



PHP+MySQL

互動式網頁程式設計班

[第五課] PHP高階語法：分割程式碼、小技巧、Cookie、檔案處理及正規表示法

國立台灣大學 資訊工程學系暨研究所

資訊系統訓練班

講師：鄔明學

將程式碼分割

- 有某些函數是常常會用到的，這時候若能將這些常用的函數寫成標準的格式，讓每個人或自己每次都能重複拿來使用，將會方便許多
- 把程式碼引入的方法：
 - `require()`
 - `include()`
 - `require_once()`
 - `include_once()`

利用require()引入檔案

- 函數：**require**(檔案)
- 回傳值：引入檔案裡所有的變數值
- 舉例：`require('function.php');`
- 將一些常用的函數、公式放在另一個檔案裡，是一個非常好的習慣，不但省去每次重新鍵入程式碼的時間，在攜帶上也十分方便，只要複製一個檔案，就可以給成千上萬的使用者使用

利用require()引入檔案的範例

```
1 <?php
2 $week = date('w');
3 $wk = array('日', '一', '二', '三', '四', '五', '六');
4 $week = $wk[$week];
5 $year = date('Y');
6 $month = date('n');
7 $day = date('j');
8 ?>
9
```



儲存成 now.php

主程式

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>require 範例</title>
5 </head>
6 <body>
7   <?php
8     require('now.php');
9
10    echo '今天是西元 ' . $year . ' 年<br>';
11    echo $month . ' 月<br>';
12    echo $day . ' 日<br>';
13    echo '星期 ' . $week;
14  ?>
15 </body>
16 </html>
```

利用include()引入檔案

- 函數：**include**(檔案)
- 回傳值：引入檔案裡所有的變數值
- 舉例：`include("function.php");`
- 用法與`require()`相同

利用include()引入檔案的範例 (1)

要被引入的程式

```
1 <?php
2 function game()
3 {
4     $getNum = rand(1,7);
5     return $getNum;
6 }
7 function show($place, $result)
8 {
9     echo '第 ' . $place . ' 個數為:' . $result . '<br><br>';
10 }
11 function check($a, $b, $c)
12 {
13     if($a == $b && $a == $c)
14     {
15         return true;
16     }
17 }
18 ?>
```



儲存成 game.php

利用include()引入檔案的範例 (2)

主程式

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>include 範例</title>
5 </head>
6 <body>
7   <?php
8     include('game.php');
9
10   for($i = 0; $i < 3; $i++)
11   {
12     $num[$i] = game();
13     show(($i+1), $num[$i]);
14   }
15   if (check($nm[0], $num[1], $num[2]))
16   {
17     echo "恭喜獨得獎金一千萬元<br>";
18   }
19   else
20   {
21     echo "也許下一次手氣會更好...<br>";
22   }
23   ?>
24 </body>
25 </html>
```

require() vs. include()

- `require()` 會將目標檔案的內容讀入，並且把自己本身代換成這些讀入的內容
 - 這個讀入與代換的動作發生在 PHP 引擎編譯程式碼的時候，而不是發生在 PHP 引擎開始執行編譯好的程式碼時
- `require()` 適合用來引入靜態的內容
 - 如版權宣告
- `include()` 則適合用來引入動態的程式碼
 - 程式內容會依其他程式碼而變動

include() 其他的能力

- 一般來說引用檔案使用 require 或 include 皆可
- 但若需經過條件判斷之後，才能決定是否引用該檔案的話，則只能使用 include
 - 例：

```
7  <?php
8  if (PHP_OS == 'WINNT')
9  {
10     include('windows.php');
11 }
12 else
13 {
14     include('other_os.php');
15 }
16 ?>
```

如果是 windows 作業系統就引用 windows.php，其他作業系統就引用 other_os.php

require_once

- require_once 的作用和 require 是幾乎相同的，唯一的差別在於：
 - require_once 會先檢查要引入的檔案是不是已經在該程式中的其他地方被引入過了；如果有的話，就不會再次重複引入該檔
 - 這項功能有時候是很重要的，比方說要引入的檔案裡面宣告了一些我們自行定義的函數，那麼如果在同一個程式重複引入這個檔案，在第二次引入的時候便會發生錯誤訊息，因為 PHP 不允許相同名稱的函數被重複宣告

include_once()

- 如同 `require_once`，`include_once` 會先檢查欲引入檔案的內容是不是在之前就已經引入過了；如果是的話，便不會再次重複引入同樣的內容

一個特殊的判斷式寫法

- 基本的判斷式：

```
<?php
if ($name == 'allan')
{
    echo 'Hi, Allan';
}
else
{
    echo 'Who are you?';
}
?>
```

- 其實可以被簡化成一行：

```
<?php
echo ($name == 'allan') ? 'Hi, Allan' : 'Who are you?';
?>
```

課堂練習 (5-1)

- 利用一維陣列存下使用帳號：
 - `$user = array('allan', 'bill', 'cindy', 'david', ...);`
- 利用二維陣列存下使用者密碼與email：
 - `$userData['allan']['password'] = 'php';`
 - `$userData['allan']['email'] = 'allan@csie.org';`
 - ...
- 寫一個 HTML 表單可以輸入使用者帳號與密碼，當帳號密碼比對正確時就秀出該使用者的 email address
- 將程式切成：
 - `login.php` - 主程式
 - `verify.php` - 驗證帳號密碼的 function
 - `user.php` - 存使用者資料的一維陣列和二維陣列

課堂練習 (5-1) 執行畫面

ID:

PW:



Hi! your email address is: allan@csie.org



```
1 <?php
2 $user = array(
3     'allan',
4     'bill',
5     'cindy'
6 );
7
8 $userData['allan'] = array(
9     'password' => '1234',
10    'email' => 'allan@csie.org'
11 );
12
13 $userData['bill'] = array(
14     'password' => '5678',
15     'email' => 'bill@csie.org'
16 );
17
18 $userData['cindy'] = array(
19     'password' => '0000',
20     'email' => 'cindy@csie.org'
21 );
22 ?>
```

如果登入者為: bill / 5678

Hi! your email address is: bill@csie.org

特殊的函數 – header()

- 重新導向網頁的寫法：
 - `header('Location:網址');`
- 讓網頁自動重新整裡的寫法：
 - `header('Refresh:時間');`

重寫導向網頁的寫法

- 函數：`header('Location:網址')`
- 回傳值：無
- 舉例：`header('Location:hello.htm');`
- 限制：
 - 不可放在文字輸出之後
 - 比方 `echo` 或是 `print`
 - 需在網頁載入之前(在`<html>`標籤之前)就執行
 - 也就是說通常要放在 `php` 程式的第一行

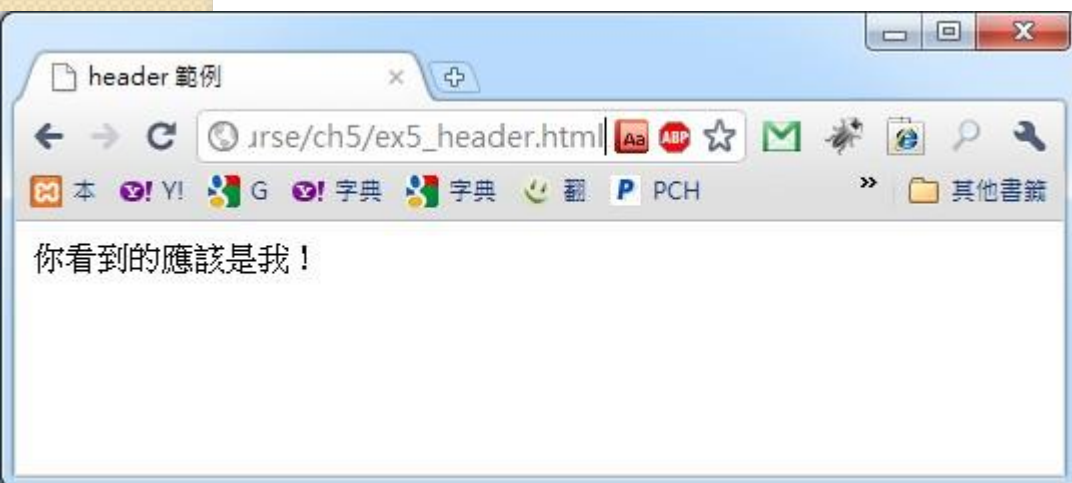

```
1 <?php
2 header('Location:ex5_header.html');
3 ?>
4
5 <html>
6 <head>
7   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
8   <title>header 範例</title>
9 </head>
10 <body>
11 你看不見我、你看不見我...
12 </body>
13 </html>
```

重新導向範例

ex5_header.php

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>header 範例</title>
5 </head>
6 <body>
7 你看到的應該是我!!
8 </body>
9 </html>
```

ex5_header.html



重新導向的應用 (1)

```
1 <?php
2 if(date('A') == 'AM')
3 {
4     header("Location:ex6_am.php");
5 }
6 else
7 {
8     header("Location:ex6_pm.php");
9 }
10 ?>
11
12 <html>
13 <head>
14     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
15     <title>header 範例</title>
16 </head>
17 <body>
18 你看不見我、你看不見我...
19 </body>
20 </html>
```

ex6_header.php



重新導向的應用 (2)

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>header 範例</title>
5 </head>
6 <body>
7 Hi! 現在是早上喔!!
8 </body>
9 </html>
```

ex6_am.php

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>header 範例</title>
5 </head>
6 <body>
7 Hi! 現在是下午喔!!
8 </body>
9 </html>
```

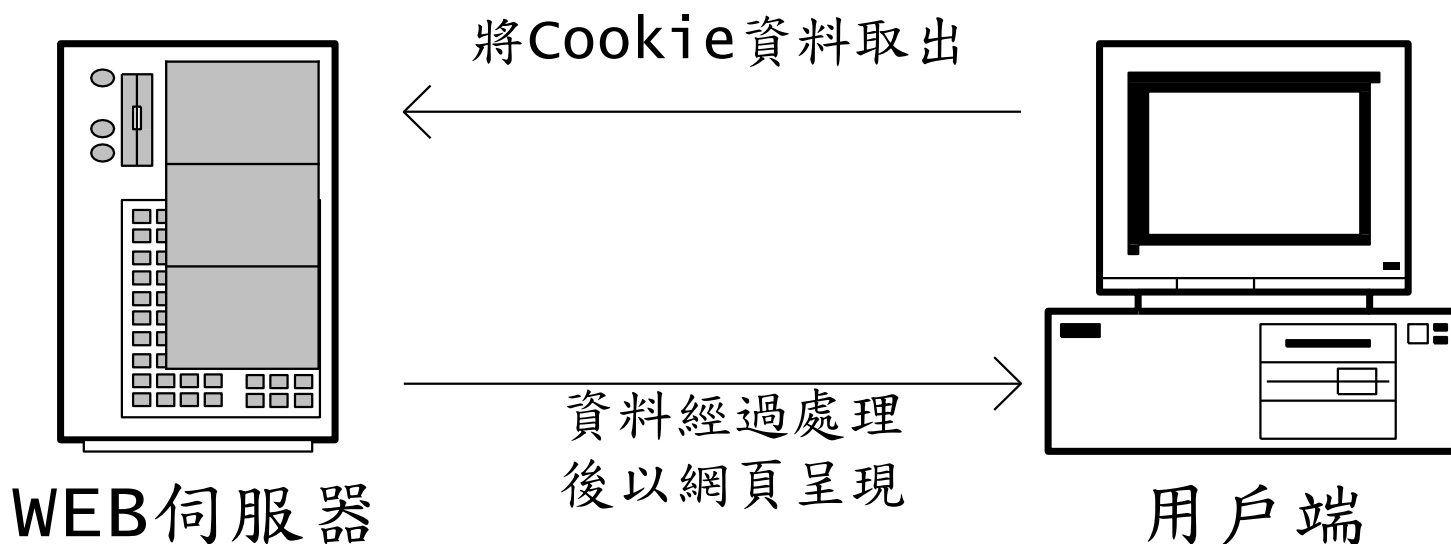
ex6_pm.php

讓網頁自動重新整裡的寫法

- 函數：`header('Refresh:時間(秒);URL=網址')`
- 回傳值：無
- 舉例：`header('Refresh:5;URL=hello.php');`
 - 則此頁過 5 秒後會自動轉移至 `hello.php`
- `refresh`的使用不可有空格

利用Cookie記錄用戶資料

- Cookie是資料存放在「用戶端」的技術

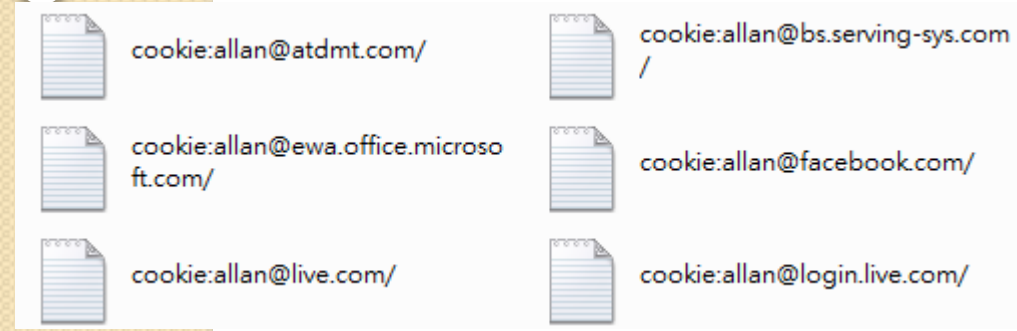


1

列印(P)
檔案(F)
縮放(Z) (100%)
安全性(S)
檢視下載(N)
管理附加元件
F12 開發者工具
前往訂選的網頁
網際網路選項
關於 Internet Explorer

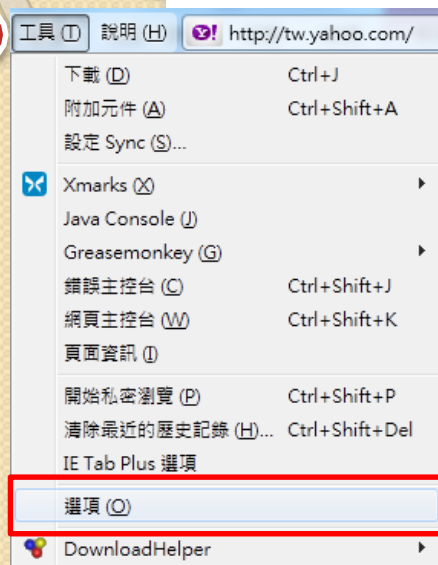
4

cookie: ...
cookie: ...
ft.com/
cookie: ...



在 Firefox 中的 Cookie

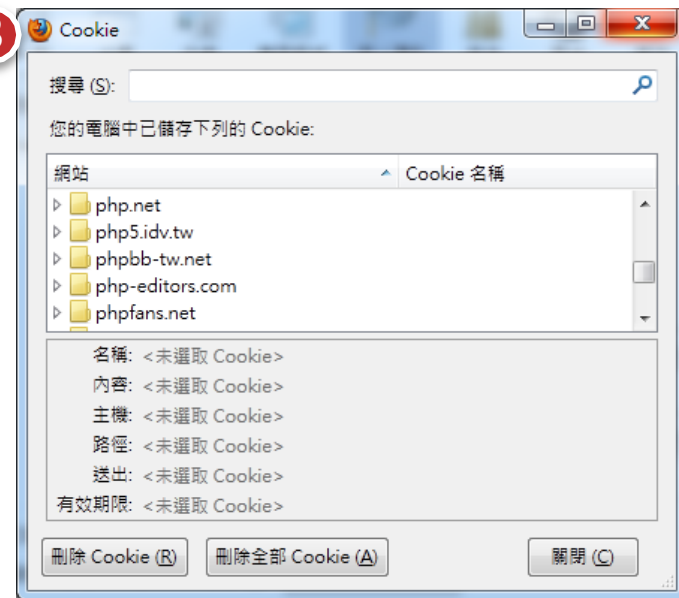
1



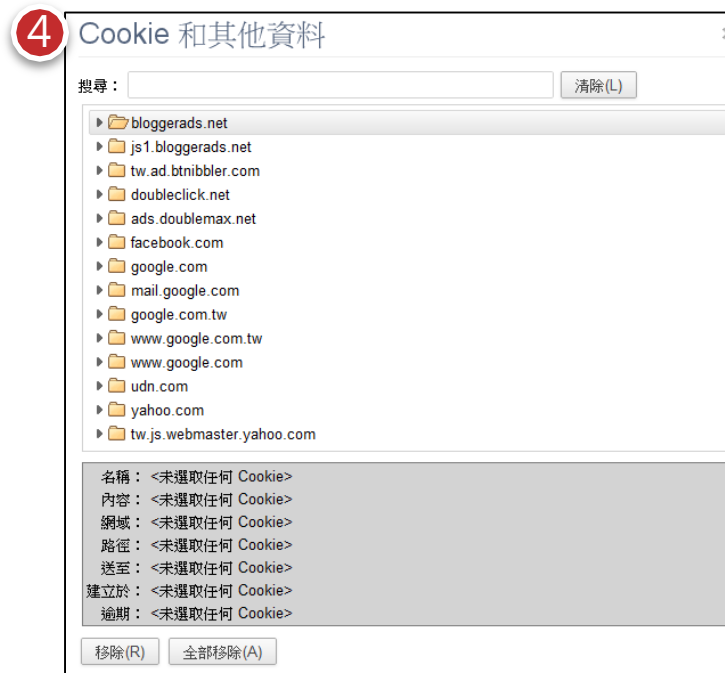
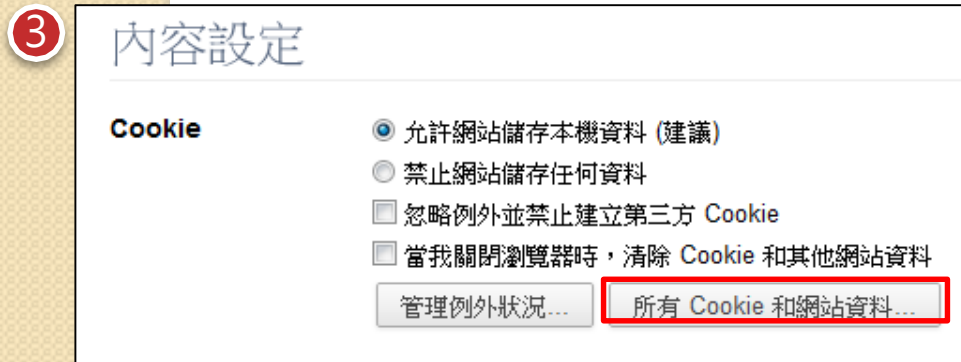
2



3



在 Chrome 中的 Cookie



Cookie的功能

- Cookie 可以紀錄一些微量的資料在使用者的瀏覽器上，以便使用者下次造訪時使用
 - 如：使用者的上站次數、上一次的上站日期
 - 或：把登入成功的使用者的帳號密碼記錄在 cookie 中，如此一來下次使用者造訪時就不需要再做一次登入網站的動作了
- 另一個重要應用場合是「購物車」
 - 使用者可能會在一段時間內在同一家網站的不同頁面中選擇不同的商品，這些資訊都會寫入 Cookie，以便在最後付款時一次付款

使用Cookie的限制

- 因為Cookie常常用來存取使用者的資訊，為了怕被拿來濫用，或是佔用太多硬碟空間，所以對Cookie做出了以下的限制：
 - 每個使用者的瀏覽器只能支援(存取)300個Cookie
 - 每個瀏覽器只能針對同一個伺服器存取20個Cookie
 - 每個Cookie的大小最多僅4k Bytes的容量
 - 大部份瀏覽器可以把Cookie的功能關掉，若關掉後Cookie就不能使用

Cookie函數

- 函數：`setCookie(name[,value] [,expire])`
- 回傳值：若順利將資料寫入Cookie，則回傳整數1
- 舉例：`setCookie('userid', 'allan');`
 - ※儲存的動作會在網頁載入之後才進行，也就是網頁全部載入之後才存入Cookie資料，所以要看儲存結果，**要等到下一回**才看得到
 - 所以通常會搭配 `header();` 函數做轉頁或更新

setCookie的使用方式

- 與header()的使用相同
 - 必須寫在<html>標記之前
 - 前面不可使用echo輸出資料
- 取得 Cookie 內容值的方式：
 - 例： `$_COOKIE['cookie名稱'];`
- 要刪除cookie的話，直接把要刪除cookie名稱的值設定為空白即可
 - 例： `setCookie('userid');`

Cookie的生命週期

- 以 `setCookie` 預設的寫法來說，如果將網頁關閉，再次開啟網頁時，會發現Cookie 的值會消失了
 - 因此 `setCookie` 第三個參數可設定Cookie的保留時間，是由絕對時間起算的秒數
 - 絕對時間：1970年1月1日
 - 以下例子為保留時間為60秒的例子
- ```
<?php
 $getTime = date('U');
 setCookie('userid', 'allan', $getTime+60);
?>
```

# 練習一下 (1)

- 寫入一組名叫 user 的 cookie 到瀏覽器中，內容值為 allan，存活60秒，关掉瀏覽器後再打開看看！

我現在還不知道你是誰，可是下次我就知道囉！

第一次  
瀏覽

Hi, 你是allan

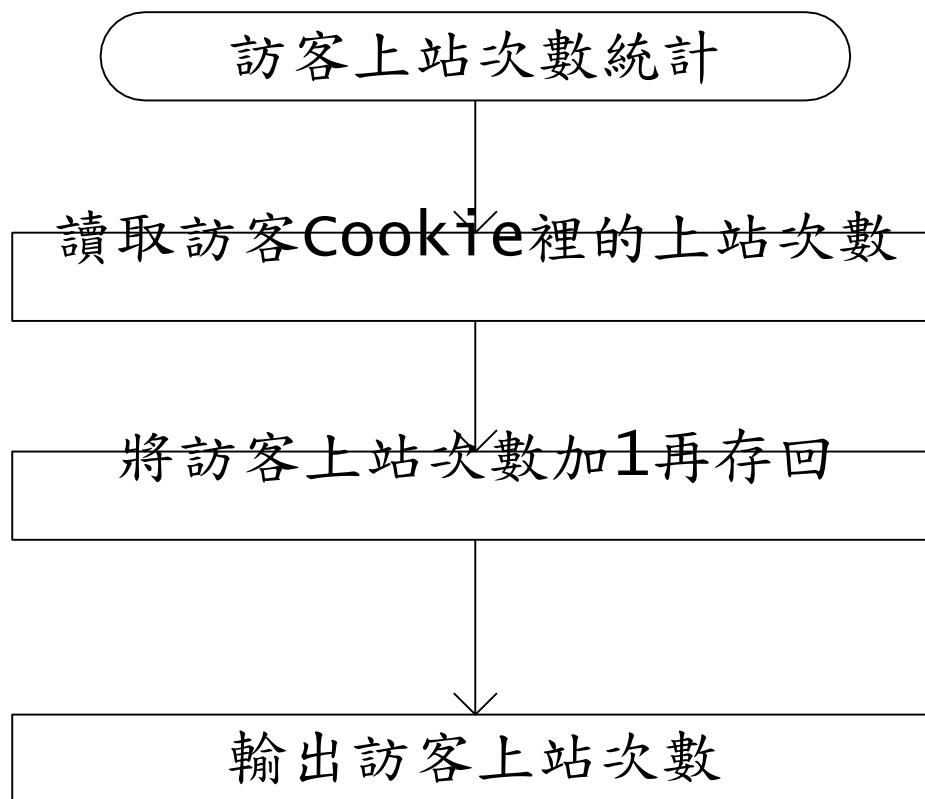
60秒內第  
二次瀏覽

# 練習一下 (2)

```
1 <?php
2 if ($_COOKIE['user'])
3 {
4 $name = 'Hi, 你是:' . $_COOKIE['user'];
5 }
6 else
7 {
8 setCookie('user', 'allan', date('U') + 60);
9 $name = '我現在還不知道你是誰，可是下次我就知道囉！';
10 }
11 ?>
12 <html>
13 <head>
14 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
15 <title>cookie 範例</title>
16 </head>
17 <body>
18 <?php
19 echo $name;
20 ?>
21 </body>
22 </html>
```

# 課堂練習 (5-2)

- 取得訪客上站次數





# 練習一下

- 按下廣告後開放檔案下載
  - 許多網站靠賣廣告看板、廣告連結營運
  - 而許多廣告商是以「點擊次數」來當做付費依據 (比方說被點擊一次付\$1)
  - 故有些網站為了生存而須讓使用者先點擊廣告後才開放部份增值服務，如檔案下載
  - 概念
    - 使用者進入網站後確認他/她是否有點過廣告
    - 有點過廣告的使用者會在 cookie 中存入一個「已點過廣告」的紀錄

# 按下廣告後開放檔案下載範例

- main.php:首頁
  - 提供三個連結，分別是「廣告連結」、「需先點廣告的檔案下載連結」、「不需先點廣告的檔案下載連結」
- adver.php:廣告商的連結
  - 瀏覽此頁後，自動在cookie加上一個瀏覽過的標記
- download.php:需要先點廣告的網頁
- free.php:不需先點廣告的網頁

# 檔案1 - main.php

～ 歡迎光臨檔案下載中心 ～

[贊助廠商廣告](#)

[付費檔案專區，請先點廣告再下載！](#)

[免費檔案專區，請自由參觀！](#)

```
1 <html>
2 <head>
3 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4 <title>cookie 應用 範例</title>
5 </head>
6 <body>
7 <?php
8 echo '～ 歡迎光臨檔案下載中心 ～

';
9 echo '贊助廠商廣告

';
10 echo '付費檔案專區，請先點廣告再下載！

';
11 echo '免費檔案專區，請自由參觀！';
12 ?>
13 </body>
14 </html>
```

# 檔案2 - adver.php

```
1 <?php
2 setCookie('check', 1);
3 ?>
4 <html>
5 <head>
6 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
7 <title>cookie 應用 範例</title>
8 </head>
9 <body>
10 <?php
11 echo '~ 廣告頁 ~

';
12 echo '謝謝你的觀看，這頁是廣告頁

';
13 echo '回首頁下載檔案！';
14 ?>
15 </body>
16 </html>
```

～廣告頁～

謝謝你的觀看，這頁是廣告頁

[回首頁下載檔案！](#)

# 檔案3 - download.php

```
1 <html>
2 <head>
3 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4 <title>cookie 應用 範例</title>
5 </head>
6 <body>
7 <?php
8 if($_COOKIE['check'] == 1)
9 {
10 echo '這是需要先點選廣告才看的到的！
';
11 }
12 else
13 {
14 echo '少年ㄟ～天下沒有白吃的午餐
';
15 echo '先瀏覽過「廣告」之後再過來下載啦！
';
16 echo '回首頁';
17 }
18 ?>
19 </body>
20 </html>
```

這是需要先點選廣告才看的到的！

少年ㄟ～天下沒有白吃的午餐  
先瀏覽過「廣告」之後再過來下載啦！  
[回首頁](#)

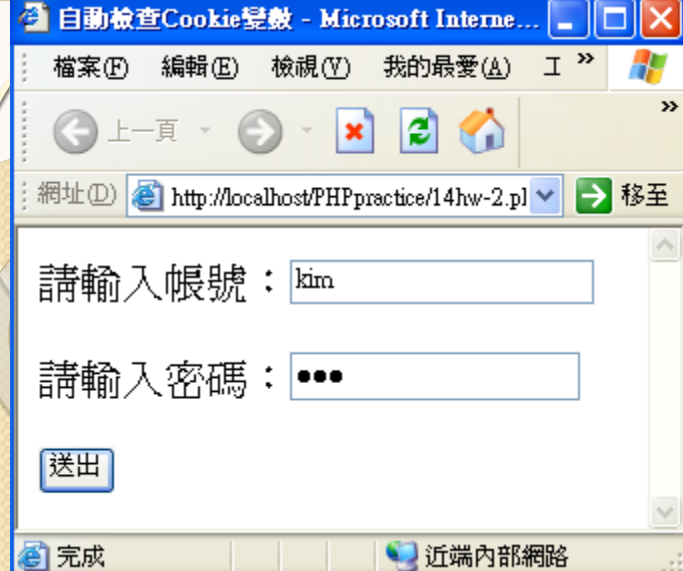
# 檔案4 - free.php

```
1 <html>
2 <head>
3 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4 <title>cookie 應用 範例</title>
5 </head>
6 <body>
7
8 這是誰都看的到的內容！
9
10 </body>
11 </html>
```

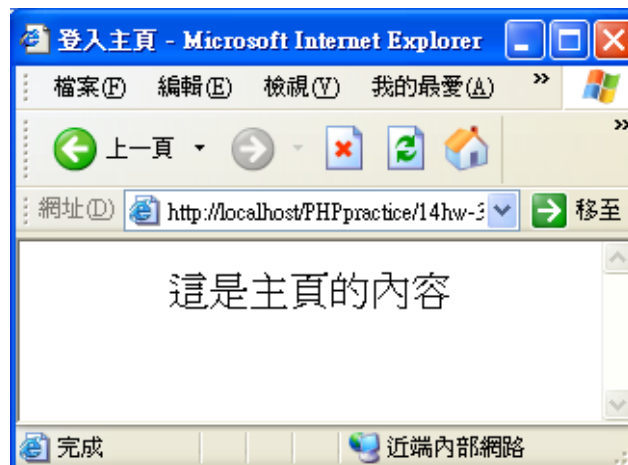
這是誰都看的到的內容！

## 課堂練習 (5-3)

- 請設計一個能將使用者帳號密碼存在 Cookie 中的網頁：
  - 表單頁：input.html
  - 登入頁：login.php
    - 登入成功則將使用者帳號密碼存在 cookie 中
    - 讓 cookie 存活 40 秒
    - 關掉瀏覽器後直接再打開 login.php 則仍然算登入成功
    - 40 秒過後則顯示登入已失效



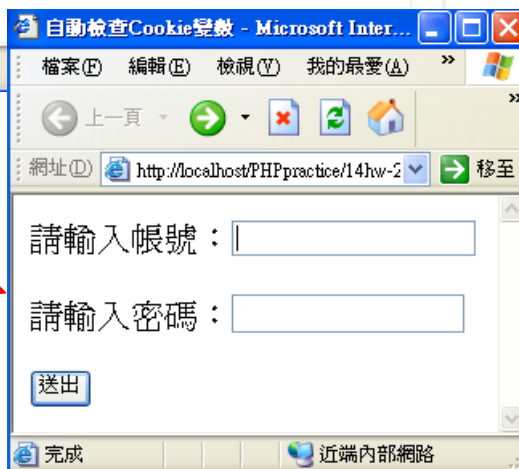
第一次



關閉網頁，40秒  
內重新開啟  
pt53-login.php

第二次以後

關閉網頁超過40  
秒重新開啟  
pt53-login.php





# PHP的檔案處理

- 為何要檔案處理？
  - 在沒有資料庫的情況下，或是不需要資料庫的情況下，檔案就是最好的資料儲存媒介之一
    - 比方說，設計一張表單，讓訪客輸入基本資料，此時，可利用函數，將所有資料儲存在檔案裡，系統管理者在遠端就可以直接將使用者的資料取出，匯集成會員資料庫
  - PHP對於檔案處理的能力非常強大，相較其他程式語言而言是非常適合做檔案處理的

# 常用的檔案相關函數

- `fopen`: 開啟檔案的函數
- `fseek`: 移動檔案讀寫指標的函數
- `fwrite`: 將資料寫入檔案的函數
- `fread`: 將資料從檔案讀出的函數
- `fclose`: 關閉檔案的函數
  
- `file_get_contents`: 直接讀取檔案
- `file_put_contents`: 直接寫入檔案

# 開啟檔案的函數

- 函數：**fopen**(\$file, Open mode)
- 回傳值：  
檔案的handle值，資料型態為整數
- 舉例：  
\$file = "sample/saved\_file";  
\$fileopen = fopen(\$file, "w+"); 其中，\$fileopen為檔案的handle值，  
\$file為檔案的完整路徑名稱，"w+"表  
開啟一個可讀可寫的檔案

# 檔案開啟模式

| 開啟模式 | 說明                                           |
|------|----------------------------------------------|
| a    | 以附加檔案方式開啟，屬性為「寫入」，且將檔案指標移至檔案最末，若檔案不存在則建立檔案   |
| a+   | 以附加檔案方式開啟，屬性為「可讀可寫」，且將檔案指標移至檔案最末，若檔案不存在則建立檔案 |
| r    | 「唯讀」模式，檔案必須存在                                |
| r+   | 「可讀可寫」，檔案必須存在                                |
| w    | 「唯寫」模式，先將檔案清除後重新寫入，若檔案不存在則建立檔案               |
| w+   | 「可讀可寫」，將檔案清除後重新寫入，若檔案不存在則建立檔案                |

# 移動檔案讀寫指標的函數

- 函數：**fseek**(\$fileopen,\$byte\_num)  
\$fileopen:欲處理檔案的handle值  
\$byte\_num:從頭起算的位元數
- 回傳值：  
回傳的資料型態為整數；  
成功回傳0；  
否則，回傳-1；  
此外，若碰到檔尾(EOF)時，不會當作失敗
- 舉例：  
\$file="sample/saved\_file";  
\$fileopen=fopen(\$file,"w+");  
fseek(\$fileopen, 0);

# 將資料寫入檔案的函數

- 函數：**fwrite**(\$fileopen,\$data[\$len])  
\$fileopen:欲處理檔案的handle值  
\$data:欲寫入檔案中的資料  
\$len：欲寫入的字元個數，此參數可省略
- 回傳值：  
回傳的資料型態為整數；  
成功回傳寫入的字元個數；  
失敗則回傳FALSE
- 舉例：  
\$data="大家好！";  
fwrite(\$fileopen, \$data);

# 將資料從檔案讀出的函數

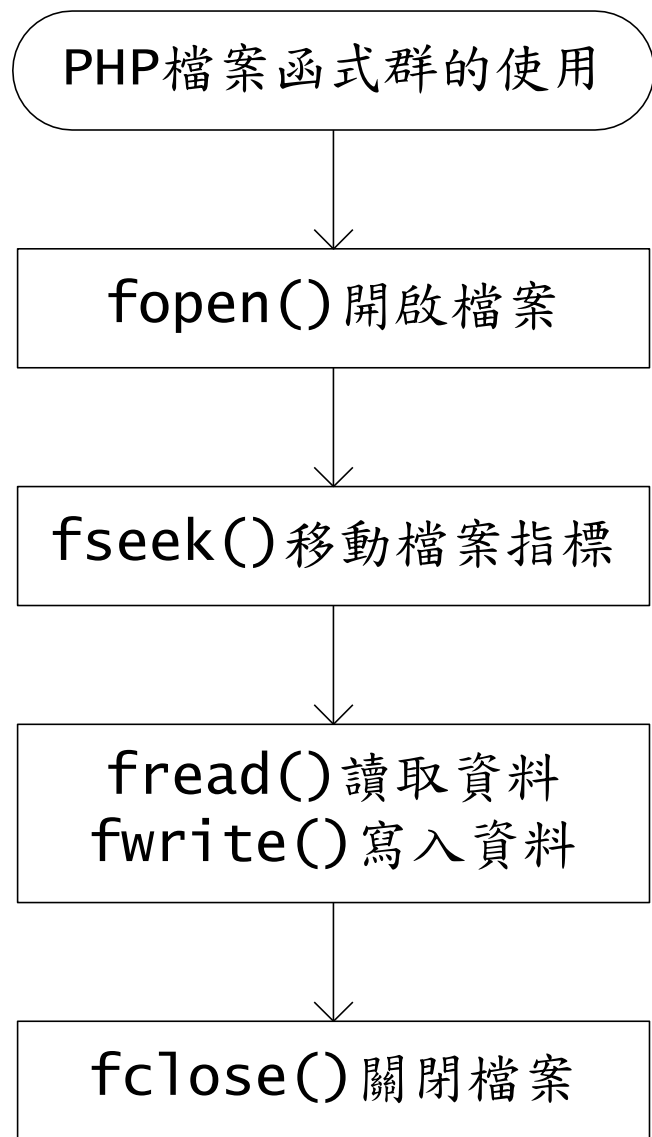
- 函數：**fread**(\$fileopen,\$length)  
    \$fileopen:欲處理檔案的handle值  
    \$length:欲從檔案中讀出的字元個數
- 回傳值：  
    回傳的資料型態為字串；  
    回傳值就是讀出的資料
- 舉例：  
    \$data=fread(\$fileopen, 10);

# 關閉檔案的函數

- 函數：**fclose**(\$fileopen)  
\$fileopen:欲關閉檔案的handle值
- 回傳值：  
回傳的資料型態為布林值；  
成功回傳TRUE；  
失敗回傳FALSE
- 舉例：  
`fclose($fileopen);`



# PHP檔案函數群的使用順序



# 確定是否有開檔成功

- 「`$fileopen = fopen($file, "w+");`」本身有回傳值，可以讓程式設計師作為除錯之用，如果成功開啟檔案則回傳`true`，反之則回傳`false`

```
if ($fileopen = fopen($file,"w+"))
{
 echo "成功開啟檔案";
}
else
{
 echo "開啟檔案失敗";
}
```

# 練習一下

```
1 <html>
2 <head>
3 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4 <title>寫入檔案 範例</title>
5 </head>
6 <body>
7
8 <?php
9 $filename = 'test.txt';
10 $somecontent = '要寫入的內容';
11
12 // 確認檔案是否可被寫入 (檔案本身非唯讀)
13 if (is_writable($filename))
14 {
15 // 開始要被寫入的檔案，使用附加模式
16 $fopen = fopen($filename, 'a');
17 if ($fopen == false)
18 {
19 echo '無法開啟檔案: ' . $filename;
20 exit;
21 }
22
23 // 如果寫入不成功
24 $fwrite = fwrite($fopen, $somecontent);
25 if ($fwrite === false)
26 {
27 echo '檔案寫入失敗: ' . $filename;
28 exit;
29 }
30
31 echo '寫入檔案成功！開啟瞧瞧吧: ' . $filename . '';
32
33 fclose($fopen);
34
35 }
36 else
37 {
38 echo '本檔案非可寫入的狀態，請確認檔案是否存在，或是否唯讀？';
39 }
40 ?>
41
42 </body>
43 </html>
```

寫入檔案成功！開啟瞧瞧吧: [test.txt](#)

要寫入的內容

# 從檔案中取出資料

```
1 <html>
2 <head>
3 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4 <title>讀取檔案 範例</title>
5 </head>
6 <body>
7
8 <?php
9 // 檔案位置
10 $filename = 'test.txt';
11
12 // 開啟檔案
13 $fopen = fopen($filename, 'r');
14
15 if ($fopen == false)
16 {
17 echo '檔案開啟失敗!';
18 exit;
19 }
20 else
21 {
22 // 將檔案指標移至最前方
23 fseek ($fopen, 0);
24
25 // 取得檔案內容
26 $getData = fread($fopen, filesize($filename));
27
28 echo '檔案的內容為 ' . $getData . '';
29
30 // 使用完則關閉連結
31 fclose($fopen);
32 }
33 ?>
34
35 </body>
36 </html>
```

檔案的內容為 要寫入的內容

## 課堂練習 (5-4)

- 將一個學生的成績寫入檔案：  
grade.txt，分別寫入學生的姓名、國文成績、英文成績及數學成績，每個欄位以逗號「,」分隔

# 更好用的檔案處理函式 - file

- 函數：**file**(\$filepath)
  - \$filepath: 只要檔案路徑，不需先開檔案
- 回傳值：
  - 回傳一個陣列，每個元素為一行資料(用 \n 做分割)
- 舉例：
  - \$array = **file**('test.txt');

# file 函數的範例

```
1 <html>
2 <head>
3 <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4 <title>file 範例</title>
5 </head>
6 <body>
7
8 <?php
9 // 檔案位置
10 $filename = 'test2.txt';
11
12 // 用 file 函數讀檔案內容
13 $content = file($filename);
14
15 // 開啟檔案
16 for ($i = 0; $i < sizeof($content); $i++)
17 {
18 echo '第'. ($i + 1) .'行內容: ';
19 echo $content[$i];
20 echo '
';
21 }
22 ?>
23
24 </body>
25 </html>
```

test2.txt - 記事本

檔案(F) 編輯(E) 格式(O) 檢視(V) 說明(H)

Allan  
Bill  
Cindy  
David  
Evelyn

第1行內容: Allan  
第2行內容: Bill  
第3行內容: Cindy  
第4行內容: David  
第5行內容: Evelyn

# 正規表示法 (1)

- 正規表示法對於字串處理非常重要
- 用途：
  - 防呆處理：電話輸入成文字、血型輸入成數字
  - 格式檢查：如身份證字號、Email格式、網址格式
  - 更改日期格式：如12/06/2006 → 2006-12-06
  - 挑出指定字串，或挑出指定字串的前後字串
  - ...
- 範例：`/[A-Z]+(abc|xyz)*/`



# 正規表示法 (2)

## 範例1：比對指定的字串

**/car/**：比對car

Carl spilt his carton of orange juice on the carpet of his new car.

If he had taken more care when opening the carton he wouldn't have had this annoying and disappointing accident.

Some car shampoo would, Carl hoped, make the carpet look as good as new.

# 正規表示法 (2)

## 範例1：比對指定的字串

**/car/**：比對car

Carl spilt his carton of orange juice on the carpet of his new car.

If he had taken more care when opening the carton he wouldn't have had this annoying and disappointing accident.

Some car shampoo would, Carl hoped, make the carpet look as good as new.

# 正規表示法 (3)

## 練習1

| /y/                                                                                                                                                         | /yes/                                                                                                                                                                                  |
|-------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <ol style="list-style-type: none"><li>1. y</li><li>2. Y</li><li>3. yes</li><li>4. yy</li><li>5. hey</li><li>6. heyes</li><li>7. x</li><li>8. good</li></ol> | <ol style="list-style-type: none"><li>1. yes</li><li>2. Yes</li><li>3. YES</li><li>4. yes!</li><li>5. yeS</li><li>6. yesterday</li><li>7. y e s</li><li>8. yes? no! yes ....</li></ol> |

## 正規表示法 (4)

範例2：比對指定的字串，且完全符合

`/bcar\b/`：比對car

Carl spilt his carton of orange juice on the carpet of his new car.

If he had taken more care when opening the carton he wouldn't have had this annoying and disappointing accident.

Some car shampoo would, Carl hoped, make the carpet look as good as new.

# 正規表示法 (5)

## 範例3：比對指定開頭的字串

**`/^If/`**：比對以if開頭的字串

Carl spilt his carton of orange juice on the carpet of his new car.

**If** he had taken more care when opening the carton he wouldn't have had this annoying and disappointing accident.

Some car shampoo would, Carl hoped, make the carpet look as good as new.

# 正規表示法 (6)

## 範例4：比對指定結束的字串

`/new$/`：比對以new結尾的字串

Carl spilt his carton of orange juice on the carpet of his new car.

If he had taken more care when opening the carton he wouldn't have had this annoying and disappointing accident.

Some car shampoo would, Carl hoped, make the carpet look as good as new.

# 正規表示法 (7)

## 練習2

| <code>/yes\$/</code> | <code>/^yes/</code> |
|----------------------|---------------------|
| 1. yes               | 1. yes              |
| 2. yesyes            | 2. Yes              |
| 3. yes no            | 3. yes no           |
| 4. no yes            | 4. no yes           |
| 5. yesterday         | 5. yey yes          |
| 6. yes, it is.       | 6. myes             |
| 7. my eyes           | 7. eyes             |

# 正規表示法 (8)

- 正規表示法 的表示字串的內容主要可以分成兩大類：
  - **Characters**：一般字元(文數字元)，所代表的意義與原字面的意義相同。
  - **Operators**：特殊字元(非文數字字元)，代表某種特殊規則的意義。若要取消其代表意義，必須使用「\」跳脫字元。



# 正規表示法 (9)

- "foo" 的 RE 表示字串 "foo" 的意思
- "[Ff]oo" 的 RE 表示字串 "Foo" 或字串 "foo" 的意思
- "F\*" 的 RE 表示F這個字元會出現0次或多次
- "F\\*" 的 RE 表示字串 "F\*" 的意思

# 正規表示法 (10)

- 「**.**」句號(period) 代表任意一個字元，如 **/.at/** 可符合 bat, cat, rat等任何字開頭，結尾是at的單字。
- 「**[ ]**」中括號(brackets)代表集合中的任 一字元，如 **/[01256]/** 代表0,1,2,5,6這個 集合中的任何一個字元。
- 「**-**」連字號(dash)在中括號內表示「範圍」，如 **/[0-9]/**代表0到9的任一個單一的 數字。
- 「**^**」(carat)在中括號內表示「否定」，**/[^aeiou]/** 代表除了a,e,i,o,u這幾個母音 之外的字元。

# 正規表示法 (11)

- 「|」(pipe)表示可選擇的，如/**cat|dog|bird**/代表cat, dog, bird 其中之一皆可(or的概念)。
- 「?」問號(question mark) 表示前面的字元或集合出現0次或1次，如/**colou?r** / 代表u這個字可出現也可不出現。
- 「+」加號(plus) 表示前面的字元或集合出現1次或多次，如/**ap+le** / 代表p這個字至少出現1次。
- 「\*」星號(asterisk)表示前面的字元或集合出現0次或多次，如/**section [0-9]\***/ 代表數字可出現也可不出現。
- 「{ }」大括號(curly-braces)表示前面的字元或集合出現的次數，如/**c{5,8}** / 代表c這個字重複出現5到8次。

# 正規表示法 (12)

## 練習3

$/[15]0^*/$

1. 1
2. 6
3. 16
4. 50
5. 600
6. 650
7. 1000
8. 5000

$/10\{2,4\}1/$

1. 101
2. 1001
3. 10001
4. 100001
5. 1000001

# 正規表示法 (13)

## 練習4

`/[k-q].[^4-9]/`

1. 5am6a4
2. p43
3. amma
4. aqk4
5. bom1

`/[^a-k]+[0-9]?./`

1. 4
2. 54
3. m0
4. s4a
5. pmq1

# PHP中的正規表示法函數

- `ereg / eregi` : PHP原創的正規表示函數
  - 在 PHP5.3.0 之後已經不再支援...
- `preg_match` : Perl based 的正規表示函數
- `preg_match_all`

# preg\_match

- 函數：`preg_match($pattern, $subject [, $matches])`
- 參數說明：
  - `$pattern`: 正規表示法的模型
  - `$subject`: 要比對的內容
  - `$matches`: 比對到的存放陣列
- 回傳值：有找到回傳 **1**，否則**0**
- 舉例：

```
preg_match('/^def/', 'abcdef', $matches);
print_r($matches);
```

# preg\_match\_all

- 函數：`preg_match_all($pattern, $subject [, $matches])`
- 用法和 `preg_match` 一樣，唯一差別在於：
  - `preg_match` 找到第一個符合模型的比對項目時就會結束比對
  - `preg_match_all` 則會比對完所有的內容



# 正規表示法範例 (1)

- 基本字串搜尋：

```
<?php
$subject = "學 PHP 好好玩！";
if (preg_match("/php/", $subject))
{
 echo "找到了！";
}
else
{
 echo "沒找到！";
}
?>
```

# 正規表示法範例 (1)

- 基本字串搜尋：

```
<?php
$subject = "學 PHP 好好玩！";
if (preg_match("/php/i", $subject))
{
 echo "找到了！";
}
else
{
 echo "沒找到！";
}
?>
```

# 正規表示法範例 (2)

- 完全字串搜尋：

```
<?php
$subject = "PHP 可以拿來寫 web 應用程式";
if (preg_match("/\bweb\b/i", $subject))
{
 echo "找到了！";
}
else
{
 echo "沒找到！";
}
?>
```

# 正規表示法範例 (2)

- 完全字串搜尋：

```
<?php
$subject = "PHP 可以拿來寫 website 應用程式";
if (preg_match("/\bweb\b/i", $subject))
{
 echo "找到了！";
}
else
{
 echo "沒找到！";
}
?>
```

# 正規表示法範例 (3)

- 取得網域名稱

```
<?php
```

```
$pattern = '/^http\:\/\/ (.+)\.*/i';
```

```
$subject = "http://www.php.net/index.html";
```

```
preg_match($pattern, $subject, $matches);
```

```
$host = $matches[1];
```

```
preg_match('/[^\.]+\.[^\.]+$/', $host, $matches);
```

```
echo "domain name is: {$matches[0]}\n";
```

```
?>
```

# 正規表示法範例 (3)

- 取得網域名稱

```
<?php
```

```
$pattern = '/^http\:\/\/(.+)\:\/\/(.*)/i';
```

```
$subject = "http://www.php.net/index.html";
```

```
preg_match($pattern, $subject, $matches);
```

```
$host = $matches[1];
```

```
preg_match('/^[^.]+\.[^.]+' . $, $host, $matches);
```

```
echo "domain name is: " . $matches[0] . "\n";
```

```
?>
```

```
domain name is: php.net
```

# 正規表示法的應用 (1)

- Email Address 的驗證：

```
<?php
$email = 'allan@csie.org';
$pattern = '/^[^@]+@[a-zA-Z0-9\._-]+\.[a-zA-Z]+$/';
if (preg_match($pattern, $email))
{
 echo 'Email 格式正確！';
}
else
{
 echo 'Email 格式錯誤！';
}
?>
```

# 正規表示法的應用 (2)

- 電話號碼格式驗證：
  - 手機號碼：0912345678 / 0912-345-678
  - 一般號碼：(02)23939889 / 02-2393-9889

```
<?php
$phone = '0912-345-678';
$pattern = '/^([\(\)\0-9\-]){6,}$/';
if (preg_match($pattern, $phone))
{
 echo 'OK';
}
else
{
 echo 'error';
}
?>
```



# 驗證身份證字號 (3-1)

- 中華民國身分證字號驗證

- 先講講規則：A123456789

- (1)第1個字元代表地區，轉換方式為：A轉換成1,0兩個字元，B轉換成1,1.....

| A  | B  | C  | D  | E  | F  | G  | H  | I  | J  | K  | L  | M  | N  | O  | P  | Q  | R  | S  | T  | U  | V  | W  | X  | Y  | Z  |
|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 34 | 18 | 19 | 20 | 21 | 22 | 35 | 23 | 24 | 25 | 26 | 27 | 28 | 29 | 32 | 30 | 31 | 33 |

- (2)第2個字元代表性別，1代表男性，2代表女性
    - (3)第3個字元到第九個字元為流水號碼
    - (4)第10個字元為檢查號碼

# 驗證身份證字號(3-2)

1

| 每個相對應數字乘上權數 |    |    |    |    |    |    |    |    |    | 檢查號碼 |
|-------------|----|----|----|----|----|----|----|----|----|------|
| A           |    | 1  | 2  | 3  | 4  | 5  | 6  | 7  | 8  | 9    |
| 1           | 0  | x8 | x7 | x6 | x5 | x4 | x3 | x2 | x1 |      |
| x1          | X9 |    |    |    |    |    |    |    |    |      |

2.將乘上權數後之積相加

$$\begin{array}{rcl}
 1 & \times & 1 = 1 \\
 0 & \times & 9 = 0 \\
 1 & \times & 8 = 8 \\
 2 & \times & 7 = 14 \\
 3 & \times & 6 = 18 \\
 4 & \times & 5 = 20 \\
 5 & \times & 4 = 20 \\
 6 & \times & 3 = 18 \\
 7 & \times & 2 = 14 \\
 +) & 8 & \times 1 = 8
 \end{array}$$

---

121

3. 相加後之值除以模數10取其餘數

$$121 / 10 = 12 \dots \text{餘數 } 1$$

4. 由模數減去餘數得檢查號碼，若餘數為0時，則設定其檢查碼為0

$$10 - 1 = 9$$

# 寫成程式看看

```
<?php
function check_roc_id($id)
{
 $first_char = array(
 'A' => 10, 'B' => 11, 'C' => 12, 'D' => 13, 'E' => 14,
 'F' => 15, 'G' => 16, 'H' => 17, 'J' => 18, 'K' => 19,
 'L' => 20, 'M' => 21, 'N' => 22, 'P' => 23, 'Q' => 24,
 'R' => 25, 'S' => 26, 'T' => 27, 'U' => 28, 'V' => 29,
 'W' => 30, 'X' => 31, 'Y' => 32, 'Z' => 33, 'I' => 34,
 'O' => 35
);

 $n0 = $first_char[strtoupper(substr($id, 0, 1))];
 $n[] = substr($n0, 0, 1);
 $n[] = substr($n0, 1, 1);

 for ($i = 1; $i < strlen($id); $i++)
 {
 $n[] = substr($id, $i, 1);
 }

 $mod = ($n[0] + ($n[1]*9) + ($n[2]*8) + ($n[3]*7) + ($n[4]*6) +
 ($n[5]*5) + ($n[6]*4) + ($n[7]*3) + ($n[8]*2) + $n[9]) % 10;

 if ($mod == 0 && $n[10] == 0)
 {
 $check = TRUE;
 }
 else
 {
 $check = (10 - $mod == $n[10]) ? TRUE : FALSE;
 }

 return($check);
}

$id = 'A123456789';
// $id = 'O223344555';
echo '身分證字號: ' . $id . '
';

if (check_roc_id($id) == FALSE)
{
 echo '你的身分證字號錯誤!';
}
else
{
 echo '正確';
}
?>
```

# 回家作業 – 成績統計程式 (1)

- 設計一個 HTML 表單，有下列元件：
  - 姓名 / 學號 / Email
  - 國文 / 英文 / 數學 / 物理 / 化學 成績
- 送出後程式可以檢查
  - 姓名不能有數字
  - 學號 (假設為 8 位數字)
  - Email 是否正確
  - 五個成績欄只接受 1~3 位數的數字

# 回家作業 – 成績統計程式 (2)

- 格式都確定無誤後將資料寫入 grade.txt 檔案中
- 檔案格式：
  - 姓名,學號,email,國文,英文,數學,物理,化學
- 自行輸入10筆可以被寫入的正確資料
- 另外寫一支程式，可以統計和分析：
  - 每個人的總分和平均分數
  - 班上排名
  - 班上各科平均
  - 不及格分數用紅色字標示出來 (<60)

# 範例畫面

grade.html

學生姓名：

學生學號：

email：

國文成績：

英文成績：

數學成績：

物理成績：

化學成績：

grade.php

你的英文成績不是數字！

寫入完成！[回上一頁繼續輸入成績](#)

grade.txt

```
1 王小明,11111111,email@mail.com,60,80,90,85,76
2 陳大福,22222222,email@mail.com,11,60,80,90,85,76
3 林阿偉,33333333,email@mail.com,11,60,80,90,85,76
4 張大頭,44444444,email@mail.com,60,80,90,85,76
5 劉小文,55555555,email@mail.com,60,80,90,85,76
```

analysis.php

| 名次     | 姓名  | 學號       | email              | 國文 | 英文 | 數學 | 物理 | 化學  | 總分  | 平均分數 |
|--------|-----|----------|--------------------|----|----|----|----|-----|-----|------|
| 1      | 劉小文 | 11111111 | email@yahoo.com.tw | 90 | 95 | 86 | 98 | 100 | 469 | 93.8 |
| 2      | 陳大福 | 22222222 | email@gmail.com    | 92 | 85 | 82 | 99 | 90  | 448 | 89.6 |
| ...    |     |          |                    |    |    |    |    |     |     |      |
| 10     | 張大頭 | 99999999 | email@ntu.edu.tw   | 75 | 55 | 68 | 59 | 60  | 317 | 63.4 |
| 全班總平均： |     |          |                    |    |    |    |    |     |     | 78.6 |

# 繳交回家作業

- 把「成績統計程式」完成後把相關檔案壓縮成一個 姓名.zip 或 姓名.rar 檔
- 把該檔案附件到 email 中寄到：
  - 主旨：「[PHP263][hw][5]姓名」
  - 收件人：[m10112101@mail.ntust.edu.tw](mailto:m10112101@mail.ntust.edu.tw)
- Deadline: 下次上課前

# 繳交課堂作業

- 把課堂練習(5-1)~課堂練習(5-n) 分別儲存：
  - pt5-1 ~ pt5-n 的資料夾中，資料夾裡存放所有你該次練習的所有檔案 (如果有圖片、html 或 php)都存在裡面
  - 把這 n 個資料夾都壓縮成 姓名.zip 或 姓名.rar 檔，把該檔案附件到 email 中寄到：
    - 主旨：「[PHP263][pt][5]姓名」
    - 收件人：[m10112101@mail.ntust.edu.tw](mailto:m10112101@mail.ntust.edu.tw)
    - Deadline:下次上課前