

# 樣式之美 CSS

前端工程開發實務訓練 (2/8)

講師：蔣定宇 / josephj



# CSS 基礎知識

很好上手，重點在了解有哪些屬性

# 行間 CSS

*Inline CSS*

直接在標籤內使用 style 屬性

# 行間 CSS

*Inline CSS*

直接在標籤內使用 style 屬性

```
<p style="屬性名稱:屬性值;">...</p>
```

# 嵌入式 CSS *Embedded CSS*

直接將 CSS 寫在 HTML 內的作法

# 嵌入式 CSS

*Embedded CSS*

直接將 CSS 寫在 HTML 內的作法

`<style></style>` 標籤

# 嵌入式 CSS

*Embedded CSS*

直接將 CSS 寫在 HTML 內的作法

<style></style> 標籤

type 可省略

```
<style type="text/css" media="顯示媒介名稱">  
選擇器 {  
    屬性名稱 : 屬性值;  
}  
</style>
```

media 在何種媒介顯示

- 印表機 = printer
- 電腦螢幕 = screen
- 預設是全部 = all

# 嵌入式 CSS

Embedded CSS

直接將 CSS 寫在 HTML 內的作法

<style></style> 標籤

type 可省略

```
<style type="text/css" media="顯示媒介名稱">  
選擇器 {  
    屬性名稱 : 屬性值;  
}  
</style>
```

CSS 內容 →

media 在何種媒介顯示

- 印表機 = printer
- 電腦螢幕 = screen
- 預設是全部 = all

# 嵌入式 CSS

Embedded CSS

直接將 CSS 寫在 HTML 內的作法

<style></style> 標籤

type 可省略

```
<style type="text/css" media="顯示媒介名稱">  
選擇器 {  
    屬性名稱 : 屬性值;  
}  
</style>
```

CSS 內容 →

media 在何種媒介顯示

- 印表機 = printer
- 電腦螢幕 = screen
- 預設是全部 = all

建議放置位置: </head> 之前

# 外部 CSS *External CSS*

引用外部 CSS 檔案 (建議附檔名為 .css)

# 外部 CSS *External CSS*

引用外部 CSS 檔案 (建議附檔名為 .css)

<link> 標籤 (無結尾)

# 外部 CSS *External CSS*

引用外部 CSS 檔案 (建議附檔名為 .css)

<link> 標籤 (無結尾)

```
<link rel="stylesheet" href="CSS 檔案路徑" media="顯示媒介名稱">
```

# 外部 CSS

*External CSS*

引用外部 CSS 檔案 (建議附檔名為 .css)

<link> 標籤 (無結尾)

必須指定



```
<link rel="stylesheet" href="CSS 檔案路徑" media="顯示媒介名稱">
```

# 外部 CSS

*External CSS*

引用外部 CSS 檔案 (建議附檔名為 .css)

<link> 標籤 (無結尾)

必須指定

```
<link rel="stylesheet" href="CSS 檔案路徑" media="顯示媒介名稱">
```

↑  
與前面相同

# 外部 CSS

*External CSS*

引用外部 CSS 檔案 (建議附檔名為 .css)

<link> 標籤 (無結尾)

必須指定

```
<link rel="stylesheet" href="CSS 檔案路徑" media="顯示媒介名稱">
```

與外部檔案連結

與前面相同

```
選擇器 {  
    屬性名稱 : 屬性值;  
}
```

※ 不需 style 標籤

# 外部 CSS

*External CSS*

引用外部 CSS 檔案 (建議附檔名為 .css)

<link> 標籤 (無結尾)

必須指定

```
<link rel="stylesheet" href="CSS 檔案路徑" media="顯示媒介名稱">
```

與外部檔案連結

與前面相同

```
選擇器 {  
    屬性名稱 : 屬性值;  
}
```

※ 不需 style 標籤

另外還有 HTML Tag 中的 style, 正常情況用不到

# 該用哪一種？

# 該用哪一種？

99% 的情況都應該使用外部 CSS

# 該用哪一種？

99% 的情況都應該使用外部 CSS

- 語意：樣式與結構能夠徹底分離。

# 該用哪一種？

99% 的情況都應該使用外部 CSS

- 語意：樣式與結構能夠徹底分離。
- 效率：寫一次可以套用到多個網頁中。

# 該用哪一種？

99% 的情況都應該使用外部 CSS

- 語意：樣式與結構能夠徹底分離。
- 效率：寫一次可以套用到多個網頁中。
- 效能：外部檔案能被快取 (cache)。

# 該用哪一種？

99% 的情況都應該使用外部 CSS

- 語意：樣式與結構能夠徹底分離。
- 效率：寫一次可以套用到多個網頁中。
- 效能：外部檔案能被快取 (cache)。
- 唯一例外：會被設為首頁的入口網站。

# 該用哪一種？

99% 的情況都應該使用外部 CSS

- 語意：樣式與結構能夠徹底分離。
- 效率：寫一次可以套用到多個網頁中。
- 效能：外部檔案能被快取 (cache)。
- 唯一例外：會被設為首頁的入口網站。
  - ❖ 頁數少、更快顯示。

# 該用哪一種？

99% 的情況都應該使用外部 CSS

- 語意：樣式與結構能夠徹底分離。
- 效率：寫一次可以套用到多個網頁中。
- 效能：外部檔案能被快取 (cache)。
- 唯一例外：會被設為首頁的入口網站。
  - ❖ 頁數少、更快顯示。

<http://developer.yahoo.com/performance/rules.html#external>

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
<div id="nav">
  <div class="bd">
    <p>選單</p>
  </div>
</div>
```

```
<div id="article">
  <div class="bd">
    <p>文章</p>
  </div>
</div>
```

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
  
<div id="nav">  
  <div class="bd">  
    <p>選單</p>  
  </div>  
</div>  
  
<div id="article">  
  <div class="bd">  
    <p>文章</p>  
  </div>  
</div>
```

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
  
<div id="nav">  
    <div class="bd">  
        <p>選單</p>  
    </div>  
</div>  
  
<div id="article">  
    <div class="bd">  
        <p>文章</p>  
    </div>  
</div>
```

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
  
<div id="nav">  
    <div class="bd">  
        <p>選單</p>  
    </div>  
</div>  
  
  
<div id="article">  
    <div class="bd">  
        <p>文章</p>  
    </div>  
</div>
```

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
  
<div id="nav">  
    <div class="bd">  
        <p>選單</p>  
    </div>  
</div>  
  
  
<div id="article">  
    <div class="bd">  
        <p>文章</p>  
    </div>  
</div>
```

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
  
<div id="nav">  
    <div class="bd">  
        <p>選單</p>  
    </div>  
</div>  
  
  
<div id="article">  
    <div class="bd">  
        <p>文章</p>  
    </div>  
</div>
```

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
}  
  
<div id="nav">  
    <div class="bd">  
        <p>選單</p>  
    </div>  
</div>  
  
  
<div id="article">  
    <div class="bd">  
        <p>文章</p>  
    </div>  
</div>
```

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
}  
  
#nav {  
    /* 針對 id="nav" 設定樣式 */  
}  
.bd {  
  
<div id="nav">  
    <div class="bd">  
        <p>選單</p>  
    </div>  
</div>  
  
<div id="article">  
    <div class="bd">  
        <p>文章</p>  
    </div>  
</div>
```

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
}  
  
<div id="nav">  
    <div class="bd">  
        <p>選單</p>  
    </div>  
</div>  
  
#nav {  
    /* 針對 id="nav" 設定樣式 */  
}  
.bd {  
    /* 針對所有 class="bd" 設定樣式 */  
}  
  
<div id="article">  
    <div class="bd">  
        <p>文章</p>  
    </div>  
</div>
```

# 常用選擇器

Selector

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
}  
  
<div id="nav">  
    <div class="bd">  
        <p>選單</p>  
    </div>  
</div>  
  
#nav {  
    /* 針對 id="nav" 設定樣式 */  
}  
.bd {  
    /* 針對所有 class="bd" 設定樣式 */  
}  
  
<div id="article">  
    <div class="bd">  
        <p>文章</p>  
    </div>  
</div>
```

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
}  
  
<div id="nav">  
    <div class="bd">  
        <p>選單</p>  
    </div>  
</div>  
  
<div id="article">  
    <div class="bd">  
        <p>文章</p>  
    </div>  
</div>  
  
#nav {  
    /* 針對 id="nav" 設定樣式 */  
}  
.bd {  
    /* 針對所有 class="bd" 設定樣式 */  
}  
div.bd {
```

# 常用選擇器

Selector

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
}  
  
#nav {  
    /* 針對 id="nav" 設定樣式 */  
}  
.bd {  
    /* 針對所有 class="bd" 設定樣式 */  
}  
  
div.bd {  
    /* 針對所有 div class="bd" 設定樣式 */  
}  
  
<div id="nav">  
    <div class="bd">  
        <p>選單</p>  
    </div>  
</div>  
  
<div id="article">  
    <div class="bd">  
        <p>文章</p>  
    </div>  
</div>
```

# 常用選擇器

Selector

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
}  
  
#nav {  
    /* 針對 id="nav" 設定樣式 */  
}  
.bd {  
    /* 針對所有 class="bd" 設定樣式 */  
}  
  
div.bd {  
    /* 針對所有 div class="bd" 設定樣式 */  
}  
  
<div id="nav">  
    <div class="bd">  
        <p>選單</p>  
    </div>  
</div>  
  
<div id="article">  
    <div class="bd">  
        <p>文章</p>  
    </div>  
</div>
```

# 常用選擇器

Selector

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
}  
  
#nav {  
    /* 針對 id="nav" 設定樣式 */  
}  
.bd {  
    /* 針對所有 class="bd" 設定樣式 */  
}  
  
div.bd {  
    /* 針對所有 div class="bd" 設定樣式 */  
}  
  
#nav .bd {
```

```
<div id="nav">  
    <div class="bd">  
        <p>選單</p>  
    </div>  
</div>  
  
<div id="article">  
    <div class="bd">  
        <p>文章</p>  
    </div>  
</div>
```

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```

<div id="nav">
  <div class="bd">
    <p>選單</p>
  </div>
</div>

<div id="article">
  <div class="bd">
    <p>文章</p>
  </div>
</div>
```

p {  
 /\* 針對所有 p 標籤設定樣式 \*/  
 }  
 #nav {  
 /\* 針對 id="nav" 設定樣式 \*/  
 }  
 .bd {  
 /\* 針對所有 class="bd" 設定樣式 \*/  
 }  
 div.bd {  
 /\* 針對所有 div class="bd" 設定樣式 \*/  
 }  
 #nav .bd {  
 /\* 只針對所有 id="nav" 下的 class="bd" 設定樣式 \*/  
 }

# 常用選擇器

Selector

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {  
    /* 針對所有 p 標籤設定樣式 */  
}  
  
#nav {  
    /* 針對 id="nav" 設定樣式 */  
}  
.bd {  
    /* 針對所有 class="bd" 設定樣式 */  
}  
  
div.bd {  
    /* 針對所有 div class="bd" 設定樣式 */  
}  
  
#nav .bd {  
    /* 只針對所有 id="nav" 下的 class="bd" 設定樣式 */  
}
```

# 常用選擇器

Selector

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```
p {
    /* 針對所有 p 標籤設定樣式 */
}

#nav {
    /* 針對 id="nav" 設定樣式 */
}

.bd {
    /* 針對所有 class="bd" 設定樣式 */
}

div.bd {
    /* 針對所有 div class="bd" 設定樣式 */
}

#nav .bd {
    /* 只針對所有 id="nav" 下的 class="bd" 設定樣式 */
}

#nav, #article {
```

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```

p {
    /* 針對所有 p 標籤設定樣式 */
}

#nav {
    /* 針對 id="nav" 設定樣式 */
}

.bd {
    /* 針對所有 class="bd" 設定樣式 */
}

div.bd {
    /* 針對所有 div class="bd" 設定樣式 */
}

#nav .bd {
    /* 只針對所有 id="nav" 下的 class="bd" 設定樣式 */
}

#nav, #article {
    /* 同時針對 id="nav" 與 id="article" 設定相同的樣式 */
}

<div id="nav">
    <div class="bd">
        <p>選單</p>
    </div>
</div>

<div id="article">
    <div class="bd">
        <p>文章</p>
    </div>
</div>

```

# 常用選擇器

*Selector*

在 HTML 中、可針對所有的標籤設定 id 與 class 屬性，除了給予 HTML 更好的語意外，也能方便我們設定 CSS 樣式

```

p {
    /* 針對所有 p 標籤設定樣式 */
}

#nav {
    /* 針對 id="nav" 設定樣式 */
}

.bd {
    /* 針對所有 class="bd" 設定樣式 */
}

div.bd {
    /* 針對所有 div class="bd" 設定樣式 */
}

#nav .bd {
    /* 只針對所有 id="nav" 下的 class="bd" 設定樣式 */
}

#nav, #article {
    /* 同時針對 id="nav" 與 id="article" 設定相同的樣式 */
}

```

# 選擇器類型

[http://www.w3.org/TR/CSS2/selector.html#pattern-matching \(CSS2\)](http://www.w3.org/TR/CSS2/selector.html#pattern-matching)

Pattern	Meaning	Described in section
*	Matches any element.	<a href="#">Universal selector</a>
E	Matches any E element (i.e., an element of type E).	<a href="#">Type selectors</a>
E F	Matches any F element that is a descendant of an E element.	<a href="#">Descendant selectors</a>
E > F	Matches any F element that is a child of an element E.	<a href="#">Child selectors</a>
E:first-child	Matches element E when E is the first child of its parent.	<a href="#">The :first-child pseudo-class</a>
E:link E:visited	Matches element E if E is the source anchor of a hyperlink of which the target is not yet visited (:link) or already visited (:visited).	<a href="#">The link pseudo-classes</a>
E:active E:hover E:focus	Matches E during certain user actions.	<a href="#">The dynamic pseudo-classes</a>
E:lang(c)	Matches element of type E if it is in (human) language c (the document language specifies how language is determined).	<a href="#">The :lang() pseudo-class</a>
E + F	Matches any F element immediately preceded by a sibling element E.	<a href="#">Adjacent selectors</a>
E[foo]	Matches any E element with the "foo" attribute set (whatever the value).	<a href="#">Attribute selectors</a>
E[foo="warning"]	Matches any E element whose "foo" attribute value is exactly equal to "warning".	<a href="#">Attribute selectors</a>
E[foo~="warning"]	Matches any E element whose "foo" attribute value is a list of space-separated values, one of which is exactly equal to "warning".	<a href="#">Attribute selectors</a>
E[lang = "en"]	Matches any E element whose "lang" attribute has a hyphen-separated list of values beginning (from the left) with "en".	<a href="#">Attribute selectors</a>
DIV.warning	<i>Language specific.</i> (In HTML, the same as DIV[class~="warning"].)	<a href="#">Class selectors</a>
E#myid	Matches any E element with ID equal to "myid".	<a href="#">ID selectors</a>

CSS3 支援更多：[http://www.w3schools.com/cssref/css\\_selectors.asp](http://www.w3schools.com/cssref/css_selectors.asp)

# 屬性

# 屬性

```
選擇器 {  
    屬性名稱1 : 屬性值;  
    屬性名稱2 : 屬性值;  
    屬性名稱3 : 屬性值;  
}
```

# 屬性

用大括號包覆



```
選擇器 {  
    屬性名稱1 : 屬性值;  
    屬性名稱2 : 屬性值;  
    屬性名稱3 : 屬性值;  
}
```

# 屬性

用大括號包覆

選擇器 {

屬性名稱1：屬性值；  
屬性名稱2：屬性值；  
屬性名稱3：屬性值；

}

半型冒號區隔屬性名稱與屬性值

# 屬性

用大括號包覆

選擇器 {

屬性名稱1：屬性值；

屬性名稱2：屬性值；

屬性名稱3：屬性值；

}

半型冒號區隔屬性名稱與屬性值

← 以分號分隔不同屬性

# 屬性

用大括號包覆

選擇器 {

屬性名稱1：屬性值；  
屬性名稱2：屬性值；  
屬性名稱3：屬性值；

← 以分號分隔不同屬性

半型冒號區隔屬性名稱與屬性值

```
<div id="nav">
    <div class="bd">
        <p>選單</p>
    </div>
</div>
<div id="article">
    <div class="bd">
        <p>文章</p>
    </div>
</div>
```

# 屬性

用大括號包覆

選擇器 {

屬性名稱1：屬性值；  
 屬性名稱2：屬性值；  
 屬性名稱3：屬性值；

```
<div id="nav">
    <div class="bd">
        <p>選單</p>
    </div>
</div>
<div id="article">
    <div class="bd">
        <p>文章</p>
    </div>
</div>
```

+

```
#nav, #article {
    background: yellow;
    padding: 20px;
    width: 100px;
    margin-bottom: 10px;
}
#nav p {
    text-decoration: underline;
    color: blue;
}
#article p {
    color: red;
    font-weight: bold;
}
```

=

# 屬性

用大括號包覆

選擇器 {

屬性名稱1 : 屬性值;  
 屬性名稱2 : 屬性值;  
 屬性名稱3 : 屬性值;

}

半型冒號區隔屬性名稱與屬性值

← 以分號分隔不同屬性

```
<div id="nav">
    <div class="bd">
        <p>選單</p>
    </div>
</div>
<div id="article">
    <div class="bd">
        <p>文章</p>
    </div>
</div>
```

+

```
#nav, #article {
    background: yellow;
    padding: 20px;
    width: 100px;
    margin-bottom: 10px;
}
#nav p {
    text-decoration: underline;
    color: blue;
}
#article p {
    color: red;
    font-weight: bold;
}
```

選單

文章

# 屬性介紹

W3School 的屬性類別列表

<http://www.w3schools.com/css/>

## CSS Styling

[Styling](#)  
[Backgrounds](#)  
[Styling Text](#)  
[Styling Fonts](#)  
[Styling Links](#)  
[Styling Lists](#)  
[Styling Tables](#)

## CSS Box Model

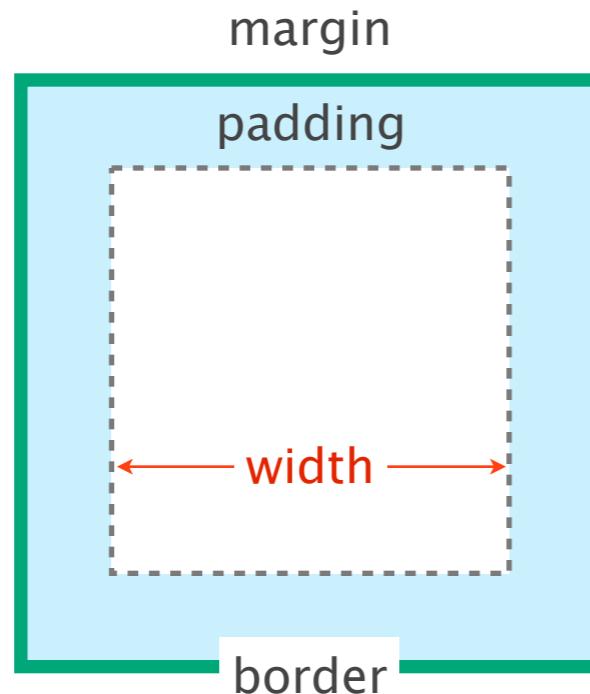
[CSS Box Model](#)  
[CSS Border](#)  
[CSS Outline](#)  
[CSS Margin](#)  
[CSS Padding](#)

## CSS Advanced

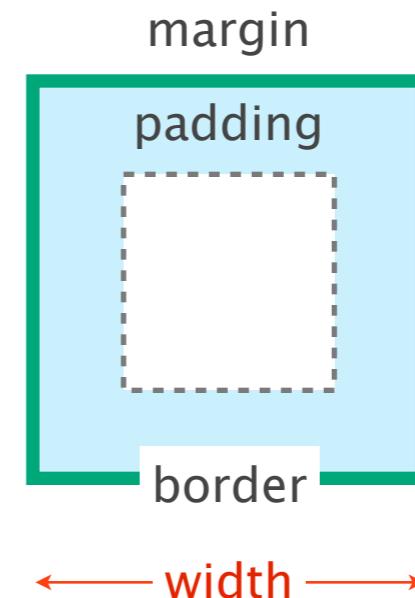
[CSS Grouping/Nesting](#)  
[CSS Dimension](#)  
[CSS Display](#)  
[CSS Positioning](#)  
[CSS Floating](#)  
[CSS Align](#)  
[CSS Pseudo-class](#)  
[CSS Pseudo-element](#)  
[CSS Navigation Bar](#)  
[CSS Image Gallery](#)  
[CSS Image Opacity](#)  
[CSS Image Sprites](#)  
[CSS Media Types](#)  
[CSS Attribute Selectors](#)

# Box Model

W3C 的標準盒模型



IE 怪異模式盒模型



width, height 算法完全不同、初學者需非常小心  
現今的網頁已經沒有怪異模式的需求了  
只要小心 DOCTYPE 的宣告即可避免

# 練習：替模組加上樣式

<http://f2e阶级.com/lab/css/module.html>

在 Linix 的 Command Line 模式下建置 Flex 開發環境 `font:bold 16px;`

大部分的人會覺得我有病吧，明明就有一個好用的 Flex Builder 可拖拉，為何還要用 CLI (Command Line Interface) 開發呢？請理解我是徒手 + 不查詢就寫 HTML / CSS / JavaScript 的硬派前端工程師出身、加上喜愛沈溺於敲鍵盤的爽感...

`background:#ffe;`

閱讀全文...

`color:#369;`

- 替此文章模組取個適合的 id
- 加上 hd, bd, ft 的 class
- 利用前面教的東西設定樣式

# CSS 浮動

排版的常用技巧

# 傳統排版

過去：常用 <table> 來做到下面的版型：

靠左文字

靠右文字

```
<table border="1">
  <tr>
    <td align="left">靠左文字</td>
    <td align="right">靠右文字</td>
  </tr>
</table>
```

# float 屬性

現在：使用 CSS 的 float 屬性來做到

靠左文字

靠右文字

```
<div id="masthead" style="border:solid 1px #000">
    <span style="float:left">
        靠左文字
    </span>
    <span style="float:right">
        靠右文字
    </span>
</div>
```

# 浮動產生的問題

但 float 之後的區塊是不會佔空間的  
實際情況會是：

靠左文字

靠右文字

**border 不會被文字的高度給撐開！**

# 對 IE 的解法

IE9 之前的工作  
在上一層設 zoom:1

```
<div id="masthead" style="zoom:1; border:solid 1px #000">  
  <span style="float:left">  
    靠左文字  
  </span>  
  <span style="float:right">  
    靠右文字  
  </span>  
</div>
```

# 通用解法

加上特別的 <br>

```
<div id="masthead" style="zoom:1; border:solid 1px #000">
    <span style="float:left">
        靠左文字
    </span>
    <span style="float:right">
        靠右文字
    </span>
    <br style="clear:both;">
</div>
```

BUT! 我們不應該為了樣式特別加一個 <br>

```
/* 作用是增加了一個有 clear:both 屬性的區塊 */
#masthead:after {
    content: "";
    display:block;           ← 與上面加的 <br> 效果相同
    clear:both;
}
```

# 練習：用 CSS 做浮動

<http://f2e阶级.com/lab/css/float.html>

border:solid 5px #ccc;

雅虎前端工程半日班 font:bold 197%;



IE : 在父節點設定 zoom ;

- 標題浮到左邊
- YUI 圖浮到右邊
- 替整個模組設 border
- 用前面的方法清除

其他瀏覽器 :

```
父節點:after {
  content: '';
  display:block;
  clear:both;
}
```

# CSS 背景

適當地將 `<img>` 改用 CSS 背景可讓讀取更順暢

# 在 Y! 所有裝飾性的圖都是 CSS 背景圖

設為首頁 保障+還本=Direct 好醫靠

11月16日星期天 台北 ☁ 19°C ~ 22°C

**YAHOO!** 奇摩

網頁 | 知識+ | 圖片 | 影片 | 部落格 | 商家 | 字典 | 商品 | 細分

熱門：表情符號 遊戲區 減肥 新社 吃到飽 田中千繪 楓之谷 賽車 隋棠

伊能靜部落格 | 搜尋榜 | 頁面選擇

服務列表

- 資訊
- 新聞
- 運動 NEW
- 股市
- 理財
- 知識+
- 生活+
- 汽車
- 工作
- 房地產
- 無名影音
- 字典
- 時尚
- 塑身健康
- 氣象
- 地圖
- 上網方案
- 下載
- 公益
- 企業
- 開店
- 關鍵字
- 網路行銷
- My»
- 消費
- 拍賣
- 購物通
- 購物中心
- ATM
- 社群
- 部落格
- 家族
- 交友
- 信箱<sup>2.0</sup>
- 即時通訊
- 無名小站
- 娛樂
- 圖鈴簡訊
- 遊戲
- 音樂
- 電影
- 星座算命
- 手機上網
- 我的
- 捷徑列
- 迷你筆
- 無名相簿

焦點新聞

- 運動
- 娛樂
- 國際
- 新奇

禁食5天 扁強制送亞東醫院  
陳前總統被收押在土城看守所，由於...

澳研發皮膚癌疫苗 5年上市  
澳洲的子宮頸癌疫苗發明者佛雷塞教...

快訊 一年一次送巧連智全套教材  
熱門 半年就精通1000個英文單字

josephj6802, 您好！ 登出

信箱 知識+ 拍賣

Yahoo奇摩關鍵字廣告

商品全部售完

讓你成為超人氣賣家！

0800-086-868 業績飆高就靠它

點我~前往活動！

怪物總動員 楓之谷

破盤 全透視NB 日本裝靚車 輕感孕洋 名模無暇腿 娥們泡湯趣

免費報名 年滿16歲/在職在學/失業轉業人士 限50名！

立即洽詢

97年產業職訓課程招生中

浪漫宮廷風襯衫 時尚百搭長版衣 大甜心人氣熱賣 雙排扣毛料外套

日系原色單寧褲 雙色刷毛連帽T 型男混搭針織衫 潮男素T下殺99

水晶鑽時尚男錶 經典雙戒環對鍊 潮流圍巾包199 簡約復古騎士靴

秋冬滋潤護手霜 超神奇粉刺剋星 保暖高領孕罩衫 嫩白保濕夯面膜

PayEasy 購物專區

牛爾青木瓜175 一夜醒來粉刺消 牛爾基礎組9折

8hr好氣色粉撲 Kevin彩妝全9折 燃23吋蠻腰美胸

kevin推薦2穿靴 靴子必搭瘦腿襪 入冬外套Top10

簡單樂活：跋山涉水輕鬆騎 從三坑老街或大溪橋

單車省錢環保行程推薦 宅男勇敢貼出告白宣言

理財專區 » 更多

# 在 Y! 所有裝飾性的圖都是 CSS 背景圖

設為首頁 保障+還本=Direct 好醫靠

11月16日星期天 台北 19°C ~ 22°C

**YAHOO!** 奇摩

網頁 知識+ 圖片 影片 部落格 商家 字典 商品 綜合 網頁搜尋

熱門：表情符號 遊戲區 減肥 新社 吃到飽 田中千繪 楓之谷 賽車 隋棠 伊能靜部落格 | 搜尋榜 頁面選擇

服務列表 My»

資訊	消費
新聞	<b>拍賣</b>
運動	購物通
股市	購物中心
理財	ATM
知識+	社群
生活+	部落格
汽車	家族
工作	交友
房地產	信箱 <sup>2.0</sup>
無名影音	即時通訊
字典	<b>無名小站</b>
時尚	娛樂
塑身健康	<b>圖鈴簡訊</b>
氣象	遊戲
地圖	音樂
上網方案	電影
下載	星座算命
公益	手機上網
企業	我的
開店	捷徑列
關鍵字	迷你筆
網路行銷	無文煩惱

焦點新聞 運動 娛樂 國際 新奇

**禁食5天 扁強制送亞東醫院**  
陳前總統被收押在土城看守所，由於...

**澳研發皮膚癌疫苗 5年上市**  
澳洲的子宮頸癌疫苗發明者佛雷塞教...

一年一次送巧連智全套教材  
半年就精通1000個英文單字

點我~前往活動！ 怪物總動員 楓之谷

格紋毛背心 淘氣美肩 搭靴聖品褲 水漾圓臘包 朵結娃娃鞋

免費報名 年滿16歲/在職在學/失業轉業人士 限50名！ 立即洽詢

97年產業職訓課程招生中

浪漫宮廷風襯衫 時尚百搭長版衣 大甜心人氣熱賣 雙排扣毛料外套

水晶鑽時尚男錶 經典雙戒環對鍊 潮流圓臘包199 簡約復古騎士靴

日系原色單寧褲 雙色刷毛連帽T 型男混搭針織衫 潮男素T下殺99

秋冬滋潤護手霜 超神奇粉刺剋星 保暖高領孕罩衫 嫩白保濕夯面膜

PayEasy 購物專區

牛爾青木瓜175 一夜醒來粉刺消 牛爾基礎組9折 kevin推薦2穿靴 靴子必搭瘦腿襪 入冬外套Top10

8hr好氣色粉撲 Kevin彩妝全9折 燃23吋蠻腰美胸

Copyright © 2012 FITPI. All rights reserved.

# 抉擇方式

何時該用背景圖？何時該用 img 標籤？

# 抉擇方式

何時該用背景圖？何時該用 img 標籤？

- ◎ 對於裝飾性、可有可無的圖片應採用背景圖。

# 抉擇方式

何時該用背景圖？何時該用 img 標籤？

- 對於裝飾性、可有可無的圖片應採用背景圖。
  - ❖ 例如圓角、小 Icon 等即使消失也不會造成無法閱讀的圖片。

# 抉擇方式

何時該用背景圖？何時該用 img 標籤？

- 對於裝飾性、可有可無的圖片應採用背景圖。
  - ❖ 例如圓角、小 Icon 等即使消失也不會造成無法閱讀的圖片。
- 而對於內容的一部分，則應該使用 img 標籤。

# 抉擇方式

何時該用背景圖？何時該用 img 標籤？

- 對於裝飾性、可有可無的圖片應採用背景圖。
  - ❖ 例如圓角、小 Icon 等即使消失也不會造成無法閱讀的圖片。
- 而對於內容的一部分，則應該使用 img 標籤。
  - ❖ 例如食記中的餐點圖片、網頁上的廣告、用圖片做的標題、網站 Logo 都是內容的一部分。

# 語法

**background:** [背景色]  
[url(圖檔路徑)]  
[x, y 位置]  
[重複];

```
background:#369 url(ico_yui.gif) left top repeat-x;  
0 100% repeat-y  
0 -12px no-repeat
```

或

```
background-color:#369;  
background-image:url(ico_yui.gif);  
background-position:left top;  
background-repeat:repeat-x;
```

# 練習：CSS 背景

<http://f2e阶级.com/lab/css/background.html>

選單 1 | 選單 2 | 選單 3 | 選單 4

- 設定背景圖為 `bg_nav.png`

# CSS Sprites

把多張背景圖合併成一張

雅虎奇摩首頁的 Sprites

<http://l.yimg.com/f/i/tw/hp/spirit1210041/sprite.png>

sprite.png

0, 0		background: url(sprite.png) 0 0 no-repeat;
0, 30		background: url(sprite.png) 0 -30px no-repeat;
0, 55		background: url(sprite.png) 0 -55px no-repeat;
0, 90		background: url(sprite.png) 0 -90px no-repeat;
0, 115		background: url(sprite.png) 0 -115px no-repeat;
0, 145		background: url(sprite.png) 0 -145px no-repeat;

# Sprites 的優缺點

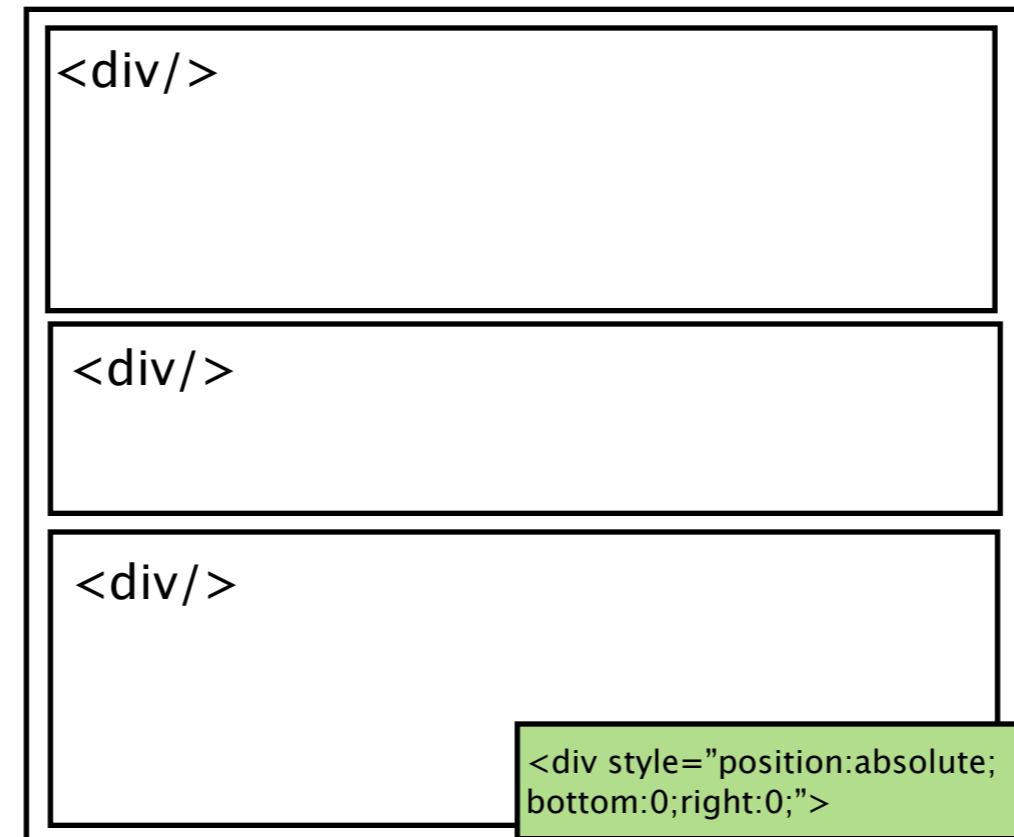
- 好處：有效減少網路請求數量
  - ❖ 太多請求對網頁的顯示是有影響的。
- 壞處：難以維護
  - ❖ x, y 的位置得打開影像處理軟體查找。
- 已經有軟體可以自動化處理 :)

# CSS 位置

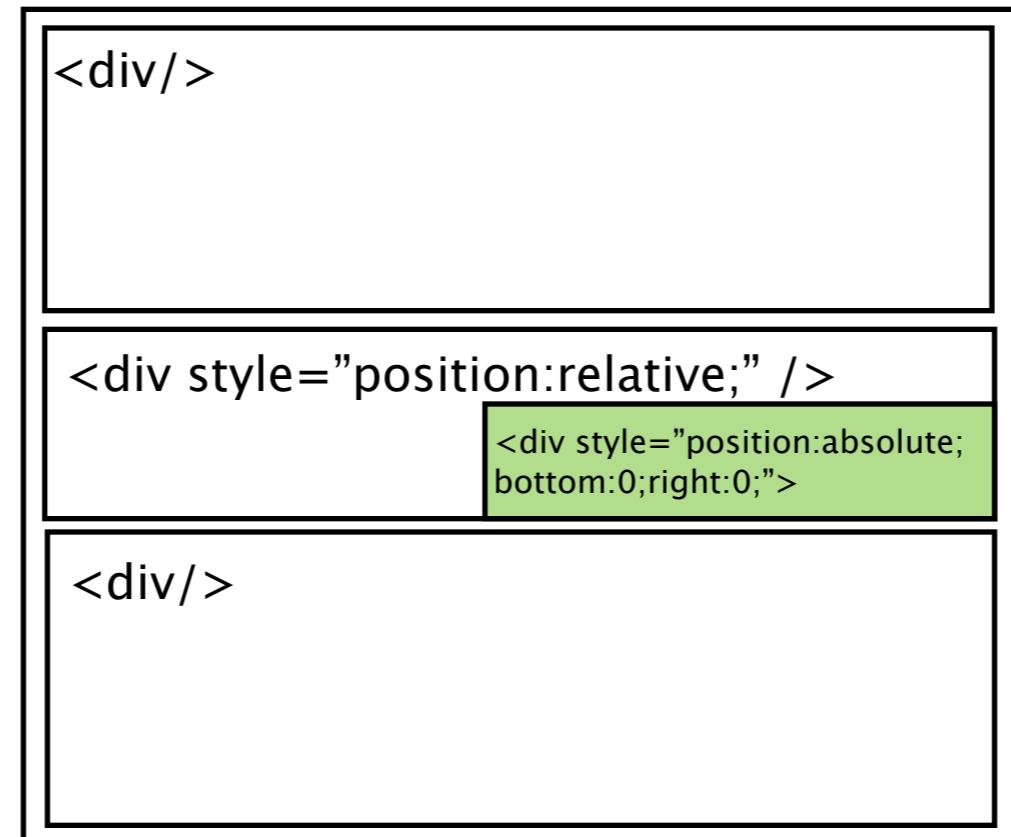
不管是做圓角、位置微調都會用到



什麼都沒設時 (position:static;)



當對某個區塊設了 position:absolute; 時  
就脫離原有區塊的範圍、不佔空間



對母區塊設定 position:relative;  
則會限制住 position:absolute; 的子區塊



# ○ position

## ◎ position

- ❖ static 預設

## ◎ position

- ❖ static 預設
- ❖ absolute 絶對定位

## ◎ position

- ❖ static 預設
- ❖ absolute 絶對定位
- ❖ relative 相對定位

## ◎ position

- ❖ static 預設
- ❖ absolute 絶對定位
- ❖ relative 相對定位
- ❖ fixed

## ○ position

- ❖ static 預設
- ❖ absolute 絶對定位
- ❖ relative 相對定位
- ❖ fixed

## ○ top, right, bottom, left 的配合

## ○ position

- ❖ static 預設
- ❖ absolute 絶對定位
- ❖ relative 相對定位
- ❖ fixed

## ○ top, right, bottom, left 的配合

- ❖ 在目前的位置往下移 5px

position:relative; top:5px;

## ○ position

- ❖ static 預設
- ❖ absolute 絶對定位
- ❖ relative 相對定位
- ❖ fixed

## ○ top, right, bottom, left 的配合

- ❖ 在目前的位置往下移 5px  
`position:relative; top:5px;`
- ❖ 整份文件中的右下角：  
`position:absolute; right:0; bottom:0;`

## ○ position

- ❖ static 預設
- ❖ absolute 絶對定位
- ❖ relative 相對定位
- ❖ fixed

## ○ top, right, bottom, left 的配合

- ❖ 在目前的位置往下移 5px

`position:relative; top:5px;`

- ❖ 整份文件中的右下角：

`position:absolute; right:0; bottom:0;`

- ❖ 可視區域的最下方 (IE6不支援) :

`position:fixed;bottom:0;`

# 練習：CSS 位置

<http://f2e阶级.com/lab/position.html>

1. 將 li 設定為 position:relative;
2. 將 img 設定為 position:absolute; 並利用 top, right 做微調



# CSS Hack

針對不同瀏覽器做樣式處理

# 常用 CSS Hack

大多是為了 IE 特別修正，其他瀏覽器較少問題

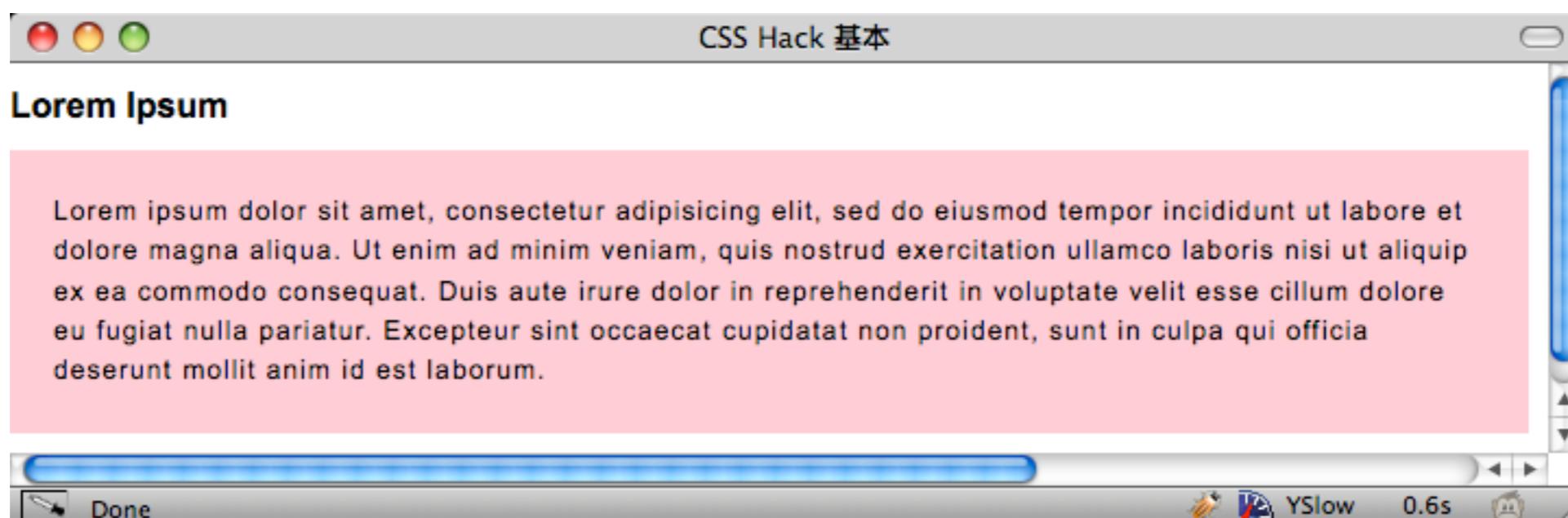
```
#article .bd {  
    background:blue;                      All Browsers  
    background:purple \0;                  IE9, IE8  
    background:pink \9;                   IE8, IE7, IE6  
    *background:green;                   IE 7, IE 6  
    _background:red;                     IE 6  
}
```

IE 6, IE7, IE8, IE9, 其他瀏覽器會有不同的底色

# 練習：CSS Hack

<http://f2e阶级.com/lab/css-hack.html>

讓 #article .bd 在 IE9 為 purple、IE8 為 green、IE7 為 red，其他瀏覽器為 pink。



# 進階知識與工具

- Reset CSS - 將樣式歸零
- CSS Grids - CSS 排版方式
- SASS / Compass - 必學工具
- CSS 3 必學屬性
- Twitter Bootstrap - 工程師也有好介面
- CSS Code Convention - 書寫規範
- CSS 繪圖

# Reset CSS

- HTML 標籤本身都有預設樣式，但是每個瀏覽器都有差異，加上大多不符合我們的需求。
- HTML 是結構、不應該有任何樣式。
- 那麼我們就歸零吧、讓每個標籤的外觀都一模一樣。
- 專業！所有的樣式都由自己重新定義。

# Reset CSS 效果

HTML with no CSS defined	HTML with only CSS Reset
<b>Heading 1</b>	Heading 1 A paragraph: Lorem ipsum dolor sit amet, pharetra libero nec mi tempor sed blandit
A paragraph: Lorem ipsum dolor sit amet, vel magna luctus malesuada at sit amet le platea dictumst.	Heading 2 A blockquote: Aliquam laoreet diam et lig tincidunt augue eu ullamcorper.
<b>Heading 2</b>	Heading 3 unordered one unordered two unordered three ordered one ordered two ordered three
A blockquote: Aliquam laoreet dia faucibus vitae egestas at, porttitor	Heading 4 Nam eu odio nec neque vestibulum conse sed posuere neque pharetra. Nam quis tei In tempor fringilla neque at tempor. Intege orci luctus et ultrices posuere cubilia Cura in cursus libero mi nec tellus. Sed iaculis e vestibulum rutrum leo semper. Nullam iaci Duis dignissim sem id nisl hendrerit id eius himenaeos. Maecenas rhoncus semper au velit, non tincidunt metus. Nulla tristique n semper purus vitae lorem placerat dapibus Duis ac ligula vitae dui euismod venenatis facilisis sit amet, scelerisque sed lacus. Fu condimentum imperdiet. Curabitur quis tel Phasellus in magna dui. Sed tellus eros, tin aptent taciti sociosqu ad litora torquent pe accumsan, risus facilisis blandit aliquam, m sit amet molestie urna imperdiet. Nam por laoreet, ante quam condimentum odio, noi
<b>Heading 3</b>	<ul style="list-style-type: none"> <li>• unordered one</li> <li>• unordered two</li> <li>• unordered three</li> </ul> <ol style="list-style-type: none"> <li>1. ordered one</li> <li>2. ordered two</li> <li>3. ordered three</li> </ol>
<b>Heading 4</b>	Nam eu odio nec neque vestibulum cons

# Reset CSS 的選擇

- Eric Meyer

<http://meyerweb.com/eric/tools/css/reset/>

- YUI CSS Reset

<http://yuilibrary.com/yui/docs/cssreset/>

直接放在頁面上就生效

```
<link rel="stylesheet" type="text/css"  
      href="http://yui.yahooapis.com/3.7.2/build/cssreset/cssreset-min.css">
```

# CSS Grids

# CSS Grids

- Grids 是指整個網頁的排版。

# CSS Grids

- Grids 是指整個網頁的排版。
- CSS Float 會是排版的一種作法。

# CSS Grids

- Grids 是指整個網頁的排版。
- CSS Float 會是排版的一種作法。
- 不建議自己開發排版，細節太多。

# CSS Grids

- Grids 是指整個網頁的排版。
- CSS Float 會是排版的一種作法。
- 不建議自己開發排版，細節太多。
- 再簡單的網頁都應該引用穩定的 Grids 系統。

# 960gs

<http://960.gs/>

# 960gs

<http://960.gs/>

**Gantry Framework — 12 col**

What Is Gantry? News Documentation Support Download

**Gantry**  
Web Platform Theme Framework

How does Gantry make theming easier and more powerful?

Gantry is a comprehensive set of building blocks to enable the rapid development and realization of a design into a flexible and powerful web platform theme.



**Key Features**

- ✓ 960 Grid System
- ✓ Stunning Administrator Interface
- ✓ XML driven and with overrides for unprecedented levels of customization
- ✓ Per-menu-item level control over any configuration parameter
- ✓ 65 base module positions. Easily add more!
- ✓ 36 possible layout combinations for mainbody and sidebars
- ✓ Up to 3 sidebars for a total of 4 column layouts
- ✓ Many built-in features such as font-size, to-top smooth slider, IE8 warning

Get to know Gantry



**Pixel Dreams — 16 col**

menu

**SHOW GRID**

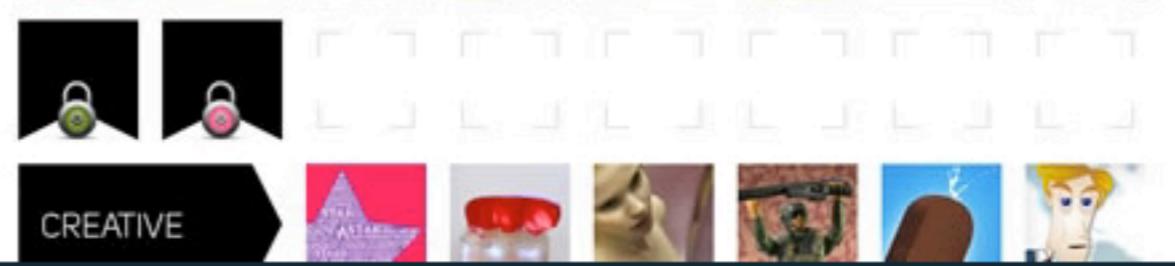
**PIXEL DREAMS**

**Hello!** We are the digital design agency super heroes. We love helping brands connect with their audience through engaging and intelligent communication design. **Need rescuing? Let's talk.**

---

**WORK**





**CREATIVE**

# 960gs

<http://960.gs/>

- 將網頁限制為 960 pixel。
- 依照 16 或 12 等分切格子。
- 好處是跟視覺設計師容易分工。
  - ❖ 不用再橋一堆對齊的問題。
- 壞處是原始碼的順序沒辦法依 SEO 調整。

# YUI 2 Grids

<http://developer.yahoo.com/yui/grids/>

The screenshot shows a web browser window displaying the Yahoo! Chimo Knowledge+ website. The page is structured using the YUI 2 Grids framework, which is highlighted by red arrows and boxes.

- Header:** A yellow box highlights the header area with the text "版面寬度 950px". The header includes the Yahoo! logo, a search bar, and navigation links like "會員登入" and "新使用者？立即註冊".
- Content Area:** The main content area is divided into two columns. The left column has a red border and contains a section titled "熱門戀愛煩惱問答" with a photo of a couple. The right column also has a red border and contains sections like "誰是天下第一團?", "結婚前分手的真正原因!", and "神奇的果汁機!". A yellow box at the bottom of this column says "平均分配 50%, 50%".
- Left Sidebar:** A sidebar on the left is outlined with a red border and contains a list of questions under categories like "信用卡", "髮型", etc., with a total of 10 items.
- Right Sidebar:** A sidebar on the right is outlined with a red border and contains sections like "信用卡" (with a credit card icon), "遊戲點數" (with a game controller icon), and a photo of a woman smiling.
- Footer:** The footer features a "Done" button and some performance-related icons like YSlow and FITPI.

http://josephj.com – josephj.com | 這樣做就對了

http://josephj.com/?page=2

# 這樣做就對了

**Recent Status**

2008-04-26 現在在石牌吃晚餐：馥香圓

2008-03-30 00:00:00

---

**[求生技能] 3/30 YDN 開發者論壇**

此篇文章最近更新時間為 2008-03-29 09:35:45 目前共有 0 篇留言

活動名稱：YDN 開發者論壇  
 時間：2008 年 3 月 30 日 (日) 下午 2:00~5:30  
 地點：新文明書店 – 台北市中山北路三段30號（民權西路農安街口，近民權西路捷運站）  
 形式：Workshop  
 主講人：YIK Engineer、Endorser  
 ---  
 這次擔綱演出的有兩個部份：  
 1. YUI 練功坊：15 分鐘，簡單介紹  
 2. ImageCropper 實際練習

**About Me**



[部落格作者] 蔣定宇、網路代號 josephj、  
 匿稱啊鳴。生活喜歡簡單跟自由、熱愛運動  
 與美食。工作對新技術有熱情、對程式有偏  
 執、愛把好東西與人分享。目前任職於  
 Yahoo! 奇摩、擔任前端工程師。

Inspect Edit : div#doc3.yui-t6 < body.index < html

Console HTML CSS Script DOM Net YSlow Options ▾ Style Layout DOM Options ▾

```
<html>
  <head>
  <body class="index yui-skin-sam">
    <div id="doc3" class="yui-t6">
      <div id="hd">
      <div id="bd">
      <div id="ft">
```

#doc4, #doc3 { josephjiang\_core... (line 31)
background: #FFFFFF none repeat scroll 0%;
margin-top: 5px;
}

#doc3 { josephjiang\_core... (line 8)
margin: auto 10px;
width: auto;
}

Find: Aptana Next Previous Highlight all Match case Done YSlow 8.038s

http://josephj.com - josephj.com | 這樣做就對了

http://josephj.com/?page=2

# 這樣做就對了

**About Me**



[部落格作者] 蔣定宇、網路代號 josephj、  
暱稱啊鳴。生活喜歡簡單跟自由、熱愛運動  
與美食。工作對新技術有熱情、對程式有偏  
執、愛把好東西與人分享。目前任職於  
Yahoo! 奇摩、擔任前端工程師。

**Recent Status**

2008-04-27 一個小時的 Presentation 真難拿捏，怒～

2008-03-30 00:00:00

---

**[求生技能] 3/30 YDN 開發者論壇**

此篇文章最近更新時間為 2008-03-29 09:35:45 | 目前共有 0 篇留言

活動名稱：YDN 開發者論壇  
時間：2008 年 3 月 30 日 (日) 下午 2:00~5:30  
地點：新文明書店 – 台北市中山北路三段30號（民權西路農安街口，近民權西路捷運站）  
形式：Workshop  
主講人：YIK Engineer、Endorser  
---  
這次擔綱演出的有兩個部份：  
1. YUI 練功坊：15 分鐘，簡單介紹  
2. ImageCropper 實際練習

Inspect Edit : div#doc3.yui-t6 < body.index < html

Console HTML CSS Script DOM Net YSlow Options ▾ Style Layout DOM Options ▾

```
<html>
  <head>
  <body class="index yui-skin-sam">
    <div id="doc4" class="yui-t3">
      <div id="hd">
      <div id="bd">
      <div id="ft">
```

#doc4, #doc3 { josephjiang\_core... (line 31)
background: #FFFFFF none repeat scroll 0%;
margin-top: 5px;
}

#doc4 { josephjiang\_core... (line 8)
width: 74.923em;
}

Find: Aptana Next Previous Highlight all Match case Done YSlow 8.038s

# YUI 2 Grids

<http://developer.yahoo.com/yui/grids/>

# YUI 2 Grids

<http://developer.yahoo.com/yui/grids/>

- 頁面寬度多種可選。

# YUI 2 Grids

<http://developer.yahoo.com/yui/grids/>

- 頁面寬度多種可選。
- 概念為「主要內容」與「側邊欄」。

# YUI 2 Grids

<http://developer.yahoo.com/yui/grids/>

- 頁面寬度多種可選。
- 概念為「主要內容」與「側邊欄」。
- 側邊欄可自由調整左右、各三種寬度。

# YUI 2 Grids

<http://developer.yahoo.com/yui/grids/>

- 頁面寬度多種可選。
- 概念為「主要內容」與「側邊欄」。
- 側邊欄可自由調整左右、各三種寬度。
- 內容寬度 = 頁面寬度 - 側邊欄寬度。

# YUI 2 Grids

<http://developer.yahoo.com/yui/grids/>

- 頁面寬度多種可選。
- 概念為「主要內容」與「側邊欄」。
- 側邊欄可自由調整左右、各三種寬度。
- 內容寬度 = 頁面寬度 - 側邊欄寬度。
- 即使重要的內容在右邊、也可以在原始碼中早點顯示。

# 練習：使用 Grid Builder

快速 DIV + CSS 版面產生器

<http://developer.yahoo.com/yui/grids/builder/>



# CSS3 與 Sprite 救星

一套命令列的軟體

讓你可以用 SASS 更結構化的方式寫 CSS  
並且提供了許多好用的功能 (mixin) !

<http://compass-style.org/>

# 巢狀式 CSS

a.scss - 你所撰寫及維護的

```

1 #navbar {
2   width: 80%;
3   height: 23px;
4   ul { list-style-type: none; }
5   li {
6     float: left;
7     a { font-weight: bold; }
8   }
9 }
10
11 .fakeshadow {
12   border: {
13     style: solid;
14     left: {
15       width: 4px;
16       color: #888;
17     }
18     right: {
19       width: 2px;
20       color: #ccc;
21     }
22   }
23 }
```

a.css - 工具自動產生的

```

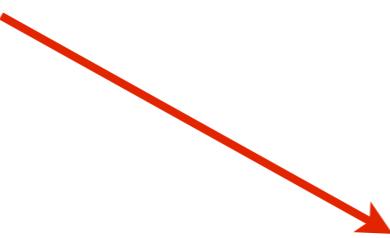
1 #navbar {
2   width: 80%;
3   height: 23px; }
4 #navbar ul {
5   list-style-type: none; }
6 #navbar li {
7   float: left; }
8 #navbar li a {
9   font-weight: bold; }
10
11 .fakeshadow {
12   border-style: solid;
13   border-left-width: 4px;
14   border-left-color: #888;
15   border-right-width: 2px;
16   border-right-color: #ccc; }
17
```

# 輕鬆寫 CSS3

**foo.scss**

```
@import "compass";

.foo {
  @include border-radius(5px);
  @include box-shadow(10px);
}
```



CSS3 支援度在每個瀏覽器還有一些差異，  
目前大多還是會冠上各自廠商的 prefix

使用 Compass，你只要寫一種就可以產出  
所有瀏覽器所需的 CSS，十分輕鬆！

**foo.css**

```
/* line 3, ../sass/foo.scss */
.foo {
  -webkit-border-radius: 5px;
  -moz-border-radius: 5px;
  -ms-border-radius: 5px;
  -o-border-radius: 5px;
  border-radius: 5px;
  -webkit-box-shadow: 10px;
  -moz-box-shadow: 10px;
  box-shadow: 10px;
}
```

# 安裝並設定 Scout

免費版跨平台 GUI 的 Compass 工具



# 練習 Compass

# 練習 Compass

- 請下載 <http://f2eclass.com/lab/css/compass.zip>

# 練習 Compass

- 請下載 <http://f2eclass.com/lab/css/compass.zip>
- 解壓縮到你的網站根目錄 C:\AppServ\www\

# 練習 Compass

- 請下載 <http://f2eclass.com/lab/css/compass.zip>
- 解壓縮到你的網站根目錄 C:\AppServ\www\
- 會有 sass, stylesheets, images 等目錄

# 練習 Compass

- 請下載 <http://f2eclass.com/lab/css/compass.zip>
- 解壓縮到你的網站根目錄 C:\AppServ\www\
- 會有 sass, stylesheets, images 等目錄
  - ❖ sass 是原始檔

# 練習 Compass

- 請下載 <http://f2eclass.com/lab/css/compass.zip>
- 解壓縮到你的網站根目錄 C:\AppServ\www\
- 會有 sass, stylesheets, images 等目錄
  - ❖ sass 是原始檔
  - ❖ stylesheets 是由 sass 轉出、實際被瀏覽器讀取的 css 檔

# 練習 Compass

- 請下載 <http://f2eclass.com/lab/css/compass.zip>
- 解壓縮到你的網站根目錄 C:\AppServ\www\
- 會有 sass, stylesheets, images 等目錄
  - ❖ sass 是原始檔
  - ❖ stylesheets 是由 sass 轉出、實際被瀏覽器讀取的 css 檔
  - ❖ images 下有我預先準備好的圖檔

# 練習 Compass

- 請下載 <http://f2eclass.com/lab/css/compass.zip>
- 解壓縮到你的網站根目錄 C:\AppServ\www\
- 會有 sass, stylesheets, images 等目錄
  - ❖ sass 是原始檔
  - ❖ stylesheets 是由 sass 轉出、實際被瀏覽器讀取的 css 檔
  - ❖ images 下有我預先準備好的圖檔
- Scout 設定好後啟動、檔案變動會自動產生。

# 練習 Compass

- 請下載 <http://f2eclass.com/lab/css/compass.zip>
- 解壓縮到你的網站根目錄 C:\AppServ\www\
- 會有 sass, stylesheets, images 等目錄
  - ❖ sass 是原始檔
  - ❖ stylesheets 是由 sass 轉出、實際被瀏覽器讀取的 css 檔
  - ❖ images 下有我預先準備好的圖檔
- Scout 設定好後啟動、檔案變動會自動產生。
- Sublime Text 2 得有 config.rb 配合 Build。

# 練習 Compass

- 請下載 <http://f2eclass.com/lab/css/compass.zip>
- 解壓縮到你的網站根目錄 C:\AppServ\www\
- 會有 sass, stylesheets, images 等目錄
  - ❖ sass 是原始檔
  - ❖ stylesheets 是由 sass 轉出、實際被瀏覽器讀取的 css 檔
  - ❖ images 下有我預先準備好的圖檔
- Scout 設定好後啟動、檔案變動會自動產生。
- Sublime Text 2 得有 config.rb 配合 Build。
- 有 normal.html, basic-sprite.html, auto-sprite.html 三個檔案請大家練習。

# Compass 參考連結

- **SASS** Syntactically Awesome StyleSheets

<http://sass-lang.com>

定義了基礎語法

- **Compass** CSS Authoring Framework

<http://compass-style.org/>

基於 SASS 所做的框架、請多參考這邊的文件

- **Scout** Compass and Sass without all hassle

<http://mhs.github.com/scout-app/>

跨平台且免費的 SASS 工具

# Twitter Bootstrap

<http://twitter.github.com/bootstrap/>

# Twitter Bootstrap

<http://twitter.github.com/bootstrap/>

- 工程師也能有好樣式

# Twitter Bootstrap

<http://twitter.github.com/bootstrap/>

- 工程師也能有好樣式
- 在頁面上引用外部 CSS、透過 Bootstrap 預先定義的 HTML 結構與 Class 名稱即可使用。例如 class="btn"

# Twitter Bootstrap

<http://twitter.github.com/bootstrap/>

- 工程師也能有好樣式
- 在頁面上引用外部 CSS、透過 Bootstrap 預先定義的 HTML 結構與 Class 名稱即可使用。例如 class="btn"
- 適合用在不需特別樣式的地方，例如 CMS 或 Prototype。但有自定樣式的地方，例如形象網站就不適合了。

# CSS3 必學屬性

# CSS3 必學屬性

- background: rgba 半透明

# CSS3 必學屬性

- background: rgba 半透明
- background-image: linear-gradient 漸層

# CSS3 必學屬性

- background: rgba 半透明
- background-image: linear-gradient 漸層
- border-radius 圓角

# CSS3 必學屬性

- background: rgba 半透明
- background-image: linear-gradient 漸層
- border-radius 圓角
- box-shadow 盒子的陰影

# CSS3 必學屬性

- background: rgba 半透明
- background-image: linear-gradient 漸層
- border-radius 圓角
- box-shadow 盒子的陰影
- text-shadow 文字的陰影

# CSS3 必學屬性

- background: rgba 半透明
- background-image: linear-gradient 漸層
- border-radius 圓角
- box-shadow 盒子的陰影
- text-shadow 文字的陰影
- transition 換場

# CSS 書寫規範

有一致的書寫規範，程式才能用得久

# CSS 書寫規範

有一致的書寫規範，程式才能用得久

- 絕對不以樣式命名 HTML 的 id 或 class 屬性。

# CSS 書寫規範

有一致的書寫規範，程式才能用得久

- 絕對不以樣式命名 HTML 的 id 或 class 屬性。
- class 與 id 命名採用小寫配合破折號連接。

# CSS 書寫規範

有一致的書寫規範，程式才能用得久

- 絕對不以樣式命名 HTML 的 id 或 class 屬性。
- class 與 id 命名採用小寫配合破折號連接。
- 屬性按照字母順序 (a-z) 做排序。

# CSS 書寫規範

有一致的書寫規範，程式才能用得久

- 絕對不以樣式命名 HTML 的 id 或 class 屬性。
- class 與 id 命名採用小寫配合破折號連接。
- 屬性按照字母順序 (a-z) 做排序。
- 選擇器後空一格放大括號開頭、大括號結尾占一行。

# CSS 書寫規範

有一致的書寫規範，程式才能用得久

- 絕對不以樣式命名 HTML 的 id 或 class 屬性。
- class 與 id 命名採用小寫配合破折號連接。
- 屬性按照字母順序 (a-z) 做排序。
- 選擇器後空一格放大括號開頭、大括號結尾占一行。
- 所有屬性皆單獨一行、以 4 個 Spaces 開頭。

# CSS 書寫規範

有一致的書寫規範，程式才能用得久

- 絕對不以樣式命名 HTML 的 id 或 class 屬性。
- class 與 id 命名採用小寫配合破折號連接。
- 屬性按照字母順序 (a-z) 做排序。
- 選擇器後空一格放大括號開頭、大括號結尾占一行。
- 所有屬性皆單獨一行、以 4 個 Spaces 開頭。
- 顏色碼一律採用小寫。

# CSS 書寫規範

有一致的書寫規範，程式才能用得久

- 絕對不以樣式命名 HTML 的 id 或 class 屬性。
- class 與 id 命名採用小寫配合破折號連接。
- 屬性按照字母順序 (a-z) 做排序。
- 選擇器後空一格放大括號開頭、大括號結尾占一行。
- 所有屬性皆單獨一行、以 4 個 Spaces 開頭。
- 顏色碼一律採用小寫。
- 屬性值後一律加上分號。

# CSS 書寫規範

有一致的書寫規範，程式才能用得久

- 絕對不以樣式命名 HTML 的 id 或 class 屬性。
- class 與 id 命名採用小寫配合破折號連接。
- 屬性按照字母順序 (a-z) 做排序。
- 選擇器後空一格放大括號開頭、大括號結尾占一行。
- 所有屬性皆單獨一行、以 4 個 Spaces 開頭。
- 顏色碼一律採用小寫。
- 屬性值後一律加上分號。
- 採用 HTML 模組化開發、對共用的部份要小心。

# CSS 書寫規範

有一致的書寫規範，程式才能用得久

- 絕對不以樣式命名 HTML 的 id 或 class 屬性。
- class 與 id 命名採用小寫配合破折號連接。
- 屬性按照字母順序 (a-z) 做排序。
- 選擇器後空一格放大括號開頭、大括號結尾占一行。
- 所有屬性皆單獨一行、以 4 個 Spaces 開頭。
- 顏色碼一律採用小寫。
- 屬性值後一律加上分號。
- 採用 HTML 模組化開發、對共用的部份要小心。

以上書寫規範只是建議，你可以自訂

# 課程回顧

# 課程回顧

- 請花點時間看過 CSS 的所有屬性。

# 課程回顧

- 請花點時間看過 CSS 的所有屬性。
- ID 選擇器：給予樣式設定命名空間。

# 課程回顧

- 請花點時間看過 CSS 的所有屬性。
- ID 選擇器：給予樣式設定命名空間。
- Class 選擇器：最常用來設定 CSS 樣式。

# 課程回顧

- 請花點時間看過 CSS 的所有屬性。
- ID 選擇器：給予樣式設定命名空間。
- Class 選擇器：最常用來設定 CSS 樣式。
- 以上兩者的命名都應該避免參雜樣式

# 課程回顧

- 請花點時間看過 CSS 的所有屬性。
- ID 選擇器：給予樣式設定命名空間。
- Class 選擇器：最常用來設定 CSS 樣式。
- 以上兩者的命名都應該避免參雜樣式
  - ❖ 例如：topnav, leftbar, red12px, bottom-ad, black-bar...

# 課程回顧

- 請花點時間看過 CSS 的所有屬性。
- ID 選擇器：給予樣式設定命名空間。
- Class 選擇器：最常用來設定 CSS 樣式。
- 以上兩者的命名都應該避免參雜樣式
  - ❖ 例如：topnav, leftbar, red12px, bottom-ad, black-bar...
- 選擇器需用結構來命名。

# 課程回顧

- 請花點時間看過 CSS 的所有屬性。
- ID 選擇器：給予樣式設定命名空間。
- Class 選擇器：最常用來設定 CSS 樣式。
- 以上兩者的命名都應該避免參雜樣式
  - ❖ 例如：topnav, leftbar, red12px, bottom-ad, black-bar...
- 選擇器需用結構來命名。
  - ❖ 例如：sidebar, nav, extra, ad-rec...

# 課程回顧

- 請花點時間看過 CSS 的所有屬性。
- ID 選擇器：給予樣式設定命名空間。
- Class 選擇器：最常用來設定 CSS 樣式。
- 以上兩者的命名都應該避免參雜樣式
  - ❖ 例如：topnav, leftbar, red12px, bottom-ad, black-bar...
- 選擇器需用結構來命名。
  - ❖ 例如：sidebar, nav, extra, ad-rec...
- Compass 讓寫 CSS 更結構化、Sprite 更自動。

# 課程回顧

- 請花點時間看過 CSS 的所有屬性。
- ID 選擇器：給予樣式設定命名空間。
- Class 選擇器：最常用來設定 CSS 樣式。
  - ❖ 例如：topnav, leftbar, red12px, bottom-ad, black-bar...
- 選擇器需用結構來命名。
  - ❖ 例如：sidebar, nav, extra, ad-rec...
- Compass 讓寫 CSS 更結構化、Sprite 更自動。
- Bootstrap 提供很棒的樣式，你的產品適合嗎？

# Q & A