

Table of Contents

CSS 概念	1
CSS 的優勢	1
程式碼比較.....	1
集體宣告.....	2
集體宣告的方法.....	2
分項宣告.....	3
分項宣告.....	3
外部呼叫.....	3
製做 CSS 檔	3
從網頁呼叫 CSS 檔	3
單獨使用.....	4
加入 STYLE 屬性	4
STYLE 屬性	4
自定類別.....	4
類別的定義.....	4
用 CSS 畫重點	5
自定類別的用法.....	5
文字屬性.....	5
文字屬性的設定.....	5
文字顏色 { COLOR }	6
文字類別 { FONT }	6
下載字型 { @FONT-FACE }	6
文字字型 { FONT-FAMILY }	6
文字大小 { FONT-SIZE }	6
文字樣式 { FONT-STYLE }	7
文字變化 { FONT-VARIANT }	7
文字粗細 { FONT-WEIGHT }	7
連結屬性.....	8
定義連結屬性.....	8
屬性分類.....	8
邊框屬性.....	8
屬性分類.....	8
段落屬性.....	10
定義段落屬性.....	10
屬性分類.....	10
滑鼠屬性.....	11

滑鼠屬性.....	11
屬性分類.....	11
背景屬性.....	11
定義背景屬性.....	11
屬性分類.....	11
網頁 CSS 性質參考資料表	13
字型.....	13
顏色與背景.....	13
文字.....	14
區塊.....	15
分類.....	17
定位.....	18

CSS 概念

- CSS 的全名是 [Cascading Style Sheet]，以字面上來解釋就是 [連(串)接樣式表]，是一種網頁的文字格式設定！

CSS 的優勢

- 通吃瀏覽器－CSS 樣式表，不管瀏覽者是使用何種瀏覽器來觀賞網頁皆能表現出網頁獨特的風格，不像 JavaScript 或 VBScript 語言會因為瀏覽器的不同而產生差異，不單是市場上使用率最高的 Microsoft Internet Explorer (IE) 與 Netscape Communicator 瀏覽器支援，甚至許多非主流的瀏覽器，如 Emacs-W3、Arena..... 等，都有支援解析的能力。
- 特立獨行的風格－一個人氣旺盛的網站之所以吸引人就是因為擁有獨一無二的風格！舉例來說：在 CSS 樣式表出現之前，幾乎所有網頁中的超鏈結文字都會加上底線 (Under Line)，而且並無特色變化，如果加上 CSS 語法那可就大大不同了！
- 網路環保－在大家全力推廣全民上網的同時，出現了一個嚴重的危機 [網路頻寬不足]，一方面是因為上網人口的急速增加造成頻寬不足，但最大的原因是網頁設計者在網頁設計過程中使用了大量的圖片，圖片檔的體積大，在傳輸的過程中消耗了大量的網路資源，而 CSS 的出現將使這個問題獲得部分紓解，CSS 可使用單純的文字來達到與圖片相同的表現效果！也就是說：一段簡短的文字就可以取代體積龐大的圖片！不但傳輸速度加快，同時也獲得與圖片相同的視覺效果。
- 網頁維護容易－為了維持特立獨行的網站風格，同時也為了維持網頁的一致性，可以建立一個獨立的 CSS 樣式表檔案，讓我們建立的每一個網頁樣式都能參照此一樣式檔案，當此一樣式檔案內容改變後，所有參照此一樣式檔案的網頁外觀也就會全部跟著改觀了，因此我們不必一一的修改各個網頁來達到網站風格的一致性，如果維護的網頁只有幾頁，或許並不會感受到它的便利性，如果是上百頁甚至數百頁，就會體會到它所帶來的便利性。
- 隱藏樣式－自行設定的樣式，可在 HTML 程式碼中用 `<!-- -->` 隱藏起來，或另外存於外部的 CSS 檔案裡呼叫出來使用。

程式碼比較

- 無 CSS 程式碼

```
<html>
<head><title>無 CSS 程式碼網頁</title></head>
<body>
<p align="center"><b><font color="#0000FF" size="5">天下無難事</font></b></p>
<p><b><i><font color="#FF0000" size="2">只怕有心人</font></i></b></p>
</body>
</html>
```

- 有 CSS 程式碼

```
<html>
<head><title>有 CSS 程式碼文件</title>
<style type="text/css">
.style1 {color: #0000FF; font-weight: bold; font-size: 18px;}
.style2 {color: #FF0000; font-style: italic; font-size: 12px; font-weight: bold;}
</style>
</head>
<body>
<p class="style1">天下無難事</p>
<p class="style2">只怕有心人</p>
</body></html>
```

集體宣告

集體宣告的方法

- 集體宣告有兩種方法，第一種就是針對一個標籤，然後一次設定好幾個樣式性質。第二種就是同時對好幾個標籤設定一樣的樣式性質。
- 針對一個標籤，然後一次設定好幾個樣式性質

```
<style type="text/css">
body{font-size:9pt;
color:red;
background:white}
</style>
```

- 同時對標籤宣告了：字形大小為 9pt、字形顏色為紅色、背景為白色，這三個樣式性質，利用分號";"來將之隔開，這樣才不會無法運作。當然，若是覺得這樣一列一列的寫出來很浪費空間，也可以寫成一列，如下：

```
<style type="text/css">
body{font-size:9pt; color:red; background:white}
</style>
```

- 數個標籤同時宣告

```
<style type="text/css">
h1,h2,td{font-size:12pt; font-color:red; font-family:標楷體}
</style>
```

- 以此例而言，同時設定了三組標籤<h1>、<h2>、<td>，換言之，凡是被這三個標籤包起來的，其文字大小都會變成 12pt、紅色字、並且以標楷體顯示。

分項宣告

分項宣告

- 分項宣告並不是什麼特別的用法，如下例：

```
<style type="text/css">
body{font-size:9pt; color:red}
body{font-family:標楷體; background:white}
</style>
```

- 上例中同時對標籤做了兩次宣告，此即分項宣告。

- 在分項宣告中，如下情形要特別注意

```
<style type="text/css">
body{font-size:9pt; font-color:red}
body{font-family:標楷體; font-color:blue}
</style>
```

- 同一個標籤來作分項宣告，結果裡面的樣式性質有一組竟是一樣的（font-color 性質一樣，但是設定值不一樣），其結果是，後出現的屬性值會蓋過先出現的屬性值，所以被標籤包起來的文字還是藍色。

外部呼叫

- 外部呼叫的優點
 - 網頁處理速度會快一點，尤其在很多網頁共用一份 CSS 表時更好用。因為不用每一頁網頁都擺同樣的 CSS 碼，網頁自然就會瘦一點、輕快一點。
 - 可以防止一些電腦程度較低的使用者直接看到 CSS 語法（就是有人不喜歡被看見語法），當然指的是無法直接看到，而非無法看到，稍微有點道行的，要查看 CSS 檔的內容簡直易如反掌。
 - 維護方便，要修改一個 CSS 檔，不管有幾千篇網頁，都會以最新修改的版本為準。

製做 CSS 檔

- 首先將原來宣告中的<style type="text/css"></style>拿掉（CSS 檔裡，不需這兩個標籤...因為，瀏覽器看副檔名是 css 就知道這是一個 css 檔了），然後利用任一文字編輯軟體輸入以下的語法：

```
body{font-size:9pt}
```

- 接著將之存檔，副檔名存成 css，例如 my_css.css，並放置於和網頁同一目錄下，然後利用任一文字編輯軟體輸入以下的語法

從網頁呼叫 CSS 檔

- 做好 CSS 檔後，呼叫的方法如下：

```
<head>
<title>CSS 外部呼叫範例</title>
<link rel=stylesheet type="text/css" href="my_css.css">
</head>
```

單獨使用

加入 STYLE 屬性

- 「單獨使用」顧名思義，就是將 CSS 宣告套用在單獨的一個網頁元件上（任一個文字、圖片、表格...等，都是網頁元件），此時，CSS 將不再以一個表來狀態來呈現，而是利用 **style** 屬性來加到標籤中。
- 需要「單獨使用」CSS 的原因是：比較靈活。例如，希望讓輸入表單的底色，不要是白色的，想將之改成淺綠色，就需要單獨使用 **Style** 了。

STYLE 屬性

- 幾乎每個 HTML 標籤都有其各自的屬性，例如 `<p align="center">` 其中的 `align` 便是標籤 `<p>` 的「`align` (對齊)屬性」，加上 `align` 屬性後，就可以設定段落要置左、置中或置右。相同的，**STYLE** 也可以當作是一個屬性，一樣加在標籤之中，以們上面的問題而言，其原先的語法應該是這樣：

```
<input type="text" value="淺綠色的底">
```

- 替這個輸入表單加入 **STYLE** 屬性，也就是在 `<input>` 標籤中加入 **style** 屬性：

```
<input type="text" value="淺綠色的底" style="background-color:#ccffcc">
```

- ◆ 「`style="*****"`」就是把 **style** 當作屬性的用法，其後一樣將 CSS 宣告放在"`"`裡面。
- ◆ `background-color:#ccffcc` 表示「背景顏色：(為) `#ccffcc`」的意思。
- 基本上，任何標籤都可以加上 **style** 屬性。所以不用擔心會有標籤不接受。
- 不過，單獨針對網頁元件加入 **style** 屬性雖然非常靈活，但是，卻喪失了 CSS 的一個優點，那就是統一性。所以在使用時，除非只有部份元件需要單獨的 CSS 格式，否則，盡量使用之前介紹的套用方法，如此，日後在維護上會比較簡單。

自定類別

類別的定義

- 在 HTML 標籤中，有個屬性叫做 **class**，稱為「類別」屬性，能夠讓我們在不同的標籤中使用相同的 CSS 設定。舉個例子如下：在讀書的時候，常常會在課本上用紅筆用螢光筆畫畫重點，如果，今天在網頁上，也想將一些比較重要的地方，用「紅色、粗體字」來呈現，把上一段拿來畫重點：

```
<font color="red"><b>類別</b></font>
```

用 CSS 畫重點

- 若是網頁中有 1000 多個所謂的「重點」，今天突然想把重點通通換成「藍色」的，那該怎麼辦？這時，就請出 CSS 的「自定類別」派上用場。

```
<style type="text/css">
.important { color:red ; font-weight:bold }
</style>
```

- 新原始碼 `類別`
- 首先，在 CSS 表中做一個叫做 `important` 的類別，然後利用 `class` 屬性，套用在網頁中，以後若是要將「紅色」改成「藍色」，那麼只要改類別裡的設定就行。

自定類別的用法

- 自定類別方法和一般的 CSS 宣告沒什麼兩樣，唯一的不同是，CSS 宣告是針對某標籤；而自定類別則不針對某標籤，而是要自己命名。樣式語法

```
<style type="text/css">
.important { color:red ; font-weight:bold }
</style>
```

- 以上為例，`.important { color:red ; font-weight:bold }`就是自定的類別，其中的 `{}`部份和一般的 CSS 宣告方法一模一樣！差別就在前面的 `.important`，`important` 就是自定的名稱，有了名稱，才能呼叫。
- 注意！前面記得在自定名稱前加一個小點「.」，這樣就完成了自定一個類別！這個類別可以利用 `class` 來呼叫並套用到任何標籤中！

文字屬性

文字屬性的設定

- 文字屬性是可以設定文字的顏色、大小、字型、粗細等等。例如

```
<style>
span.c1 {color:red;position:relative;font-size:12px;font-family:arial;font-weight:700}
span.c2 {color:blue}
span.c3 {color:green}
span.c4 {color:pink}
</style>
<span class=c1>a</span>
<span class=c2>b</span>
<span class=c3>c</span>
<span class=c4>d</span>
```

文字顏色{ COLOR }

- 語法：{ color: (color) }
- 範例

```
<style>
span.test {color:red}
</style>
<span class=test>a b c d</span>
```

文字類別{ FONT }

- 設定字體樣式、大小寫變化、粗細、大小、文字行列高度、字型
- 語法：
{ font : (font-style) | (font-variant) | (font-weight) | (font-size) | (font-family) | /
(line-height) }
- 範例

```
<style>
span.test {font:italic bold 細明體 12pt/18pt}
</style>
<span class=test>a b c d</span>
```

下載字型{ @FONT-FACE }

- 語法：@font-face { font-family : (font-family) ; src : url (url) }
- 範例

```
<style>
@font-face { font-family : (未型) ; src : url (下載地址) }
</style>
```

文字字型 { FONT-FAMILY }

- 語法：{ font-family : (font-name) | (generic-family) }
- 範例

```
<style>
span.test {font-family:"arial black"}
</style>
<span class=test>a b c d</span>
```

文字大小{ FONT-SIZE }

- 設定字體大小 (可設單位屬性：點 pt、英寸 in、公分 cm、像素 px、百分比 %)
- 語法：
{ font-size : large | medium | smaller | (length) | (percentage) }

- 範例

```
<style>
span.test1 {font-size:large}
span.test2 {font-size:10pt}
span.test3 {font-size:50%}
</style>
<span class=test>a b c d</span>
<span class=test1>e f g h</span>
<span class=test2>i j k l</span>
```

文字樣式 { FONT-STYLE }

- 設定字體樣式（分爲：正常、斜體）語法：
{ font-style : normal | italic }
- 範例

```
<style>
span.test1 {font-style:italic}
</style>

<span class=test>a b c d</span>
```

文字變化 { FONT-VARIANT }

- 設定字體變化（分爲：正常、小字體）
語法：{ font-variant : normal | small-caps }
- 範例

```
<style>
span.test1 {font-variant:small-caps}
</style>

<span class=test>a b c d</span>
```

文字粗細 { FONT-WEIGHT }

- 設定字體粗細（分爲：正常、粗字體）
語法：{ FONT-WEIGHT : NORMAL | BOLD }
- 範例

```
<style>
span.test1 {font-weight:bold}
</style>
```

```
<span class=test>a b c d</span>
```

連結屬性

定義連結屬性

- 以往 HTML 文件裡的超連結文字經過瀏覽器的解譯後都會以加底線的方式來顯現，並沒有動態的效果，透過 CSS 之 Anchor Pseudo Classes，可以將超連結文字的顯現方式做幾種不同的變化。
- 例如：`HongKong.com`
 - 以上的超連結經過在瀏覽器畫面上是以加底線的方式顯示。但如果利用 CSS 之 Anchor Pseudo Classes 則可以做出動態的效果，其標籤如下：記得兩個屬性間以 ";" 隔開，詳細屬性值設定請參閱 CSS 常用屬性 (Properties) 與設定值一覽表。

屬性分類

- 範例

```
a:link{color:blue; font-size:9pt; text-decoration:none}
a:visited{color:blue; font-size:9pt; text-decoration:none}
a:hover{color:red; font-size:9pt; text-decoration:underline}
a:active{color:red; font-size:9pt; text-decoration:underline}
```

- a:link 是代表普通狀態的連結
- a:visited 是代表造訪過的連結
- a:hover 是代表滑鼠移到連結
- a:active 是代表按下去連結
- color 是代表連結顏色
- font-size 是代表連結文字大小，適用 point(pt) 或是 pixels(px)
- text-decoration 是代表文字樣式，none 代表沒有底線

邊框屬性

- 定義邊框屬性：在 CSS 中，幾乎所有的元件都可以使用框線設定，不像 HTML 般，只有表格才可使用框線。CSS 不但可任意控制框線粗細、顏色，就連框線的形式都可以設定。

屬性分類

- BORDER

語法：{ border : (border-width) | (border-style) | (color) }

- (color) - 可設為顏色名稱 (16 種) 或 16 進制表示法 - #rrggbb (00 暗 ~ ff 亮)、#rgb (0 暗 ~ f 亮) 代表紅綠藍強度

- (border-style) - 可設無、實線、雙線、溝線、脊線、嵌入線、浮出線
- (border-width) - 可設薄、普通、厚、長度-單位屬性：點 pt、英寸 in、公分 cm、像素 px
- border-color：設定邊框顏色，可設為顏色名稱（16 種）或 16 進制表示法 - #rrggbb (00 暗 ~ ff 亮)、#rgb (0 暗 ~ f 亮) 代表紅綠藍強度
語法：{ border-color : (color) }
- border-style：設定邊框樣式（可設無、實線、雙線、溝線、脊線、嵌入線、浮出線）
語法：{ border-style : none | solid | double | groove | ridge | inset | outset }
- border-width：設定邊框寬度（可設薄、普通、厚、長度-單位屬性：點 pt、英寸 in、公分 cm、像素 px）
語法：{ border-width : thin | medium | thick | (length) }
- border-top：設定上邊框
語法：{ border-top : (border-top-width) | (border-style) | (color) }
- border-top-color：設定上邊框顏色，可設為顏色名稱（16 種）或 16 進制表示法 - #rrggbb (00 暗 ~ ff 亮)、#rgb (0 暗 ~ f 亮) 代表紅綠藍強度
語法：{ border-top-color : (color) }
- border-top-style：設定上邊框樣式（可設無、實線、雙線、溝線、脊線、嵌入線、浮出線）
語法：{ border-top-style : none | solid | double | groove | ridge | inset | outset }
- border-top-width：設定上邊框寬度（可設薄、普通、厚、長度-單位屬性：點 pt、英寸 in、公分 cm、像素 px）
語法：{ border-top-width : thin | medium | thick | (length) }
- border-right：設定右邊框
語法：{ border-right : (border-right-width) | (border-style) | (color) }
- border-right-color：設定右邊框顏色，可設為顏色名稱（16 種）或 16 進制表示法 - #rrggbb (00 暗 ~ ff 亮)、#rgb (0 暗 ~ f 亮) 代表紅綠藍強度
語法：{ border-right-color : (color) }
- border-right-style：設定右邊框樣式（可設無、實線、雙線、溝線、脊線、嵌入線、浮出線）
語法：{ border-right-style : none | solid | double | groove | ridge | inset | outset }
- border-right-width：設定右邊框寬度（可設薄、普通、厚、長度-單位屬性：點 pt、英寸 in、公分 cm、像素 px）
語法：{ border-right-width : thin | medium | thick | (length) }
- border-bottom：設定下邊框
語法：{ border-bottom : (border-bottom-width) | (border-style) | (color) }
- border-bottom-color：設定下邊框顏色，可設為顏色名稱（16 種）或 16 進制表示法 - #rrggbb (00 暗 ~ ff 亮)、#rgb (0 暗 ~ f 亮) 代表紅綠藍強度

語法：{ border-bottom-color : (color) }

- border-bottom-style：設定下邊框樣式（可設無、實線、雙線、溝線、脊線、嵌入線、浮出線）

語法：{ border-bottom-style : none | solid | double | groove | ridge | inset | outset }

- border-bottom-width：設定下邊框寬度（可設薄、普通、厚、長度-單位屬性：點 pt、英寸 in、公分 cm、像素 px）

語法：{ border-bottom-width : thin | medium | thick | (length) }

- border-left：設定左邊框

語法：{ border-left : (border-left-width) | (border-style) | (color) }

- border-left-color：設定左邊框顏色，可設為顏色名稱（16 種）或 16 進制表示法 - #rrggbb（00 暗 ~ ff 亮）、#rgb（0 暗 ~ f 亮）代表紅綠藍強度

語法：{ border-left-color : (color) }

- border-left-style：設定左邊框樣式（可設無、實線、雙線、溝線、脊線、嵌入線、浮出線）

語法：{ border-left-style : none | solid | double | groove | ridge | inset | outset }

- border-left-width：設定左邊框寬度（可設薄、普通、厚、長度-單位屬性：點 pt、英寸 in、公分 cm、像素 px）

語法：{ border-left-width : thin | medium | thick | (length) }

段落屬性

定義段落屬性

- 利用這些設定可以輕易的控制字距、行高、縮排、凸排、水平對齊、垂直對齊等。這些性質對網頁設計的排版非常有用！

屬性分類

- letter-spacing：設定文字間隙（可設單位屬性：點 pt、英寸 in、公分 cm、像素 px）

語法：{ letter-spacing : normal | (length) }

- line-height：設定文字行列高度（可設單位屬性：點 pt、英寸 in、公分 cm、像素 px、百分比 %）

語法：{ line-height : normal | (number) | (length) | (percentage) }

- text-align：設定調整文字（可設左邊、右邊、置中、整齊）

語法：{ text-align : left | right | center | justify }

- text-decoration：設定文字裝飾（分為：無、加底線、加橫線、加頂線）

語法：{ text-decoration : none | underline | line-through | overline }

- text-indent：設定文字縮排（可設單位屬性：點 pt、英寸 in、公分 cm、像素 px、百分比 %）

語法：{ text-indent : (length) | (percentage) }

- text-transform：設定文字改變（可設無、第一個字母大寫、大寫、小寫）

語法：{ text-transform : none | capitalize | uppercase | lowercase }

滑鼠屬性

滑鼠屬性

- 可以設定滑鼠游標有很多類形可以使用（十字、箭頭、手形、移動、雙箭頭、文字、等待、求助等等）

屬性分類

NO	語法	顯示	NO	語法	顯示
01	cursor : n-resize	指北箭頭	02	cursor : s-resize	指南箭頭
03	cursor : w-resize	指西箭頭	04	cursor : e-resize	指東箭頭
05	cursor : sw-resize	指西南箭頭	06	cursor : se-resize	指東南箭頭
07	cursor : ne-resize	指東北箭頭	08	cursor : nw-resize	指西北箭頭
09	cursor : hand	手形符號	10	cursor : move	十字箭頭
11	cursor : help	問號箭頭	12	cursor : text	工字符號
13	cursor : wait	漏斗符號	14	cursor : crosshair	十字符號
15	cursor : default	一般箭頭	16	cursor : url(mycursor.cur)	自定游標

背景屬性

定義背景屬性

- 背景性質設定可說是 CSS 最強悍的地方，有了這些 CSS，甚至可以在一段文字，或一格表格中，套用不同的背景。

屬性分類

- background：設定背景圖片、顏色、混合、透空、捲動、位置、重複
語法：{ background : transparent | (scroll) | (color) | url (url) | (position) | (repeat) }
- background-attachment：設定背景圖片是否捲動（分為：捲動、固定）
語法：{ background-attachment : scroll | fixed }
- background-color：設定背景顏色、透明
語法：{ background-color : transparent | (color) }
 - (color) 可為顏色名稱 或 16 進制表示法 - #rrggbb (00 暗 ~ ff 亮) 、 #rgb (0 暗 ~ f 亮) 代表紅綠藍強度。
- background：設定背景圖片、顏色、混合、透空、捲動、位置、重複
語法：{ background : transparent | (scroll) | (color) | url (url) | (position) | (repeat) }
- background-position：設定背景位置（可設單位屬性：點 pt、英寸 in、公分 cm、

像素 px、百分比 %)

語法：{ background-position : top | middle | bottom | left | center | right | (length) |
(position) }

- background-repeat : 設定背景重複填滿方式

語法：

{ background-repeat : repeat | repeat-x | repeat-y | no-repeat }

附錄－網頁 CSS 性質參考資料表

- 參考連結

<http://www.w3.org/TR/REC-CSS2/indexlist.html>

<http://www.w3.org/TR/REC-CSS2/cover.html>

字型

性質名稱	可用值	作用範圍	是否可繼承	%百分比
font-family	[[<family-name> <generic-family>],]* [<family-name> <generic-family>] 預設值：依瀏覽器而定	所有元素	是	不可使用
font-style	normal italic oblique	所有元素	是	不可使用
font-variant	normal small-caps	所有元素	是	不可使用
font-weight	normal bold bolder lighter 100 200 300 400 500 600 700 800 900	所有元素	是	不可使用
font-size	<absolute-size> <relative-size> <length> <percentage> 預設值：medium	所有元素	是	相對於親元素的字形尺寸
font (簡寫性質)	[<font-style> <font-variant> <font-weight>]? <font-size> [/<line-height>]? <font-family> 預設值：未定義	所有元素	是	可使用在 <font-size> 與 <line-height> 兩部分

顏色與背景

性質名稱	可用值	作用範圍	是否可繼承	%百分比
color	<color> 預設值：依瀏覽器而定	所有元素	是	不可使用
background-color	transparent	所有元素	否	不可使用

	<color>			
background-image	none <url>	所有元素	否	不可使用
background-repeat	repeat repeat-x repeat-y no-repeat	所有元素	否	不可使用
background-attachment	scroll fixed	所有元素	否	不可使用
background-position	[<percentage> <length>]{1,2} [top center bottom] [left center right] 預設值：0% 0%	區塊等級與替代元素	否	相對於該元素本身的尺寸大小
background (簡寫性質)	<background-color> <background-image> <background-repeat> <background-attachment> <background-position> 預設值：未定義	所有元素	否	可使用在 <background-position>

文字

性質名稱	可用值	作用範圍	是否可繼承	%百分比
word-spacing	normal <length>	所有元素	是	不可使用
letter-spacing	normal <length>	所有元素	是	不可使用
text-decoration	none [underline overline line-through blink] 預設值：none	所有元素	否	不可使用
vertical-align	baseline sub super top text-top middle bottom text-bottom <percentage>	行內元素	否	相對於該元素本身的行高
text-transform	none capitalize	所有元素	是	不可使用

	uppercase lowercase			
text-align	left right center justify 預設值：依瀏覽器而定	區塊等級元素	是	不可使用
text-indent	<length> <percentage> 預設值：0	區塊等級元素	是	相對於親元素的寬度
line-height	normal <number> <length> <percentage>	所有元素	是	相對於該元素本身的字型尺寸大小

區塊

性質名稱	可用值	作用範圍	是否可繼承	%百分比
margin-top	<length> <percentage> auto 預設值：0	所有元素	否	相對於親元素的寬度
margin-right	<length> <percentage> auto 預設值：0	所有元素	否	相對於親元素的寬度
margin-bottom	<length> <percentage> auto 預設值：0	所有元素	否	相對於親元素的寬度
margin-left	<length> <percentage> auto 預設值：0	所有元素	否	相對於親元素的寬度
margin (簡寫性質)	[<length> <percentage> auto]{1,4} 預設值：未定義	所有元素	否	相對於親元素的寬度
padding-top	<length> <percentage> 預設值：0	所有元素	否	相對於親元素的寬度
padding-right	<length> <percentage> 預設值：0	所有元素	否	相對於親元素的寬度
padding-bottom	<length> <percentage> 預設值：0	所有元素	否	相對於親元素的寬度
padding-left	<length> <percentage>	所有元素	否	相對於親元素的寬

	預設值：0			度
padding	[<length> <percentage>]{1,4} 預設值：0	所有元素	否	相對於親元素的寬度
border-top-width	medium thin thick <length>	所有元素	否	不可使用
border-right-width	medium thin thick <length>	所有元素	否	不可使用
border-bottom-width	medium thin thick <length>	所有元素	否	不可使用
border-left-width	medium thin thick <length>	所有元素	否	不可使用
border-width (簡寫性質)	[thin medium thick <length>]{1,4} 預設值：未定義	所有元素	否	不可使用
border-color	<color>{1,4} 預設值：「color」性質的值	所有元素	否	不可使用
border-style	none dotted dashed solid double groove ridge inset outset	所有元素	否	不可使用
border-top (簡寫性質)	<border-top-width> <border-style> <color> 預設值：未定義	所有元素	否	不可使用
border-right (簡寫性質)	<border-right-width> <border-style> <color> 預設值：未定義	所有元素	否	不可使用
border-bottom (簡寫性質)	<border-bottom-width> <border-style> <color> 預設值：未定義	所有元素	否	不可使用
border-left (簡寫性質)	<border-left-width> <border-style> <color> 預設值：未定義	所有元素	否	不可使用

border (簡寫性質)	<border-width> <border-style> <color> 預設值：未定義	所有元素	否	不可使用
width	auto <length> <percentage>	區塊等級 與替代元 素	否	相對於親 元素的寬 度
height	auto <length>	區塊等級 與替代元 素	否	不可使用
float	none left right	所有元素	否	不可使用
clear	none left right both	所有元素	否	不可使用

分類

性質名稱	可用值	作用範圍	是否可繼承	%百分比
display	block inline list-item none	所有元素	否	不可使用
white-space	normal pre nowrap	區塊等級元素	是	不可使用
list-style-type	disc circle square decimal lower-roman upper-roman lower-alpha upper-alpha none	所有 「display」性 質的值是 「list-item」的 元素	是	不可使用
list-style-image	none <url>	所有 「display」性 質的值是 「list-item」的 元素	是	不可使用
list-style-position	outside inside	所有 「display」性 質的值是 「list-item」的 元素	是	不可使用
list-style (簡寫性質)	<keyword> <position> <url> 預設值：未定義	所有 「display」性 質的值是 「list-item」的	是	不可使用

		元素		
--	--	----	--	--

定位

性質名稱	可用值	作用範圍	是否可繼承	%百分比
position	static absolute relative	所有元素	否	不可使用
left	auto <length> <percentage>	所有「position」性質的值是「absolute」或者「relative」的元素	否	相對於親元素的寬度與高度。如果親元素的高度的值是「auto」，則百分比型式的值視為未定義
top	auto <length> <percentage>	所有「position」性質的值是「absolute」或者「relative」的元素	否	相對於親元素的寬度與高度。如果親元素的高度的值是「auto」，則百分比型式的值視為未定義
z-index	auto <integer>	所有可定位的元素	否	不可使用
width	auto <length> <percentage>	區塊等級與替代元素，以及所有「position」性質的值是「absolute」的元素	否	相對於親元素的寬度與高度。如果親元素的高度的值是「auto」，則百分比型式的值視為未定義
height	auto <length> <percentage>	區塊等級與替代元素，以及	否	相對於親元素的寬度與

		所有「 position 」性質的值是「 absolute 」的元素		高度。如果親元素的高度的值是「 auto 」，則百分比型的值視為未定義
clip	auto <shape>	所有「 position 」性質的值是「 absolute 」的元素	否	不可使用
overflow	visible hidden scroll auto	所有「 position 」性質的值是「 absolute 」的元素	否	不可使用
visibility	inherit visible hidden	所有元素	如果此性質的值是「 inherit 」，便繼承親元素的值，否則此性質沒有可繼承性	不可使用