

CSS

長這樣

```
html, body {  
  position: relative;  
  padding: 0; margin: 0;  
  overflow: hidden;  
  font-family: '文泉驛微米黑', 'LiHei Pro', '微軟正黑體', sans-serif;  
  font-weight: 200;  
  letter-spacing: -0.02em;  
  color: #eee; width: 100%; height: 100%;  
  min-height: 400px;  
}  
  
table {  
  border-collapse: collapse;  
  border-spacing: 0;  
}
```

結合 CSS 技術到 HTML 中

- CSS (Cascading Style Sheet) 樣式表，主要用來裝飾、編排網頁內容，與 HTML 相輔相成。
- HTML 用來表示結構性的資料，CSS 決定網頁的樣式 (顏色、文字大小、位置等等)。

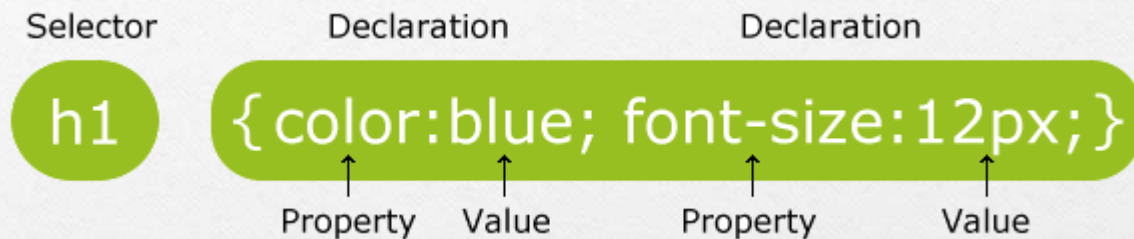
為什麼要CSS

- HTML 的 `` 標籤可以決定文字的顏色和大小，``、`<i>`、`<u>` 等標籤都有改變文字樣式的能力。
- 在 CSS 尚未被廣泛採用之前，這些 HTML 標籤是主要決定網頁樣式的方法。

為什麼要CSS

- 隨著分工細膩，我們希望把美化網頁的工作和設計 HTML 文件的工作獨立分開運作。
- 如果能讓同樣的 HTML 文件，透過套用不同的樣式來達到改變外觀的效果，那是最理想的。
- => CSS 可以做得到。
- 例如：部落格樣式。

CSS 語法 CSS Syntax



- CSS 語法的基本構成是一個 CSS 屬性名稱搭配屬性的內容。寫法如下：`color:red`
- 可以一次定義很多屬性，中間用分號分開。

`color:red;font-size:14px`

Source: [W3Schools](https://www.w3schools.com/css/css_syntax.asp)

在 HTML 中加入 CSS

- 最簡單的方式就是為 HTML 標籤加上一個 style 屬性。style 屬性的內容必須以 CSS 語法撰寫，用法如下
- `<div style="CSS語法">Hello World</div>`
- Example
`<div style="color:red;font-size:14px">Hello World</div>`

使用時機

- 所有 HTML 標籤都可以套用 CSS 語法，CSS 所描述的特性會套用在該標籤的內文中。

```
<h3 style="color:red">
```

這是寫在 h3 標籤中 CSS 屬性的套用範圍</h3>

font 相關常用屬性

- **font-size**: 字型大小，可使用的單位包括 px, pt, em, %。

font-size:14px

font-size:120%

font-size:1.2em

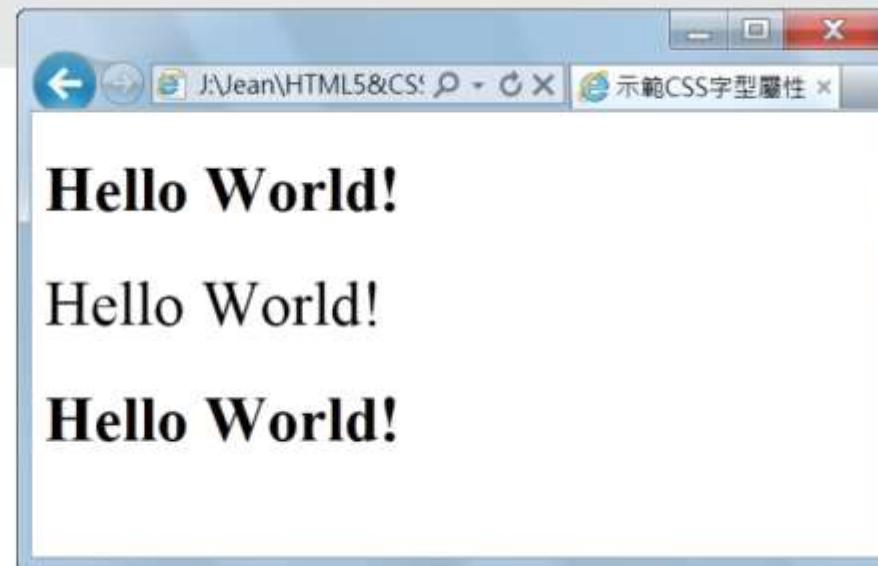
- **font-weight**: 粗細，可控制字型粗細

font-weight:bold

font-weight:normal

```
<p style="font-size:20px">生日快樂 Happy Birthday</p>
<p style="font-size:20pt">生日快樂 Happy Birthday</p>
<p style="font-size:1cm">生日快樂 Happy Birthday</p>
<p style="font-size:xx-small">生日快樂 Happy Birthday</p>
<p style="font-size:x-small">生日快樂 Happy Birthday</p>
<p style="font-size:small">生日快樂 Happy Birthday</p>
<p style="font-size:medium">生日快樂 Happy Birthday</p>
<p style="font-size:large">生日快樂 Happy Birthday</p>
<p style="font-size:x-large">生日快樂 Happy Birthday</p>
<p style="font-size:xx-large">生日快樂 Happy Birthday</p>
```

```
<body>
  <h1 style="font-weight:bold">Hello World!</h1>
  <h1 style="font-weight:normal">Hello World!</h1>
  <h1 style="font-weight:bolder">Hello World!</h1>
</body>
```



粗體與斜體

- font-style: normal | italic
- font-weight: normal | bold | 數字

HTML ▾

```
<p>Grumpy wizards make toxic brew for the evil Queen and Jack</p>
```

```
<p style="font-style: italic;"> Grumpy wizards make toxic brew for the evil Queen and Jack</p>
```

```
<p style="font-weight: 600;">600: Grumpy wizards make toxic brew for the evil Queen and Jack</p>
```

```
<p style="font-weight: 700;">700: Grumpy wizards make toxic brew for the evil Queen and Jack</p>
```

Grumpy wizards make toxic brew for the evil Queen and Jack

Grumpy wizards make toxic brew for the evil Queen and Jack

600: Grumpy wizards make toxic brew for the evil Queen and Jack

700: Grumpy wizards make toxic brew for the evil Queen and Jack

底線、刪除線、大寫小字

- text-decoration: none | underline | line-through
- font-variant: small-caps

HTML ▾

```
<p>
  Grumpy
  <span style="text-decoration: underline;">wizards
  </span> make
  <span style="text-decoration: line-through;">
    toxic brew
  </span> for the evil
  <span style="font-variant: small-caps;">
    Queen and Jack
  </span>
</p>
```

Grumpy wizards make ~~toxic brew~~ for the evil QUEEN AND JACK

text 相關常用屬性

- **text-decoration**: 文字裝飾。
text-decoration:underline
text-decoration:overline
text-decoration:line-through
- **text-align**: 對齊方式。
text-align:left
text-align:center
text-align:right

```
<body>
```

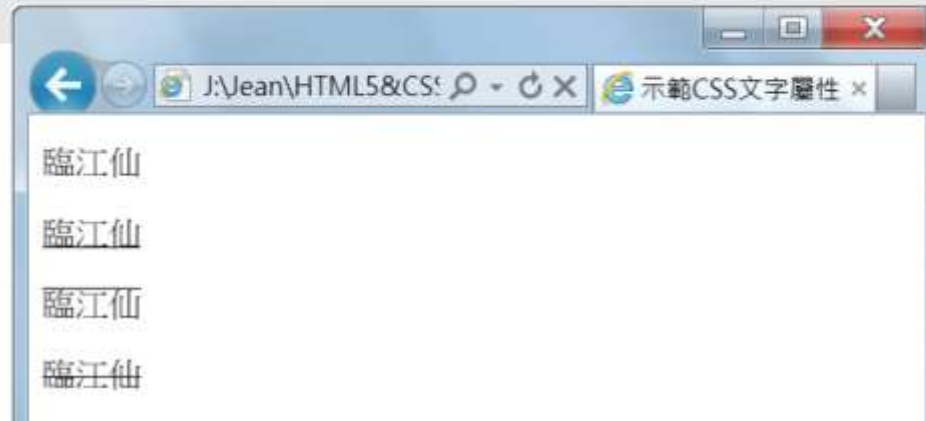
```
<p style="text-decoration:none"> 臨江仙 </p>
```

```
<p style="text-decoration:underline"> 臨江仙 </p>
```

```
<p style="text-decoration:overline"> 臨江仙 </p>
```

```
<p style="text-decoration:line-through"> 臨江仙 </p>
```

```
</body>
```



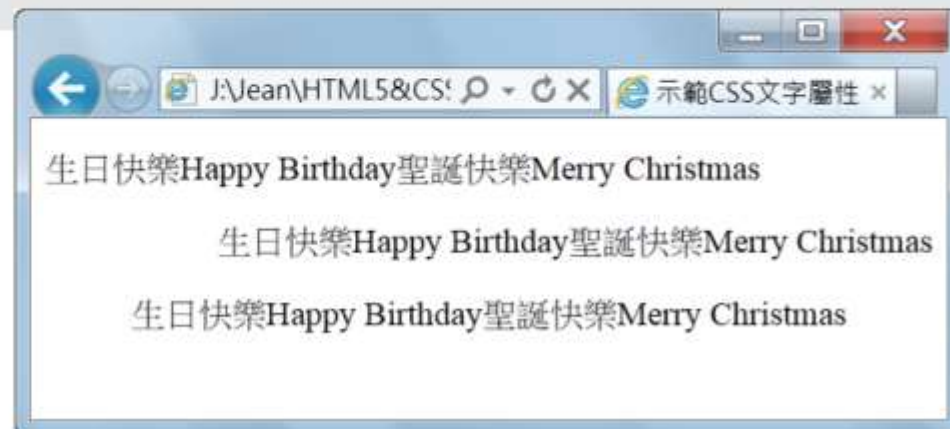
```
<body>
```

```
<p style="text-align:left"> 生日快樂 Happy Birthday 聖誕快樂 Merry Christmas</p>
```

```
<p style="text-align:right"> 生日快樂 Happy Birthday 聖誕快樂 Merry Christmas</p>
```

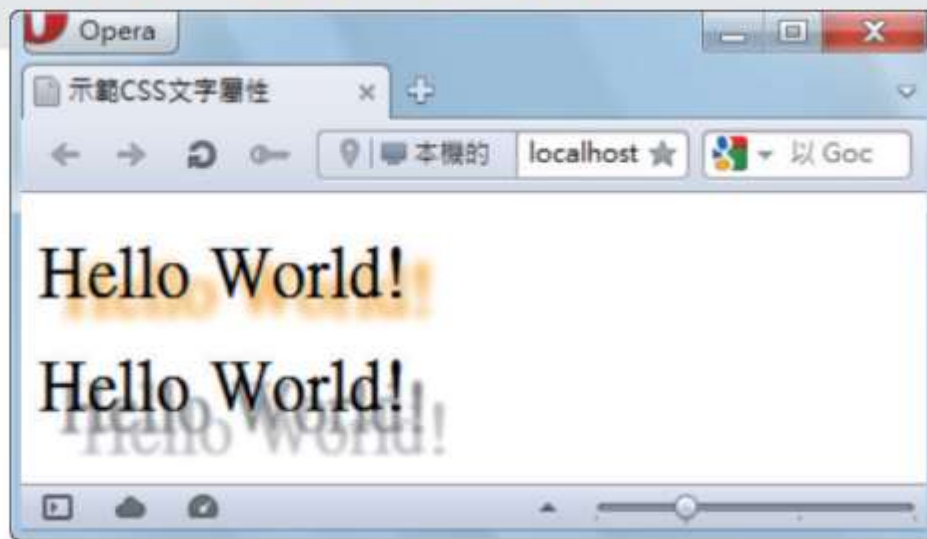
```
<p style="text-align:center"> 生日快樂 Happy Birthday 聖誕快樂 Merry Christmas</p>
```

```
</body>
```



-
- Text-shadow:右位移 下位移 模糊程度 顏色

```
<body>  
  <h1 style="text-shadow:12px 8px 5px orange">Hello World!</h1>  
  <h1 style="text-shadow:10px 10px 2px gray, 20px 20px 2px silver">Hello World!</h1>  
</body>
```



font-family

```
font-family: '文泉驛微米黑', 'LiHei Pro', '微軟正黑體', sans-serif;
```

- 如果字體在使用者的電腦裡沒有 → 用下一個
- 最後一個通常放 Generic font family ◦
- 常用之 font-family

Web Font

HTML ▾

```
<html>
<head>
<link href="http://fonts.googleapis.com/css?
family=Lobster+Two" rel="stylesheet"
type="text/css">
</head>
<body>
<h1>The quick brown fox jumps over the lazy dog.
</h1>
</body>
</html>
```

The quick brown fox jumps over the lazy dog.

- Server 提供字體給 Browser 來抓
- 通常抓下來不能直接裝在電腦上，只能顯示
- Google Web Font 英文的 Webfont
- @font-face generator 上傳字型檔後，會轉檔為 web font 並提供範例

大小與行高

- font-size: 12px
- line-height: 1.4
- font-family: Arial, sans-serif

```
/* All-in-one shorthand */  
font: 12px/1.4 Arial, sans-serif;
```

- line-height：純數字（幾倍於字體大小）或 數字px
- [了解更多](#)

實機練習

- 小試牛刀，利用 CSS 控制網頁的文字大小和文字樣式。

color 相關屬性

- **color**: 設定文字顏色，可以用代表顏色的英文字或是色碼 (以 # 開頭)

color:green

color:#0000ff

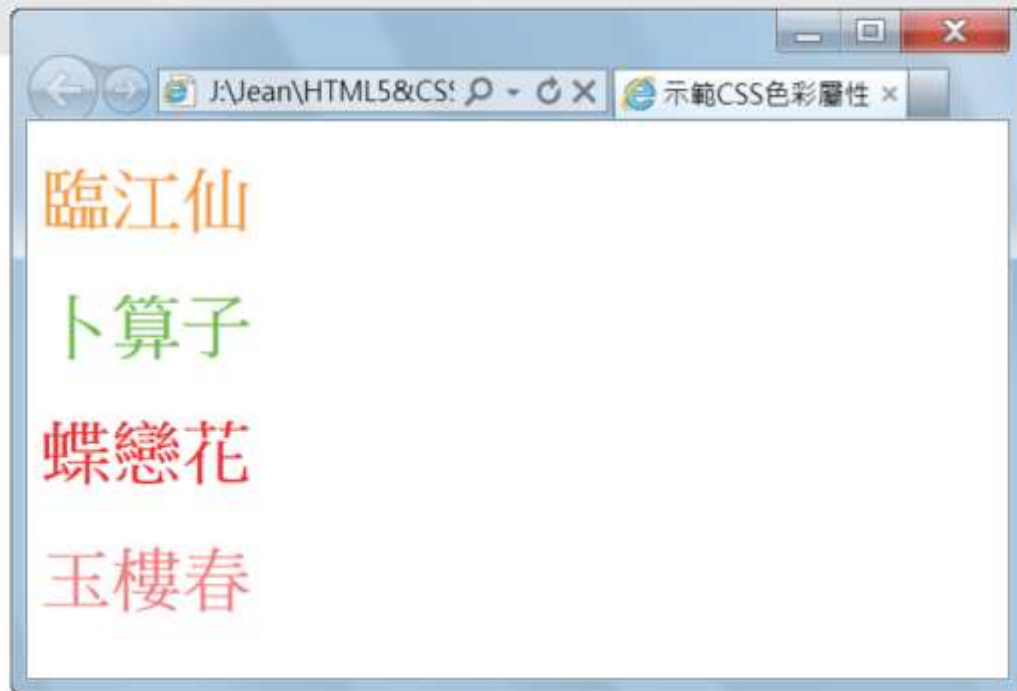
- **background-color**: 設定區塊背景顏色

background-color:red

background-color:#008800

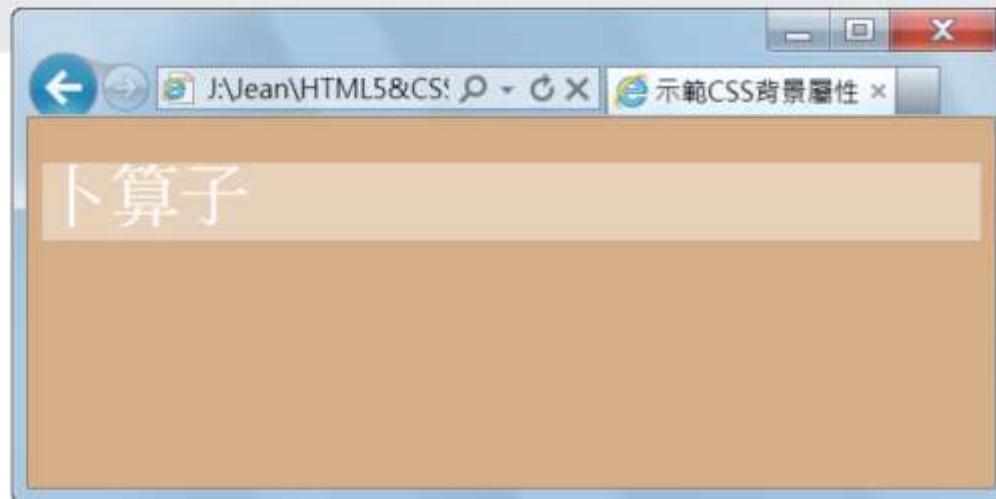
前景色彩

```
01:<body>
02:  <h1 style="color:orange"> 臨江仙 </h1>
03:  <h1 style="color:#00ff00"> 卜算子 </h1>
04:  <h1 style="color:rgb(255, 0, 0)"> 蝶戀花 </h1>
05:  <h1 style="color:rgba(255, 0, 0, 0.5)"> 玉樓春 </h1>
06:</body>
```



背景色彩

```
<body style="background-color:burlywood">  
  <h1 style="color:white; background-color:rgba(255, 255, 255, 0.5)">卜算子 </h1>  
</body>
```

















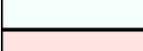

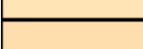


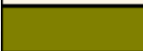




Color Table


Color	Name	RGB Value
	AliceBlue	#F0F8FF
	AntiqueWhite	#FAEBD7
	Aqua	#00FFFF
	Aquamarine	#7FFFD4
	Azure	#F0FFFF
	Beige	#F5F5DC
	Bisque	#FFE4C4
	Black	#000000
	BlanchedAlmond	#FFEBCD
	Blue	#0000FF
	BlueViolet	#8A2BE2
	Brown	#A52A2A
	BurlyWood	#DEB887
	CadetBlue	#5F9EA0
	Chartreuse	#7FFF00
	Chocolate	#D2691E
	Coral	#FF7F50
	CornflowerBlue	#6495ED
	Cornsilk	#FFF8DC
	Crimson	#DC143C
	Cyan	#00FFFF
	DarkBlue	#00008B
	DarkCyan	#008B8B
	DarkGoldenrod	#B8860B













	DarkGray	#A9A9A9
	DarkGreen	#006400
	DarkKhaki	#BDB76B
	DarkMagenta	#8B008B
	DarkOliveGreen	#556B2F
	DarkOrange	#FF8C00
	DarkOrchid	#9932CC
	DarkRed	#8B0000
	DarkSalmon	#E9967A
	DarkSeaGreen	#8FBC8F
	DarkSlateBlue	#483D8B
	DarkSlateGray	#2F4F4F
	DarkTurquoise	#00CED1
	DarkViolet	#9400D3
	DeepPink	#FF1493
	DeepSkyBlue	#00BFFF
	DimGray	#696969
	DodgerBlue	#1E90FF
	FireBrick	#B22222
	FloralWhite	#FFFAF0
	ForestGreen	#228B22
	Fuchsia	#FF00FF
	Gainsboro	#DCDCDC
	GhostWhite	#F8F8FF
	Gold	#FFD700
	Goldenrod	#DAA520

	Gray	#808080
	Green	#008000
	GreenYellow	#ADFF2F
	Honeydew	#F0FFFF
	HotPink	#FF69B4
	IndianRed	#CD5C5C
	Indigo	#4B0082
	Ivory	#FFFFFF
	Khaki	#F0E68C
	Lavender	#E6E6FA
	LavenderBlush	#FFF0F5
	LawnGreen	#7CFC00
	LemonChiffon	#FFFACD
	LightBlue	#ADD8E6
	LightCoral	#F08080
	LightCyan	#E0FFFF
	LightGoldenrodYellow	#FAFAD2
	LightGreen	#90EE90
	LightGrey	#D3D3D3
	LightPink	#FFB6C1
	LightSalmon	#FFA07A
	LightSeaGreen	#20B2AA
	LightSkyBlue	#87CEFA
	LightSlateGray	#778899
	LightSteelBlue	#B0C4DE
	LightYellow	#FFFFE0

Color Table

	Lime	#00FF00
	LimeGreen	#32CD32
	Linen	#FAF0E6
	Magenta	#FF00FF
	Maroon	#800000
	MediumAquamarine	#66CDAA
	MediumBlue	#0000CD
	MediumOrchid	#BA55D3
	MediumPurple	#9370DB
	MediumSeaGreen	#3CB371
	MediumSlateBlue	#7B68EE
	MediumSpringGreen	#00FA9A
	MediumTurquoise	#48D1CC
	MediumVioletRed	#C71585
	MidnightBlue	#191970
	MintCream	#F5FFFA
	MistyRose	#FFE4E1
	Moccasin	#FFE4B5
	NavajoWhite	#FFDEAD
	Navy	#000080
	OldLace	#FDF5E6
	Olive	#808000
	OliveDrab	#6B8E23
	Orange	#FFA500
	OrangeRed	#FF4500
	Orchid	#DA70D6

	PaleGoldenrod	#EEE8AA
	PaleGreen	#98FB98
	PaleTurquoise	#AFEEEE
	PaleVioletRed	#DB7093
	PapayaWhip	#FFEFD5
	PeachPuff	#FFDAB9
	Peru	#CD853F
	Pink	#FFC0CB
	Plum	#DDA0DD
	PowderBlue	#B0E0E6
	Purple	#800080
	Red	#FF0000
	RosyBrown	#BC8F8F
	RoyalBlue	#4169E1
	SaddleBrown	#8B4513
	Salmon	#FA8072
	SandyBrown	#F4A460
	SeaGreen	#2E8B57
	Seashell	#FFF5EE
	Sienna	#A0522D
	Silver	#C0C0C0
	SkyBlue	#87CEEB
	SlateBlue	#6A5ACD
	SlateGray	#708090
	Snow	#FFFAFA
	SpringGreen	#00FF7F

	SteelBlue	#4682B4
	Tan	#D2B48C
	Teal	#008080
	Thistle	#D8BFD8
	Tomato	#FF6347
	Turquoise	#40E0D0
	Violet	#EE82EE
	Wheat	#F5DEB3
	White	#FFFFFF
	WhiteSmoke	#F5F5F5
	Yellow	#FFFF00
	YellowGreen	#9ACD32

樣式選擇器

- 目前為止我們只會在 HTML 標籤中加入 style 屬性以套用 CSS 樣式。
- 我們有更方便的選擇：利用 HTML 標籤的
 - class 屬性
 - class 選擇器

利用 class 選擇器(屬性)

1. 為你想要套用樣式的標籤設定 class 屬性。

```
<div class="title">My Homepage</div>
```

2. 將 CSS 文件透過一個特別的標籤 `<style>` 撰寫在 `<head>` 和 `</head>` 之間。

```
<style type="text/css">  
  div {color: blue;}  
</style>
```

Output

My Homepage

利用 class 選擇器

用 `.` 加上 `class` 屬性的名稱，再用大括弧包住 CSS 語法。括弧中的 CSS 是給 class 設定為該名稱的 HTML 標籤用的。

```
<style type="text/css">  
  
.title{  
    font-size:25px;font-weight:bold  
}  
  
.subtitle{font-size:20px}  
</style>
```

宣告

宣告

樣式規則 — . 類別選擇器 { 屬性 : 值 [; 屬性 2 : 值 2 [; …]] }

```
01: <!doctype html>
02: <html>
03:   <head>
04:     <meta charset="utf-8">
05:     <title> 唐詩欣賞 </title>
06:     <style>
07:       .heading {font-family: 華康粗黑體 ; font-size:30px ; color:maroon}
08:       .content {font-family: 標楷體 ; font-size:25px; color:olive}
09:     </style>
10:   </head>
11:   <body>
12:     <h1> 唐詩欣賞 </h1>
13:     <p class="heading"> 春曉 </p>
14:     <p class="content"> 春眠不覺曉，處處聞啼鳥。夜來風雨聲，花落知多少？
15:     <p class="heading"> 竹里館 </p>
16:     <p class="content"> 獨坐幽篁裡，彈琴復長嘯。深林人不知，明月來相照。
17:   </body>
18: </html>
```

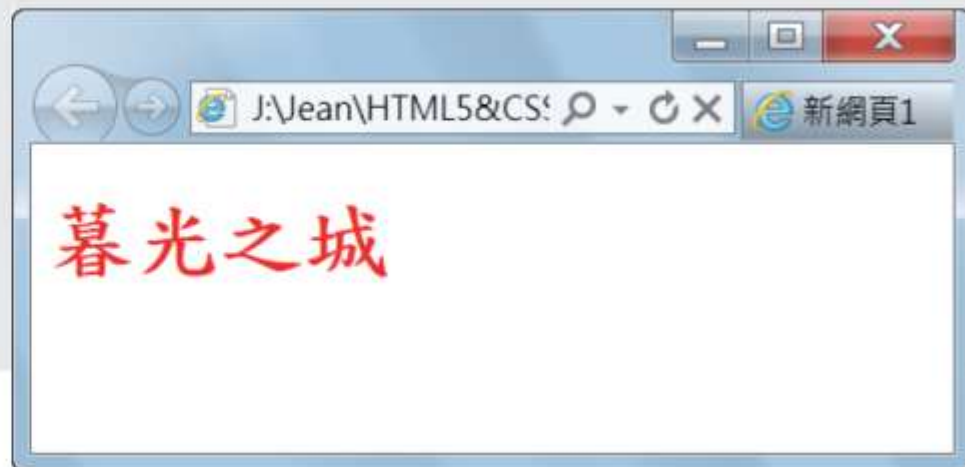


CSS 樣式規則與選擇器

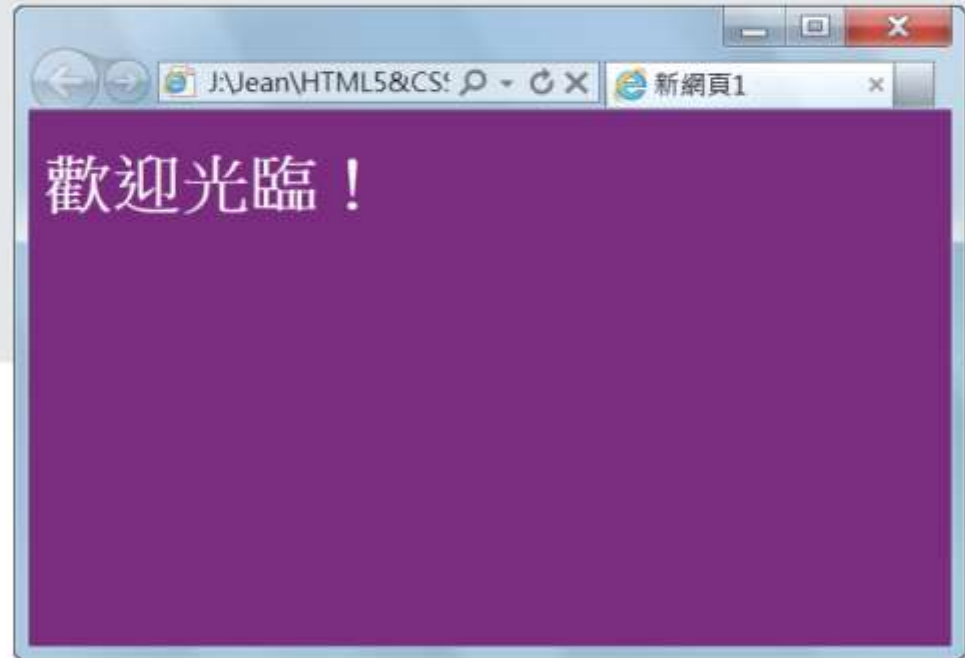
`<style>` 可放在 `<head>` 或 `<body>` 均可。



```
01: <!doctype html>
02: <html>
03:   <head>
04:     <meta charset="utf-8">
05:     <title> 新網頁 1</title>
06:     <style type="text/css">
07:       h1 {color:red; font-family: 標楷體 }
08:     </style>
09:   </head>
10:   <body>
11:     <h1> 暮光之城 </h1>
12:   </body>
13: </html>
```




```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title>新網頁 1</title>
    <style>
      body {color:white; background:purple}
    </style>
  </head>
  <body>
    <h1>歡迎光臨！ </h1>
  </body>
</html>
```



CSS Box Model

- 每個 HTML 標籤都代表一個方型的作用區塊，我們稱為 Box，CSS 對這個作用區塊可以做非常細緻的控制
- 以下先了解 CSS 對於 Box 作用區塊的設計和術語，接著在——介紹相關的屬性。

Box 與 display 屬性

```
<p>
```

```
This is a paragraph with
```

```
<em>emphasized words</em>
```

```
embedded in it.
```

```
</p>
```

```
<p>
```

```
And another...
```

This is a paragraph

with emphasized

words embedded

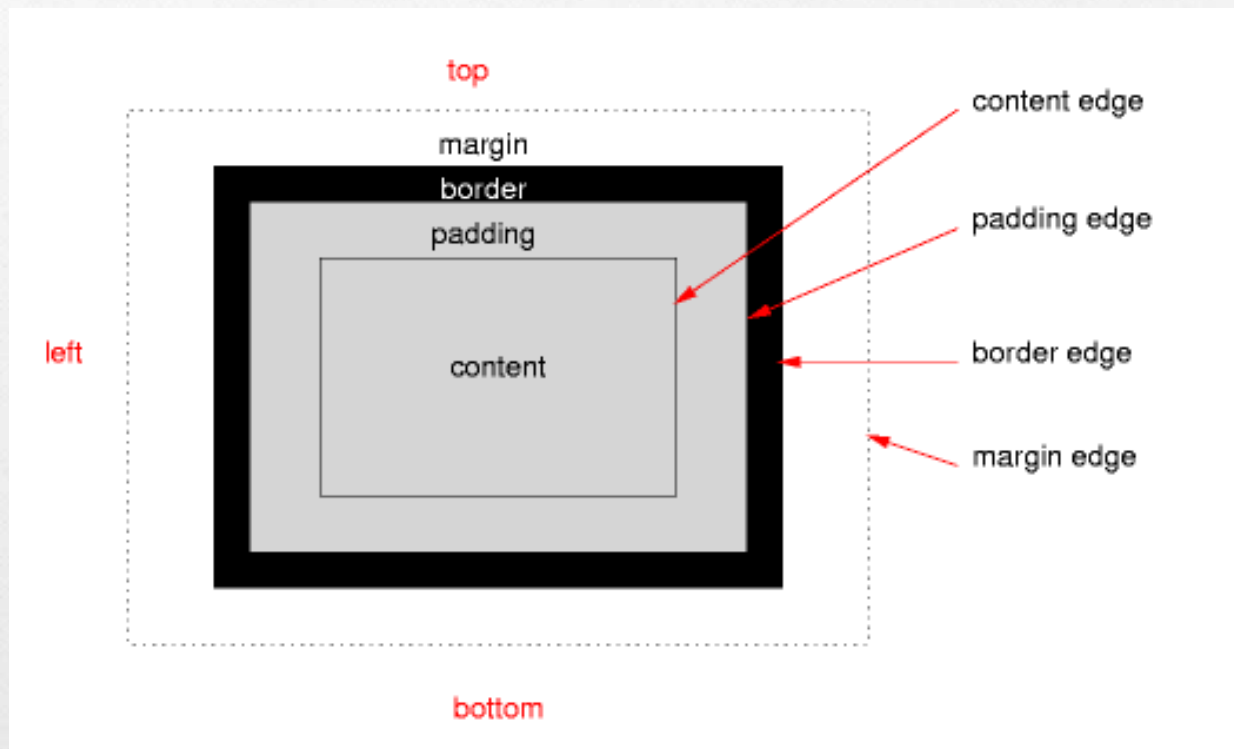
in it.

And another...

- 排文字與元素時
- 蓋成長方形 box

CSS Box Model

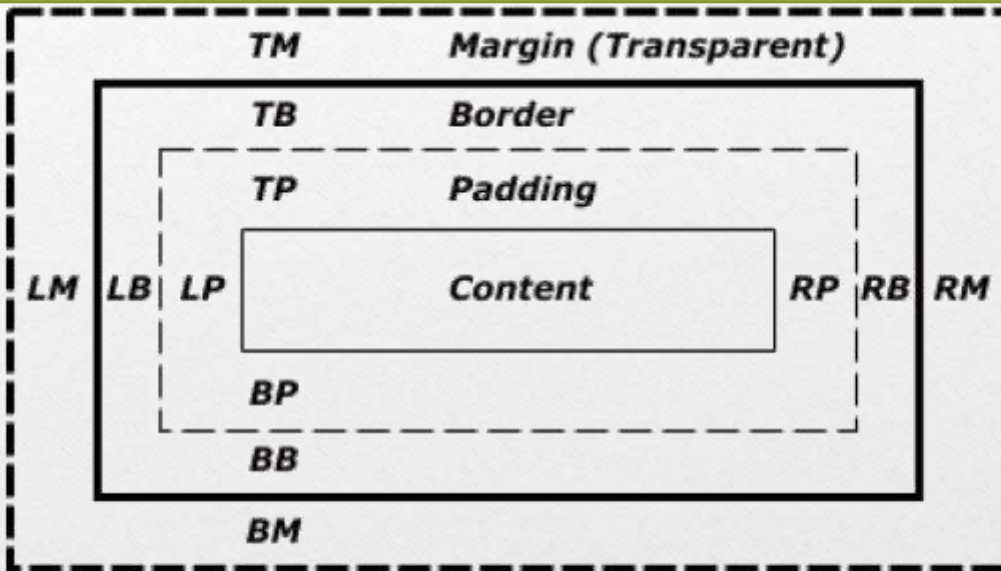
margin, border, padding



圖片來源：[W3C Box Model](#)

Box Model

- ◎ Box Model 是CSS 針對HTML 元素所產生的矩形方塊，由內容 (content)、留白 (padding)、框線 (border) 與邊界 (margin) 所組成，如下圖。



- Margin edge
- Border edge
- - - Padding edge
- Content edge

心室肥大

注意 `.heart` 的寬度與高度。

HTML ▾

```
<!DOCTYPE html>
```

```
 Lorem ipsum dolor sit amet, consectetur  
 adipiscing elit, sed do eiusmod tempor  
 incididunt ut labore et dolore magna  
 aliqua. Ut enim ad minim veniam, quis  
 nostrud exercitation ullamco laboris  
 nisi ut aliquip ex ea commodo  
 consequat.
```

```
<span class="heart">H e a r t</span>  
Duis aute irure dolor in reprehenderit  
in voluptate velit esse cillum dolore  
eu fugiat nulla pariatur. Excepteur  
sint occaecat cupidatat non proident,  
sunt in culpa qui officia deserunt  
mollit anim id est laborum.
```

CSS ▾

```
.heart{  
  display: inline-block;  
  border : 15px solid red;  
  background: pink;  
  width: 150px; height: 50px;  
  font-size: 20px;  
  margin: 20px;  
  padding: 10px;  
}
```

Output

Run with JS

Auto-run JS

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.



Duis aute irure dolor in

reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

margin 相關屬性

- **margin**: 設定邊界，可搭配 left, right, top, bottom 使用。

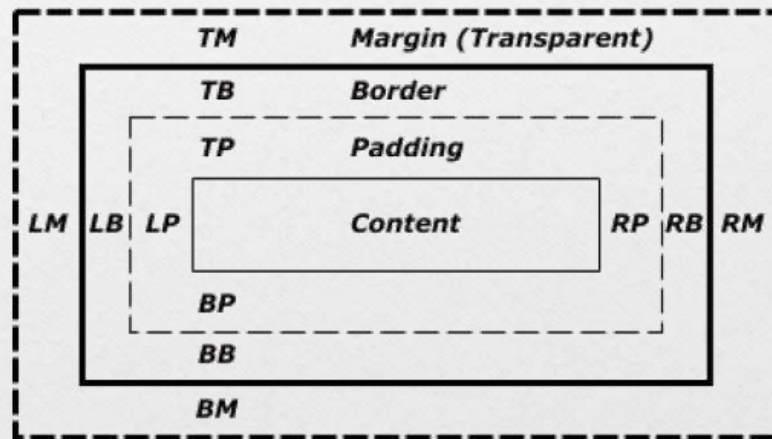
margin:5px

margin-top:5px

margin-left:3px

margin-right:10px

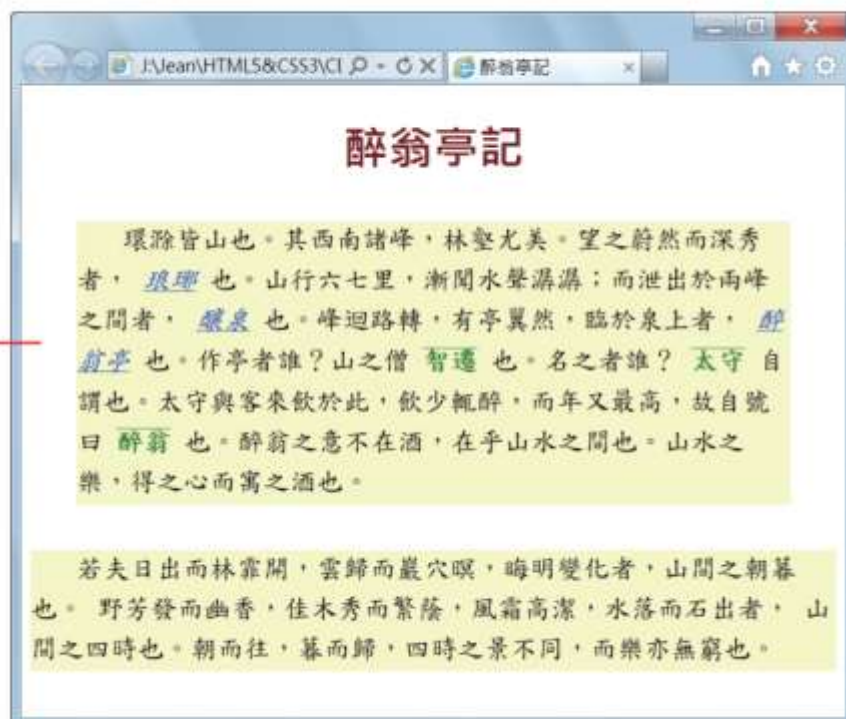
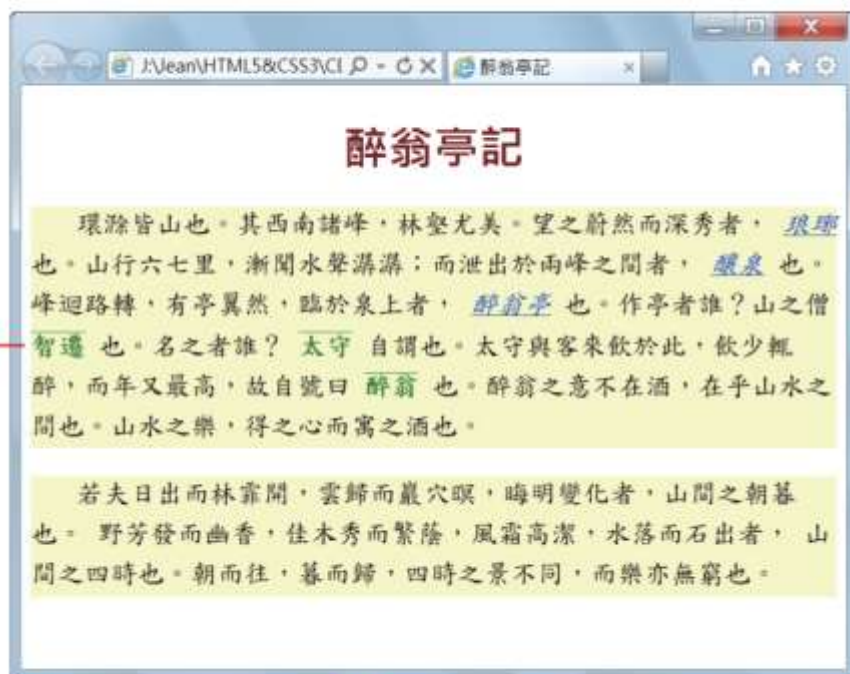
margin-bottom:5px



- Margin edge
- Border edge
- - - Padding edge
- Content edge

邊界

```
<p style="margin:1cm">
```



a 預設的邊界 **b** 上下左右邊界均為 1 公分

padding 相關屬性

- padding: 設定留白，可搭配 left, right, top, bottom 使用。

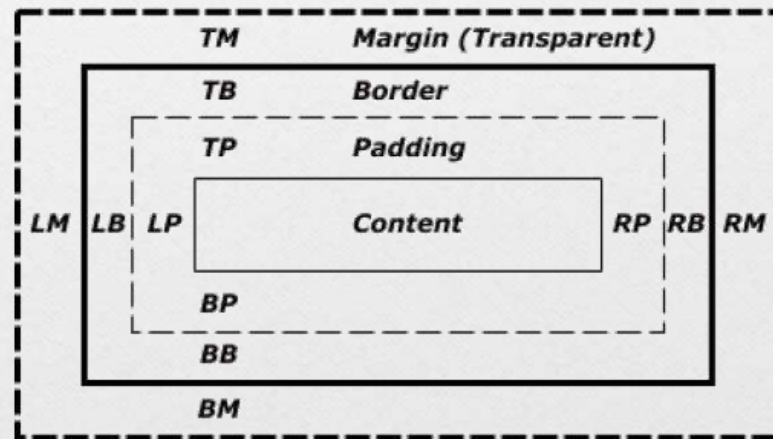
padding:5px

padding-top:5px

padding-left:3px

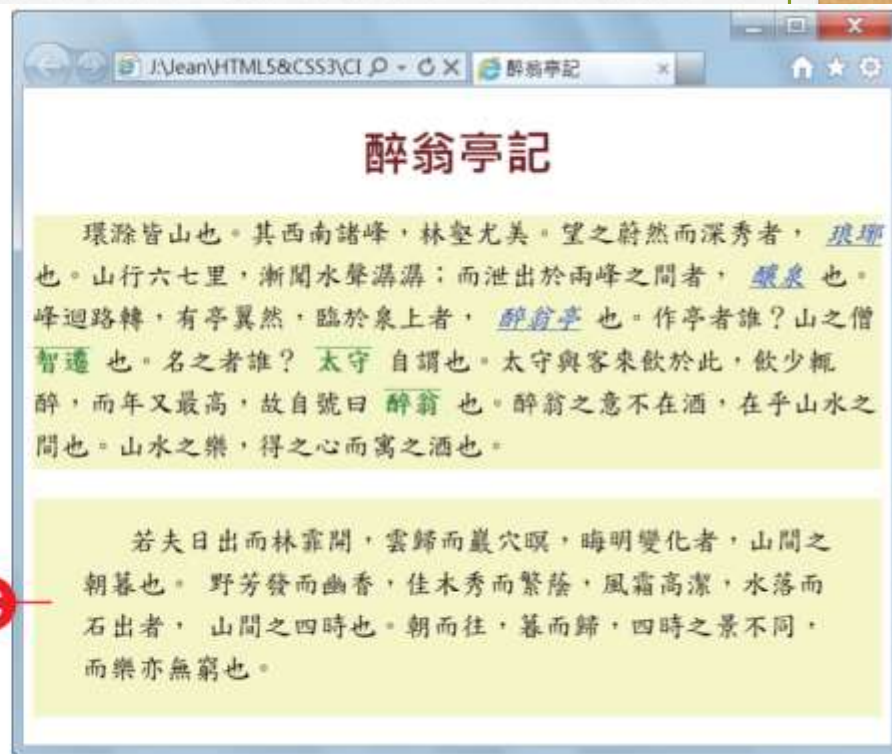
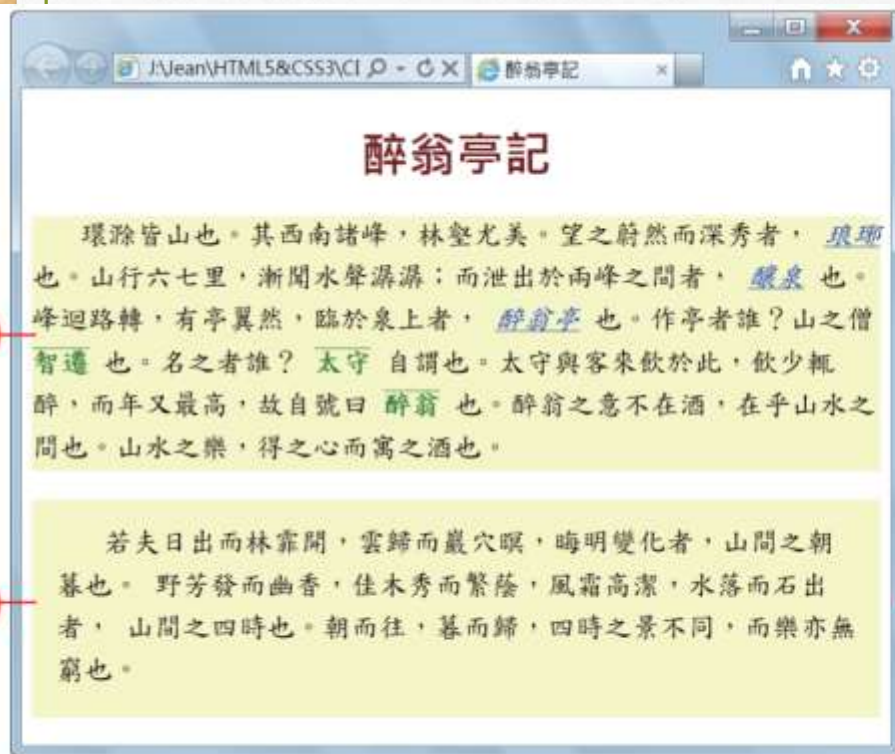
padding-right:10px

padding-bottom:5px



- Margin edge
- Border edge
- - - Padding edge
- Content edge

留白屬性



a 預設的留白 **b** 上下左右留白均為 0.5 公分 **c** 上下留白為 0.5 公分，左右留白為 1 公分

border 相關屬性

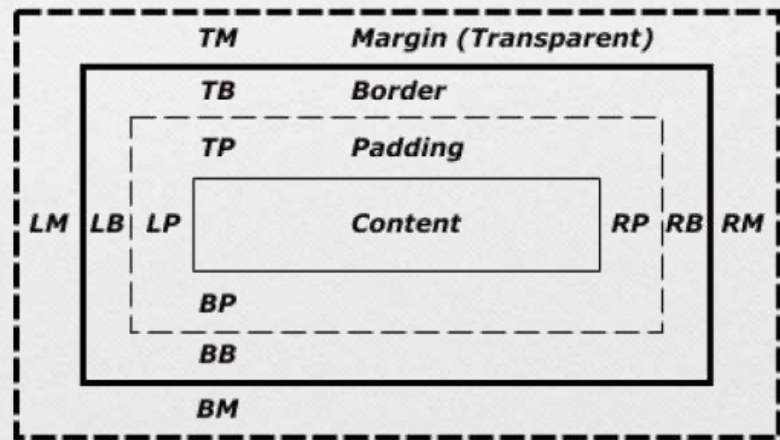
- **border**:設定邊框。此屬性的內容較複雜，有三個部分，邊框粗細(**width**)、邊框樣式(**style**)，邊框顏色(**color**)。

border:1px solid #ff0000

border-width:1px

border-style:solid

border-color:#ff0000



- Margin edge
- Border edge
- Padding edge
- Content edge

border-style (框線樣式)

```
01:<!doctype html>
02:<html>
03:  <head>
04:    <meta charset="utf-8">
05:    <title> 示範框線屬性 </title>
06:  </head>
07:  <body>
08:    
09:  </body>
10:</html>
```

- border-top-style
- border-bottom-style
- border-left-style
- border-right-style
- border-style



border-color (框線色彩)

- 我們可以使用下列屬性指定HTML 元素的框線色彩：
- border-top-color
- border-bottom-color
- border-left-color
- border-right-color
- border-color



```
<body>
```

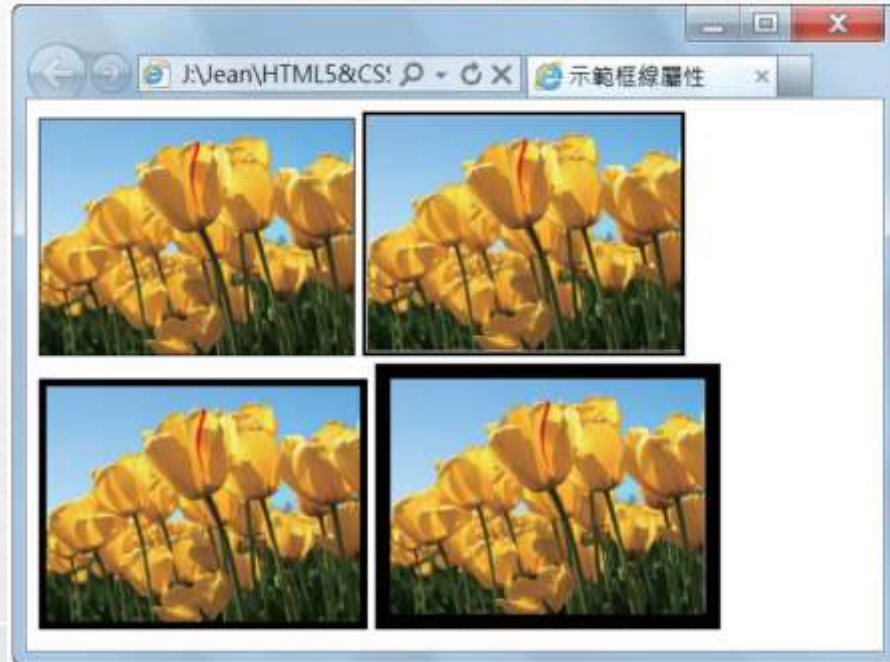
```
  <h1 style="border-style:double; border-color:green">美麗的鬱金香 </h1>
```

```
  
```

```
</body>
```

border-width (框線寬度)

- 我們可以使用下列屬性指定HTML元素的框線寬度：
- border-top-width
- border-bottom-width
- border-left-width
- border-right-width
- border-width



```
<body>
```

```

```

```
<br>
```

```

```

```

```

```
</body>
```

單位

- **px** : 1px = 螢幕上的像素高
- **em** : 1em = 當前字型大小
- **%** : 父元素值的百分比

HTML ▾

```
<div style="width: 300px;">width: 300px</div>  
<div style="height: 50%">height: 50%</div>  
<div style="font-size: 120%">font-size: 120%</div>  
<div style="width: 10em;">國國國國國國國國國國</div>
```

width: 300px

height: 50%

font-size: 120%

國國國國國國國國國國

Auto-run JS

Run with JS

小提醒

- 屬性中如果有關長度、寬度、高度、或者是粗細的資料，請務必加上單位。
- padding:5px 正確，padding:5 不正確。
- 某些瀏覽器接受不加單位，會預設為 px。但在 Firefox 則會出現錯誤。

課堂練習

- 請利用 CSS 樣式，搭配表格，做出以下畫面中的效果。

美國職棒大聯盟，美聯東區戰績

球隊	勝	負
紐約洋基	10	8
波士頓紅襪	9	8
巴爾狄摩金鶯	5	12

CSS Display

- 每個 HTML 標籤都會有一個顯示模式，在 CSS 中可用 `display` 屬性控制，這個屬性主要分為兩種，`block` 和 `inline`。
- `display:block` 標籤區塊會佔據整行的空間。會自動換行。
- `display:inline` 標籤區塊在行內，不會自動換行

display Property

- `<p>` 的 box 呈上下排列
- `` 的 box 排在字裡行間

和 display 有關

```
<p>  
This is a paragraph with  
<em>emphasized words</em>  
embedded in it.  
</p>  
<p>  
And another...
```

This is a paragraph
with emphasized
words embedded
in it.
And another...

- 預設 **display: block;**
`<p>` , `<h1>`~`<h6>` , `<div>`
- 預設 **display: inline;**
`` , `` , ``

CSS Display

- 每個 HTML 標籤都有預設的顯示模式。常見的如下。
- 預設 **block**: `<div>`、`<p>`、`<h1>` ~ `<h6>`、`<table>`、`<form>` (自動換行)
- 預設 **inline**: ``、`<a>`、``、``、`<i>`、`<u>` (不會自動換行)
- 無論預設值是甚麼，每個標籤都可以透過 CSS 強制修改成你想要的顯示模式。

其他 display 值

- none: 連 box 都沒有。完全沒有容身之處。
- table 系列：當成 table 來排版，見 [SPEC](#)。

HTML ▾

```
<!DOCTYPE html>
<div class="none">
  Totally not visible.
</div>
```

```
<div class="table">
  <div class="tr">
    <div class="td">
      1
    </div>
    <div class="td">
      2
    </div>
  </div>
```

```
<div class="tr">
  <div class="td">
    3
  </div>
  <div class="td">
    4
  </div>
</div>
</div>
```

CSS ▾

```
/*.none{ display: none; }*/
/*.table{ display: table; }
.tr{ display: table-row; }
.td{ display: table-cell; }*/
```

CSS ▾

```
/*.none{ display: none; }*/
.table{ display: table; }
.tr{ display: table-row; }
.td{ display: table-cell; }
```

CSS ▾ Saved

```
.none{ display: none; }
.table{ display: table; }
.tr{ display: table-row; }
.td{ display: table-cell; }
```

Output

Totally not visible.

1
2
3
4

Output

Totally not visible.

12
34

Output

12
34

border 與 outline

- **border** 會佔有空間、推開周圍元素；**outline** 不會。

HTML ▾

```
<!DOCTYPE html>
<div>
  Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.
  <span class="heart">Heart</span>
  Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.
</div>
```

CSS ▾

```
.heart{
  display: inline-block;
  border : 15px solid red;
  background: pink;
  width: 150px; height: 50px;
  font-size: 20px;
  margin: 20px;
  padding: 10px;
}
```

Output Run with JS Auto-run JS

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.

Heart Duis aute

irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

width 和 height

- 可利用 width 和 height 屬性設定標籤作用區塊的寬高。
- 只有顯示模式為 **block** 的標籤才有用。
(此規則為標準作法，某些瀏覽器允許顯示模式為 inline 的標籤設定寬高，但不建議使用)

width:100px;height:200px

width、height、top、bottom、left、right
(Block Box 的寬度與高度、上下左右位移量)

```
<h1 style="background-color:burlywood; color:white; width:300px; height:200px">  
木柵動物園Taipei Zoo</h1>
```



a 當視窗縮小時，區塊的寬度不會自動變小 **b** 當視窗放大時，區塊的寬度也不會自動變大

max-width、min-width、max-height、min-height
(Block Box 的寬度與高度的最大值及最小值)

```
<h1 style="background-color:lightgreen; color:white; min-width:400px; max-width:600px">  
木柵動物園Taipei Zoo</h1>
```



- a** 當視窗縮小時，區塊的寬度仍不得小於寬度的最小值，故文字不會折到下一行
- b** 當視窗放大時，區塊的寬度亦不得大於寬度的最大值

利用 CSS 為標籤區塊定位

- 每個 HTML 標籤都佔有一個區塊，如何為這些區塊排版呢？
- 最常見的就是按照 HTML 文件中標籤出現的**先後順序**由上到下、由左至右排。
- 而我們**還有其他**方式。

利用 CSS 為標籤區塊定位

- 我們可以利用 CSS 的 **position** 屬性指定標籤的定位模式。有以下幾種定位模式可以選擇。
- ◎ 其語法如下：

```
position:static | relative | absolute | fixed
```

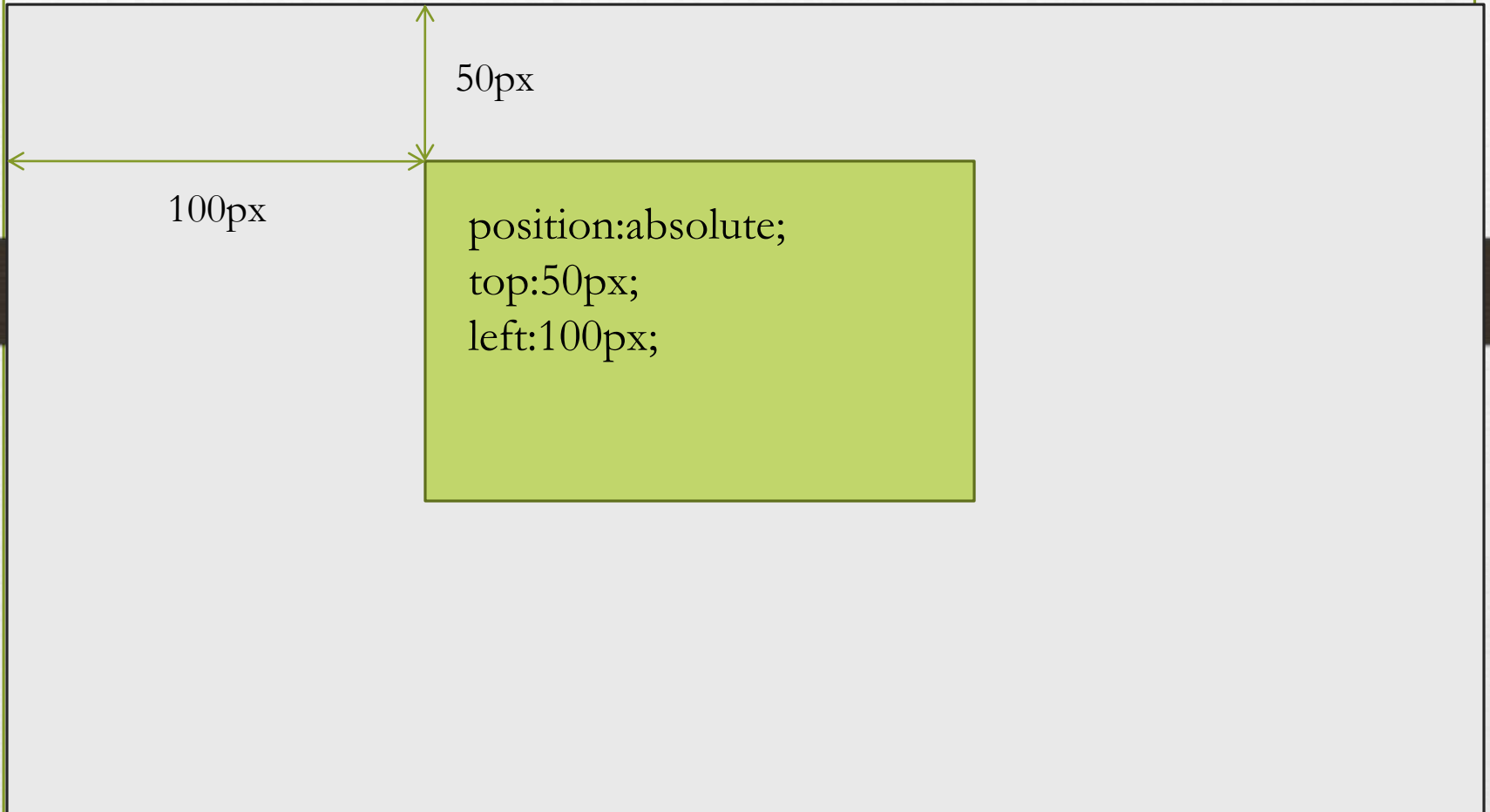
- static: **預設**的定位模式 (由上到下、由左到右)
- relative: **相對**於預設的位置，作位移。
- absolute: **絕對定位**，直接指定標籤區塊出現的座標位置。

使用絕對定位

- 將 position 設定為 absolute，並且利用 top, left 屬性指定座標位置

```
position: absolute; top: 50px; left: 100px
```

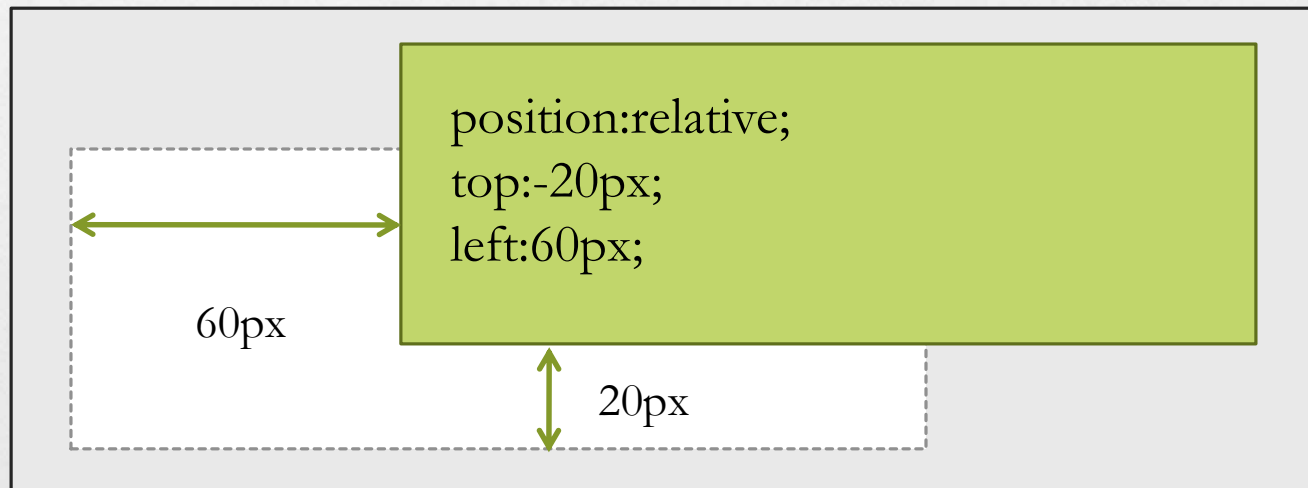
使用絕對定位



使用相對定位

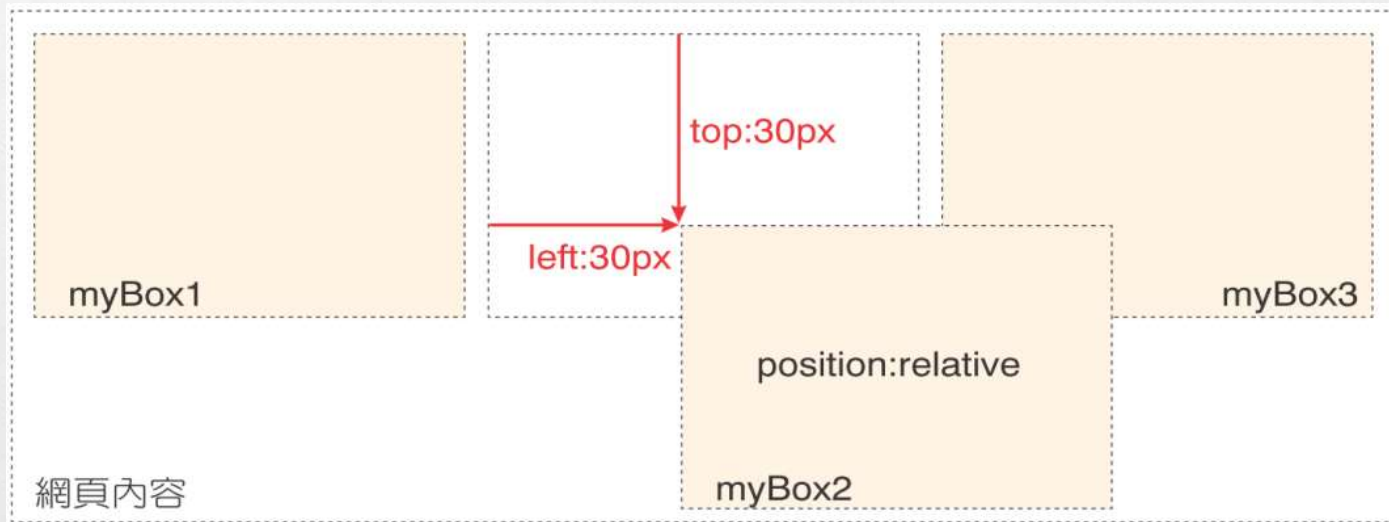
- 將 position 設定為 relative，並且利用 top, left 屬性指定與原先位置的差距。

Position: **relative**; **top**: -20px; **left**: 60px



使用相對定位

```
#myBox2 {  
  position:relative;  
  top:30px;  
  left:30px;  
}
```



實機操作

- 練習使用絕對定位及相對定位功能。

z-index (重疊順序)



標題 1 區塊的 z-index 屬性比較大，故會重疊在圖片上面。

```
<!doctype html>
<html>
  <head>
    <meta charset="utf-8">
    <title> 示範定位方式 </title>
  </head>
  <body>
    
    <h1 style="background-color:rgba(255, 255, 0, 0.3); width:472px;
      position:absolute; top:100px; left:10px; z-index:2"> 豪斯登堡 </h1>
  </body>
</html>
```

設定標籤區塊的背景

- background-color: 設定背景顏色
- background-image: 設定背景圖片

background-image:url("bg.gif");

- Background-position: 設定背景圖片位置，先設定上下對齊、再設定左右。

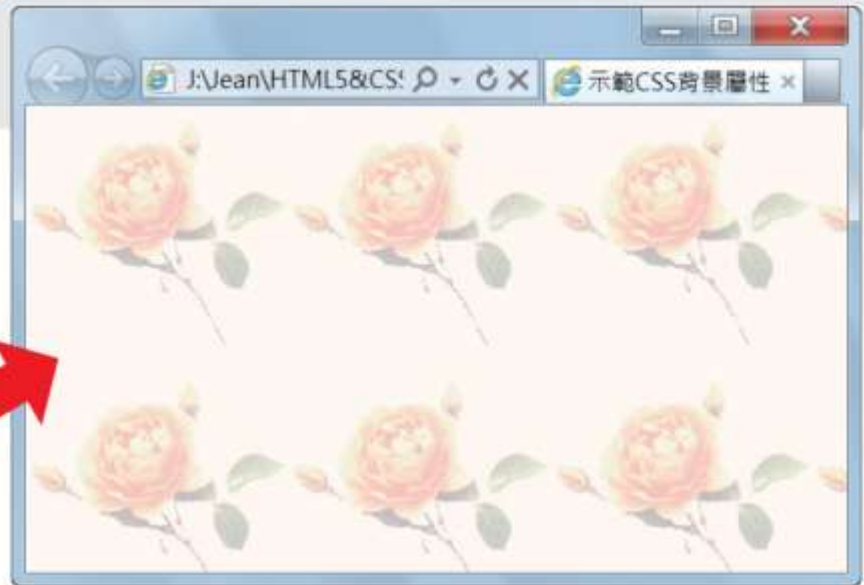
background-position:top left;

background-position:top center;

background-position:bottom left;

背景圖片

```
01:<!doctype html>
02:<html>
03:  <head>
04:    <meta charset="utf-8">
05:    <title> 示範 CSS 背景屬性 </title>
06:  </head>
07:  <body style="background-image:url (rose.gif) ">
08:  </body>
09:</html>
```



設定標籤區塊的背景

- `background-repeat`: 設定重複填滿模式，圖片比區塊小時，可以設定如何用現有的圖片填滿區塊

background-repeat: no-repeat

background-repeat: repeat-x

background-repeat: repeat-y

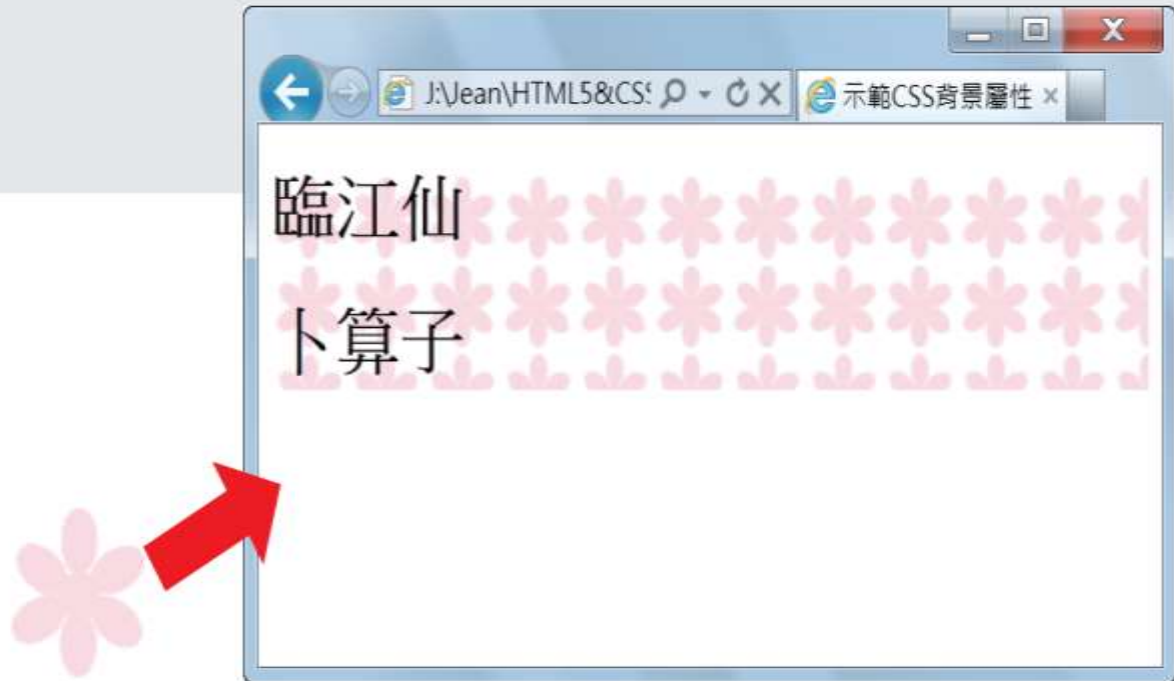
background-repeat: repeat

- `background-attachment`: 設定圖片是否固定

background-attachment: scroll / fixed

background-repeat (背景圖片重複排列方式)

```
<div style="background-image:url (flower.gif) ; background-repeat:repeat" >  
  <h1> 臨江仙 </h1>  
  <h1> 卜算子 </h1>  
</div>
```



background-attachment (背景圖片是否隨內容捲動)

```
<body>
```

```
<pre style="background-image:url(bg06.gif); background-repeat:no-repeat;  
background-position:right top; background-attachment:fixed">
```

泥娃娃，泥娃娃，一個泥娃娃，
她有那眉毛，也有那眼睛，
眼睛不會眨。

泥娃娃，泥娃娃，一個泥娃娃，
她有那鼻子，也有那嘴巴，
嘴巴不說話。

她是個假娃娃，不是個真娃娃。
她沒有親愛的爸爸，也沒有媽媽。

泥娃娃，泥娃娃，一個泥娃娃。
我做她爸爸，我做她媽媽，
永遠愛著她。

—「泥娃娃」

```
</pre>
```

```
</body>
```



① 將內容向下捲動 ② 背景圖片依然顯示在右上方

設定標籤區塊的背景

- 利用 background 屬性一次指定所有跟背景相關的屬性。

- background:

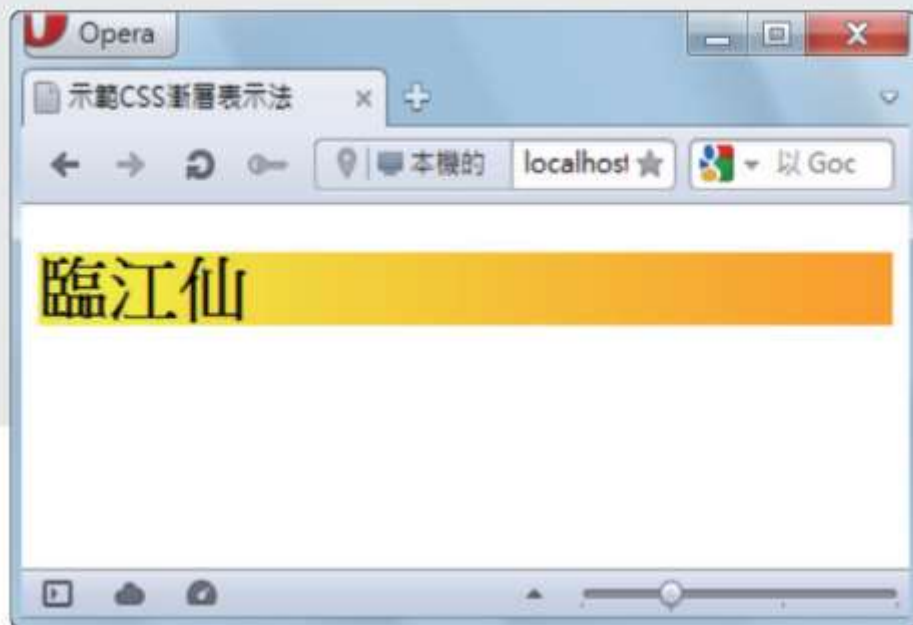
color image repeat attachment position

```
background:#ffffff url("bg.gif") repeat-x scroll top left;
```

漸層表示法

- CSS 3 新增下列漸層表示法 (gradient notation) :
- `linear-gradient()` : 指定線性漸層。
- `radial-gradient()` : 指定放射狀漸層。
- `repeating-linear-gradient()` : 指定重複線性漸層。
- `repeating-radial-gradient()` : 指定重複放射狀漸層。

```
01:<!doctype html>
02:<html>
03:  <head>
04:    <meta charset="utf-8">
05:    <title> 示範 CSS 漸層表示法 </title>
06:    <style>
07:      h1 {background:-o-linear-gradient(0deg, yellow, orange);          /* Opera11.10+ */
08:          background:-moz-linear-gradient(0deg, yellow, orange);      /* FireFox 3.6+ */
09:          background:-webkit-linear-gradient(0deg, yellow, orange);   /* Webkit */
10:          -ms-filter:"progid:DXImageTransform.Microsoft.gradient(startColorstr='yellow',
11:                           endColorstr='orange', GradientType=1)";    /* IE8+ */
12:          background:linear-gradient(0deg, yellow, orange);           /* 標準寫法 */
13:    }
14:  </style>
15:  <body>
16:    <h1> 臨江仙 </h1>
17:  </body>
18:</html>
```



課堂練習

- 設定背景圖片與漸層練習

linear-gradient(to bottom, #xxxxxx 0%,#xxxxxx50%,#xxxxxx 51%,#xxxxxx 100%);

美國職棒大聯盟，美聯東區戰績

球隊	勝	負
坦帕灣光芒	12	6
紐約洋基	10	8
波士頓紅襪	9	8
巴爾狄摩金鶯	5	12

object-fit (配合Box 調整物件大小)

```

```

```

```

```

```



fill



contain

引入外部的 CSS 樣式表

- 實務上，我們經常把 CSS 的程式碼從原本的 HTML 網頁中分開。獨立成一個新的檔案。
- 使用 `<link>` 這個標籤，一般會放在 `<head>` 標籤的內文中，作法如下。

`<link rel="stylesheet" type="text/css" href="CSS檔案的位址" >`

```
<head>
...
<link rel="stylesheet" type="text/css" href="path/to/file.css">
...
</head>
```



```
body {  
  color:white;  
  background:purple  
}
```

body.css

```
<!doctype html>  
<html>  
  <head>  
    <meta charset="utf-8">  
    <title>新網頁 1</title>  
    <style>  
      @import url("body.css");  
    </style>  
  </head>  
  <body>  
    <h1>歡迎光臨！ </h1>  
  </body>  
</html>
```

linkcss3.html

課堂練習

- 練習樣式選擇器
- 將 CSS 樣式表分開
- 引入外部的 CSS 樣式表

CSS 看起來怪怪的嗎？

- 使用

Ctrl + Shift + i

The screenshot shows the browser's developer tools interface. The 'Styles' panel is open, displaying the following CSS rules:

```
element.style {  
}  
  
body {  
  background: purple;  
}  
  
body {  
  display: block;  
  margin: 8px;  
}
```

The 'background' property is checked and highlighted with a purple color swatch. The 'margin' property is highlighted with a value of 8px. The 'user agent stylesheet' is also visible.

On the left, the 'Elements' panel shows the HTML structure, with the <body> element selected. The HTML includes a script, a table with cellpadding="3", and another script.

On the right, a diagram illustrates the box model with the following values:

- margin: 8
- border: -
- padding: -
- Content area: 1243 x 637,438

The bottom of the screenshot shows the breadcrumb 'html > body' and a search bar with the text 'Find in Styles'.

Chrome DevTool Elements 頁籤

- Styles

- Computed Styles

```
Styles Computed Event Listeners DOM Breakpoints Properties »
element.style {
}
.page-index.chrome.chrome-32.mac.extra- inspector-stylesheet:1
large-window {
}
.extra-large-window, .large-window {
}
body {
  margin: 0;
  position: relative;
  z-index: 0;
  background-color: #f0f0f0;
  overflow-y: auto;
}
html, body, input, select, textarea, button {
  font-size: 14px;
  line-height: 18px;
  color: #393939;
}
```

- 根據 selector 分類規則
- 點兩下可修改

```
Styles Computed Event Listeners DOM Breakpoints Properties »
1440 x 705
-webkit-background-clip: border-box;
-webkit-background-origin: padding-box;
-webkit-background-size: auto;
background-attachment: scroll;
background-clip: border-box;
background-color: rgb(240, 240, 240);
background-image: none;
background-origin: padding-box;
background-size: auto;
color: rgb(57, 57, 57);
display: block;
font-family: 'Helvetica Neue', Arial, Helvetica, sans-serif;
font-size: 14px;
html, body, input, select, textarea, button - 14px
html - 100%
html, body, input, select, textarea, button - 14px
font-style: normal;
```

- 根據 CSS property 分類，方便追蹤。

特殊值 inherit

所有 CSS 屬性，均可設為「inherit」。

- 部分 CSS 屬性預設不會繼承自父元素，明確指定 inherit 才會繼承。

HTML ▾

```
<p>Paragraph <span>Span</span></p>
```

CSS ▾

```
p{  
  color: red;  
  font-size: 32px;  
  border: 1px solid blue;  
}  
span{  
  /*border: inherit;*/  
}
```

Paragraph Span

HTML ▾

```
<p>Paragraph <span>Span</span></p>
```

CSS ▾

```
p{  
  color: red;  
  font-size: 32px;  
  border: 1px solid blue;  
}  
span{  
  border: inherit;  
}
```

Paragraph Span

特殊值 inherit (續)

- `input` , `textarea` , `select` , `button` 等 `replaced element` , 樣式不會自動 `inherit` 自父元素。

HTML ▾

```
<button>Button</button> and body text.
```

CSS ▾

```
body /*, button */{  
  color: purple;  
}  
/* button{  
  color: inherit;  
}*/
```

Button and body text.

HTML ▾

```
<button>Button</button> and body text.
```

CSS ▾

```
body /*, button */{  
  color: purple;  
}  
button{  
  color: inherit;  
}
```

Button and body text.

總整理

tag, .class, #id

Selector

h1

Declaration

{color:blue; font-size:12px;}

Declaration

Property

Value

Property

Value

HTML

```
<div>I am div!</div>
<div class="italic alert">I am
a bold div!</div>

<p class="italic">I am also
bold</p>

<span id="uline"
class="italic">I am a bold span
with underline</span>
```

CSS

```
div { /* div element */
  color: blue;
}
.italic { /* Elements with class "italic" */
  font-style: italic;
}
.alert { /* Elements with class "alert" */
  border: 1px solid red;
}
#uline { /* Elements with ID = u */
  text-decoration: underline;
}
div.italic { /* div elements with class "italic" */
  font-size: small;
}
```

Output

Run with JS

Auto-run JS

I am div!

I am a bold div!

I am also bold

I am a bold span with underline

進階版

tag, .class, #id

- div.tall
 - 有 tall class 之 div
- #target.highlight
 - 有 highlight class 且 ID 為 target 之 element
- .box.widget
 - 同時有 box 和 widget 兩個 class 之 element

多個 Selector 套用相同樣式

- 用逗號，將 Selector 分開

```
<!DOCTYPE html>
<html>
<head>
<meta charset=utf-8 />
<title>JS Bin</title>
</head>
<body>
  <button class="btn-primary">
    OK
  </button>
  <button class="btn-inverse">
    Cancel
  </button>
</body>
</html>
```

```
CSS
.btn-primary, .btn-inverse{
  font-size: 16px; padding: 0.5em 1em;
  outline: 0;
}
.btn-primary{
  background: #ccc; color: #333;
}
.btn-inverse{
  background: #333; color: #ccc;
}
body{
  padding-top: 3em;
}
```

OK

Cancel

Auto-run JS

Run with JS

Descendant & Child Selector

- `.root div` : `.root` 的子孫中為 `div` 者。
- `.root>div` : `.root` 的孩子裡為 `div` 者。

```
CSS ▼  
div{ padding: 3px; }  
.root>div{  
  /* .root div{ */  
  border: 1px solid green;  
}
```

HTML ▼

```
<div class="root">  
  Parent div has  
  <div>Child div 1</div> and  
  <div>Child div 2, which has some  
    <div>Young kids</div>  
  </div>  
</div>
```

Parent div has

Child div 1

and

Child div 2, which has some
Young kids

Universal Selector

- * : 任意元素。
- .root *: .root 的子孫。

CSS ▾ Saved

```
div{ padding: 3px; }  
.root *{  
  border: 1px solid green;  
}
```

HTML ▾

```
<div class="root">  
  Parent div has  
  <div>Child div 1</div> and  
  <div>Child div 2, which has some  
    <div>Young kids</div>  
    <p>Another <span>kid</span></p>  
    <span>Third kid</span>  
  </div>  
</div>
```

Parent div has

Child div 1

and

Child div 2, which has some

Young kids

Another kid

Third kid

Attribute Selector

中括號； [屬性 運算子 值]

- https://developer.mozilla.org/en-US/docs/Web/CSS/Attribute_selectors

HTML ▾

```
Login: <input type="text" name="txtName">
<br>
```

```
Password: <input type="password"
name="txtPassword"><br>
```

```
<input type="checkbox"
name="optCheckbox">Remember Me
```

Login:

Password:

Remember Me

CSS ▾

```
input[type=text]{
  border: 1px solid blue;
}
[name^=txt]{
  /* 'name' starting with 'txt' */
  background: pink;
}
```


Pseudo-Class

依狀態套用的「假的」class

- 幾乎所有元素都可用 :hover
- a 元素 : :link, :visited, :active
- 避免定義相互覆蓋的順序 : LoVe HAte

HTML ▾

```
<a href="html.html">HTML  
tutorial</a>
```

CSS ▾

```
a:link{  
  text-decoration: none;  
  color: blue;  
}  
a:visited{ color: blue; }  
a:hover{  
  text-decoration: underline;  
  color: red;  
}  
a:active{ color: yellow; }
```

HTML tutorial

Auto-run JS

Run with JS

Pseudo-Class

HTML ▾

```
<table>
  <tr>
    <td>Elephant</td>
    <td>3kg</td>
  </tr>
  <tr>
    <td>Eggplant</td>
    <td>300g</td>
  </tr>
  <tr>
    <td>Eggplant</td>
    <td>300g</td>
  </tr>
  <tr>
    <td>Eggplant</td>
    <td>300g</td>
  </tr>
  <tr>
    <td>Earplug</td>
    <td>10g</td>
  </tr>
</table>
```

CSS ▾

```
tr{
  background: #ccc;
}
tr:nth-child(4) {
  background: #0f0;
}
tr:first-child {
  background: #0ff;
}
tr:last-child {
  background: #ff0;
}
```

Output

Elephant	3kg
Eggplant	300g
Eggplant	300g
Eggplant	300g
Earplug	10g

- 老大 : `first-child`
- 老么 : `last-child`
- 老 N : `nth-child()`

Pseudo-Element

HTML ▾

```
<blockquote>臨投影片涕泣，不知  
講師所云。</blockquote>
```

CSS ▾

```
blockquote::before{  
  content: '“';  
}  
blockquote::after{  
  content: '”';  
}
```

“臨投影片涕泣，不知
講師所云。”

- 需有 **content** 屬性。
- **::before** 像是一個 span 元素
加在 start tag 之後、內容之前。
- **::after** 像是一個 span 元素
加在 end tag 之前、內容之後。
- 亦可只用一個冒號（:before, :after）

Sibling Selector

- `:checked` : 「被勾選」之 pseudo-class ◦

HTML ▾

```
<input type="checkbox" id="tos">
<label for="tos">I accept term of service.
</label>
<input type="submit" value="Submit">
```

I accept term of service.

I accept term of service.

CSS ▾

```
input:checked + label{
  color: green;
}
input[type=submit]{
  opacity: 0.3;
}
input:checked ~
input[type=submit]{
  opacity: 1;
}
```

- `input + *` 選出 `input` 的最大的弟弟妹妹
- `input ~ *` 選出 `input` 的所有弟弟妹妹

小結

- Basic Selector
 - tag, #id, .class, [attr], *
- Pseudo-elements & Pseudo-classes
 - :before, :hover
- Combinator
 - A B, A>B, A+B, A~B
- [MDN Reference](#)

兩個定義打架

- Value 不同的兩個定義，被套用在同一個元素上時.....

HTML ▾

```
<div class="loud">  
  What color is this?  
</div>
```

CSS ▾

```
.loud{  
  color: red;  
}  
div{  
  color: blue;  
}
```

What color is this?

HTML ▾

```
<div class="loud">  
  What color is this?  
</div>
```

CSS ▾

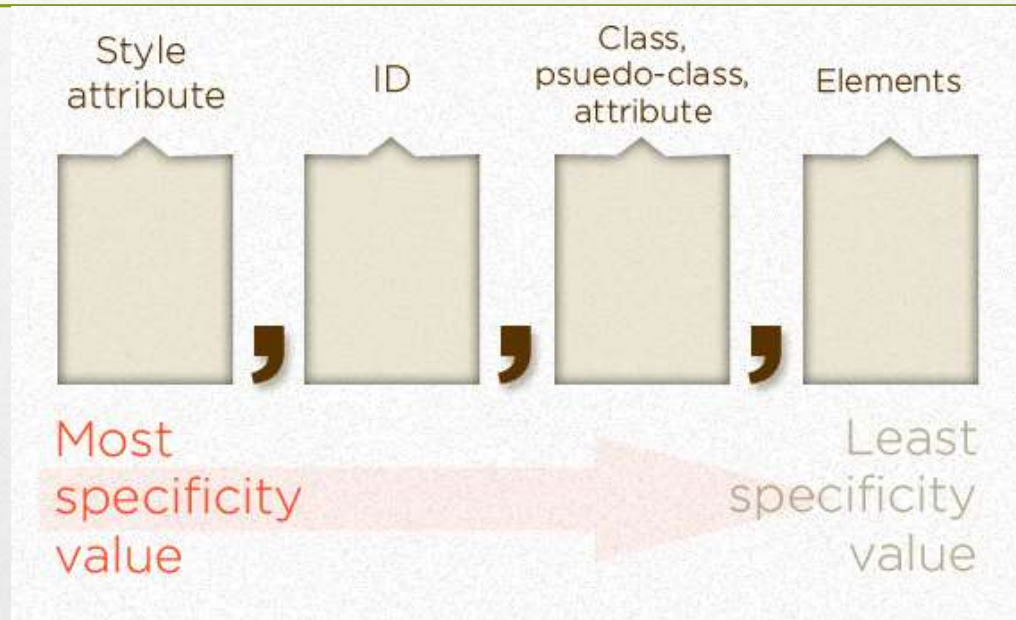
```
div{  
  color: blue;  
}  
.loud{  
  color: red;  
}
```

What color is this?

Source: [CSS Tricks](#)

CSS Specificity

Specificity : Selector 精準與否



- 用字典順序比較 Specificity 。
- Specificity 相同的話，後定義者蓋過之前定義者。

打架調解委員會

HTML ▾

```
<div class="loud">
  What color is this?
</div>
```

CSS ▾

```
.loud{
  color: red;
}
div{
  color: blue;
}
```

What color is this?

- 此 **div** 會被 selector **.loud** 和 **div** 選中
- 兩 CSS 定義內均設有 **color**
- Specificity of **.loud** : 0,0,1,0
- Specificity of **div** : 0,0,0,1
- $0,0,1,0 > 0,0,0,1$
- 故套用 **.loud** 對 **color** 的定義

Matched CSS Rules

<pre>.loud { color:  red; }</pre>	css.html:1
<pre>div { color:  blue; }</pre>	css.html:4

Takeaway

Specificity 越小越好

- 盡量用可重複使用的 class

```
.btn, .panel, .media-box
```

- ID 用於分開不重複區塊

```
#header .title
```

- Decendent / Child selector 別太深

```
#header .banner nav ul li .title /* Avoid! */
```


小練習

HTML ▾

```
<div class="loud">  
  What color is this?  
</div>
```

CSS ▾

```
.loud{  
  color: red;  
}  
div{  
  color: blue;  
}
```

What color is this?

- 請將本範例中的css檔案中加入一名為abc的id
- 並將顏色設定為綠色
- 看其文字顏色是否會變綠色

Media Queries

Media Queries 就是要在支援 CSS3 的瀏覽器中才能正常工作

Media Queries 的套用方法千變萬化

要套用在什麼樣的裝置中，都可以自己來定義

<http://mediaqueri.es/>

http://www.w3schools.com/cssref/css3_pr_mediaquery.asp

Media Query

- 給不同 device 不同的 CSS

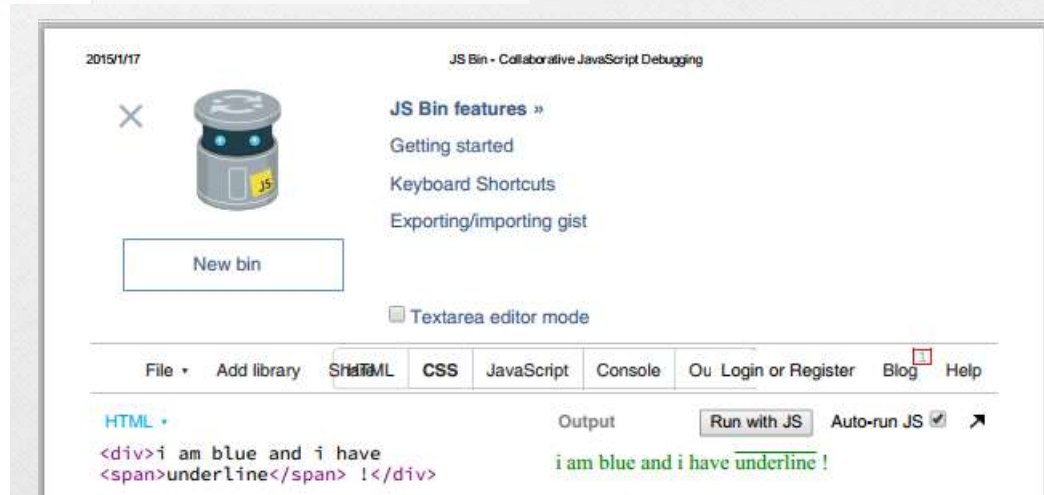
HTML ▾

```
<div>i am blue and i have  
<span>underline</span> !</div>
```

CSS ▾

```
div{  
  color: blue;  
}  
  
span{  
  text-decoration: underline;  
}  
  
@media print{  
  div{  
    color: green;  
  }  
  span{  
    text-decoration: overline;  
  }  
}
```

i am blue and i have underline !



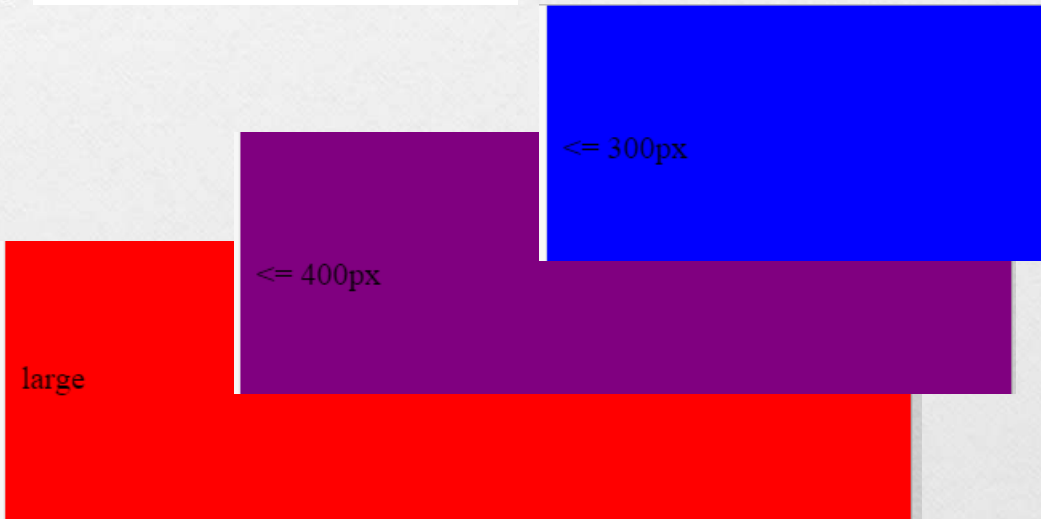
- 實例：[履歷](#)、[Web Designer Wall 精選](#)

螢幕寬度

- 「max-width: Npx」 就是 「width <= Npx」
- 「min-width: Npx」 就是 「Npx <= width」

HTML ▾

```
<div></div>
```



CSS ▾

```
body{
  background: red;
  transition-property: background-color;
  transition-duration: 0.5s;
  margin-top: 4em;
}
div:before{
  content: 'large';
}

@media screen and (max-width: 400px){
  body{
    background: purple;
  }
  div:before{
    content: '<= 400px';
  }
}

@media screen and (max-width: 300px){
  body{
    background: blue;
  }
  div:before{
    content: '<= 300px';
  }
}
```