

CHAPTER 06

STYLE與CSS

HORAZON

互動媒體設計

什麼是 CSS?

CASCADING STYLE SHEETS (階層式樣式表)

負責網頁的外觀、排版與視覺效果 (皮膚與衣服)

HTML 是骨架 (內容)，CSS 是樣式 (裝飾)。

沒有 CSS 的網頁，就像沒有裝潢的毛胚屋。

CSS 的三種寫法

1. 行內樣式 (Inline)：直接寫在 HTML 標籤上 (不推薦)。

```
<h1 style="color: red;">標題</h1>
```

2. 內部樣式 (Internal)：寫在 `<head>` 的 `<style>` 標籤內。

3. 外部樣式 (External)：寫在獨立的 `.css` 檔案 (最推薦)。

```
<link rel="stylesheet" href="style.css">
```

CSS 語法結構

```
選擇器 {  
    屬性: 設定值;  
    屬性: 設定值;  
}
```

- **選擇器 (Selector)**：選到誰？(例如 `h1` , `p` , `.box`)
- **屬性 (Property)**：改什麼？(例如 `color` , `font-size`)
- **設定值 (Value)**：改多少？(例如 `red` , `16px`)

```
h1 {  
    color: blue;  
    font-size: 24px;  
}
```

選擇器 (SELECTORS)

決定 CSS 要套用在哪些元素上。

選擇器	符號	範例	說明
標籤選擇器	無	p { ... }	選取所有 <p> 標籤
類別選擇器	.	.title { ... }	選取 class="title" 的元素 (最常用!)
ID 選擇器	#	#header { ... }	選取 id="header" 的元素 (唯一)
全域選擇器	*	* { ... }	選取網頁上所有元素

常用文字屬性

- `color` : 文字顏色 (`red` , `#FF0000` , `rgb(255, 0, 0)`).
- `font-size` : 文字大小 (`16px` , `1.5rem`).
- `font-weight` : 文字粗細 (`bold` , `700`).
- `text-align` : 文字對齊 (`center` , `left` , `right`).
- `line-height` : 行高 (閱讀舒適度關鍵).

```
p {  
  color: #333;  
  font-size: 18px;  
  line-height: 1.6;  
}
```

盒子模型 (BOX MODEL)

每個 HTML 元素都是一個盒子，由內而外包含：

1. **Content** (內容)：文字或圖片本身。
2. **Padding** (內距)：內容與邊框之間的距離 (留白)。
3. **Border** (邊框)：盒子的框線。
4. **Margin** (外距)：盒子與盒子之間的距離。

盒子模型示意圖

```
.box {  
    width: 300px;  
    padding: 20px; /* 內距 (藍色背景部分變大) */  
    border: 5px solid blue; /* 邊框 */  
    margin: 20px; /* 外距 (推開別人) */  
}
```



排版神器：FLEXBOX

現代網頁排版最常用的工具 (`display: flex`)。

它可以輕鬆讓元素並排、置中或平均分配。

```
.container {  
  display: flex; /* 開啟 Flex 模式 */  
  justify-content: center; /* 水平置中 */  
  align-items: center; /* 垂直置中 */  
}
```

常用屬性：

- `justify-content` : `center` (置中), `space-between` (兩側推開), `flex-start` (靠左)。
- `align-items` : `center` (垂直置中), `flex-end` (靠底)。

響應式設計 (RWD)

讓網頁在手機、平板、電腦上都好看。

主要使用 Media Queries (@media)。

```
/* 電腦版樣式 */
.box {
    width: 50%;
}

/* 當螢幕寬度小於 600px (手機) */
@media (max-width: 600px) {
    .box {
        width: 100%; /* 手機版變全寬 */
    }
}
```

實作練習：個人卡片

製作一張個人簡介卡片，包含：

1. 一個外框 (`div`)：設定邊框、陰影、圓角。
2. 圓形大頭貼 (`img`)：使用 `border-radius: 50%` 。
3. 名字與簡介 (`h2` , `p`)：設定置中與適當間距。

```
<div class="card">
  
  <h2>Horazon</h2>
  <p>前端工程師</p>
</div>
```

實作練習：CSS 參考

```
.card {  
    width: 300px;  
    padding: 20px;  
    border: 1px solid #ccc;  
    border-radius: 10px; /* 圓角 */  
    text-align: center; /* 文字置中 */  
    box-shadow: 0 4px 8px rgba(0,0,0,0.1); /* 陰影 */  
    margin: 50px auto; /* 整個卡片置中 */  
}  
  
.avatar {  
    width: 100px;  
    height: 100px;  
    border-radius: 50%; /* 變圓形 */  
    object-fit: cover;  
}
```

Responsive Web Design

隨著手機上網的人越來越多，網頁必須能自動適應各種螢幕尺寸。

核心概念：

1. 流動網格 (Fluid Grids)：寬度使用 `%` 而不是 `px`，讓區塊隨螢幕伸縮。
2. 彈性圖片 (Flexible Images)：`max-width: 100%`，防止圖片撐破版面。
3. 媒體查詢 (Media Queries)：針對不同尺寸設定不同 CSS。

MEDIA QUERY RWD 範例

讓我們練習一個最簡單的 RWD 效果：

「當螢幕變窄時，背景變色」

```
/* 1. 預設（電腦版大螢幕） */
.box {
    background-color: lightblue;
}

/* 2. 當螢幕寬度小於 768px（手機/平板） */
@media (max-width: 768px) {
    .box {
        background-color: pink; /* 變成粉紅色 */
    }
}
```

實作與測試：

開發人員工具 (F12) -> 切換手機模式 -> 觀察顏色變化！

常見的 RWD 製作方法

現代開發有幾種主流做法：

1. CSS FLEXBOX (彈性盒子)

最推薦！直接用 `flex-wrap: wrap` 讓元素在小螢幕自動換行。

2. CSS GRID (網格系統)

適合大型排版，可以精準控制「幾欄幾列」。

3. CSS 框架 (FRAMEWORKS)

站在巨人的肩膀上！

- Bootstrap：最老牌，用 class (`col-6`, `col-md-4`) 來排版。
- Tailwind CSS：現代主流，寫法極簡 (`w-1/2`, `md:w-1/3`)。

實戰：用 FLEX 做 RWD

不用寫 Media Query 也能做 RWD？

```
.container {  
    display: flex;  
    flex-wrap: wrap; /* 空間不夠時自動換行 */  
}  
  
.box {  
    flex: 1 1 300px; /* 彈性縮放，最小寬度 300px */  
}
```

- **大螢幕**：空間夠，三個箱子並排 (300px + 300px + 300px)。
- **手機**：空間不夠 (< 600px)，箱子自動掉下來變垂直排列。

這就是 無痛 RWD！

1. CSS 負責網頁的「樣式」，讓 HTML 變漂亮。
2. 推薦使用**外部樣式表** (External Style)。
3. 善用 Class (.) 來管理樣式。
4. 解 **Box Model** (Padding/Margin/Border) 是排版基礎。
5. **Flexbox** 是最強大的排版工具。

重點回顧

1. 選擇器：`.class` 最常用，`#id` 只有一個。
2. 顏色：`color` (字), `background-color` (底)。
3. 距離：`padding` (內), `margin` (外)。
4. 排版：`display: flex` 讓元素乖乖排好。
5. RWD：`@media` 讓手機板也好用。