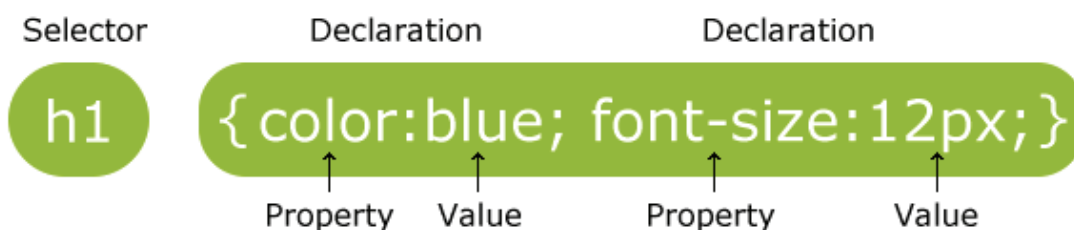


Cascading Style Sheets, CSS--串接樣式表

優點：

將 content 與 style 分開，由 html 負責 content，CSS 負責定義 Style；分開的好處是可用一套 CSS 定義在多組 Content 上，設計與維護均可簡化。

CSS 語法說明：



Selector 選擇子：

可為 Tag, Id, Class 或 Compound

1 · 當 Selector 為 Tag，代表重新定義此 Tag 之 CSS style，之後網頁中使用到此 Tag 均會套用此 CSS style

2 · 當 Selector 為 Id，代表定義此 Id 之 CSS style，Id 在此網頁中具有唯一性，不得有重複的 Id 出現，在 CSS 時定義 Id 開頭必須用“#”，後面為 Id 名稱，Id 名稱開頭不得為數字。

3 · 當 Selector 為 Class，代表定義此 Class 的“群組”之 CSS style，使用 Class 的時機是當有許多“群”的 Content 均需套用某 CSS 語法時。在 CSS 時定義 Class 開頭必須用“.”，後面為 Class 名稱，Class 名稱開頭不得為數字。

4 · 有時需要使用兩種以上的 Selector，就會定義成 Compound Selector(複合選擇子)，例如某 Tag 之下的某 Class 或某 id 下的某 Tag。

Declaration 宣告：

用{}括起，屬性設定結束一定要用“；”結束，屬性設定方式為“屬性：值”。常見的屬性群組如下：

1. CSS Background

.....

.....

.....

.....

2. CSS Text

.....

.....

.....

.....

3. CSS Fonts

.....

.....

.....

4. CSS Links

.....

.....

.....

5. CSS Lists

.....

.....

.....

.....

6. CSS Tables

CSS Box Model

在 CSS 中，所有的 Html 標籤元素都可以被想成 Boxes，所以 CSS Box Model 是在學習設計 CSS 時很重要的觀念，在一個 Box 中，最外圍的是”邊界”，margin，再來是邊框, border，文字 (Content)與邊框的”留白”稱之為 Padding。如下圖所示。



1. CSS Border

2. CSS Outline

3. CSS Margin

4. CSS Padding

CSS 的使用方式(如何套用至 Html 檔, CSS How to...)

有三種方式插入 CSS 樣式

1. External Style Sheet · 外部樣式檔：

將 CSS 語法獨立成一個樣式檔(副檔名.css)，在 Html 檔的<head></head>宣告連結至此樣式檔，語法如下：

```
<head>
<link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```

優點：獨立出樣式檔的最大優點就是可重覆使用至其他的 Html 檔，修改維護時也很方便，當大量使用到 CSS 語法時，建議使用此種方式。

2. Internal Style Sheet · 內部樣式檔：

將 CSS 語法以<style></style>包住，放在 Html 檔的<head></head>之中，使用此種方式 CSS 語法會內含在 Html 檔案內，語法如下：

```
<head>
<style>
hr {color:sienna;}
p {margin-left:20px;}
body {background-image:url("images/back40.gif");}
</style>
</head>
```

缺點：將 CSS 語法與 Html 檔綁成一個檔，其他 Html 檔無法套用，除非獨立出來，用於少量 CSS 設定時。

3. Inline Style · 內嵌語法：

將 CSS 語法變成 Html 標籤的屬性之一，用法如下：

```
<p style="color:sienna;margin-left:20px;">This is a paragraph.</p>
```

此三種方式的 CSS 套用順序如下：

意即網頁會先套用瀏覽器內定樣式，第二再套用外部樣式表，第三再套用內部樣式表，最後是套用內嵌樣式表。所以，若有某樣式重複定義，最後會呈現的是 Inline Style 所定義的樣式，前三種會被蓋掉。

1. Browser default
2. External Style Sheet
3. Internal Style Sheet
4. Inline Style

CSS Selector Reference

Selector	Example	Example description	CSS
<u>.class</u>	.intro	Selects all elements with class="intro"	1
<u>#id</u>	#firstname	Selects the element with id="firstname"	1
<u>*</u>	*	Selects all elements	2
<u>element</u>	p	Selects all <p> elements	1
<u>element,element</u>	div,p	Selects all <div> elements and all <p> elements	1
<u>element element</u>	div p	Selects all <p> elements inside <div> elements	1
<u>element>element</u>	div>p	Selects all <p> elements where the parent is a <div> element	2
<u>element+element</u>	div+p	Selects all <p> elements that are placed immediately after <div> elements	2
<u>[attribute]</u>	[target]	Selects all elements with a target attribute	2
<u>[attribute=value]</u>	[target=_blank]	Selects all elements with target="_blank"	2
<u>[attribute~=value]</u>	[title~=flower]	Selects all elements with a title attribute containing the word "flower"	2
<u>[attribute =value]</u>	[lang =en]	Selects all elements with a lang attribute value starting with "en"	2
<u>:link</u>	a:link	Selects all unvisited links	1
<u>:visited</u>	a:visited	Selects all visited links	1
<u>:active</u>	a:active	Selects the active link	1
<u>:hover</u>	a:hover	Selects links on mouse over	1
<u>:focus</u>	input:focus	Selects the input element which has focus	2
<u>:first-letter</u>	p:first-letter	Selects the first letter of every <p> element	1
<u>:first-line</u>	p:first-line	Selects the first line of every <p> element	1

Selector	Example	Example description	CSS
<u>:first-child</u>	p:first-child	Selects every <p> element that is the first child of its parent	2
<u>:before</u>	p:before	Insert content before the content of every <p> element	2
<u>:after</u>	p:after	Insert content after every <p> element	2
<u>:lang(<i>language</i>)</u>	p:lang(it)	Selects every <p> element with a lang attribute equal to "it" (Italian)	2
<u><i>element1</i>~<i>element2</i></u>	p~ul	Selects every element that are preceded by a <p> element	3
<u>[<i>attribute</i>^=<i>value</i>]</u>	a[src^="https"]	Selects every <a> element whose src attribute value begins with "https"	3
<u>[<i>attribute</i>\$=<i>value</i>]</u>	a[src\$=".pdf"]	Selects every <a> element whose src attribute value ends with ".pdf"	3
<u>[<i>attribute</i>*=<i>value</i>]</u>	a[src*="w3schools"]	Selects every <a> element whose src attribute value contains the substring "w3schools"	3
<u>:first-of-type</u>	p:first-of-type	Selects every <p> element that is the first <p> element of its parent	3
<u>:last-of-type</u>	p:last-of-type	Selects every <p> element that is the last <p> element of its parent	3
<u>:only-of-type</u>	p:only-of-type	Selects every <p> element that is the only <p> element of its parent	3
<u>:only-child</u>	p:only-child	Selects every <p> element that is the only child of its parent	3
<u>:nth-child(<i>n</i>)</u>	p:nth-child(2)	Selects every <p> element that is the second child of its parent	3
<u>:nth-last-child(<i>n</i>)</u>	p:nth-last-child(2)	Selects every <p> element that is the second child of its parent, counting from the last child	3
<u>:nth-of-type(<i>n</i>)</u>	p:nth-of-type(2)	Selects every <p> element that is the second <p> element of its parent	3

Selector	Example	Example description	CSS
<u>:nth-last-of-type(<i>n</i>)</u>	p:nth-last-of-type(2)	Selects every <p> element that is the second <p> element of its parent, counting from the last child	3
<u>:last-child</u>	p:last-child	Selects every <p> element that is the last child of its parent	3
<u>:root</u>	:root	Selects the document's root element	3
<u>:empty</u>	p:empty	Selects every <p> element that has no children (including text nodes)	3
<u>:target</u>	#news:target	Selects the current active #news element (clicked on a URL containing that anchor name)	3
<u>:enabled</u>	input:enabled	Selects every enabled <input> element	3
<u>:disabled</u>	input:disabled	Selects every disabled <input> element	3
<u>:checked</u>	input:checked	Selects every checked <input> element	3
<u>:not(<i>selector</i>)</u>	:not(p)	Selects every element that is not a <p> element	3
<u>::selection</u>	::selection	Selects the portion of an element that is selected by a user	3

Measurement Values

Unit	Description
%	percentage
in	inch
cm	centimeter
mm	millimeter
em	1em is equal to the current font size. 2em means 2 times the size of the current font. E.g., if an element is displayed with a font of 12 pt, then '2em' is 24 pt. The 'em' is a very useful unit in CSS, since it can adapt automatically to the font that the reader uses
ex	one ex is the x-height of a font (x-height is usually about half the font-size)
pt	point (1 pt is the same as 1/72 inch)
pc	pica (1 pc is the same as 12 points)
px	pixels (a dot on the computer screen)