

Web應用程式開發 Web Application Development

層疊樣式表 (Cascading Style Sheets, CSS)

授課教師：林器弘

1

Outline

- CSS 是什麼？
- 開始使用 CSS
- 為我們的文件加入CSS
- 添加分類（class）
- CSS 實際上是怎麼運作的？

2

2

CSS 是什麼？

- CSS（階層式樣式表）可以讓您建立出好看的網頁，但是它背後是怎麼運作的？在這個主題裡，藉由簡單的語法範例來說明 CSS 是什麼，以及含蓋這個語言的一些關鍵項目。
- 在 HTML 入門單元中，我們含蓋了什麼是 HTML 以及它是如何被用來標記文件。這些文件能夠被瀏覽器讀取，標題的文字會看起來比一般段落更大，段落之間會換行並帶有間隔。
- 連結會帶有顏色及底線，讓它與其它一般的文字有區別。您所看到的這些是瀏覽器的預設樣式，用來確保當作者沒有指定任何樣式的狀況下，仍有一些非常基本的樣式被套用上，好讓內容基本上能夠被閱讀（如下圖所示）。

3

3

CSS 是什麼？

Browser defaults

The browser will style HTML documents using an internal stylesheet. This ensures that headings are larger than normal text, links are highlighted and structures such as lists and tables are understandable.

Paragraphs are spaced out. List items get a bullet or number, Links are highlighted and underlined.

- Item One
- Item Two

A level 2 heading

You can change all of this with CSS.

4

4

CSS 是什麼？

- 您能使用 CSS 對 HTML 元件的樣子作更多控制，將這些標記以任何您喜歡的設計作調整。
- 如同我們前面所提到的，CSS 是一種用來指定文件該用什麼方式呈現的語言，可以定義它們的樣式、布局...等。
- 文件通常指的是使用標記語言的文字檔案，HTML 是最常見的，但是您也可能遇到其它例如 SVG 或 XML 的標記語言。

5

5

CSS 是作什麼用的？

- 所謂的呈現文件，指的是將文件轉換為你的讀者可用的形式。像是 Firefox、Chrome 或 Edge 這類的瀏覽器，是設計來將文件視覺化，再呈現電腦螢幕、投影機上或是由列表機列印出來。
- CSS 可以用在很基本文字樣式上頭，像是改變標題和連結的顏色及尺寸。
- 可以用在建立布局，像是將原本單欄的文字內容加入布局，劃分出主要的內容以及包含相關資訊的側邊欄。它甚至可以用在建立動畫效果。點進上面的連結，看看相關的例子。

6

6

CSS 語法

- CSS 是一種基於規則的語言，您對網頁裡特定或一群元素指定一系列的規則。舉例來說：「我要讓頁面裡的主標題，以紅色且大號的字體呈現」。
- 下面這段語法是為了實現上面的需求，用簡單 CSS 規則示範：

```
h1 {  
    color: red;  
}
```

7

7

CSS 語法

- 樣式規則以一個選擇器開始。它選擇了您預計改變樣式的 HTML 元素。在這個例子中，我們要調整的是第一級的標題元素（<h1>）。跟著一組大括號 { }，裡面是一到多個聲明，它的形式是一對一對屬性名稱和屬性內容的組合。
- 每一對聲明會將我們選中元素的屬性，付予我們所想要設定的內容（或數值）。
- 在冒號 (:) 前面的是屬性的名稱，後面的是屬性內容（值）。
- CSS 的屬性依照其類型可以使用的值而有所不同。在例子中，有個 color 屬性，它可以設定各種顏色值。

8

8

CSS 語法

- 一個 CSS 樣式表包含了許多這樣子的規則，一個接著一個。

```
h1 {  
    color: red;  
}  
  
p {  
    color: black;  
}
```

- 你將會發覺有些值很容易學會，而另一些則需要查資料確認。

9

9

CSS 規範

- 網路標準技術（HTML、CSS、JavaScript...等）都被定義在稱為定義（specifications 或簡稱 specs）文件中，由像是 W3C、WHATWG、ECMA 或 Khronos 之類的標準組織所發布，並且很精確地定義這些技術的行為方式。
- CSS 並沒有什麼不同，它由 W3C 一個被稱為 CSS SIG 的團體所發展。這個團體是由對 CSS 感興趣的瀏覽器供應商和其它公司的代表所組成。

10

10

CSS 規範

- 新的 CSS 特性被 CSS SIG 所發展、定義。有時候是因為特定瀏覽器對某個功能有興趣，而有時候是因為網站設計師與開發人員的要求，還有一些時候是工作組本身定義的需求。CSS 正不斷發展，新的可用特性正在出現。然而，每個人很努力達到的 CSS 重要方針，是不要往會破壞舊網站的方向進行改變。一個在 2000 年建立的網站，使用了當時能用的 CSS 特性，應該到今天仍能夠在瀏覽器上使用。
- 作為一個 CSS 新手，你會發現 CSS 的規範不勝枚舉，它們是用來給開發用戶端程式的開發者實作功能所使用，而不是讓網站開發人員閱讀來了解 CSS。

11

11

CSS 規範

- 被定義好的 CSS 特性，只有被一個或更多瀏覽器實作出來之後，才會在我們開發網頁上面有所幫助。
- 這意味著已經編寫了程式，可以將 CSS 檔案裡的設定轉換為輸出在畫面上的結果。我們將在 CSS 工作原理中詳細介紹這個過程
- 一個（新）特性被所有瀏覽器同時實作出來是不常見的，通常會缺了幾個，CSS 某些部分您可以在某些瀏覽器上使用，然而在其它瀏覽器則沒有作用。基於這個原因，確認特性被實作的狀況是有用的。

12

12

開始使用 CSS

- 在這個主題中，我們將 CSS 套用到一個簡單的 HTML 文件上，在過程中學習這個語言一些實際的東西。
- 通常我們會將CSS的檔案放置於static 資料夾中

```
Website/
├── css.html
├── static/
│   └── styles.css
```

13

13

開始使用 CSS

- 由某個 HTML 開始
 - 我們先產生一個 HTML 文件。在您電腦上的目錄中，開啟一個 CSS1.html 為檔名儲存。
 - 在Body中加入這段程式

```
<h1>I am a level one heading</h1>
```

```
<p>第一段內容可參考 <a href="result.html">link</a>. 與<span>span 元素</span> </p>
```

```
<p>這是第二段 <em> 強調元素 </em> 表現</p>
```

```
<ul>
```

```
<li>Item one</li>
```

```
<li>Item <em> two </em></li>
```

```
</ul>
```

```
<ol>
```

```
<li>technologists</li>
```

```
<li>thinkers</li>
```

```
</ol>
```

```
<!DOCTYPE html>
```

```
<html>
```

```
<head>
```

```
<meta charset="utf-8" />
```

```
<title>CSS測試</title>
```

```
</head>
```

```
<body>
```

```
</body>
```

```
</html>
```

14

14

為我們的網頁加入CSS

- 首先，第一個步驟是告訴 HTML 文件我們有些 CSS 規則要加入。你通常有三種不同的方式來將 CSS 套用到 HTML 文件上，然而，現在我們先將焦點放在其中最常見並且實用的方式：在您文件標頭的位置將 CSS 連結進去。
- 建立一個檔案，存在與您 HTML 文件同一個目錄之中或放置static資料夾中，並命名為 styles.css。副檔名 .css 會讓它被當作一個 CSS 檔案。
- 為了連結 styles.css 到 CSS1.html 中，在 HTML 文件的 <head> 中，加入下面這行：

```
<link rel="stylesheet" href="/static/styles.css">
```

15

15

為我們的網頁加入CSS

- 在 HTML 文件的 <head> 中，加入下面這行：

```
<link rel="stylesheet" href="/static/styles.css">
```

- <link> 元素使用 rel 屬性告訴瀏覽器我們有一個樣式表，接著 href 屬性指向到這個樣式表的位置。您可以在 styles.css 裡加入規則，來測試它是否能夠運作。
- 使用程式碼編輯器，在您的 CSS 檔案中加入下面這段規則：

```
h1 {  
    color: red;  
}
```

16

16

為我們的網頁加入CSS

- 將您的 HTML 與 CSS 檔案儲存後，重新整理瀏覽器裡的頁面。文件中最上頭的第一級標題現在應該會是紅色。如果是，恭喜！您成功地套到了一些 CSS 設定到 HTML 中了。如果結果不是如此，仔細檢查您的每段文字並確認都輸入正確。
- 你可以在你電腦裡的 styles.css 上繼續，或是使用我們下的互動編輯器來繼續本教學指引。互動編輯器的行為，就像是這個 CSS 已經連結到 HTML 中，就如同我們在前面剛剛完成的一樣。

17

17

為我們的網頁加入CSS

- 裝飾 HTML 元素
- 藉由的將我們的標題設定為紅色，已經證明了我們可以指定並修飾 HTML 元素。我們透過一個元素選擇器來達成，它會指定名稱相符的 HTML 元素。想要指向所有段落元素的話，您會將選擇器設為 p。
- 要把所有段落文字改成綠色，您可以這麼作：

```
p {  
  color: green;  
}
```

18

18

為我們的網頁加入CSS

- 如果我想要讓所有清單項目都變藍色，我的規則會寫成：

```
ul{  
  color: blue;  
}
```

19

19

為我們的網頁加入CSS

- 藉由逗號的分隔，您可以一次設定多個選擇器。如果我想要讓所有段落與順序清單項目都變綠色，我的規則會寫成：

```
p, ol {  
  color: green;  
}
```

20

20

改變元素的預設樣式

- 看到瀏覽器藉由加上一些預設的樣式讓 HTML 變得可閱讀。
- 標題字體較大且較粗，而清單的項目前添加上符號。因為瀏覽器裡有包含預設樣式的內部樣式表，它預設會套用到所有的頁面。少了它們的話，所有的內容會全擠在一起，而我們得從頭一個一個設定樣式。所有現代的瀏覽器，預設以很相似的方式來呈現 HTML 內容。
- 然而，您通常會想要一些不同於瀏覽器預設的選擇。可以通過選擇想改變的 HTML 元素，使用 CSS 規則來改變它的外觀來達成。拿我們的 `` 為例，它是一個無序清單。如果我決定不要顯示項目前的清單符號，可以如此移除它們：

```
li {  
    list-style-type: none;  
}
```

21

21

改變元素的預設樣式- Syntax

- 在 `list-style-type` 屬性中能夠設定許多內容
- 除了移除項目符號之外，也可以改變項目符號的樣式。
- `/* Keyword value */`
 - `list-style-type: none;`
- `/* Global values */`
 - `list-style-type: inherit;`
 - `list-style-type: initial;`
 - `list-style-type: revert;`
 - `list-style-type: revert-layer;`
 - `list-style-type: unset;`

22

22

改變元素的預設樣式 - Syntax

- `/* Partial list of types */`
 - `list-style-type: disc;`
 - `list-style-type: circle;`
 - `list-style-type: square;`
 - `list-style-type: decimal;`
 - `list-style-type: georgian;`
 - `list-style-type: trad-chinese-informal;`
 - `list-style-type: kannada;`
- `/* <string> value */`
 - `list-style-type: "-";`

23

23

添加分類（class）

- 到目前為止，我們依元素的名稱來設定過樣式。
- 當您希望同類型的元素在您的文件中都以相同的方式呈現，這種作法就行得通。
- 但絕大多數的時候並非如此，所以您將需要一個能選擇某部分的元素、以避免更動到其它部分樣式的方法。
- 最常見的作法，是在您要改變的 HTML 元素上設定 class（分類）並藉由它來定義要修改的範圍。

24

24

添加分類（class）

- 在您的 HTML 文件中，加上一個 class 屬性 到第二個清單項目上。您的清單會變得如同底下這樣：

```
<ol>
  <li class="special"></li>>technologists</li>
  <li>thinkers</li>
</ol>
<ul>
  <li>Item one</li>
  <li>Item <em class="special">two</em></li>
</ul>
```

- 您可以使用英文句點 . 作開頭，在您的 CSS 中指定 special 分類來建立選擇器。加上底下這些到您的 CSS 檔案中：

```
.special {
  color: orange;
  font-weight: bold;
}
```

25

25

添加分類（class）

- 您可以將 special 分類套用到任何您希望與這個清單項目相同方式呈現的元素。
- 舉例來說，您可能想要在段落中的 元素一樣以橘色粗體呈現。
- 嘗試添加 class 屬性並設定為 special 到它上頭，然後重新載入頁面看看發生了什麼。

26

26

添加分類（class）

- 有些時候，您會看到選擇器中，將元素名稱與分類合在一起：

```
li.special {  
  color: orange;  
  font-weight: bold;  
}
```

- 這個語法表示「選擇任何帶有 special 分類屬性的 li 元素」。如果您改用了這個選擇器，樣式將不會被套用到具有一樣分類的 `` 及其它元素上。

27

27

添加分類（class）

- 要維持原本的呈現結果，您應該再增加選擇器把元素包含進規則當中：

```
li.special,  
em.special {  
  color: orange;  
  font-weight: bold;  
}
```

- 您可以想像，有些分類的設定可能會被套用在多種元素上，而且您不希望當有新的元素要套用相同樣式的時候，得回頭修改選擇器。
- 因此選擇器應該只指定分類（class）而不指定元素名稱，除非要為了這個分類下的特定元素設定樣式，並確保不會套用到其它元素上。

28

28

添加分類（class）

- 在ol 外面增加一個span 並設定class 為special

```
<span class="special">  
  <ol>  
    <li>first</li>  
    <li>second</li>  
  </ol>  
</span>
```

29

29

依所在位置指定樣式

- 有些時候，您會想要一些在不同位置的東西看起來有所差別。
- 有許多選擇器可以作到，但現在我們只介紹其中兩種。在我們的文件中有兩個 元素，一個在段落中另一個在清單項目裡。要只選擇到被包含在 元素裡的 元素，我可以使用被稱後裔連結器（descendant combinator）的語法，它是用空格（一到多個）連結兩個選擇器。

30

30

依所在位置指定樣式

- 在您的樣式表加入以下規則。

```
li em {  
    color: rebeccapurple;  
}
```

- 這個選擇器將會指向任何在 `` 內部的 `` 元素。
- 所以在範例裡中，您應該會發現在第二個清單項目裡的 `` 變成紫色，但是在段落裡的則沒有改變。

31

31

依所在位置指定樣式

- 您可能會想嘗試對直接跟在標題後面的那個段落設定樣式。
- 要作到這件事就使用一個 `+`（相鄰連結器 adjacent sibling combinator）放在選擇器的中間。
- 試著加入面這個規則

```
h1 + p {  
    font-size: 200%;  
}
```

32

32

練習一

- 試試看使用list-style-type把項目符號改為”喬治亞語”。
- 試著將h1算中的段落<p>當中的span 元素變成紅色
- 但其他span元素還是維持原來顏色

33

33

CSS 實際上是怎麼運作的？(1/2)

- 每當瀏覽器要顯示一份文件時，它得先為文件內容穿上樣式，這會歷經許多程序，我們已經列在下方了。記得喔，這只是非常簡化的版本，不同的瀏覽器會有自己的做法，不過原則上就是這樣。
 1. 瀏覽器載入 HTML (比如從網路上接收(receive))。
 2. 它將 HTML 轉換成 DOM (Document Object Model，文件物件模型)，這東西是文件在電腦記憶體中的表示形式，詳情我們下個小節再說。
 3. 瀏覽器蒐集所有 HTML 文件連到的資源，像是嵌入網頁的圖片和影片等等，當然，裡面也包含 CSS！JavaScript 也是其中的一種資源，在此步驟的稍後就會處理，但我們先不要把事情弄得這麼複雜，這邊暫且不講。

34

34

CSS 實際上是怎麼運作的？(2/2)

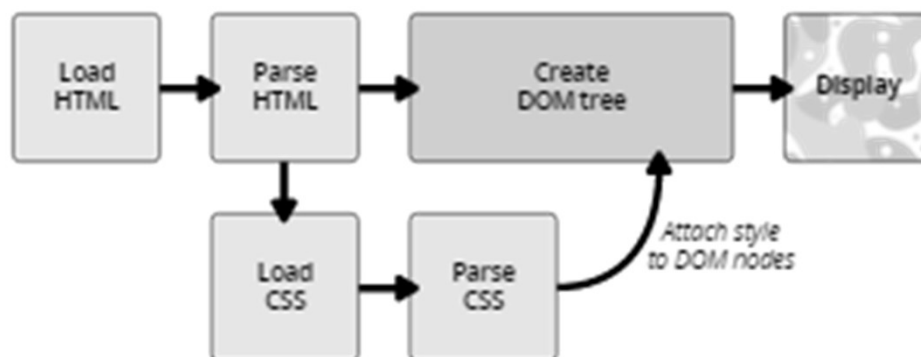
4. 瀏覽器解析 (parse) CSS，先按照選擇器的類型(如元素、類別、ID 等等)，將規則放入相對應的「桶子(buckets)」裡。接著再依找到的選擇器，推算哪些規則應該要套用在哪些 DOM 節點上，並將樣式附著上去，最後產生的東西叫做轉譯樹(render tree)。
5. 當規則都套用完畢後，開始按照網頁結構布局(layout)轉譯樹。
6. 網頁被呈現在螢幕上，這個步驟稱為繪製(painting)。

35

35

CSS 實際上是怎麼運作的？

- 下面是此流程的示意圖。



36

36

CSS 實際上是怎麼運作的？

- 關於 DOM
- DOM 有著一個樹狀結構，每個標記語言中的元素、屬性，以及文字片段都會是這個樹狀結構裡的節點。
- 每個節點與其他節點間的關係都有定義：若節點有子節點(child)，則自己是他們的父節點(parent)；若子節點為複數，則這些子節點稱彼此為兄弟/姊妹節點(sibling)。
- 了解 DOM 對於設計、除錯以及維護 CSS 有相當大的助益，因為 DOM 正是 CSS 與文件內容的交會之處。當你要利用瀏覽器的開發者工具(DevTools)來查看元素套用的規則時，你就會見到它們。

37

37

CSS 實際上是怎麼運作的？

- 直接看個簡單的例子，來瞭解 HTML 片段是如何轉換成 DOM
- 以下列 HTML 原始碼為例：

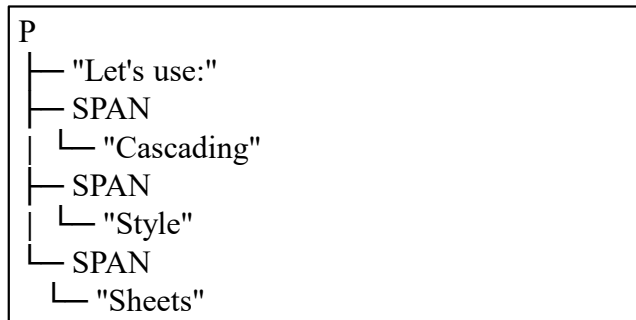
```
<p>
  Let's use:
  <span>Cascading</span>
  <span>Style</span>
  <span>Sheets</span>
</p>
```

38

38

CSS 實際上是怎麼運作的？

- 在 DOM 中，<p> 元素對應到的節點是一個父節點，它的子節點有一個純文字節點以及三個 元素節點，而 SPAN 節點也是有著純文字子節點的父節點：



39

39

CSS 實際上是怎麼運作的？

- 這就是瀏覽器如何解析上段的 HTML 片段的 — 它轉譯了以上的 DOM 樹，並產生了以下的輸出：

Let's use: Cascading Style Sheets

40

40

CSS 實際上是怎麼運作的？

- 將 CSS 套用至 DOM
- 讓我們在上例中加入一些 CSS 來增添樣式。同樣地，HTML 如下：

```
<p>
  Let's use:
  <span>Cascading</span>
  <span>Style</span>
  <span>Sheets</span>
</p>
```

- 假設我們把以下 CSS 套用上去：

```
span {
  border: 1px solid black;
  background-color: lime;
}
```

41

41

CSS 實際上是怎麼運作的？

- 瀏覽器會先解析 HTML 並產生 DOM 樹，然後再解析 CSS。因為這個 CSS 中只有使用 span 選擇器，所以瀏覽器可以很快地完成分類！接著它會將這個規則套用到每一個 上，並在螢幕上繪製出最終的畫面。
- 現在輸出變成這樣：

Let's use: Cascading Style Sheets

42

42

瀏覽器遇到不認識的 CSS 時會發生什麼事？

- 瀏覽器並不會一次實作全部的新 CSS。此外，很多人都不是使用最新版的瀏覽器。要知道 CSS 是與時俱進的，會超出瀏覽器可辨認的範圍是很正常的事，所以啦，你可能會很好奇，當瀏覽器遇到它看不懂的 CSS 選擇器或宣告時會發生什麼事呢？
- 答案就是裝作沒看到，繼續往下解析其它的 CSS！
- 如果瀏覽器在解析規則時，遇到它不認識的屬性或值，它會忽略它，並繼續解析下一個宣告。因此當它這麼做的時候，如果不是你拼錯字了，那就是那個屬性或值太新奇了，所以你的瀏覽器還沒有支援它。
- 同樣地，如果瀏覽器遇到一個它不認識的選擇器時，它會忽略整條規則，並繼續解析其他規則。

43

43

瀏覽器遇到不認識的 CSS 時會發生什麼事？

- 下面的例子使用英式英語來拼寫 color (也就是 colour)，進而導致該屬性失效，因為現在瀏覽器看不懂它了。也因此下面的段落無法以藍字顯示，不過其他的 CSS 還是成功地套用上去了，只有無效的會被忽略掉。

```
<p> I want this text to be large, bold and blue.</p>
```

```
p {  
  font-weight: bold;  
  colour: blue; /* incorrect spelling of the color  
property */  
  font-size: 200%;  
}
```

44

44

瀏覽器遇到不認識的 CSS 時會發生什麼事？

- 這樣做有個很大的好處，就是你可以放心地利用新 CSS 做出很炫的效果，而不用擔心瀏覽器不支援時會出錯——反正差別只在於那個新特性有或沒有而已。
- 再加上 CSS 層疊 (cascade) 的天性，只要你提供兩條具有相同具體程度 (specificity) 的規則，就能讓不支援的瀏覽器套用另一條規則。

45

45

文件主體元素

- 代表 HTML 文件的內容。
- 一個文件中只能有一個 `<body>` 元素，當中在 CSS 中也可以進行主體文件的設定，如下所示。
 - background-attachment
 - background-clip
 - background-color
 - background-image
 - background-origin
 - background-position
 - background-repeat
 - background-size

styles.css

```
body {  
    font-family: Arial;  
    background-color: gold;  
}
```

46

46

自定義容器

- 可以透過以下 HTML 範例，將某一個範圍進行框選，並定義成某一個具有圖像、標題和一些文字的卡片組合。

styles.css

```
.container {  
  max-width: 500px; /* 最大寬度 */  
  margin: auto; /* 水平置中 */  
  background: #fff; /* 白色背景 */  
  border: 1px solid #ddd; /* 1px 灰色邊框 */  
}
```

CSS1.html

```
<div class="container">  
  <h2>Card title</h2>  
  <p>Card content</p>  
</div>
```

47

47

依狀態指定樣式

- 我們要看的最後一種類型，依照元素的狀態來設定樣式的能力。最直接的例子就是設定連結的樣式。當我們設定連結的樣式，我們的目標是 <a>（錨點 anchor）元素。它具有不同的狀態來表示其連結：還沒被訪問過、訪問過、滑鼠停留其上方、由鍵盤取得焦點或是正被點擊中。
- 您可以使用 CSS 來指定不同的狀態，下面的 CSS 設定連結（錨點元素）在未訪問過時為粉紅色，訪問過為綠色。

```
a:link {  
  color: pink;  
}  
a:visited {  
  color: green;  
}
```

48

48

依狀態指定樣式

- 您可以改變連結當滑鼠停留在上頭（或稱為懸停）時的樣子，假如要把連結的底線移除、變波浪或是改顏色，可以用接下來的規則作到：

```
a:hover {  
  text-decoration: none;  
}  
  
a:hover {  
  text-decoration: green wavy underline;  
}  
  
a:hover {  
  text-decoration-color: #21ff21;  
}
```

- 但是在實際運作的網站上，值得注意的是您要確保使用網站的人能分判哪些元素是連結。
- 留著底線會是一個讓人意識到段落中的某些文字以可點擊的重要線索，這是約定俗成的習慣，所以建議使用其他操作以增加注意。

49

49

組合選擇器和連結器

- 您也可以使用不同種類的選擇器和連結器，試著把下面的規則加到您的 CSS 中：

```
body h1 + p .special {  
  color: yellow;  
  background-color: black;  
  padding: 5px;  
}
```

- 這個規則會對任何在 <body> 內，跟在 <h1> 後 <p> 裡的 special 分類元素設定樣式！
- 在們提供的 HTML 文件中，樣式只會套用在 元素上。
- 如果這現在看起來有很複雜，別擔心，隨著開始撰寫更多 CSS，您很快就會掌握它了。

50

50

練習二

- 使用組合選擇器和連結器，將段落<p>底下的ul 中的special變更顏色為橘色，並將背景更改為黑色。
- 在body的元素底下，建立一個你學號的容器名稱，並設定此容器最大寬度250、紫色背景、水平置中。

51

51

Q&A

52

52