

# 網頁設計入門

## (HTML+CSS)

童莉雯

[andytung@iii.org.tw](mailto:andytung@iii.org.tw)

# 課程大綱

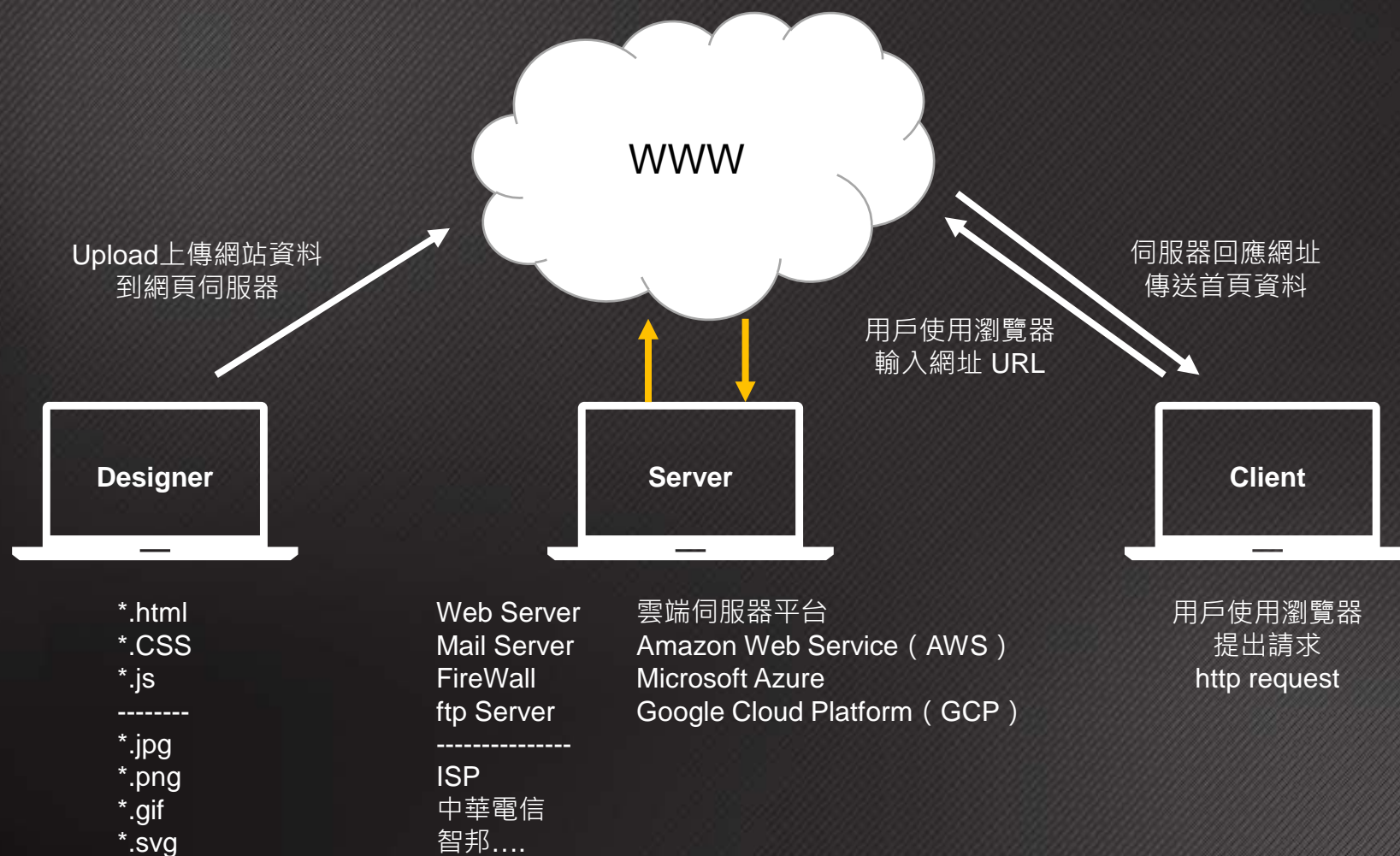
- Module 1 HTML語言基礎
- Module 2 CSS的使用和基本語法
- Module 3 表格、表單/CSS
- Module 4 HTML5
- Module 5 CSS基礎屬性
- 參考網站
  - <https://www.w3.org/html/>
  - <https://www.w3schools.com/>



# Module 1

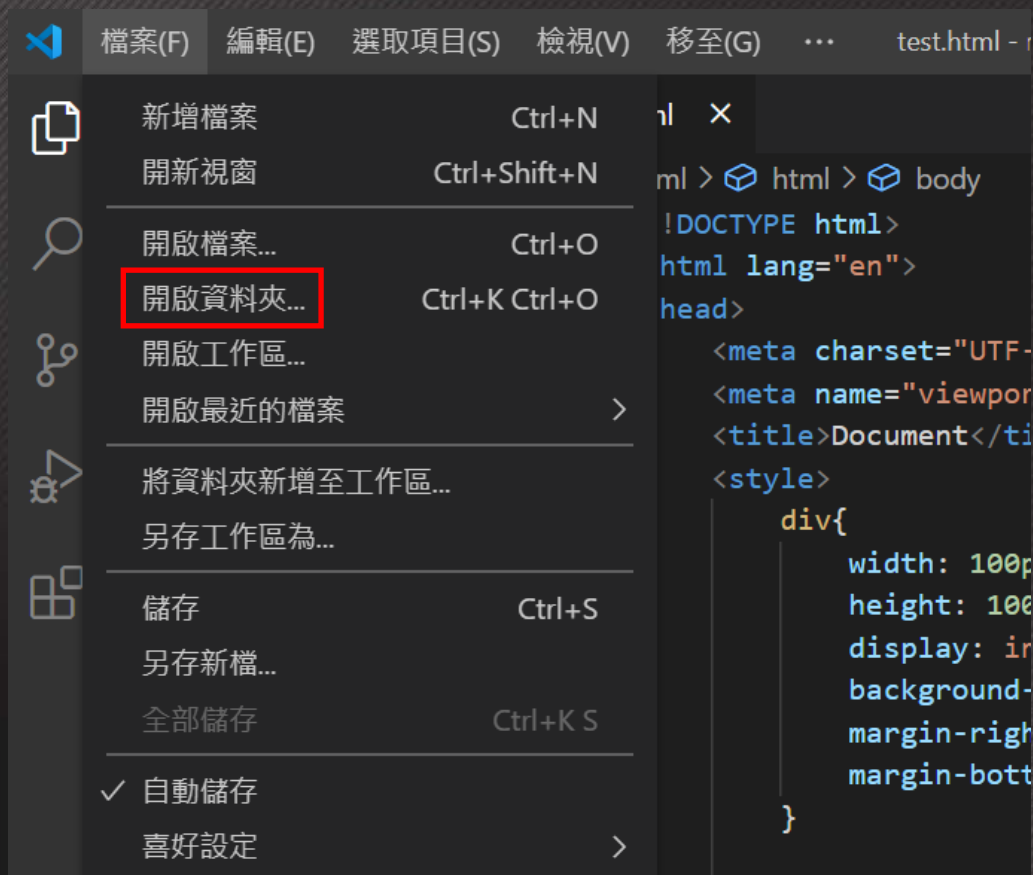
## HTML語言基礎

# 網站運作原理



# 設定本機站台

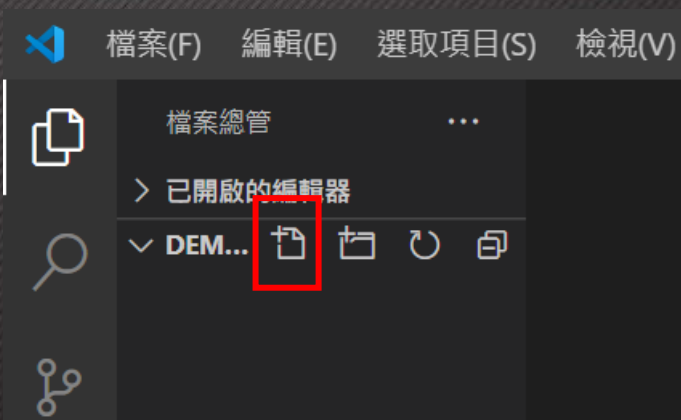
選單→檔案→開啟資料夾



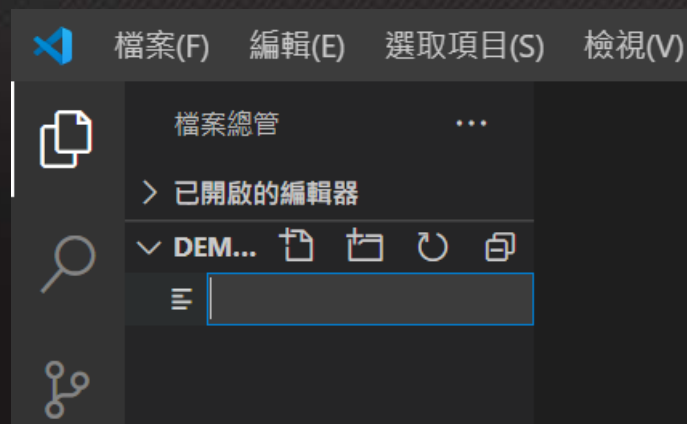


# 新增空白檔案

選單→檔案→開新檔案

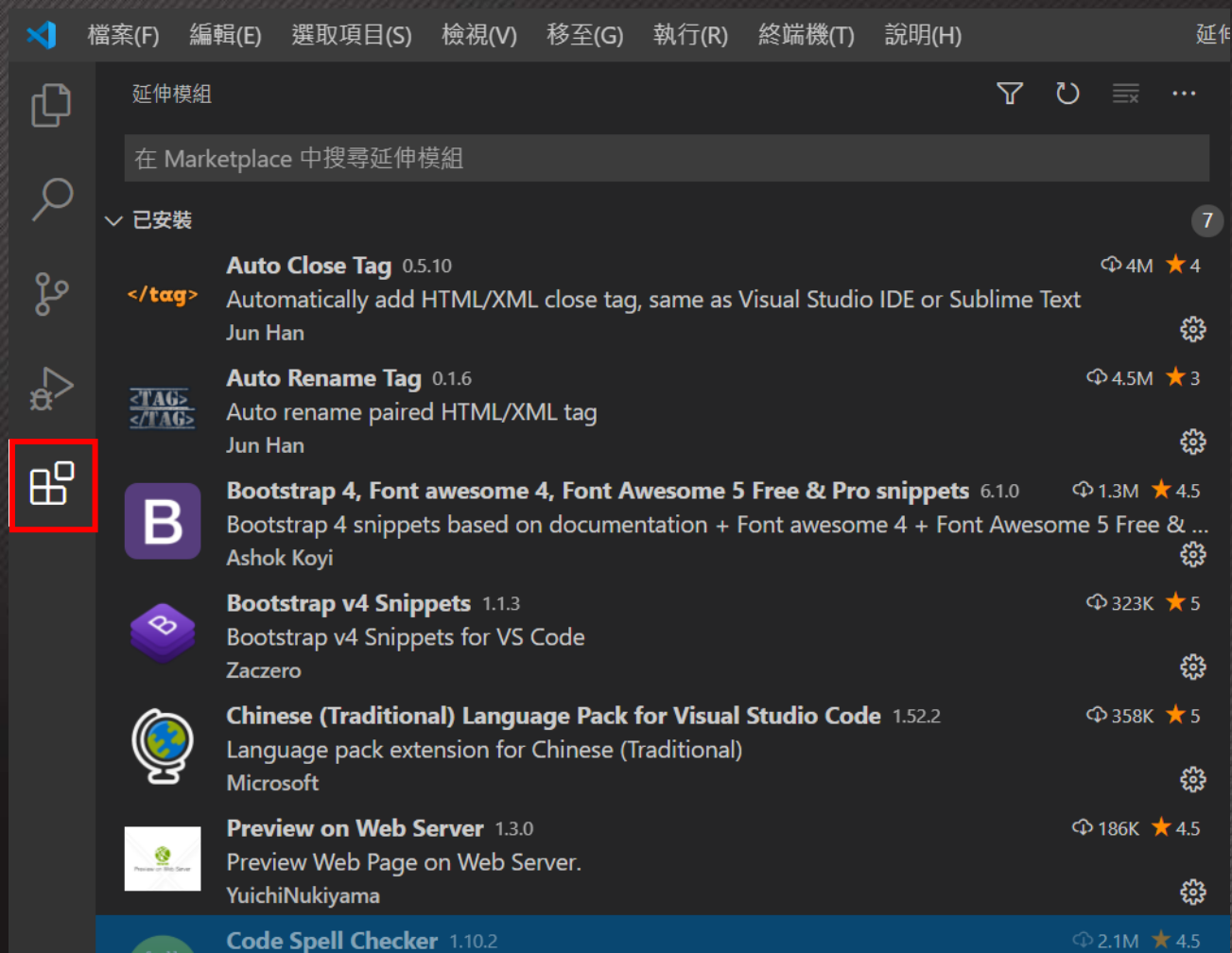


點選新增檔案

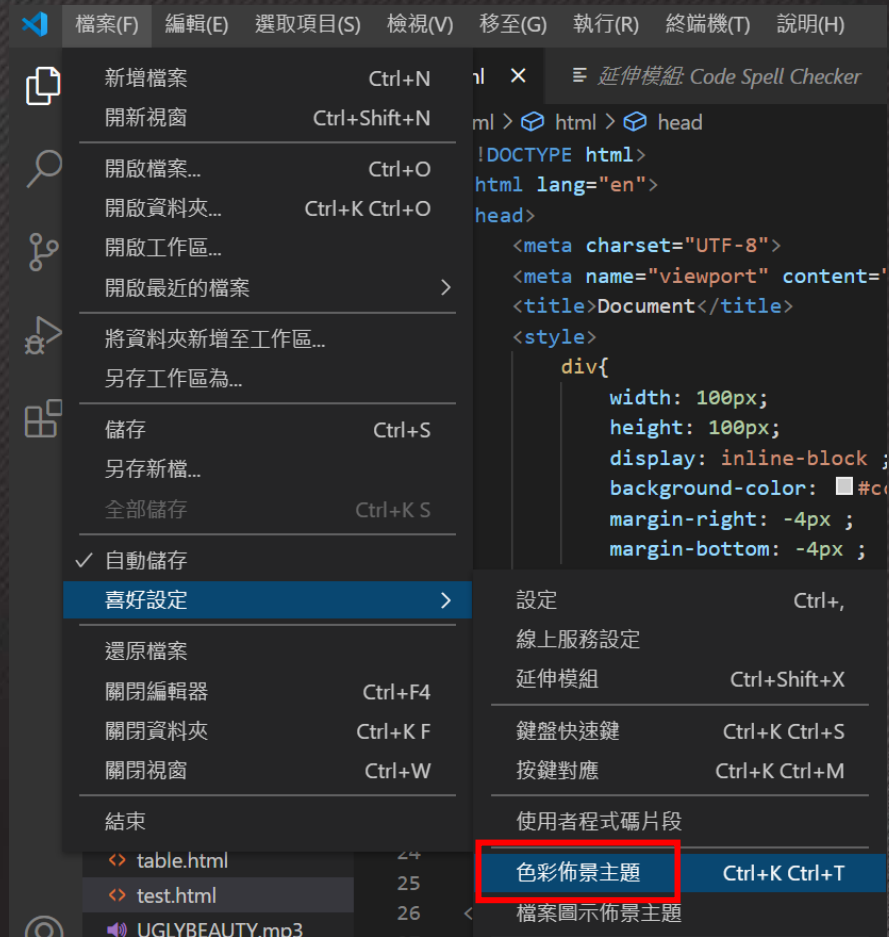
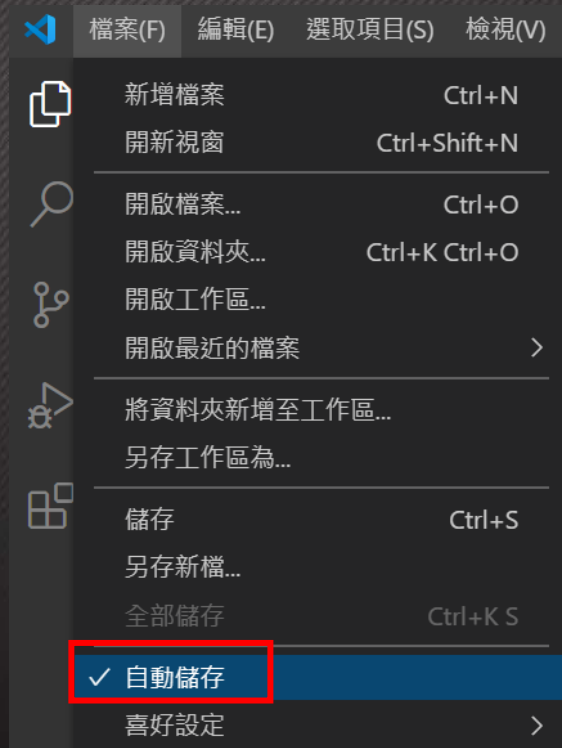


輸入 \*.html 檔名

# 設定延伸模組



# 其他





# HTML與CSS

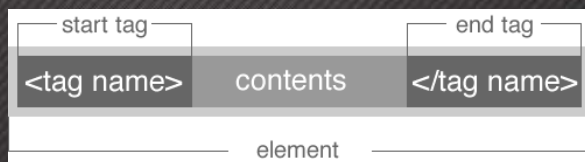
- HTML(HyperText Markup Language)
  - 網頁上的文字、圖片、超連結、表格、表單都是由HTML標籤所製作出來的
  - 每一個HTML標籤都有一個預設的樣式
- CSS(Cascading Style Sheet)
  - 用來修改HTML標籤的預設樣式

HTML標籤完成網頁資料結構

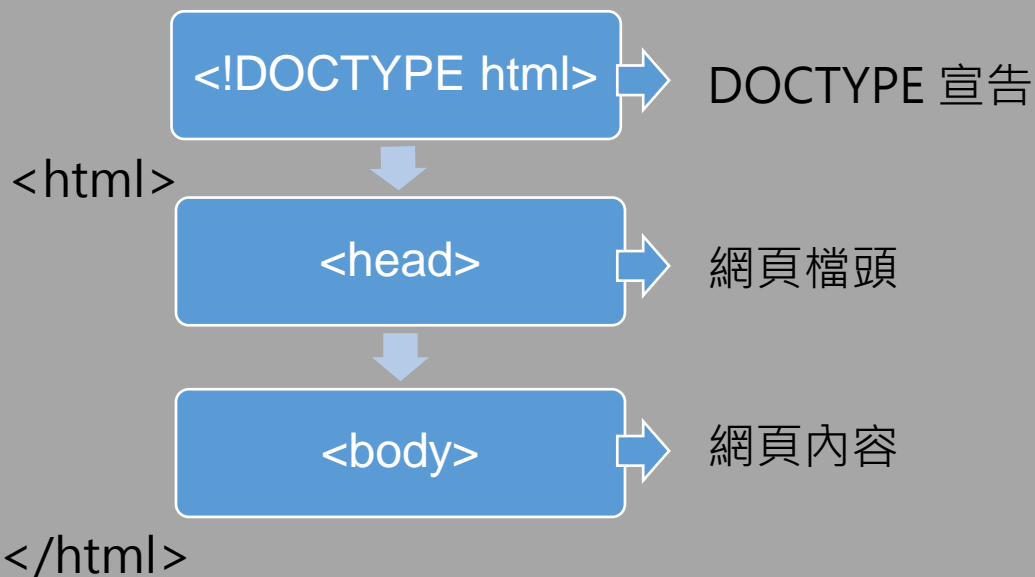
CSS視覺化標籤樣式

# HTML 標籤與結構

- 一個正確的HTML標籤必須包含「<」、「>」
- 長相如 `<h1>`



HTML 網頁分3大部分



# 什麼是 DOCTYPE

- 放在網頁的最上面，告知瀏覽器，此網頁所使用的HTML標籤是什麼版本
- HTML4時有三種版本寫法
  - 完整版或嚴格版 (Strict版)—不支援被列為負面標籤與屬性
  - 過渡版或寬鬆版(Transitional版)--支援被列為負面標籤與屬性，目前最普及的版本
  - 框架版(Frameset版)--支援框架結構標籤



# DOCTYPE 的定義方式

## HTML4

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01//EN" "http://www.w3.org/TR/html4/strict.dtd">
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
```

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.01 Frameset//EN" "http://www.w3.org/TR/html4/frameset.dtd">
```

## XHTML1.0

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Frameset//EN"  
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-frameset.dtd">
```

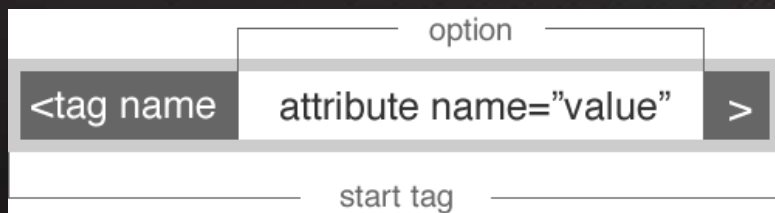
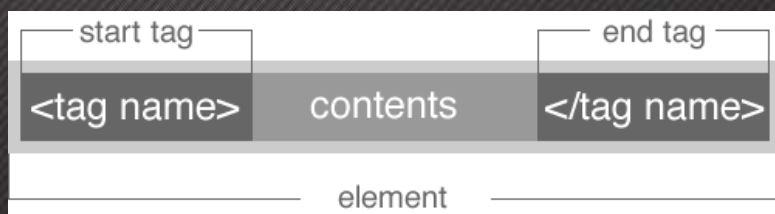
# HTML元素

- 網頁的根元素
- 內含head及body子元素
- lang屬性，指定網頁所使用的語言

```
<html lang="zh-tw">
```

```
</html>
```

# 標籤、元素、屬性與屬性值





# head 元素

- 用來放一些描述網頁的資料(metadata)
- 不會顯示在網頁上
- 內含的子元素常見有
  - title元素：文件標題，會出現在瀏覽器視窗左上方的標題區域
  - meta元素：描述HTML文件相關資訊，資訊內容由name與content屬性來決定
  - style元素：樣式表宣告
  - script元素：程式碼宣告

```
<head>  
  <meta charset="UTF-8" />  
  <title>Example  
document</title>  
  <style>CSS 寫在這裡</style>  
  <script>JS 寫在這裡</script>  
</head>
```

# meta 元素

- name屬性
  - author 、 description 、 generator 、 keywords
- http-equiv屬性
  - content-language 、 content-type 、 refresh 、 set-cookie
- charset屬性
- content屬性



# meta 元素用法

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

```
<title>HTML 教學網</title>
```

```
<meta name="author" content="sherman"/>
```

```
<meta name="keywords" content="HTML,CSS,JavaScript"/>
```

```
<meta name="date" content="2009/4/30"/>
```

```
<meta name="description" content="HTML 教學網站" />
```

```
<meta name="generator" content="Microsoft Expression Web">
```

```
<meta http-equiv="Refresh" content="300">
```

網頁是找車云

```
<meta http-equiv="Refresh" content="20; URL=page4.html">
```



# body 元素

- 主要用來呈現網頁的內容
  - 文字與段落元素
  - 清單元素
  - 超連結元素
  - 圖片元素
  - 其它(見後面範例)
    - 表格
    - 表單
    - .....
    - .....

```
<body>
  <h1>範例練習</h1>
  <p>我是說明文字</p>
  <a href=" " >我是有超連結的文字</a>
  
</body>
```

# 網頁基礎內容元素

- 文字
  - `<h1>` ~ `<h6>` 標題文字
  - `<p>` 內文
  - `<br>` 換行
  - `<span>` , `&nbsp;`
- 圖片
  - `<img>`
- 區塊元素
  - `<div>`
- 超連結
  - `<a href= "" >`
- 清單
  - `<ul>` `<ol>` `<dl>`
  - `<li>` `<dt>` `<dd>`

# 網頁基礎內容元素(h1,p)

- 文字

- <h1> ~ <h6> 為HTML內的標題元素
- 作為網頁內容資訊重要性標示
- h1最重要依序向下區分等級
- 瀏覽器對不同等級會有相對應的文字級數大小
- p通常為文章段落的意思，用來作為文章內容的標籤

```
<h1 style="color:#0a6;">我是標題</h1>
```

```
<p style="width:200px;">
```

HTML 標籤 **p** 元素用來定義 **paragraph**段落，在HTML內的  
空白 間隔對瀏覽器無意義，瀏覽器顯示資訊時會自動移除  
</p>



- 文字
  - `<br>` 換行
  - `<span>` , `&nbsp;` 半形空白

<p style="width:200px;">

HTML標籤 p 元素用來定義 paragraph段落，在HTML內的有效空白          間隔對瀏覽器無意義，瀏覽器顯示資訊時<span style=  
不會自動移除  
</span>



# 特殊字元的使用

- 版權所有的符號 © `&copy;`
- 空白 `&nbsp;`
- AND `&`
- 大於 `>`
- 小於 `<`
- 雙引號 `"`
- Registered trademark<sup>®</sup> `&reg;`
- Trademark <sup>™</sup> `&#8482;`

更多的字符

[https://www.w3schools.com/html/html\\_entities.asp](https://www.w3schools.com/html/html_entities.asp)

# 圖片(image)元素

- 插入圖片

```

```

- src = “...” 必要屬性，指定圖片來源的url
- alt= “...” 不顯示圖形時以此文字代替
- title = “...” 當滑鼠移到圖片上時出現的說明文字(IE7之前)
- height, width 圖高與圖寬

- 圖片超連結

- <a href =“http://...”> <img src=“...”>圖檔連結</a>



# 網頁基礎內容元素(div)

- 區塊元素
  - <div>
  - 區塊元素特性無法併排
  - 預設無寬高顏色等樣式
  - 用來將網頁內各種元素分類排列

我是標題

我是內文

```
<div style="width:300px;height:300px;background-color:#dda;">  
  <h1 style="color:#0a6;">我是標題</h1>  
  <p>我是內文</p>  
</div>
```

# 超連結 (Hyperlink)元素

- 連結到其他網站

`<a href= "http://www.iiiedu.org.tw">資策會</a>`

- 連結到同一網站內的網頁

`<a href= "product.html">資策會</a>`

- 連結到同一網頁內的特定位置(書籤)

- 使用id屬性建立書籤 `<p id="bookmark1">ooo</p>`
- 再做超連結 `<a href= "#bookmark1">標題一 </a>`

- 連結電子郵件位址

- `<a href= "mailto:@iii.org.tw "> 寄信給我 </a>`
- `<a href= "mailto:@iii.org.tw ?subject=test"> 寄信給我 </a>` 加上主旨



# 清單元素

- 項目清單 `<ul></ul>`
  - `type = " disc, circle ,square "`
  - 單一清單項目 `<li></li>`
- 序號清單 `<ol></ol>`
  - `type = " 1, a, A, i, l "`
  - 單一清單項目 `<li></li>`
- 定義清單
  - 整個清單 `<dl></dl>`
  - 清單項目 `<dt></dt>`
  - 清單項目說明 `<dd></dd>`
  - 可多個 `<dd></dd>`
$$dl > (dt + dd * 2) + (dt + dd * 2)$$

```
<ul>
  <li>咖啡</li>
  <li>茶</li>
  <li>果汁</li>
</ul>
<ol>
  <li>咖啡</li>
  <li>茶</li>
  <li>果汁</li>
</ol>
```

- 咖啡
- 茶
- 果汁

1. 咖啡
2. 茶
3. 果汁

`ul>li{ITEM$}*3`

```
<dl>
  <dt>咖啡</dt>
    <dd>美式咖啡</dd>
    <dd>拿鐵咖啡</dd>
  <dt>茶</dt>
    <dd>水果茶</dd>
    <dd>黑醋栗茶
</dd>
</dl>
```

咖啡	美式咖啡
	拿鐵咖啡
茶	水果茶
	黑醋栗茶



# 區塊元素 / 行內元素

- 區塊元素

- <h1> ~ <h6>
- <p>
- <ul><li><ol><dl><dt><dd>
- <div>
- .....

- 特性

- 無法並排
- 可直接設定寬高

- 行內元素

- <span>
- <img>
- <a>
- .....

- 特性

- 可以並排
- 無法直接設定寬高

# 相對路徑/絕對路徑

- 檔案命名原則
  - 不用中文(亂碼)
  - 不使用空白鍵(會產生 %符號)
  - 不用特殊符號(伺服器設定容易連結不到)
  - 建議英文小寫或大小寫(盡量有意義命名)
- 相對路徑
  - `work.html` 同一層目錄
  - `../contact.html` 向上一層目錄
- 絕對路徑
  - `file:///C:/Users/Andytung/Desktop/my_site/600x400/Blue-Flower-with-Sharp-Thorns-600x400.jpg`



# emmet

- <https://emmet.io/>
- <https://docs.emmet.io/cheat-sheet/>
  - Emmet是一套面向文字編輯器的外掛程式，它允許通過內容輔助高速度的編寫和編輯HTML、XML、XSL和其他結構化的代碼格式。此專案2008年由Vadim Makeev發起，並由Sergey Chikuyonok和其他Emmet用戶基於Zen Coding 2.0的概念繼續積極開發。

## Syntax

### Child: >

```
nav>ul>li
<nav>
  <ul>
    <li></li>
  </ul>
</nav>
```

### Sibling: +

```
div+p+bq
<div></div>
<p></p>
<blockquote></blockquote>
```

### Climb-up: ^

```
div+div>p>span+emAbq
<div></div>
<div>
  <p><span></span><em></em></p>
  <blockquote></blockquote>
</div>
```

### Multiplication: \*

```
ul>li*5
<ul>
  <li></li>
  <li></li>
  <li></li>
  <li></li>
  <li></li>
</ul>
```

### Item numbering: \$

```
ul>li.item$*5
<ul>
  <li class="item1"></li>
  <li class="item2"></li>
  <li class="item3"></li>
  <li class="item4"></li>
  <li class="item5"></li>
</ul>
```

```
h$[title-item$]{Header $}*3
<h1 title="item1">Header 1</h1>
```



# Module 2

## CSS語言基礎

# 甚麼是 CSS

- CSS用來定義網頁呈現的樣式和版型
- CSS是用來延伸html而非取代html,是用來補html的不足
- CSS的演進
  - 1996/12 CSS1
  - 1998/5 CSS2
  - 2004/2 CSS2.1
  - 2010~ CSS3



# 關於 CSS3

- CSS3還不是標準
- 每個模組的研發進度不同
  - <http://www.w3.org/Style/CSS/current-work>
- 要開始用嗎?
  - Progressive enhancement
  - 使用瀏覽器前綴
    - -ms-(Internet Explorer) 、 -moz-(Firefox) 、
    - -webkit-(Safari 、 Chrome 、 opera)



# CSS優點

- 網頁內容與顯示樣式分開
  - 加快網頁傳輸的速度
  - 一致性共享樣式設定
  - 修改時只需針對樣式修改即可大幅減低維護網頁的困難度
- 排版的效果

# CSS基本語法

Selector { 屬性：屬性值； 屬性：屬性值 }

屬性樣式的設定，必須以大括號{ }包起來

屬性名稱與屬性值之間，必須以冒號隔開

每一組屬性設定之間，必須以分號隔開

- 最後一組屬性設定的結束，可以不用加分號
- 在CSS中註解的表示方式是 `/* */`，將註解的內容寫在裡面，例如：`/* 這裡面是註解 */`



# CSS寫法

- 方法一寫在標籤內  
`<h1 style="color:#0a6">我是標題</h1>`
- 方法二寫在檔頭內<head>  
`<head>  
 <style> h1 {color:#f00 ;} </style>  
</head>`
- 方法三在檔頭內link css檔案  
`<head>  
 <link rel=stylesheet type="text/css" href="mystyle.css" >  
</head >`



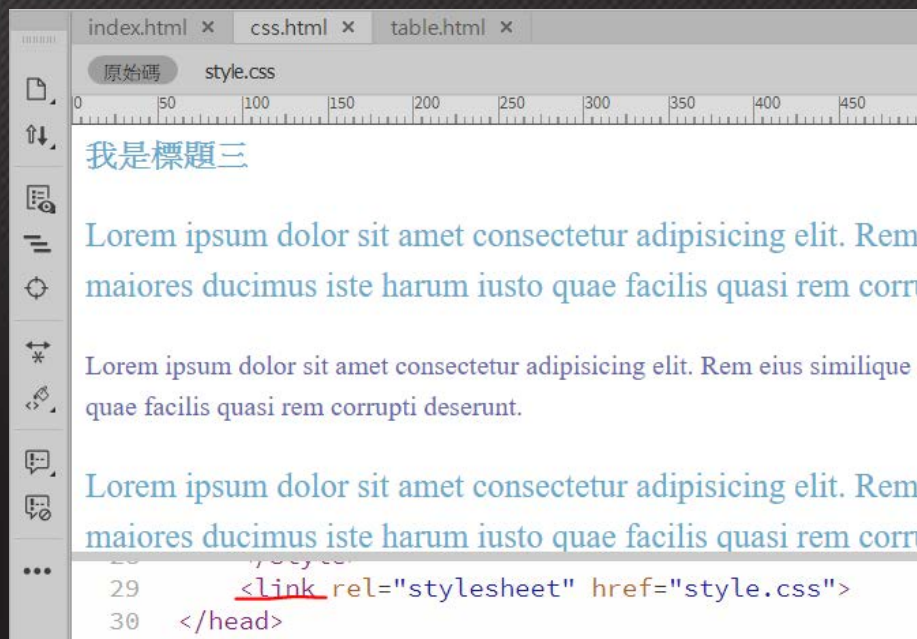
# CSS寫法之三

- 方法三在檔頭內link css檔案

<head>

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

</head >



# CSS 使用的大小單位

- 1in = 2.54cm = 25.4mm = 72pt = 12pc
- 瀏覽器預設文字(Font-Size)的大小
  - 1em=100%=12pt=16px

































	Recommended	Occasional use	Not recommended
Screen	em, px, %	ex	pt, cm, mm, in, pc
Print	em, cm, mm, in, pt, pc, %	px, ex	

# CSS使用顏色的設定方式

p {color: white ;}

p {color: #fff ;}

p {color: rgb(255,255,255) ;}

Named	Numeric	Color name	Hex rgb	Decimal
		<b>black</b>	#000000	0,0,0
		<b>silver</b>	#C0C0C0	192,192,192
		<b>gray</b>	#808080	128,128,128
		<b>white</b>	#FFFFFF	255,255,255
		<b>maroon</b>	#800000	128,0,0
		<b>red</b>	#FF0000	255,0,0
		<b>purple</b>	#800080	128,0,128
		<b>fuchsia</b>	#FF00FF	255,0,255
		<b>green</b>	#008000	0,128,0
		<b>lime</b>	#00FF00	0,255,0
		<b>olive</b>	#808000	128,128,0
		<b>yellow</b>	#FFFF00	255,255,0
		<b>navy</b>	#000080	0,0,128
		<b>blue</b>	#0000FF	0,0,255
		<b>teal</b>	#008080	0,128,128
		<b>aqua</b>	#00FFFF	0,255,255

更多顏色設定方式：

[https://www.w3schools.com/css/css\\_colors.asp](https://www.w3schools.com/css/css_colors.asp)



# CSS Selectors

- Selectors(選取器)

CSS的樣式是需要套用到某一個HTML標籤上，所以要怎麼找到您想要修改的HTML標籤，就是要透過選取器。

- 常用的選取器

- 標籤(Type)選取器
- 類別(Class)選取器
- 物件(ID)選取器

- 更多好用的選取器

- 後面有其它Selectors的範例說明
- 完整說明請參考 <http://www.w3.org/TR/selectors/>

# 標籤(Type)選取器

- 標籤(Type)選取器
- 如 Body, h1 ~h6, p .....
- 直接選取要改變的標籤樣式設定css屬性

```
h1 { color:#f00 ; }  
p { width: 500px; color: #777; }
```

---

<h1>我是大標題</h1>

<p>Lorem ipsum dolor sit amet consectetur adipisicing elit. Vel eos, ullam, consequatur nam deleniti quos at recusandae, inventore amet alias aut. Totam maxime sint ad tempore enim corrupti, ducimus eveniet.</p>



# 類別(class)選取器(一)

- 類別選擇器的用法有二種:
- (一)可套用到不同標籤上—
- CSS 設定樣式 ,定義類別名稱時要在前面加上“ . ”  
    .類別名 { 樣式1 ; 樣式2 ; .....}  
    例如 : .st1 { color:red;text-transform:uppercase}
- HTML套用  
    <h1 class= “st1”>apple</h1>  
    <h2 class=“st1” >lemon</h2>  
    這樣子表示<h1>及<h2>都會套用.st1的樣式



# 類別(class)選取器(二)

- (二) 只讓特定的標籤套用—

CSS 設定樣式

tag.類別名 { 樣式1 ; 樣式2 ; ..... }

例如 : **h2.st1** { color:green;text-transform:uppercase }

HTML 套用

<h1 class= "st1" >apple</h1>

<h2 class= "st1" >lemon</h2>

因為指定<h2>才能被.st1套用,所以只有<h2>會有效果

# ID物件選取器

- ID物件只能套用一次,不像類別選擇器可同時用在多個標籤上,定義ID名稱時要在前面加上” # ”
  - CSS 設定樣式

```
#ID名 { 樣式1 ; 樣式2 ; ..... }
```
  - 例如 : `#theH1 {text-transform:capitalize}`
  - HTML套用

```
<h1 id="theH1">this is id</h1>
```
  - Javascript使用時

```
theH1.style.color="blue";
```
- \* class or ID 命名時不可以使用標準的HTML標籤名稱



# CSS執行順序

- 同一層級中，先定義的會被後定義的覆蓋過去

Ex.

```
h1 { color:#f00 ; }
```

...

...

```
h1 { color:#00f ; }
```



# CSS執行順序範例

- 範例
  - cssSelector.html

## 我是大標題

LOREM IPSUM DOLOR SIT AMET, CONSECTETUR ADIPISICING ELIT. QUAM, NISI EXPEDITA EIUS AT LAUDANTIUM QUIDEM ILLUM MOLLITIA RATIONE IUSTO SIMILIQUE IPSUM, QUOD LABORUM ADIPISCI NEMO DISTINCTIO OPTIO, TEMPORA CUPIDITATE ENIM?

lorem ipsum dolor sit amet, consectetur adipisicing elit. quam, nisi expedita eius at laudantium quidem illum mollitia ratione iusto similique ipsum, quod laborum adipisci nemo distinctio optio, tempora cupiditate enim?

LOREM IPSUM DOLOR SIT AMET, CONSECTETUR ADIPISICING ELIT. QUAM, NISI EXPEDITA EIUS AT LAUDANTIUM QUIDEM ILLUM MOLLITIA RATIONE IUSTO SIMILIQUE IPSUM, QUOD LABORUM ADIPISCI NEMO DISTINCTIO OPTIO, TEMPORA CUPIDITATE ENIM?

lorem ipsum dolor sit amet, consectetur adipisicing elit. quam, nisi expedita eius at laudantium quidem illum mollitia ratione iusto similique ipsum, quod laborum adipisci nemo distinctio optio, tempora cupiditate enim?

# 與超連結有關的選取器

- *a:link* 定義預設超連結樣式
- *a:visited* 定義瀏覽過的超連結樣式
- *a:hover* 定義滑鼠移至的超連結樣式
- *a:active* 定義選取的超連結樣式

```
a:link{text-decoration:none;}
```

```
a:visited{color:red;}
```

```
a:hover{text-decoration:underline;color:green;}
```

```
a:active{color:yellow;}
```

# 多個超連結樣式設定

- 使用類別class制定
- 兩兩一組( a, a:visited )( a:hover, a:action )
- 加上類別作變化設定

```
a:link, a:visited{text-decoration:none;}
```

```
a:hover, a:active{text-decoration:underline;color:green;}
```

```
.box a:link, .box a:visited{text-decoration:none;}
```

```
.box a:hover, .box a:active{text-decoration:underline;color:#0cf;}
```

```
<a href="">我是第一種連結樣式</a>
```

```
<a href="" class="box">我是第二種連結樣式</a>
```



# 清單樣式的屬性

- 清單前使用的符號
  - list-style-type
  - none (無) | disc (實心圓) | circle (空心圓) | square (實心方形) | decimal (阿拉伯數字) | decimal-leading-zero (以 0 開頭的 10 進位數字) | lower-alpha (小寫英文字母) | upper-alpha (大寫英文字母) | lower-roman (小寫羅馬字母) | upper-roman (大寫羅馬字母)
- 清單前使用圖片
  - list-style-image : url ('圖片路徑與檔名')
- 清單前使用符號的位置
  - list-style-position : inside | outside

# 清單樣式sample1

- 將清單符號改符號

```
<style>
  li {
    list-style-type: square;
  }
</style>
```

---

```
<ul>
  <li>新聞</li>
  <li>汽車</li>
  <li>娛樂</li>
</ul>
```

# 清單樣式sample2

- 將清單符號改圖片

```
<style>
.s1 {
  list-style-type:none;
  list-style-image:url('xxx.jpg');
}

.s2 {
  list-style-type:none;
  list-style-image:url('xxx.jpg');
}
</style>
```

```
<ul class="s1">
  <li>咖啡
    <ul class="s2">
      <li>美式</li>
      <li>拿鐵</li>
      <li>卡布奇諾</li>
    </ul>
  </li>
  <li>紅茶</li>
  <li>果汁</li>
</ul>
```



# Module 3

## 表格、表單/CSS

# 表格的基本架構

- 表格的開始<table>與結束</table>
- tr 列(row)
- th 存放表格標題的儲存格
- td 存放資料的儲存格(column)
- thead,tbody,tfoot 群組化儲存格
- caption 表格標題
- colspan , rowspan 儲存格的合併

Product ID	Product Name	Unit Price
001	iPhone	20000
002	netbook	15000



# 表格範例 – css設定

檔案 – table-css.html

```
<table>
  <caption><h4>表格練習</h4></caption>
  <thead>
    <tr><th>a</th><th>b</th><th>c</th><th>d</th> </tr>
  </thead>
  <tbody>
    <tr><td>1</td><td>2</td><td>3</td><td>4</td></tr>
    <tr><td>5</td><td>6</td><td>7</td><td>8</td></tr>
    <tr><td>9</td><td>10</td><td>11</td><td>12</td></tr>
  </tbody>
</table>
```

表格練習

a	b	c	d
1	2	3	4
5	6	7	8
9	10	11	12



# 美化表格

```
Table {  
    width:400px;  
    border:3px solid green;  
    border-collapse:collapse;  
    margin:0 auto;
```

```
}
```

```
td, th{  
    border:1px solid red;  
    padding:4px;  
    text-align:center;  
    vertical-align:bottom;
```

```
}
```

```
tbody tr:nth-child(2n){background-color:yellow} 偶數列
```

```
tbody tr:nth-child(2n+1){background-color:silver} 奇數列
```

表格練習

a	b	c	d
1	2	3	4
5	6	7	8
9	10	11	12

# 表單

會員登入

首次購買

Email

請輸入Email帳號

密碼

請輸入密碼

☒ 記住Email

[忘記密碼](#)

登 入

註冊

名字

姓氏

電子信箱地址

@yahoo.com

[我想使用我自己的電子信箱](#)

密碼

+886 ▼

手機號碼

出生月份 ▼

日

年

性別 (選填)

繼續

已經有帳號了? [登入](#)

我同意接受 [Yahoo服務條款](#)和[隱私權](#)

# 表單結構

- 表單的開始<form>與結束</form>
- 重要屬性
  - action : 它的值是伺服器上等待接收資料程式的URL
  - method : 傳送表單資料的方式, 它的值有兩種  
get (預設值)及post
  - enctype : 資料傳送時的編碼設定
    - application/x-www-form-urlencoded(預設值)
    - multipart/form-data(上傳檔案時需要使用)
    - text/plain

```
<form action="伺服器上的某支程式" method="post">
```

所有使用者輸入，要傳送到伺服器端的資料都要放在這個裡面

```
</form>
```



# 表單 – 文字輸入控制項1

- 單行文字輸入
- Input 元素中type屬性的屬性值設定成text

```
<label for="account">姓名:</label>  
<input type="text" id="account" name="account" value="guest">
```

- 密碼輸入
  - Input 元素中type屬性的屬性值設定成password

```
<label for="account">姓名:</label>  
<input type="text" id="account" name="account" value="guest">
```

- 常用屬性
  - name、id、value(使用者輸入的資料)
  - size(寬)、maxlength(最多輸入的字元)

姓名:	<input type="text" value="guest"/>
密碼:	<input type="password"/>

# 表單 – 文字輸入控制項2

- 多行文字輸入
  - 使用 `textarea` 元素

```
<label for="memo">意見:</label>  
<textarea cols="40" rows="5" id="memo" name="memo" >  
</textarea>
```

- 常用屬性：
  - `name`、`id`、`value`
  - `rows`(高)、`cols`(寬)

意見:

# 表單 — 選擇控制項1

- 單選控制項
  - Input 元素中type屬性的屬性值設定成radio

```
<input type="radio" name="gender" value="male">男  
<input type="radio" name="gender" value="female">女
```

☐ 男 ☐ 女

- 複選控制項
  - Input 元素中type屬性的屬性值設定成checkbox

```
<input type="checkbox" name="hobby" value="music">音樂  
<input type="checkbox" name="hobby" value="sport">運動  
<input type="checkbox" name="hobby" value="reading">閱讀  
<input type="checkbox" name="hobby" value="movie">電影
```

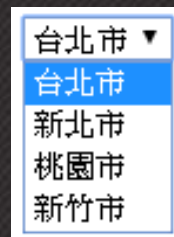
☐ 音樂 ☐ 運動 ☐ 閱讀 ☐ 電影



# 表單－選擇控制項2

- 下拉式選單控制項，可單選或者是複選
  - 使用select元素，在select元素內加上option元素

```
<select name="LivingIn">  
  <option value="TPE">台北市</option>  
  <option value="TPH">新北市</option>  
  <option value="TYC">桃園市</option>  
  <option value="HSC">新竹市</option>  
</select>
```



- select元素常用屬性
  - name、id、multiple(複選，沒有加上此屬性為單選)
- option元素常用屬性
  - value、selected(預設被選擇的項目)

# 表單 — 按鈕控制項1

- 送出按鈕
  - Input 元素中type屬性的屬性值設定成submit
  - 會將表單中使用者輸入的資料送到伺服器端
- 清除資料
  - Input 元素中type屬性的屬性值設定成reset
  - 會清除表單中使用者輸入的資料

A rectangular button with a light gray border and a light gray background, containing the text "送出" in black.A rectangular button with a light gray border and a light gray background, containing the text "清除" in black.

```
<input type="submit" value="送出">  
<input type="reset" value="清除">
```



# 表單 – 按鈕控制項2

- 一般按鈕
  - Input 元素中type屬性的屬性值設定成button
  - 使用button元素
  - 沒有任何預設功能，用來讓開發人員透過JavaScript程式去自訂一些功能。

```
<input type="button" name="button1" id="button1" value="一般按鈕">  
<button type="button" name="button1" id="button1" >一般按鈕</button>
```

- 常用屬性
  - name、id、value
  - disabled(取消按鈕功能)





# 表單 – 檔案上傳控制項

- 上傳檔案
  - Input 元素中type屬性的屬性值設定成file

```
<label class="title">照片:</label><input type="file" name="file1">
```

照片:  選擇檔案 未選擇任何檔案

HTML5提供更多的表單輸入類型及屬性

# 表單 - 範例表單標籤

```

<form action="#" method="post">
<fieldset>
<legend>個人資料一<em>(必填)</em></legend>
  <div class="st">
    <label class="title">姓名:</label>
    <input type="text" name="account" value="guest" size="10">
  </div>
  <div class="st">
    <label class="title">密碼:</label>
    • <input type="password" name="pwd" size="10">
  </div>
  <div class="st">
    <label class="title">性別:</label>
    <input type="radio" name="gender" value="male">男
    <input type="radio" name="gender" value="female">女
  </div>
</fieldset>
</form>

```



# 表單 - 範例表單標籤

```
<style>
fieldset {
  width:500px;
  margin:15px;
  border:1px solid #ACD6FF;
  border-radius:10px;
}
legend{
  border:1px solid #ACD6FF;
  border-radius:5px;
  padding: 5px 15px ;
}
em {color:red;}
```

```
.st {
  width:450px;
  margin:20px;
  border-bottom:1px solid #F0F0F0;
  padding-bottom:10px;
}
.title {
  width:100px;
  float:left;
  text-align:right;
  padding-right:3px;
}
</style>
```



# 表單 - 範例表單標籤

個人資料一 (必填)

姓名:

密碼:

性別: ☐ 男 ☐ 女

# Module 4

# HTML5

Structural Semantics Elements  
Form Elements

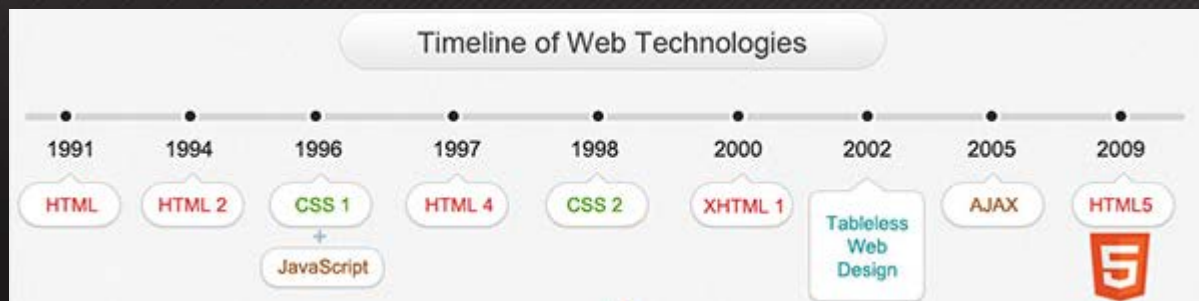
# HTML的歷史

- 最初由Tim Berners-Le於1991年所發明
- 它是由W3C組織推薦使用的國際標準
- HTML = HyperText Markup Language  
= 超文字標記(標籤)語言
- 最初設計的目的，是要將電腦中文字與圖片用一種固定的方式來呈現，並透過超連結的方式，將這些資料整合在一起。



# HTML的演進

- HTML 4.0 published on 18 December 1997
- Revised 24 April 1998
- HTML 4.01 24 December 1999(Subversion of HTML 4)
- HTML 5



# 關於 HTML5

- 2014/10/28 HTML5已經成為標準
  - <http://www.w3.org/TR/html5/>
- HTML5 = HTML + CSS + JS APIs
- 不是網頁文件了，是網頁應用程式平台
- 瀏覽器支援狀態
  - <http://html5test.com/>
  - <http://caniuse.com>



# HTML5 新元素

- Structural Semantic Elements
  - section
  - header
  - footer
  - aside
  - nav
  - article
- 這些新元素沒有預設樣式，要自行用CSS來設定



# HTML5 語意標籤

<header>(公司logo)

<hgroup>(公司名稱、標語....)

<nav>(網站導覽列)

<article>(主要內容)

<h1> ~ <h6>

<p>

<figure> 圖片

<header>

<footer>

<aside>

<section>

<section>

<nav>

<section>(主要內容)

<article>

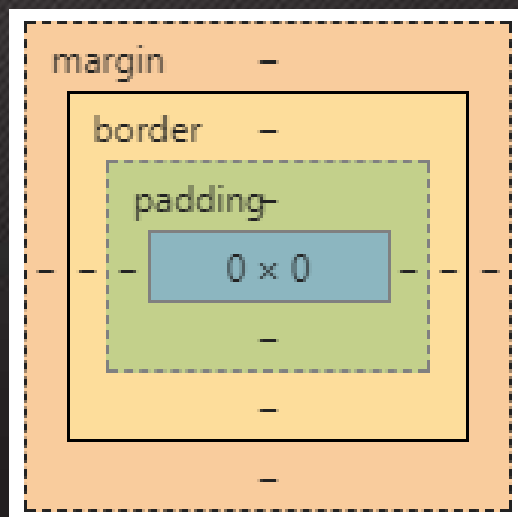
<footer>(copyright)

<nav>

<address>

# 方塊模型(box model)

- 什麼是方塊模型
- 方塊模型樣式



# 方塊模型與元素關係

- 元素分為行內元素與區塊元素
- 區塊元素直接有方塊模型設定，行內元素則無
- 行內元素與區塊元素
- 行內元素都能設定邊界參數(margin, padding, border)
  - 行內元素天生可以併排
  - 行內元素無法直接使用CSS設定寬高
- 區塊元素
  - h1, p, li, table, td.....
  - 區塊元素天生無法並排
  - 區塊元素可以直接使用CSS設定寬高



# 內置框架 - iframe

- 可將別人的網頁嵌入到我們的網頁中
  - Ex. Google Map

- 常用屬性

- src、width、height
- 捲軸 scrolling，可以設定值為yes、no及auto
- 隱藏框線 frameborder=0



# Media - Audio

- HTML5 `<audio>` 元素指定了在網頁中嵌入音頻的標準方法
  - `<source>` 元素允許您指定瀏覽器可以選擇的備用音頻檔案。
  - `<controls>` 屬性添加了播放bar，會有播放、暫停和音量控制

```
<audio controls>
```

```
  <source src="horse.mp3" type="audio/mpeg">
```

```
  Your browser does not support the audio element.
```

```
</audio>
```



# Media - Video

- HTML5 `<video>` 元素指定了在網頁中嵌入視訊檔案的標準方法
  - `<source>` 元素允許您指定瀏覽器可以選擇的備用音頻檔案。
  - `<controls>` 屬性添加了播放bar，會有播放、暫停和音量控制
  - `<autoplay>` 屬性會自動播放，不顯示播放bar

```
<video width="320" height="240" autoplay>  
  <source src="movie.mp4" type="video/mp4">  
  <source src="movie.ogg" type="video/ogg">  
</video>
```



# Media - Video

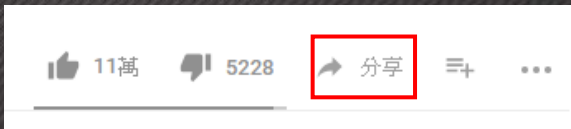
- Chrome 在 Win10 之後的版本
  - Video屬性 autoplay需多加 muted

```
<video autoplay muted>  
  <source src="video.mp4" type="video/mp4"></source>  
</video>
```

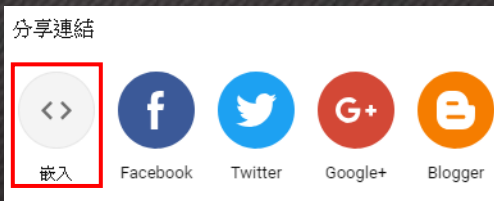
# Media - Youtube

- 找到合法授權影片

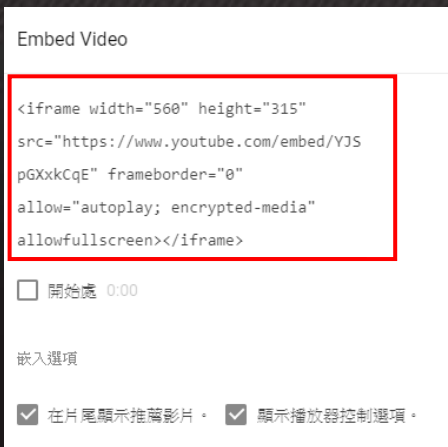
• 1.



• 2.



• 3.



# Module 5

## CSS基礎屬性



# CSS基礎屬性

- 背景
- 邊界
- 邊框
- 文字
- 列表
- 並排 - 浮動
- display
- 定位
- 溢位

# 背景(background)

- 背景顏色
- 背景圖片
- 背景呈現方式
- 背景位置及尺寸
- 背景漸層(CSS3課程會講解)



# 背景顏色

- background-color(設定背景顏色)
  - 語法：{background-color : transparent | ( color ) }
  - background-color:#000000 使用十六進制位。
  - background-color:black 直接指定顏色的名稱。
  - background-color:rgb(128,128,255) 利用RGB的顏色設定法。
- 超連結、文字、表格等，也可以有自己的背景顏色，例如：
  - <font style="background-color:rgb(128,128,255)">資策會</font>



# 背景圖片

- **background-image**背景圖片,是指定 HTML 中任何元件的背景圖片。在 CSS 還未出現時只有整頁 可以指定背景圖片，現在有了 CSS 任何元件皆有自己的背景圖片了！
- 語法 :`{background-image : none | URL ( url ) }`
- 範例 :`{background-image:URL(1.JPG)}`
- 多重背景(Multiple Background)
- 先指定的背景，會在最上面
- 盡量使用背景為透明的圖片

# 背景呈現方式

- 背景附著(background-attachment)
- 語法：{background-attachment : scroll | fixed }
- background-attachment:scroll  
background-attachment:fixed
- 背景重複(background-repeat)
- 語法：{background-repeat- : repeat | repeat-x | repeat-y | no-repeat }
- background-repeat:repeat (一般貼的滿滿的狀況)  
background-repeat:repeat-x (只往x方向貼滿)  
background-repeat:repeat-y (往Y方向貼滿)  
background-repeat:no-repeat (不重複貼，也就是只有一張小圖)



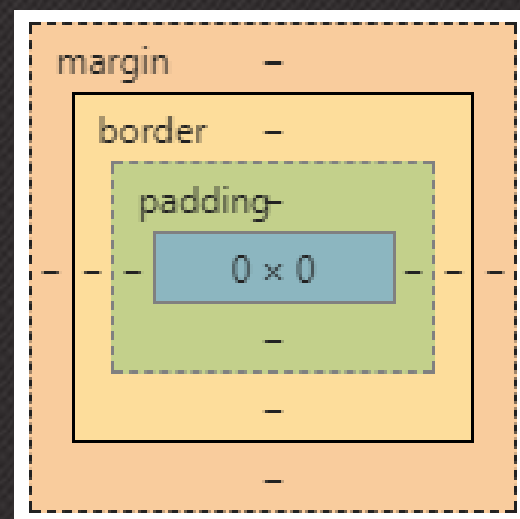
# 背景位置

- 語法 : `background-position: top | bottom | left | center | right | ( length ) | ( 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% )
- 範例  
    `{ background-position : 5% 20% ; }`
- 背景尺寸  
    `{ background-size: 110% 90%; }`



# 邊界

- 方塊模型(box model)
- margin
- padding
- border



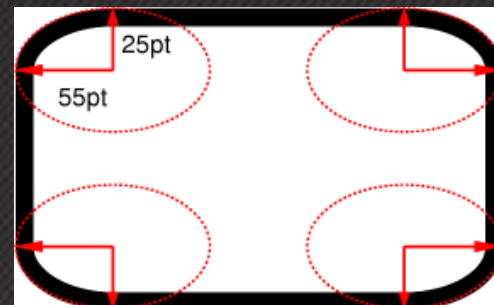
# 邊框

- 指定四個邊角都呈現圓角

```
{ border-radius: 20px;  
  -moz-border-radius: 20px;  
  -webkit-border-radius: 20px; }
```

```
{ border-radius: 5px 10px 15px 20px; } /*左上 右上 右下 左下*/
```

```
{ border-top-left-radius: 55pt 25pt;  
  border-top-right-radius: 55pt 25pt;  
  border-bottom-right-radius: 55pt 25pt;  
  border-bottom-left-radius: 55pt 25pt; }
```



# CSS常見樣式 – 文字

- 字型樣式
- 文字樣式



# 字型樣式(1)

- 字型
  - font-family : 新細明體, 標楷體, 細明體.....
- 字型粗細
  - font-weight : 100(細)~900(粗) | normal(400) | bold(700) | bolder | lighter
- 字型類型
  - font-style : normal | italic | oblique

# 字型樣式(2)

- 字型大小
- **font-size** : <絕對大小> | <相對大小> | <長度值> | <比例值>
- 絕對大小 : xx-small 、 x-small 、 small 、 medium 、 large 、 x-large 、 xx-large, 每個間隔1.2倍
- 相對大小 : larger 、 smaller
- 長度值 : in 、 cm 、 mm 、 pt 、 pc 、 em 、 ex 、 px
- 比例值 : em 、 %
- 下載字型

```
@font-face{  
  font-family: JustOldFashion;  
  src: url(fonts/JustOldFashion.ttf);  
}
```



# 文字樣式(1)

- 文字字距： **letter-spacing**
  - 設定文字字距，可調整文字與文字之間的距離  
例如： **letter-spacing：2px**
- 行高： **line-height**
  - 行高設定也就是行距的設定，行距設定的單位和字級大小設定一樣  
例如： **line-height：20px**

```
<p class="st">文字間距行高練習  
<br>有無字間行高差距  
</p>  
<p>文字間距行高練習  
<br>有無字間行高差距  
</p>
```

```
.st {  
  color: #f00;  
  letter-spacing: 20px;  
  line-height: 30px;  
}
```

文 字 間 距 行 高 練 習  
有 無 字 間 行 高 差 距

文字間距行高練習  
有無字間行高差距



# 文字樣式(2)

- 文字縮排：`text-indent`
    - 可以設定文字作縮排的效果
    - 例如：`text-indent : 15px`
  - 文字水平對齊
  - 文字垂直對齊：在DIV區塊沒有作用，只能作用於儲存格(td)
- 例如：`vertical-align : bottom | middle | top`

# 文字樣式(3)

- 文字修飾

text-decoration : none | underline | overline |  
line-through

- 文字大小轉換

text-transform : none | capitalize | uppercase |  
lowercase

- 文字色彩

color : green



# 浮動(Float)

- 行內元素不需要 float，使用<span>測試
- float 浮動 與 clear 清除
  - 區塊元素, 行內元素特性
  - 何時使用?

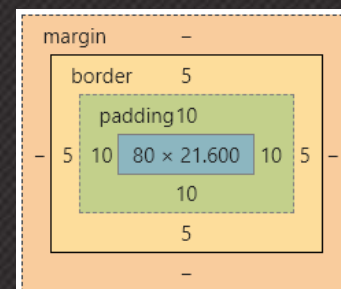
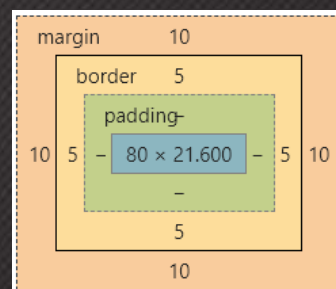
```
<div class="red">  
  我是紅盒子  
</div>  
<div class="blue">  
  我是藍盒子  
</div>  
<p>我跟在後面喔!</p>
```

```
.red { color:#f00 ; }  
.blue { color:#00F ; }  
div { float: left; }  
p { clear:left ;}
```



# 浮動(Float)

- 加上邊界參數，margin, padding, border
- 到瀏覽器查看邊界



```
<div class="red">
  我是紅盒子
</div>
<div class="blue">
  我是藍盒子
</div>
<p>我跟在後面喔!</p>
```

```
.red { color:#f00 ; margin:10px ; border:5px
solid #aaa ;}
.blue { color:#00F ; padding:10px ;
border:5px solid #fa6 ;}
div { float: left; }
p { clear:left ;}
```

# 基礎排版- display

- `display:inline-block` 可同時帶有行內與區塊元素特性，可以並排並使用CSS設定寬高
- 缺點：`margin-right`, `margin-bottom`會產生4px 間距，必須額外處理

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

```
div {
    width: 100px;
    height: 100px;
    display: inline-block ;
    background-color: #ccc ;
    margin-right: -4px ;
    margin-bottom: -4px ;
}
```



# display樣式(1)

- **inline**: 將元素顯示為內聯元素（如<span>）。任何高度和寬度屬性將不起作用播放»
- **block**: 塊將元素顯示為塊元素（如<p>）。它開始一個新的行，佔據了整個寬度播放它»
- **flex**: 將一個元素顯示為塊級Flex容器播放»
- **Inline-block**: 將內容顯示為內嵌級塊容器。元素本身被格式化為內聯元素，但是您可以應用高度和寬度值播放»
- **inline-flex**: 顯示一個元素作為內聯級Flex容器播放»
- **inline-table**: 顯示為內嵌級表播放»
- **list-item**: 讓元素的行為像一個<li>元素播放它»
- **run-in**: 根據上下文以塊或內聯方式顯示元素播放»
- **table**: 讓元素的行為像一個<table>元素播放»



# display樣式(2)

- table-caption: 讓元素的行為像一個<caption>元素播放»
- table-column-group: 讓元素的行為像一個<colgroup>元素播放»
- table-header-group: 讓元素的行為像一個<thead>元素播放»
- table-footer-group: 讓元素的行為像一個<tfoot>元素播放它»
- table-row-group: 讓元素的行為像一個<tbody>元素播放»
- table-cell: 讓元素的行為像一個<td>元素播放»
- table-column: 讓元素表現得像一個<col>元素播放它»
- table-row: 讓元素的行為像一個<tr>元素播放»
- none: 被完全刪除播放»
- initial: 將此屬性設置為其默認值。閱讀有關初始播放»
- inherit: 從其父元素繼承此屬性。閱讀關於繼承

# 邊框(border)

- 邊框樣式 - border : ( border-width ) | ( border-style ) | ( color )
- 邊框圓角 - border-radius
- 邊框圖片 – border-image (CSS3課程會講解)
- 陰影 - Box Shadow



# 邊框樣式(border)

- 功能：設定邊框
- 語法：{border : ( border-width ) | ( border-style ) | ( color ) }
- ( 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-style ) - 可設無(none)、實線(solid)、雙線(double)、點線(dotted)、虛線(dashed)、溝線(groove)、脊線(ridge)、嵌入線(inset)、浮出線(outset)
- (border- 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 ) 表紅綠藍強度
- 範例 :{border :thick double red}



# 定位(position)

- position : absolute|relative|static(預設值)|fixed
  - absolute(絕對位置)-- 父元素內容區邊界
  - relative(相對位置)--原本應該在的位置
  - static(靜態位置)--依照原始碼的排列順序來定位
  - fixed(固定位置)--捲動捲軸時不會移動
  - bottom 、 left 、 right 、 top : auto|<length>|<percentage>
- 目標元素與邊界的距離
  - z-index : auto|<integer>
  - 值大表示越上層

# 溢位(overflow)

- **overflow**：屬性指定在元素的內容太大而無法放入指定區域時是剪輯內容還是添加滾動條。
  - **visible**：默認。溢出不會被裁剪。內容在元素框外渲染
  - **hidden**：溢出被裁剪，其餘內容將不可見
  - **scroll**：剪輯了溢出，並添加了滾動條以查看其餘內容
  - **auto**：與相似**scroll**，但僅在必要時添加滾動條



# 範例

- 簡易區塊排版置中對齊
  - 區塊邊界 – width, height, border, margin, padding
  - 併排 - Float
  - Ul, li – list-style
  - 文字 – 標題, 內文, 字形,
  - 選項超連結
- CSS 優先順序
  - 套用 – 標籤內套用 > 內部載入 > 外部載入
  - 選取器優先序 – id > class > 標籤元素
  - 撰寫順序 – 後面 > 前面