

Bootstrap

王維駿

Bootstrap簡介

- <https://getbootstrap.com/>
- 包含許多預設樣式與功能的Framework
- RWD(Responsive Web Design)
- 便於快速開發
- 支援主流瀏覽器及IE10~/Edge

開始使用

- CDN
- Download
- SASS

開始使用-基礎樣板

- HTML Template
- <http://getbootstrap.com/docs/4.3/getting-started/introduction/>
- 台灣繁體中文版網站lang="zh-Hant-TW"
- <http://www.iana.org/assignments/language-subtag-registry/language-subtag-registry>
- 確保RWD正確作用的meta tag
- CSS放在<head>區塊
- JS放在</body>前，注意載入順序

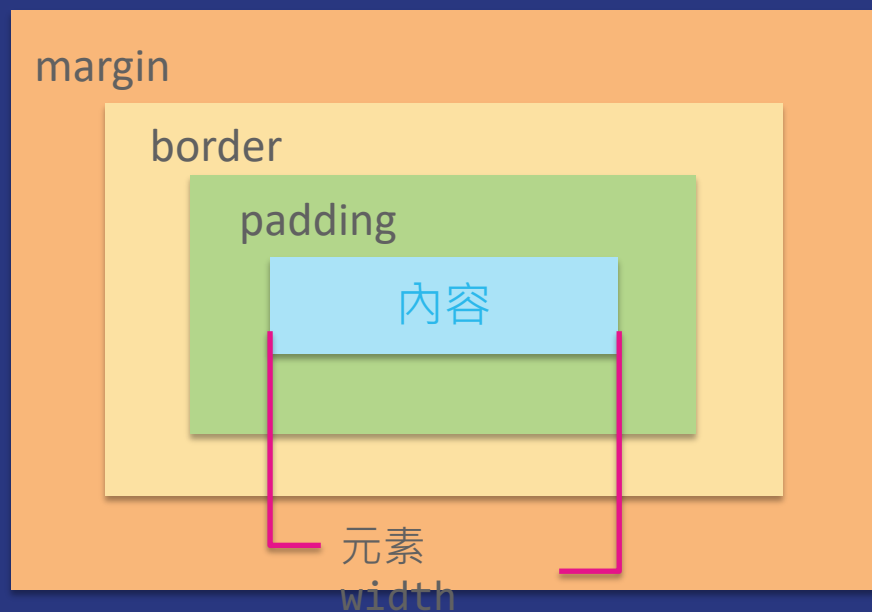
bootstrap預設值

- 一定要HTML5
- 使用Reboot統一預設樣式
- <https://getbootstrap.com/docs/4.3/content/reboot/>
- Box-sizing :
 - `box-sizing: border-box;`

box-sizing

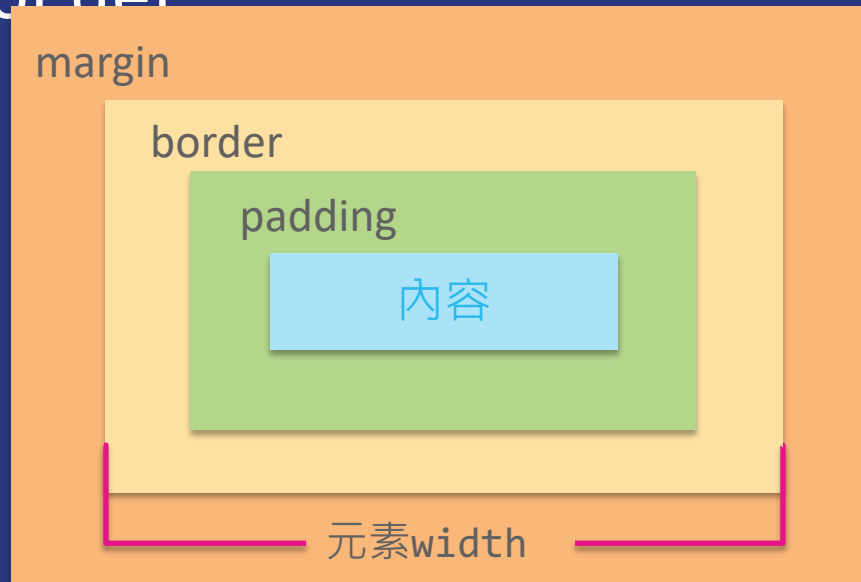
- 網頁預設

- 元素width =
內容的width



- Bootstrap

- 元素width =
內容的width + padding +
border



建立Layout與版面切割

- Bootstrap有兩種外層容器
- <https://getbootstrap.com/docs/4.3/layout/overview/#containers>
- 寬度有限的置中容器.container，最大寬度為1140px
- 滿版容器.container-fluid
- <https://getbootstrap.com/docs/4.3/utilities/colors/#background-color>

建立Layout與版面切割

- Bootstrap的RWD
- Mobile first
- 用四個預設斷點配合關鍵字來達成

主要對象	小螢幕手機 Portrait phones	大螢幕手機 Landscape phones	平板 Tablets	電腦螢幕 Desktops	大電腦螢幕 Large desktops
尺寸	<576px	>=576px	>=768px	>=992px	>=1200px
寬度關鍵字	無 (單純指尺寸時常寫為xs)	sm	md	lg	xl

建立Layout與版面切割

- Grid
- 便於整齊、劃分版面中的內容
- 網頁設計中較常見12欄、6欄
- Bootstrap使用12欄
- 每欄兼具30px
- 第一欄及最後一欄離邊寬15px

建立Layout與版面切割-Flexbox

- Flexbox
- 基本構造由一個容器與作為內容物的第一層子元素構成透過不同的屬性來控制內容物在容器中的尺寸大小及排列方式
- Bootstrap中
- 容器是.row
- 子元素則以.col-開頭

建立Layout與版面切割

- 如何在不同的瀏覽器寬度下使用不同的欄位數？
- .使用.col-{寬度關鍵字}-{數字}
- 同樣的可以用.col-{寬度關鍵字}使該區塊在指定的瀏覽器中自動分配寬度

建立Layout與版面切割-練習

- 打開 `layout-1.html`，透過加入Bootstrap的預設排版樣式使內容在不同寬度的瀏覽器中會自動改變排版

建立Layout與版面切割

- Flexbox垂直對齊
- <https://getbootstrap.com/docs/4.3/utilities/flex/#align-items>
- <https://getbootstrap.com/docs/4.3/utilities/flex/#align-self>

指定對象	容器	子元素
作用對象	全部子元素	套用子元素
置頂	.align-items-start	.align-self-start
置中	.align-items-center	.align-self-center
置下	.align-items-end	.align-self-end
依baseline對齊	.align-items-baseline	.align-self-baseline
與容器同高	.align-items-stretch	.align-self-stretch

備註：

- .align-self-baseline 套用的子元素首行對齊baseline
- 皆可搭配寬度關鍵字使用：
 - .align-items-{寬度關鍵字}-{排列方式} .align-self-{寬度關鍵字}-{排列方式}
- 若指定子元素樣式與指定容器的樣式相衝突時，以指定子元素樣式為主

建立Layout與版面切割-Flex-21

- Flexbox水平對齊
- <https://getbootstrap.com/docs/4.3/utilities/flex/#justify->

指定對象	容器
作用對象	全部子元素
置左	.justify-content-start
置中	.justify-content-center
置右	.justify-content-end
平均分配元素與兩側邊距間的空位	.justify-content-around
平均分配元素間的空位	.justify-content-between
備註：	
<ul style="list-style-type: none">• 可搭配寬度關鍵字使用：<ul style="list-style-type: none">• .justify-content-{寬度關鍵字}-{排列方式}	

建立Layout與版面切割

- Flexbox多行排列
- <https://getbootstrap.com/docs/4.3/utilities/flex/#align-content>

指定對象	容器
作用方式	控制子元素有多行時的排列方式，在子元素只有單行時無作用
<code>.align-content-start</code>	密集置頂，行與行間沒有間隔
<code>.align-content-end</code>	密集置底，行與行間沒有間隔
<code>.align-content-center</code>	密集置中，行與行間沒有間隔
<code>.align-content-between</code>	行與行間均分間距分散置中，第一行靠容器頂、最末行靠容器底
<code>.align-content-around</code>	行與行及上下容器邊界間均分間距分散置中，內容上下都有空隙
<code>.align-content-stretch</code>	每行自動延展至填滿容器的高度

備註：

- 可搭配寬度關鍵字使用：
 - `.align-content-{寬度關鍵字}-{排列方式}`

建立Layout與版面切割

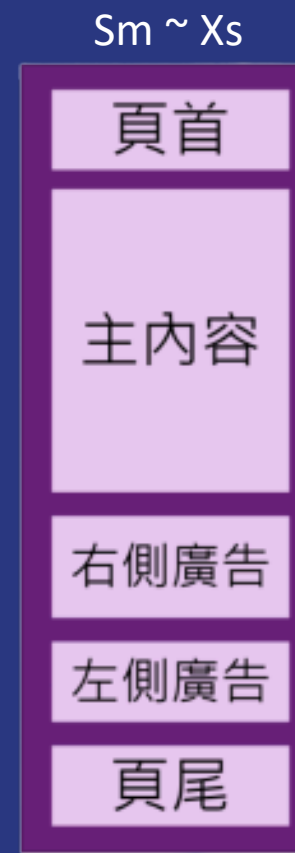
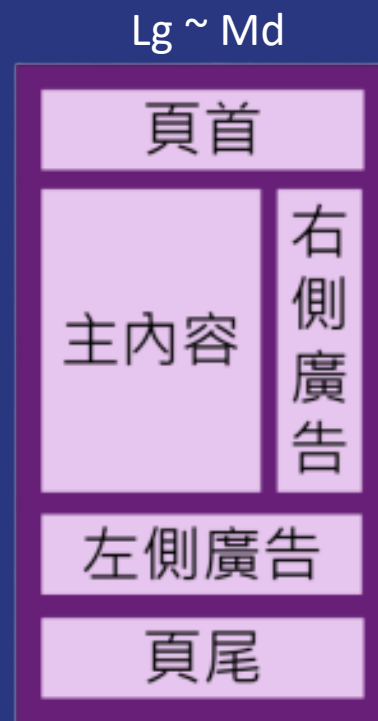
- 依Grid格線右推
- <https://getbootstrap.com/docs/4.3/layout/grid/#offset-classes>
- 可使用.offset-{格線數}將區塊向右側推
- 格線數範圍由1~11
- 指定於子元素
- 小心爆版
- 可搭配寬度關鍵字使用：.offset-{寬度關鍵字}-{格線數}
- 搭配寬度關鍵字時格線數範圍為0~11，可用於重設offset的值

建立Layout與版面切割

- 巢狀Flexbox
- 在Flexbox的子元素中可以新增.row容器製作第二層的Flexbox
- Grid會依新寬度重新切分，與上層不同

建立Layout與版面切割-練習

- 打開 [layout-2.html](#)，使用前面學過的方法先處理主要的區塊劃分
- 注意區塊順序



建立Layout與版面切割

- Flexbox 延展與縮減
- 指定子元素是否可延展填滿空間或在必要時縮減尺寸

	grow-延展	shrink-縮減
0-保持固有尺寸	.flex-grow-0	.flex-shrink-0
1-填滿空間或縮減尺寸	.flex-grow-1 (延展)	.flex-shrink-1 (縮減)

- 可搭配寬度關鍵字使用：.flex-{寬度關鍵字}-{延展或縮減}-0或1

建立Layout與版面切割

- Flexbox的子元素換行
- 指定於容器，控制子元素在容器寬度不足時的換行行為
- `.flex-nowrap`即使容器寬度不足也不換行
- `.flex-wrap`容器寬度不足時自動換行
- `.flex-wrap-reverse`容器寬度不足時換行，但先填滿靠下的行數

建立Layout與版面切割

- Bootstrap 中提供數個預設方向的 `margin` 和 `padding` 值可供選擇
- Class 寫法為 {類型}{方向}-{尺寸}
- 類型
- M 表示 `margin`
- p 表示 `padding`

建立Layout與版面切割

- `class` 寫法為{類型}{方向}-{尺寸}
- 尺寸
- 0 : margin 或 padding 的值為 0
- 1 : margin 或 padding 的值為 0.25rem
- 2 : margin 或 padding 的值為 0.5rem
- 3 : margin 或 padding 的值為 1rem
- 4 : margin 或 padding 的值為 1.5rem
- 5 : margin 或 padding 的值為 3rem
- auto : 此值只適用於margin，設margin為auto

建立Layout與版面切割

- `class` 寫法為{類型}{方向}-{尺寸}
- 方向
- `t`：上方
- `r`：右方
- `b`：下方
- `l`：左方
- `x`：左右
- `y`：上下
- 不寫：上下左右全部

建立Layout與版面切割

- 尺寸、方向

尺寸	0	1	2	3	4	5	auto
說明	0	0.25rem	0.5rem	1rem	1.5rem	3rem	設margin: auto; 此值只適用margin
方向	t	r	b	l	x	y	不寫
說明	上方	右方	下方	左方	左右	上下	上下左右 (全部)

- 可搭配寬度關鍵字使用：{類型}{方向}-{寬度關鍵字}-{尺寸}

- 範例：

- .mr-0
- .p-5
- .my-auto

內容樣式

- Color
- <https://getbootstrap.com/docs/4.3/utilities/colors/>
- `.text-{\顏色關鍵字}` 文字、連結顏色
- `.bg-{\顏色關鍵字}` 區塊背景顏色
 - 可使用透明背景

內容樣式

- Images
- <https://getbootstrap.com/docs/4.3/content/images/>
- 在 `` 標記使用 `.img-fluid` 可使圖片不超過100%寬度且維持原比例
- 若在 `` 標記使用 `.img-thumbnail` 則可製作有圓角邊框的圖片
- 使用 `.rounded` 可製作有圓角的圖片

內容樣式

- Figures
- <https://getbootstrap.com/docs/4.3/content/figures/>
- `<figure>` 可以提供加上圖說的圖片結構

通用樣式

- Sizing

- <https://getbootstrap.com/docs/4.1/utilities/sizing/>

- Bootstrap 提供少量尺寸預設值，大多還是要自己額外設定 `css`

- 預設值

寬度預設值	class名稱	.w-25	.w-50	.w-75	.w-100	.w-auto
	佔容器寬度	25%	50%	75%	100%	auto
高度預設值	class名稱	.h-25	.h-50	.h-75	.h-100	.h-auto
	佔容器高度	25%	50%	75%	100%	auto

- `.mw-100` 最大寬度不超過容器

- `.mh-100` 最大高度不超過容器

通用樣式

- Float
- <https://getbootstrap.com/docs/4.3/utilities/float/>
- 用來指定元素浮動的方向
- 浮動的元素會已指定的方向依序如流水般的互相連接排列下去
- 浮動的元素會影響後方的元素，即使該元素並未設定浮動
- 若一個元素內的所有子元素都設定浮動時，該元素會無法取得子元素的高度

通用樣式

- Float
- .float-left 向左浮動
- .float-right 向右浮動
- .float-none 不浮動
- 可搭配關鍵字使用：`float-{寬度關鍵字}-{浮動值}`

通用樣式

- Clearfix
- <https://getbootstrap.com/docs/4.3/utilities/clearfix/>
- 解決子元素全數浮動時容器無法取得子元素高度的問題
- CSS中無此屬性

通用樣式

- **Position**
- 可以使用下列屬性來推移元素（必須另外寫css）

	相對定位	絕對定位
top	從上往下推的距離	距離基準容器上緣的距離
bottom	從下往上推的距離	距離基準容器下緣的距離
left	從左往右推的距離	距離基準容器左緣的距離
right	從右往左推的距離	距離基準容器右緣的距離

- 以上預設值都是auto;

通用樣式

- Position
- 依視窗置頂與置底
- .fixed-top
- .fixed-bottom
- 捲動時固定於指定位置
- .sticky-top