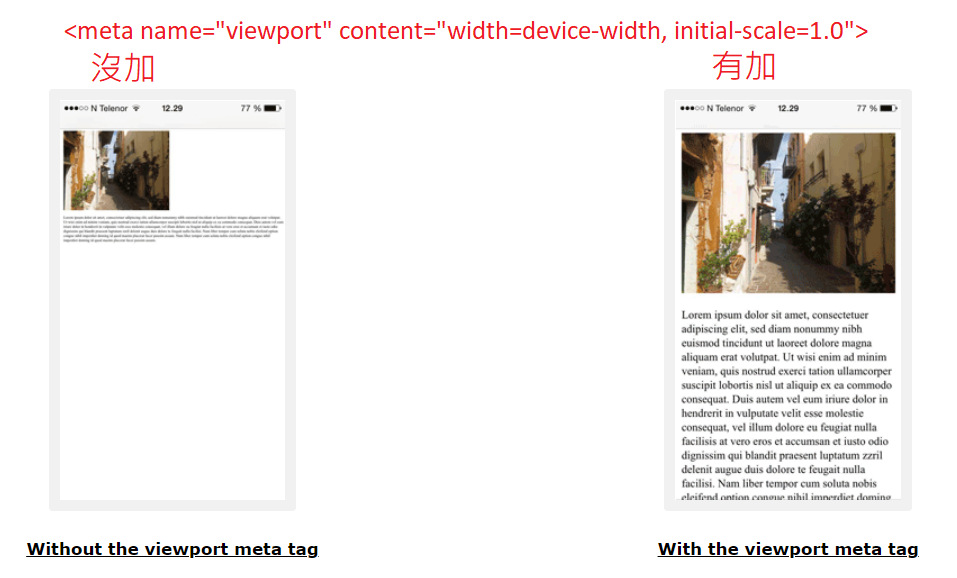
}

```

前面的@media是專門給手機，平板，有螢幕用的。

也有@printer不過很少用

RWD Viewport



```

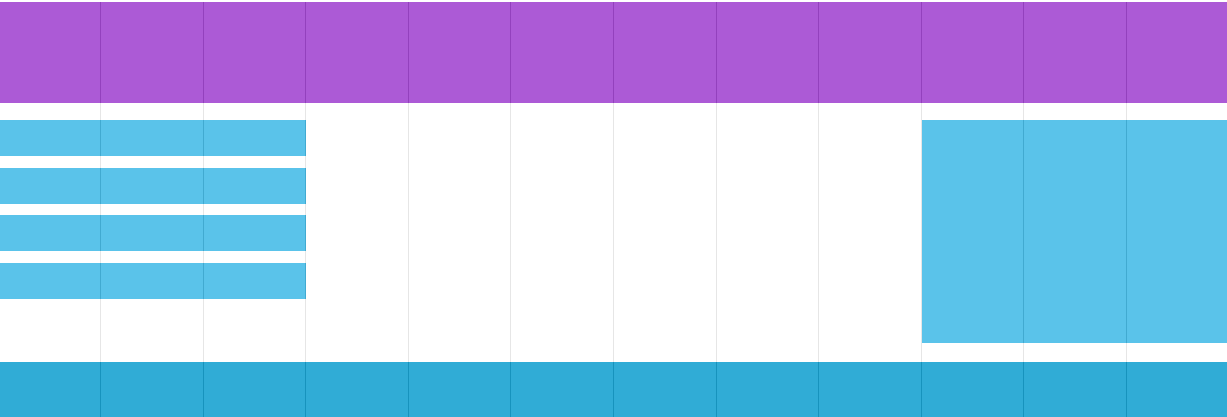
width=device-widh //代表this的寬度變成裝置寬度

initial-scale=1.0 //代表初始設定為100%

```

RWD Grid(網格) View

把網頁用網格分開，好做排版



RWD Imges

利用下面的控制選項，可以讓圖片(或選項)自動調整大小

```

img {

width: 100%;

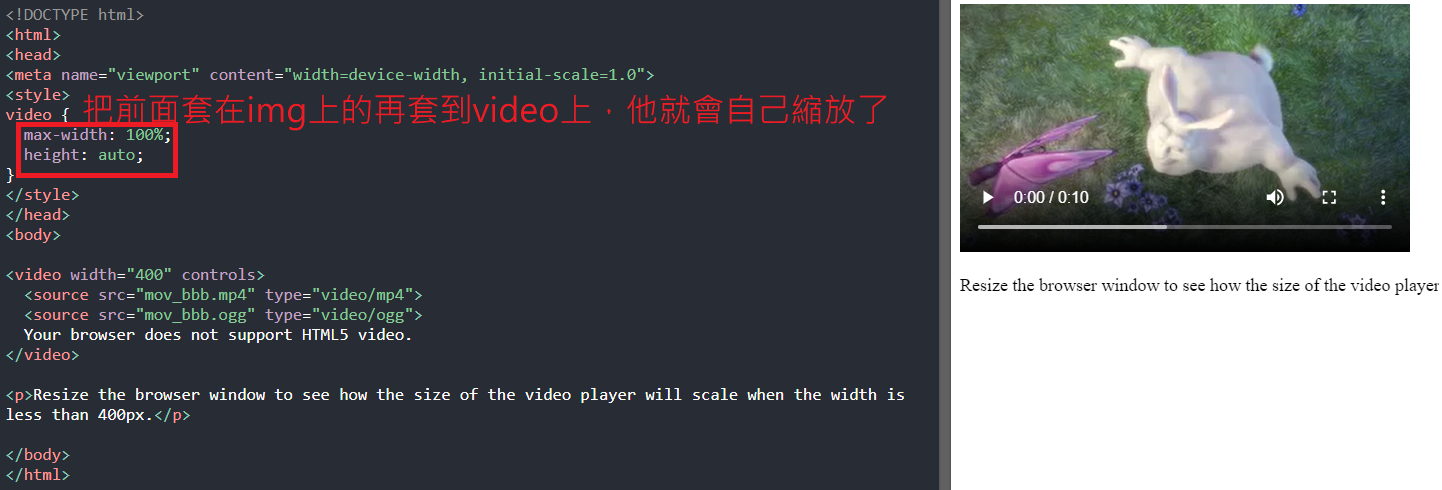
height: auto;

}

```

再配合前面的@media，就可以讓網頁觀賞性大提升

RWD Videos



Video標籤:

```

<video width="320" height="240" controls> //可改成autoplay或muted

<source src="movie.mp4" type="video/mp4">

<source src="movie.ogg" type="video/ogg">

Your browser does not support the video tag. //不支援時顯示

</video>

```

Audio標籤

```

<audio controls> //可改成autoplay或muted

<source src="horse.ogg" type="audio/ogg">

<source src="horse.mp3" type="audio/mpeg">

Your browser does not support the audio element. //不支援時顯示

</audio>

```

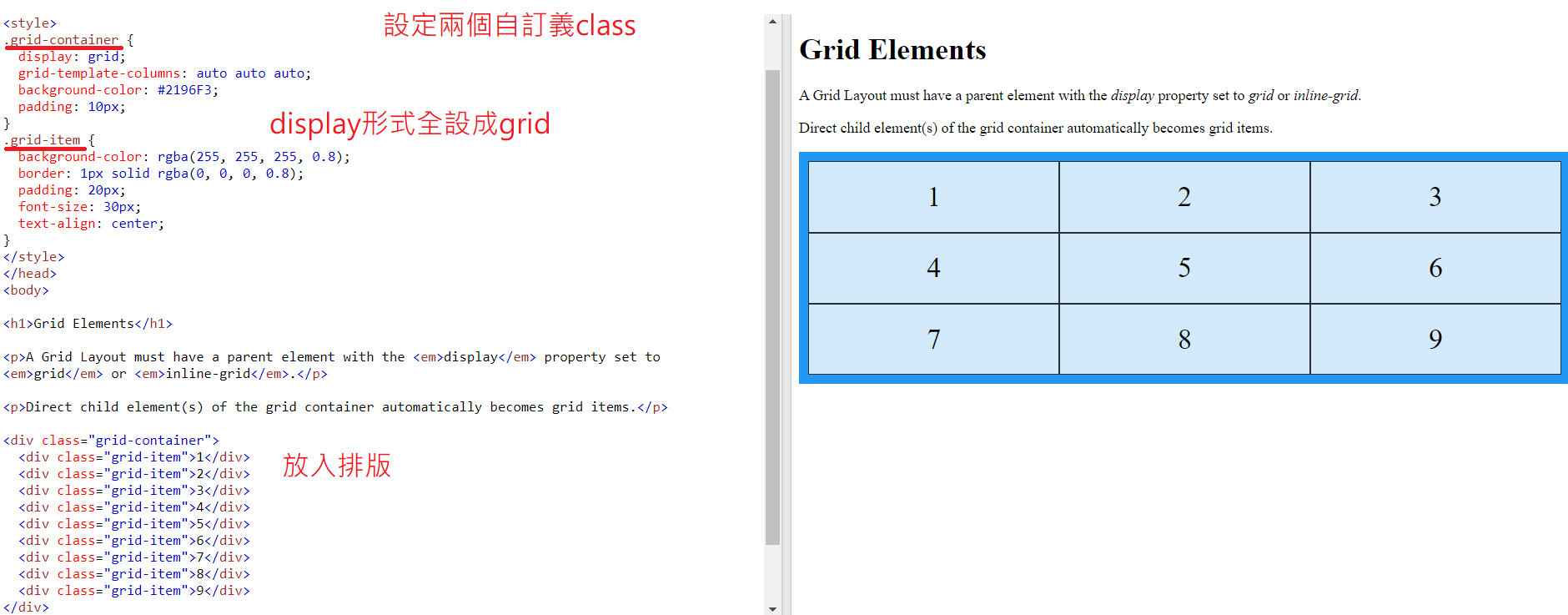
RWD Frameworks, Templates

這兩個基本上算是在介紹套件，網路上的RWD通常都設計得還不錯，但要用需要標明出處喔!!

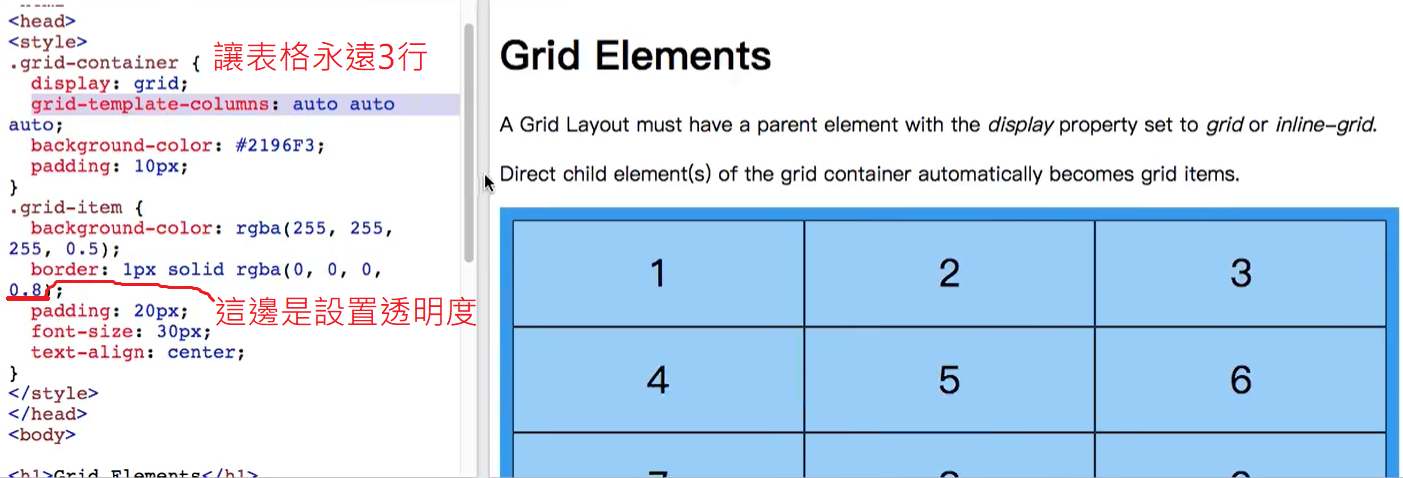
## CSS Grid

Grid intro

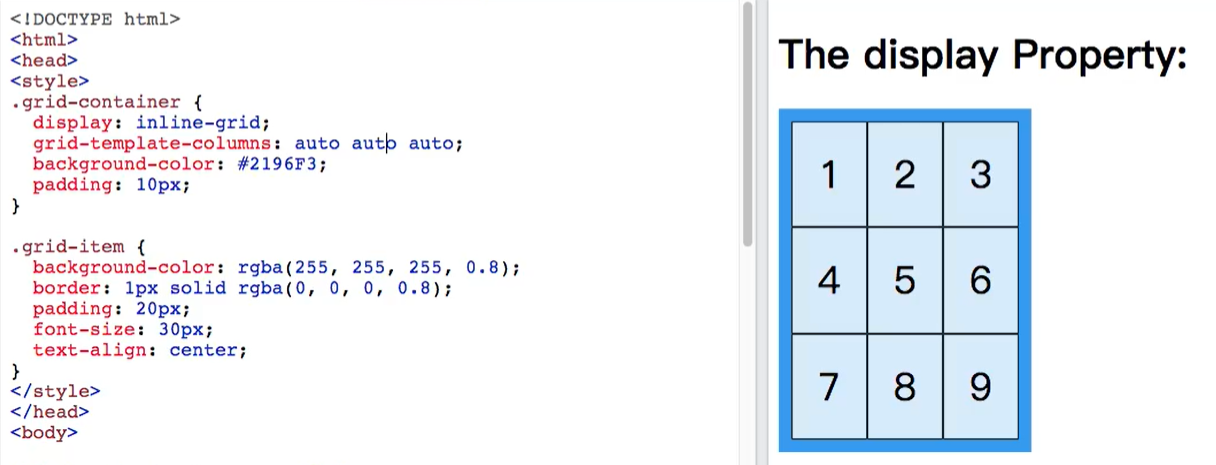
Grid layout是比較新的寫法，通常搭配div來做使用



可以設置其他元素改變樣式

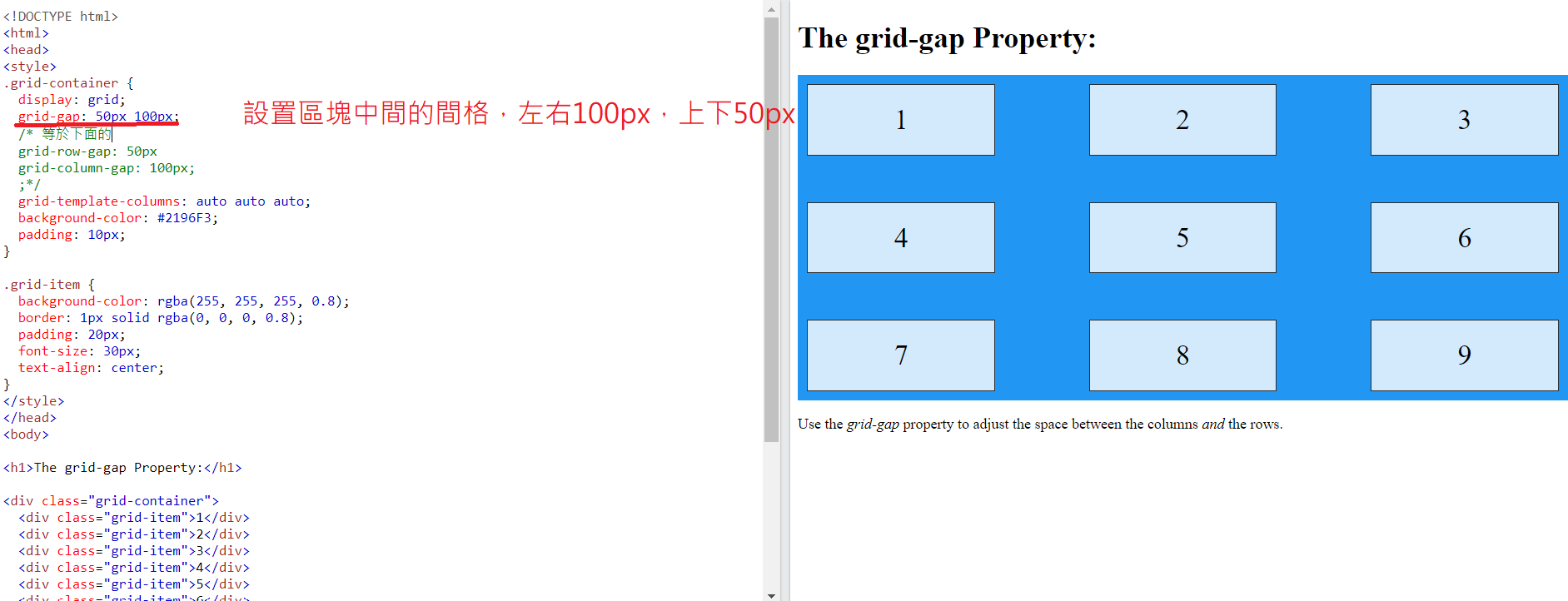


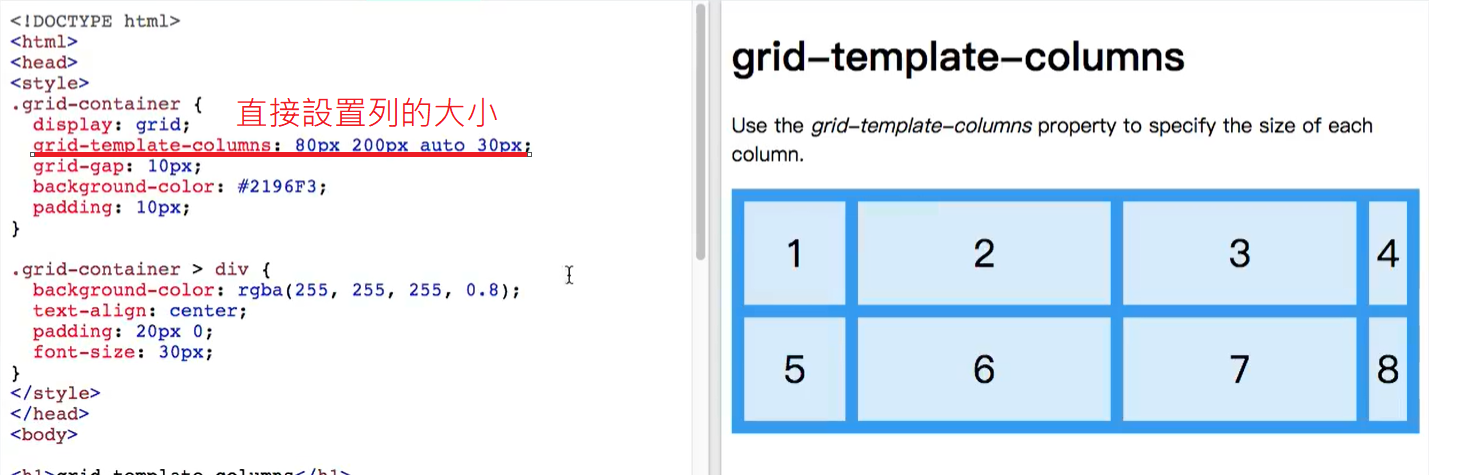
設置inline-grid不會填滿螢幕，會固定大小



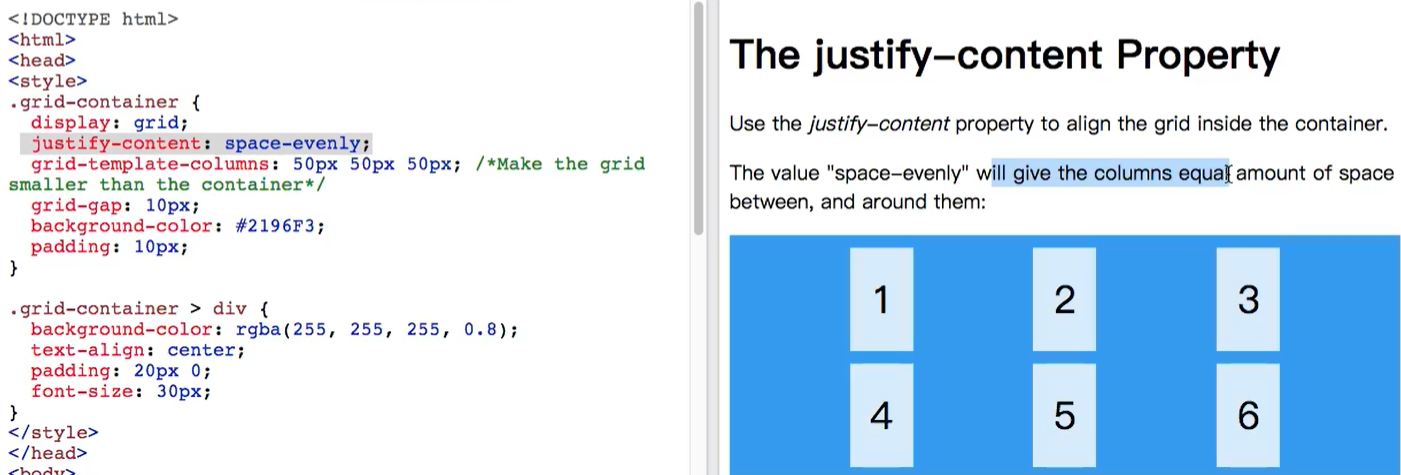
Grid Container

可以設置容器物件間格





也可以讓每個間格都一樣



```

justify-content: start; //全部靠左

justify-content: end; //全部靠右

align-content: center; //全部靠中

align-content: start; //全部靠上

align-content: end; //全部靠下

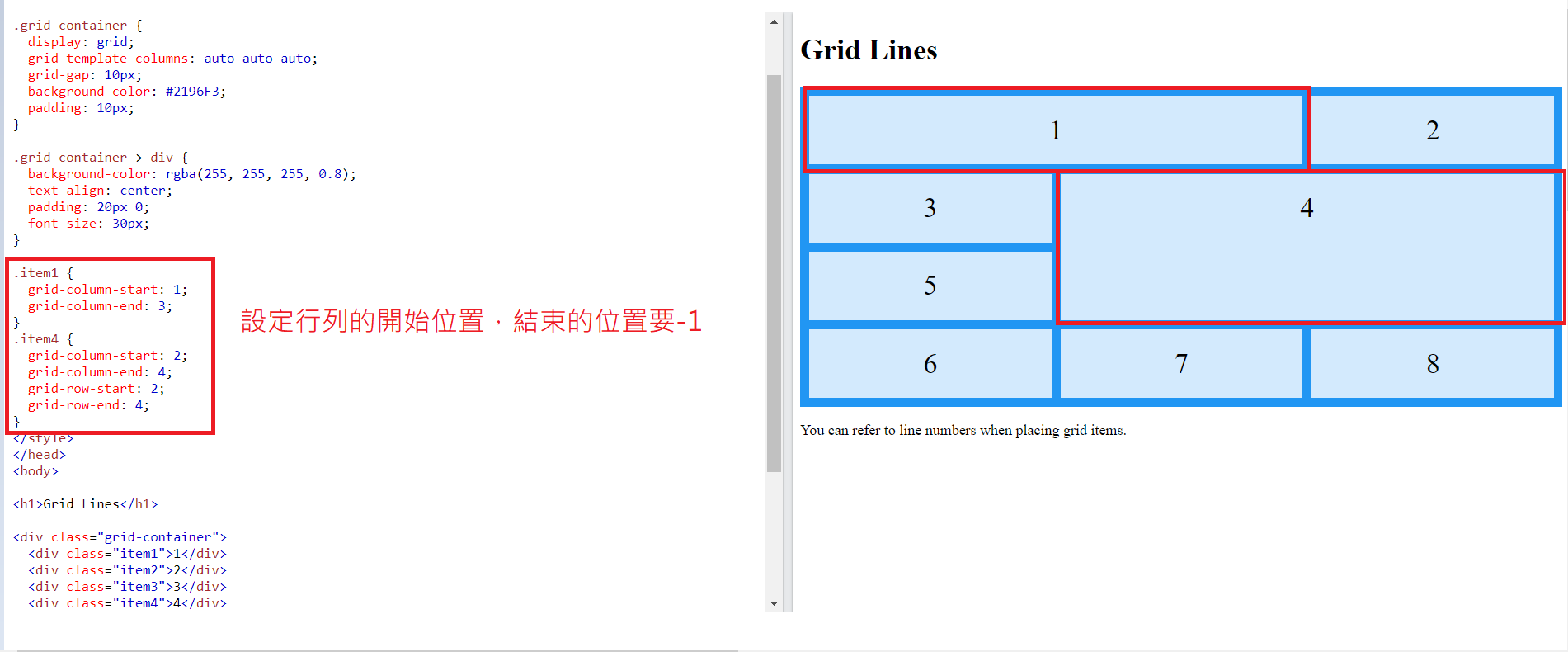
align-content: space-evenly; //平均分配

align-content: space-around; //上下正常，中間空很大

```

Grid Item

可以指定物體位置



這邊也可以用

```

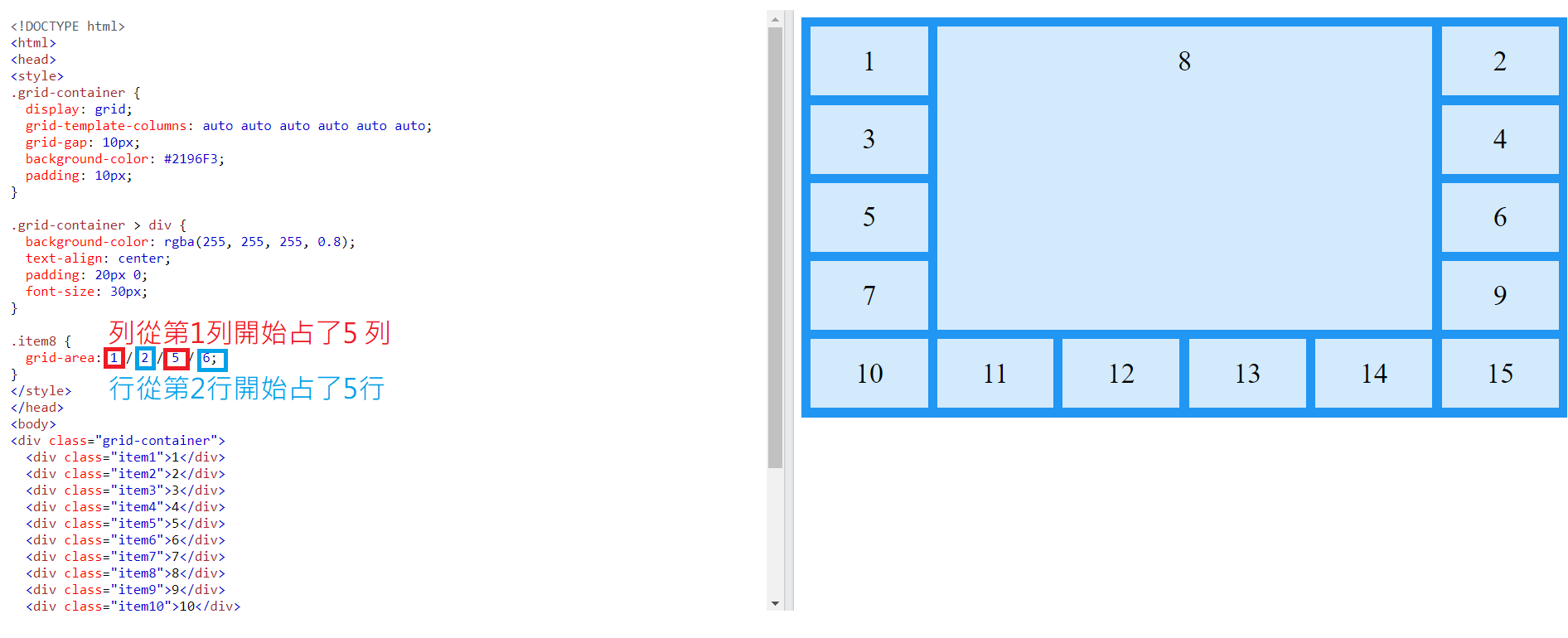
grid-column: 1 / 3; //從1到2

grid-column: 1 / span 2 //從1開始，在後兩格

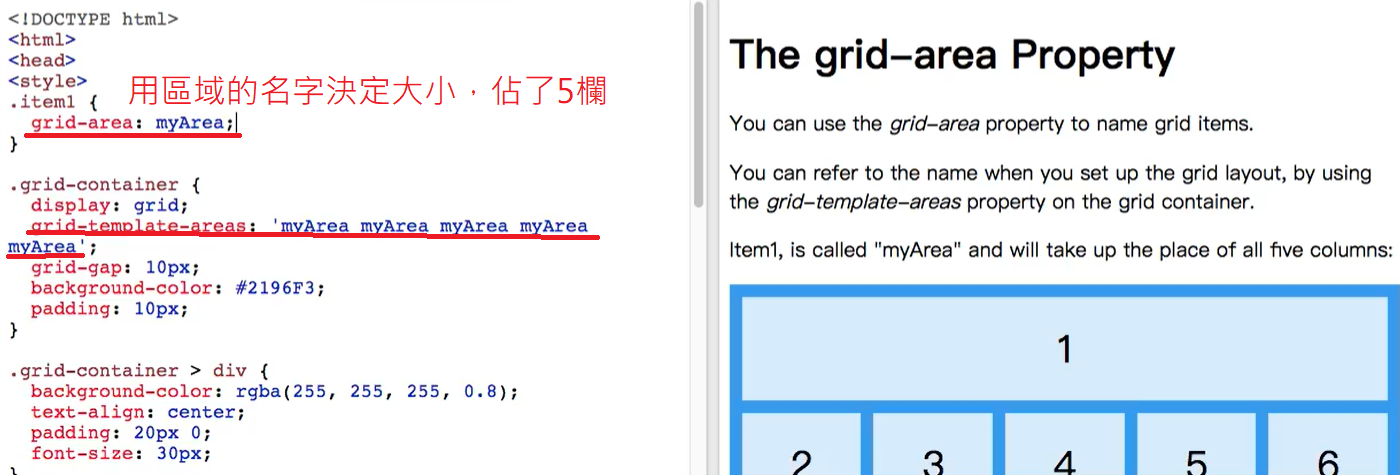
grid-row: 2/ 4; //從2到3

grid-column: 2 / span 2 //從2開始，在後兩格

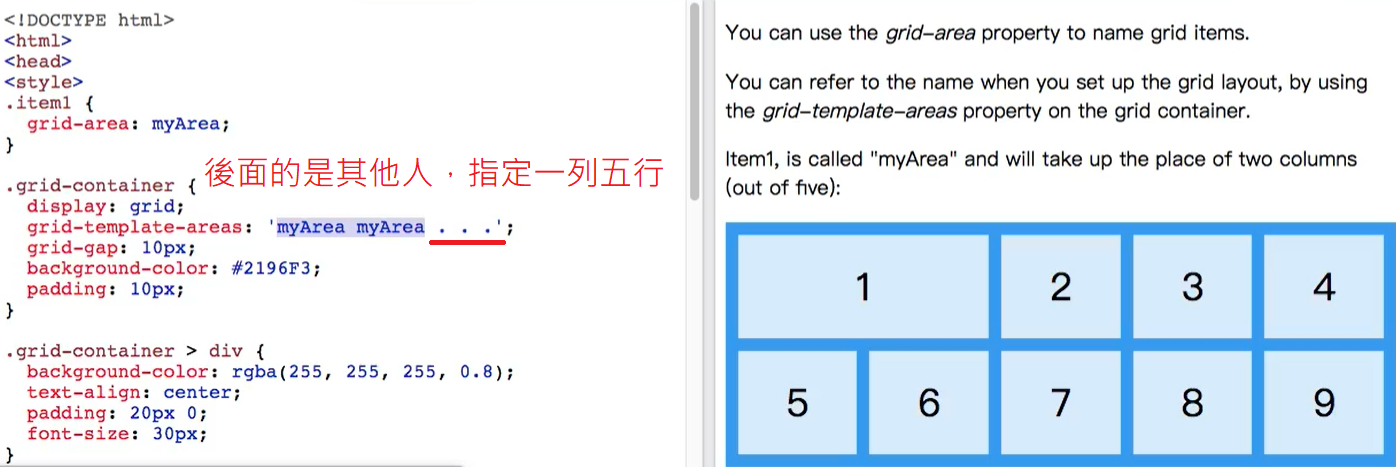
```



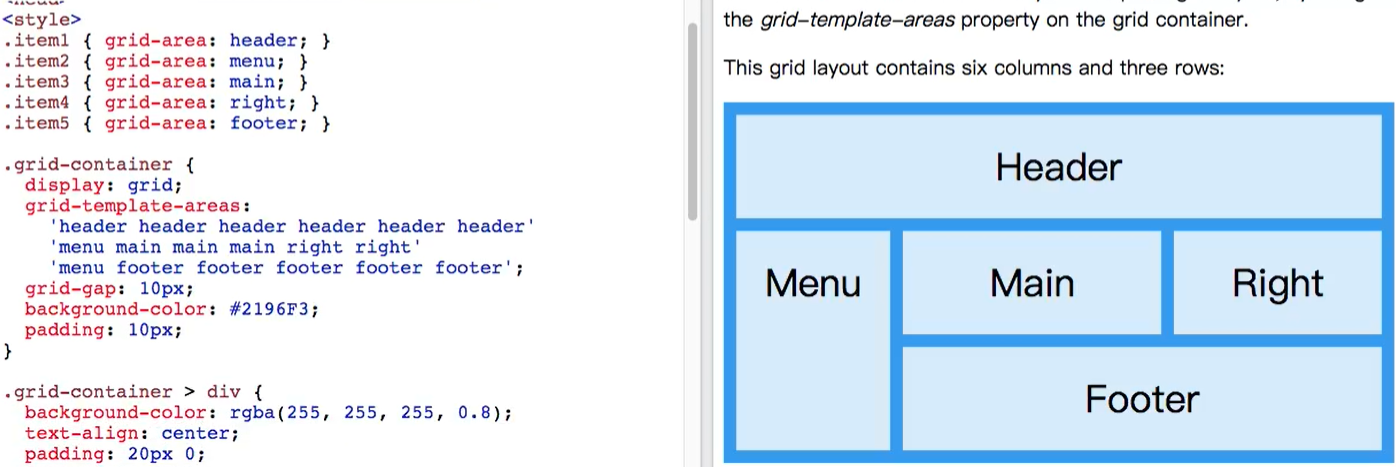
下面的例子是把收到這個class的區域取名子，再帶入到grid裡



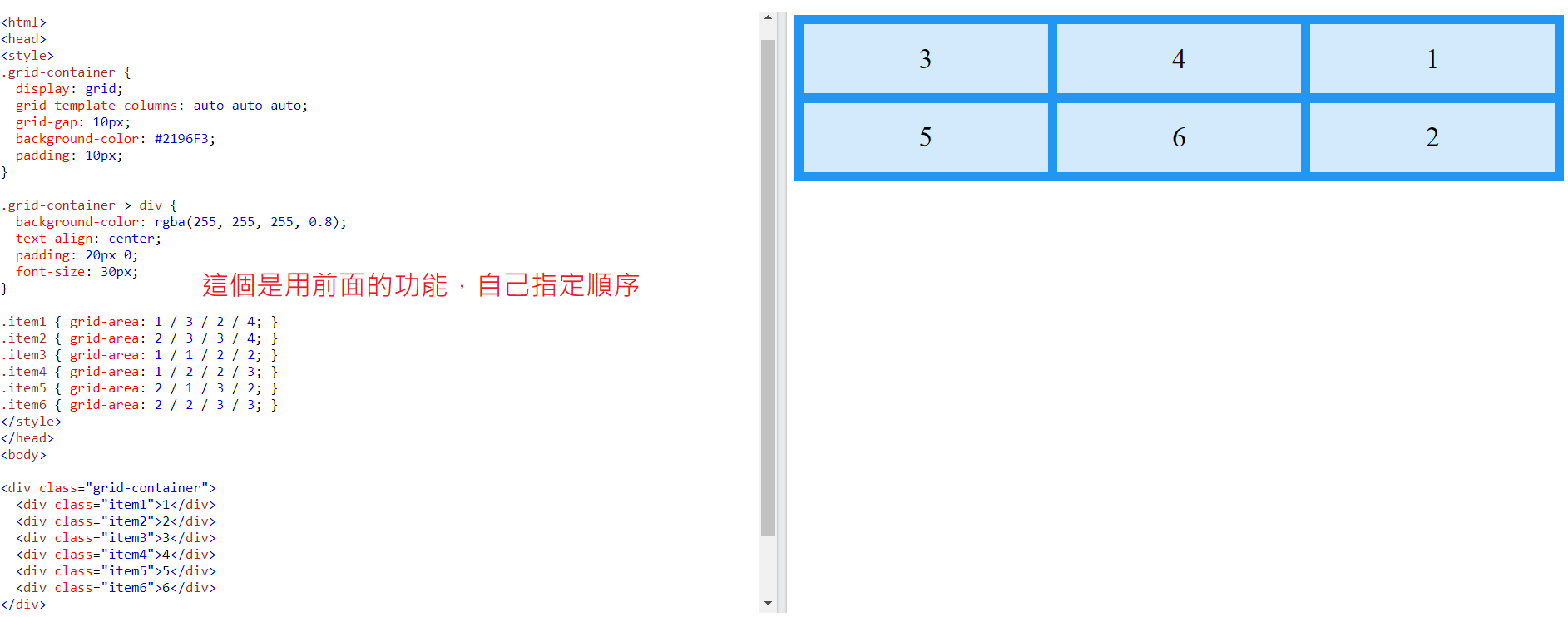
.和.中間要有空格!







指定位置上大小



## CSS SASS

如果要寫大網站，或是很複雜的CSS，可以用這個套件，比較好維護，可以宣告變數。

這個必須要經過編譯，才可以轉成CSS。

[SASS編譯]( <https://sass-lang.com/install>)

可以在VScode裡面打下面的指令行安裝:

```

npm install -g sass //用node.js安裝

choco install sass //用choco安裝

brew install sass/sass/sass //make電腦安裝

```

如果安裝好了，要編譯就打以下指令

```

sass source/stylesheets/index.scss build/stylesheets/index.css

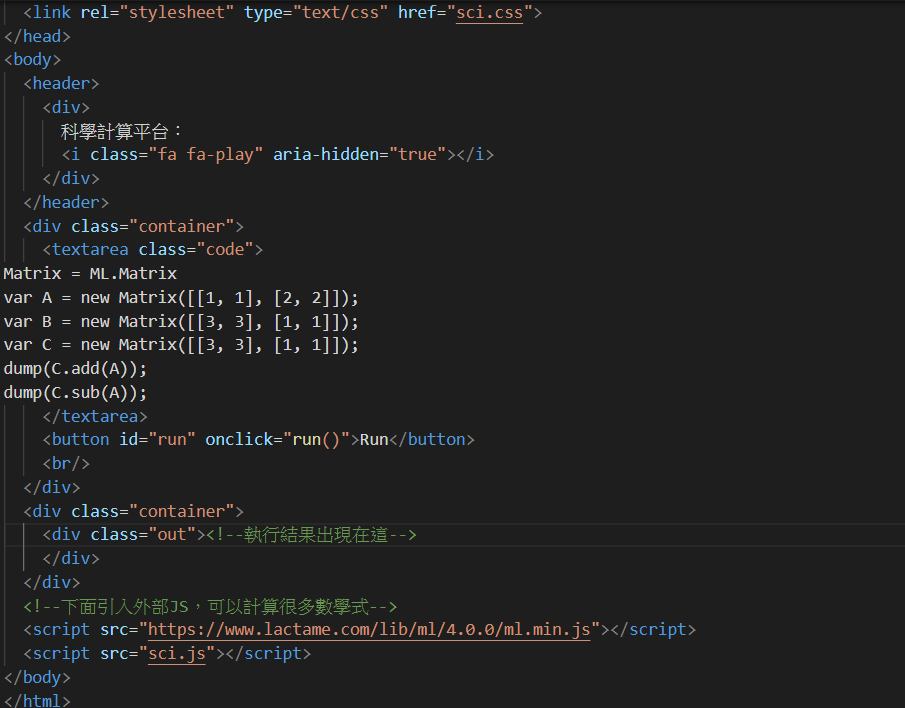
```

可以寫.sass版本的code和.scss的code。

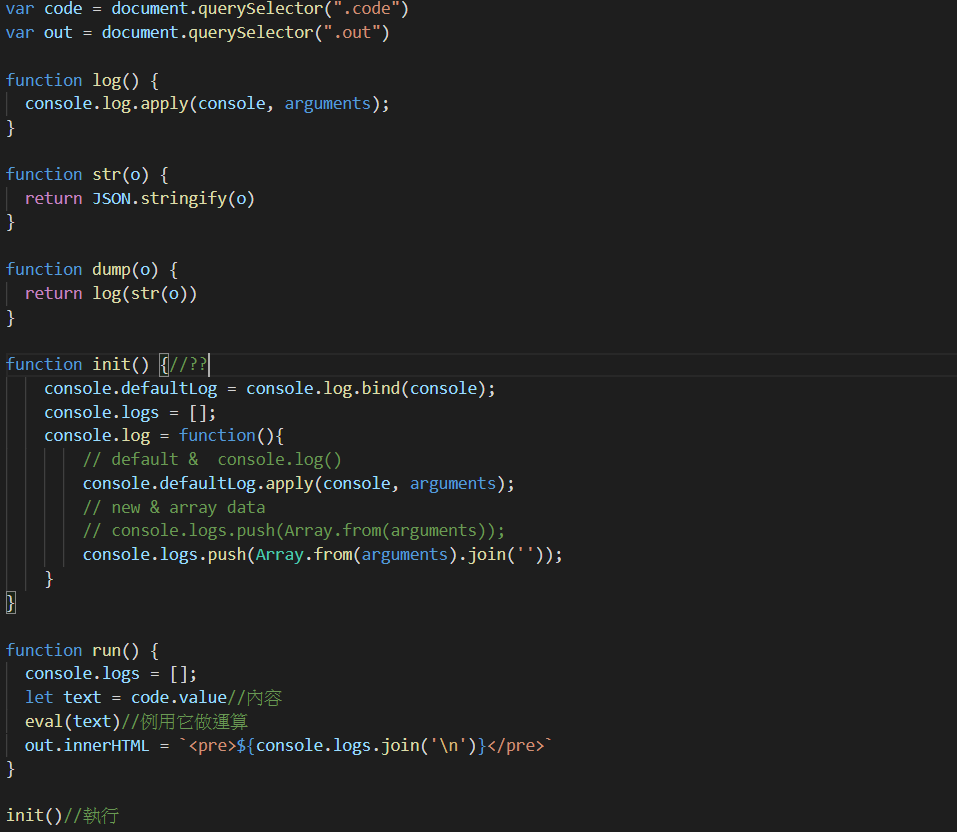
sass不寫{}或;之類的符號，而scss需要，感覺scss可讀性較高

## APP

sci



數學式運算，引用外部資源



Init()其實我不太了QQ，不過好像沒他就沒法執行

pos-report

如果要使用localStorage，這個是很好的參考!

這部分..恩，我只能說這部份程式碼很大，我稍微整理:

```

const Report = {};

const Pos = { order: { totalPrice: 0, records: [], submitted: false } };

const Lib = {};

const Setting = {};

const Shop = { name: '茶舖子', address: '金門縣金寧鄉安美村湖南 33 號', tel: '082-333333', items: {'紅茶': 20, '綠茶': 20, '珍珠奶茶': 35 }, addons: {'去冰': 0, '半糖': 0, '熱': 0, '加鮮奶': 10 }, orderCount: 0, };

const Ui = {};

//先把每個資料夾的主要東西列出來，這些都是大物件，用pos讓我們選單，再把單送到shop矩陣的物件裡面，再用localStorge去存(沒伺服端的替代方法)，report是把shop的東西放到裡面讓我們可以看到，setting把shop的物件列出，再使我們可以修改，甚至增加裡面的內容，lib式顯示時間(定單)，ui是版面的變動控制。

```

這是一個很有系統化的程式，可以看看他的[原始碼]( https://gitlab.com/ccc109/wp/-/tree/master/11-app/pos2/pos6)