

CSS Grid Layout Cheat Sheet

小水豚教你做網站! 2021.9 Ver1.0

※嚴禁在網路上任意轉載、轉發 © Capybara Design

1 | 設定網格容器

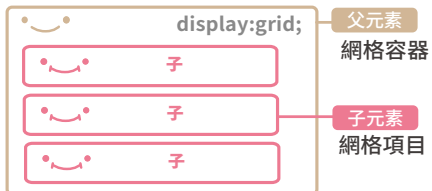
HTML

```
<div class="container">
  <div class="item01"> 項目1 </div>
  <div class="item02"> 項目2 </div>
  <div class="item03"> 項目3 </div>
</div>
```

CSS

在父元素設定 `display: grid;`
`.container {display: grid;}`

父元素 ... 成為網格容器
直接子元素 ... 成為網格項目



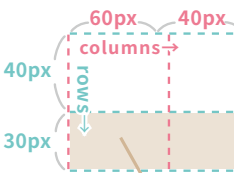
只設定 `display: grid;` 不會產生變化

2 | 設定格線軌道

```
.container {
  grid-template-columns: 60px 40px;
  grid-template-rows: 40px 30px;
}
```

注意這裡要加上 `s` !

分割後的空間 (列或欄) 稱作格線軌道



應用重點

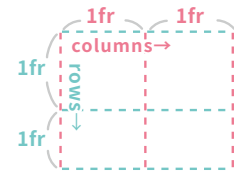
單位 1fr

這裡導入了構成網格的新單位 `fr` (fraction)。

```
.container {
  grid-template-columns: 1fr 1fr;
  grid-template-rows: 1fr 1fr;
}
```

設定了 `fr` 的軌道會依照空間調整大小，
並盡可能擴大。

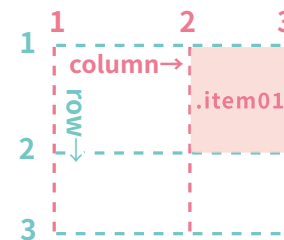
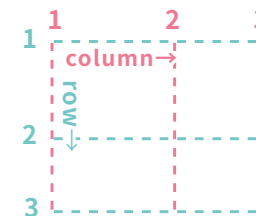
例如設定了多個 `1fr` 時，剩餘的空間會
依照該數量平均分割。`2fr` 的空間是 `1fr`
的兩倍。



3 | 配置網格項目

設定了網格軌道之後，
會自動繪製格線。

格線會依照順序編號，
可以利用編號配置網格
項目。



```
.item01 {
  grid-column-start: 2;
  grid-column-end: 3;
  grid-row-start: 1;
  grid-row-end: 2;
}
```

簡寫

grid-column: 2 / 3;

grid-row: 1 / 2;

沒有加上 `s` !

應用重點

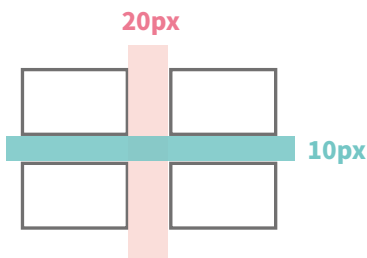
關於 gap

`gap` 是用來設定格線軌道的間隔。

```
.container {
  column-gap: 20px;
  row-gap: 10px;
}
```

簡寫

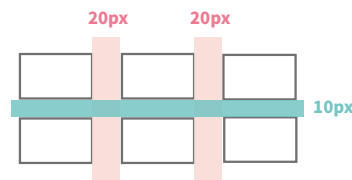
`gap: 10px 20px;`



margin 與 gap 的差異

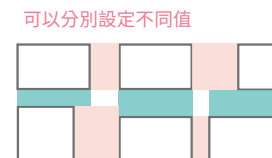
margin

- 設定網格容器 (父元素)
- 決定格線軌道的間隔
- 變成相同間隔 (可以調整 `column` 與 `row`)



margin

- 設定網格項目
- 決定格線軌道的間隔
- 可以分別設定不同值



※ 由於格線與網格項目之間的留白
會改變，所以要注意網格項目的
大小也會產生變化

CSS Grid Layout Cheat Sheet

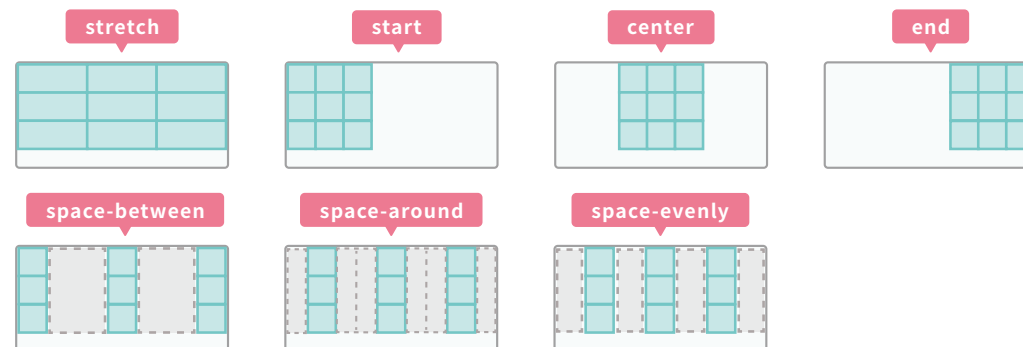
小水豚教你做網站! 2021.9 Ver1.0

※嚴禁在網路上任意轉載、轉發 © Capybara Design

設定網格容器

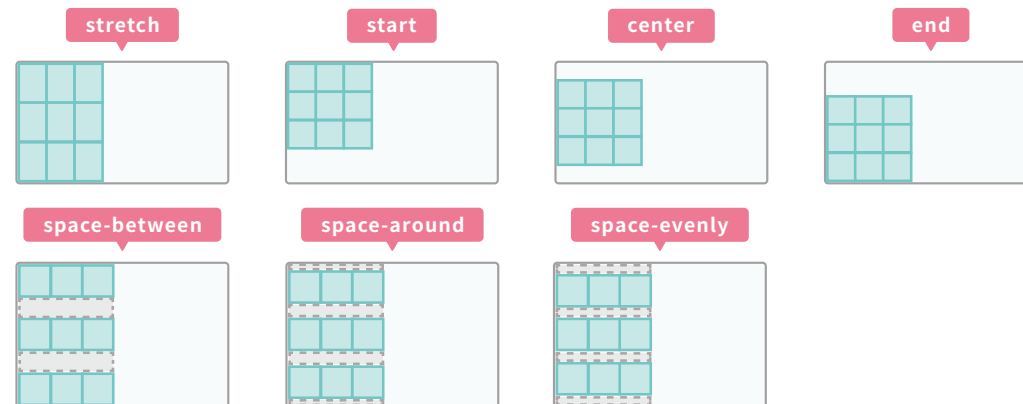
justify-content

設定「整個網格項目」的水平(橫向)位置



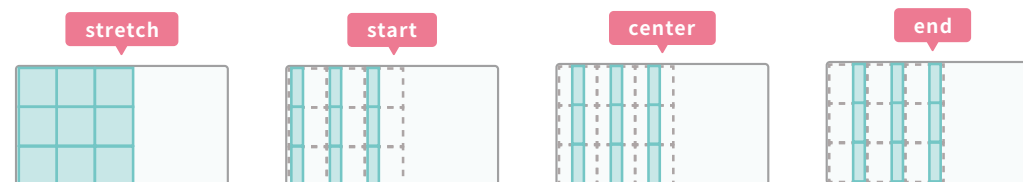
align-content

設定「整個網格項目」的垂直(縱向)位置



justify-items

針對「所有網格項目」的網格儲存格設定水平(橫向)位置



align-items

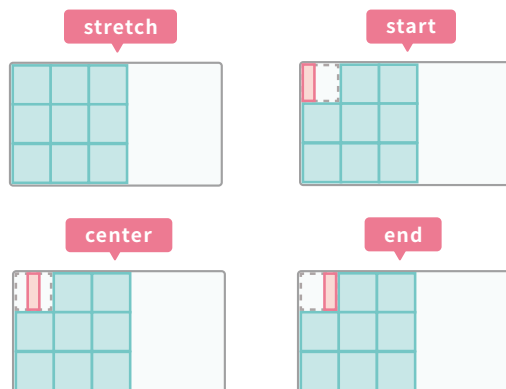
針對「所有網格項目」的網格儲存格設定垂直(縱向)位置



網格容器

justify-self

針對「任意網格項目」的網格儲存格設定水平(橫向)位置



align-self

針對「任意網格項目」的網格儲存格設定垂直(縱向)位置

