

# PHP+MySQL

## 互動式網頁程式設計班

---

[第四課] PHP進階語法：二維陣列、函數  
及檔案上傳

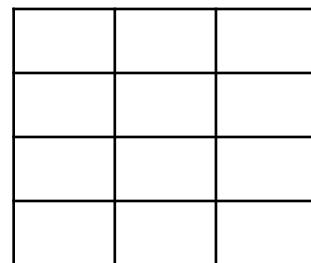
國立台灣大學 資訊工程學系暨研究所  
資訊系統訓練班  
講師：鄭明學

# 二維陣列

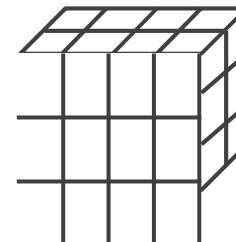
- 陣列是具有維度的
  - 維度 (dimension) 即是時常聽到的 2D 、 3D 的 D
  - 1D 為直線 (x 軸)



- 2D 為平面 (x, y 軸)



- 3D 為立體 (x, y, z 軸)



# 陣列與資料

- 一維陣列通常儲存相同類型的資料
  - 如全班同學的數學成績

1號	2號	3號	4號	5號	6號	7號	8號	...
85	95	57	83	64	92	50	75	...

- 二維陣列則適合儲存表格資料
  - 如全班同學的各科成績

	1號	2號	3號	4號	5號	6號	7號	...
數學	85	95	57	83	64	92	50	...
國文	80	92	63	87	72	90	68	...
英文	90	82	73	77	85	91	84	...

# 初始化二維陣列 (1)

- 二維陣列的初始化如同一維陣列一樣，可以分為兩種，以下列例子來說：

	1號	2號	3號	4號	5號	6號	7號	...
數學	85	95	57	83	64	92	50	...
國文	80	92	63	87	72	90	68	...
英文	90	82	73	77	85	91	84	...

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>二維陣列 範例</title>
5 </head>
6 <body>
7   <?php
8   $grade = array(
9     'math' => array(85, 95, 57, 83, 64, 92, 50),
10    'chn' => array(80, 92, 63, 87, 72, 90, 68),
11    'eng' => array(90, 82, 73, 77, 85, 91, 84)
12  );
13  echo '座號 3 號的同學國文成績為：' . $grade['chn'][2];
14 ?
15 </body>
16 </html>
```

# 初始化二維陣列 (2)

- 直接寫入法：

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>二維陣列 範例2</title>
5 </head>
6 <body>
7   <?php
8   $grade['math'][0] = 85;
9   $grade['math'][1] = 95;
10  $grade['math'][2] = 57;
11  $grade['math'][3] = 83;
12  $grade['math'][4] = 64;
13  $grade['math'][5] = 92;
14  $grade['math'][6] = 50;
15
16  $grade['chn'][0] = 80;
17  $grade['chn'][1] = 92;
18  $grade['chn'][2] = 63;
19  $grade['chn'][3] = 87;
20  $grade['chn'][4] = 72;
21  $grade['chn'][5] = 90;
22  $grade['chn'][6] = 68;
23
24  $grade['eng'][0] = 90;
25  $grade['eng'][1] = 82;
26  $grade['eng'][2] = 73;
27  $grade['eng'][3] = 77;
28  $grade['eng'][4] = 85;
29  $grade['eng'][5] = 91;
30  $grade['eng'][6] = 84;
31
32 echo '座號 3 號的同學國文成績為：' . $grade['chn'][2];
33 ?>
34 </body>
35 </html>
```

# 初始化二維陣列 (3)

- 由於二維陣列中第一維的內容值也是陣列，因此也可以像這樣使用變化型：

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>二維陣列 範例3</title>
5 </head>
6 <body>
7   <?php
8     $grade['math']= array(85, 95, 57, 83, 64, 92, 50);
9     $grade['chn']= array(80, 92, 63, 87, 72, 90, 68);
10    $grade['eng'] = array(90, 82, 73, 77, 85, 91, 84);
11
12  echo '座號 3 號的同學國文成績為：' . $grade['chn'][2];
13 ?>
14 </body>
15 </html>
```

- 三種寫法結果都相同：

座號 3 號的同學國文成績為：63

# 初始化二維陣列 (4)

- 當然，也可以將第二維的索引改為字串：

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>二維陣列 範例4</title>
5 </head>
6 <body>
7   <?php
8     $grade = array(
9       'math' => array('一號' => 85, '二號' => 95, '三號' => 57, '四號' => 83, '五號' => 64, '六號' => 92, '七號' => 50),
10      'chn' => array('一號' => 80, '二號' => 92, '三號' => 63, '四號' => 87, '五號' => 72, '六號' => 90, '七號' => 68),
11      'eng' => array('一號' => 90, '二號' => 82, '三號' => 73, '四號' => 77, '五號' => 85, '六號' => 91, '七號' => 84)
12    );
13    echo '座號 3 號的同學國文成績為：' . $grade['chn']['三號'];
14  ?>
15 </body>
16 </html>
```

雖然PHP是支援中文字  
當索引的，不過還是盡  
量用英文做索引比較好

# 初始化二維陣列 (5)

- 或是第一維與第二維都使用預設的索引：

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>二維陣列 範例5</title>
5 </head>
6 <body>
7   <?php
8     $grade = array(
9       array(85, 95, 57, 83, 64, 92, 50),
10      array(80, 92, 63, 87, 72, 90, 68),
11      array(90, 82, 73, 77, 85, 91, 84)
12    );
13   echo '座號 3 號的同學國文成績為：' . $grade[1][2];
14 ?>
15 </body>
16 </html>
```

# 計算二維陣列的元素個數

- 二維陣列時，`sizeof()`會計算下一個維度的個數
  - 假設陣列的內容如下：
    - `$array[0][0]=1;`
    - `$array[1][0]=1; $array[1][1]=2;`
    - `$array[2][0]=1; $array[2][1]=2;`
  - 則
    - `sizeof($array)=3;`
    - `sizeof($array[0])=1;`
    - `sizeof($array[1])=2;`
    - `sizeof($array[2])=2;`

# 課堂練習 (4-1)

- 利用二維陣列的概念，設計一個「業績統計表」網頁
  - input.html: 利用表單的方式，設計出可輸入二個營業員個別三個月的業績
  - output.php: 統計出每個營業員的總營業額，並印出何者最高

# 課堂練習 (4-1) 範例畫面

季業績輸入表 - Microsoft Internet Explorer

檔案(F) 編輯(E) 檢視(V) 我的最愛(A) 工具(I) 說明(H)

上一頁 → 前一頁 后一頁 下一頁 搜尋 我的最愛 媒體

網址(①) http://localhost/PHPpractice/ch8hw-1.htm 移至

請輸入營業員「阿寶」的三個月業績(單位：千元)：

第一個月：

第二個月：

第三個月：

請輸入營業員「阿花」的三個月業績(單位：千元)：

第一個月：

第二個月：

第三個月：

季統計輸出表 - Microsoft Internet Explorer

檔案(F) 編輯(E) 檢視(V) 我的最愛(A) 工具(I) 說明(H)

上一頁 → 前一頁 后一頁 下一頁 搜尋 我的最愛 媒體

網址(①) http://localhost/PHPpractice/ch8hw-2.php 移至

這季阿寶的總業績是：180千元  
這季阿花的總業績是：210千元

這季阿花比較棒哦！阿寶請加油～

近端內部網路

# 函數的種類

- PHP的內建函數：
  - PHP在安裝好的時候就已經內建在裡面的函數，是編譯在 PHP 核心內的，我們可以直接拿來使用，如前面章節曾介紹過的：
    - echo: 輸出到畫面
    - sizeof / count: 計算陣列元件數目
    - array: 建立陣列
- 程式設計者的自訂函數：
  - 我們也可以自己設計屬於自己的函數，來減少許多重覆要使用到的功能

# 函數 – 自訂函數

- 語法：

```
function 函數名稱(參數)
```

```
{
```

```
    // 函數要做些什麼事
```

```
    return 回傳值;
```

```
}
```

- 呼叫： 在程式中直接呼叫函數名稱即可，但函數跟 呼叫它的主程式需要寫在同一支檔案裡 (後面章節會再介紹如何把程式切開再引入執行)

# 函數的範例

- 範例 (以剛剛的打八折後售價為例) :

```
function discount($price)
```

```
{
```

```
    $result = $price * 0.8;  
    return $result;
```

```
}
```

```
$price = 1000;
```

```
$discount_price = discount($price);
```

```
echo $discount_price; //會輸出 800
```

# 更聰明的寫法

- 範例 (以剛剛的打八折後售價為例) :

```
function discount($price)
```

```
{
```

```
    return $price * 0.8;
```

```
}
```

```
echo discount(1000); //會輸出 800
```

# 參數也可以是陣列型式

- 範例 (以剛剛的打八折後售價為例) :

```
function discount($price)
{
    $return_price = array();
    for ($i = 0; $i < sizeof($price); $i++)
    {
        $return_price[$i] = $price[$i] * 0.8;
    }
    return $return_price;
}
```

主程式 :

```
$price = array(1000, 800, 2000, 1500);
$discount_price = discount($price);
for ($i = 0; $i < sizeof($discount_price); $i++)
{
    echo $discount_price[$i];
}
// 依序輸出 : 800, 640, 1600, 1200
```

# 多個參數的函數

- 函數可以接受一個以上的參數，如：

```
function discount($price, $discount)
{
    return $price * $discount;
}
```

這樣的程式就可以算出各種折扣後的價錢了  
例如：

```
echo discount(1000, 0.6); // 600
echo discount(800, 0.5) // 400
```

# 沒有參數的函數

- 實際上函數也是可以沒有參數的，例

```
function hello_world()
```

```
{
```

```
    return 'Hello World!';
```

```
}
```

- 呼叫時：

```
echo hello_world();
```

```
// 會直接印出 Hello World!
```

# 沒有回傳值的函數

- 實際上函數也是可以沒有回傳值的，例：

```
function hello_world()  
{  
    echo 'Hello World!';  
}
```

- 呼叫時：

```
hello_world();  
// 不用再寫一次 echo 就會印出 Hello  
World!
```

# 函數中包含了函數

- 前面的例子中，其實 `discount()` 已經被 `echo` 函數所包含了，可以看做為：
  - `echo (discount(1000, 0.8));`
  - 簡單的說，先做 $1000 * 0.8$ 得到計算結果 (800) 後，再把 800 傳到 `echo` 函數中，`echo` 函數再把 800 印在畫面中

# 函數中的變數生命週期

- 在PHP的程式設計裡，函數大部分都必須靠變數來做資料的傳遞，但是，大部分函數裡的變數都不是永遠存在的，**函數內的所有變數皆會在此函數內生與滅**
- 函數內的變數，其生命週期在完成任務後即結束
- 若要變數在函數結束後，仍能存在，需要變數宣告在所有函數外面，變成一個全域變數，這樣，即使函數結束後，變數仍然存在，直到整個程式結束，該全域變數才會被釋放掉

# 生命週期的範例

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>自訂函數 範例</title>
5 </head>
6 <body>
7   <?php
8   function discount($price)
9   {
10     $new_price = $price * 0.8;
11     echo '商品目前的價格是：$'. $price . '<br><br>';
12     return $new_price;
13   }
14   echo '商品打八折後的價格是：$'. discount(800). '<br><br>';
15   echo '商品打折前的價格是：$'. $price. '<br><br>';           // 無法顯示
16 ?>
17 </body>
18 </html>
```

商品目前的價格是：\$800

商品打八折後的價格是：\$640

商品打折前的價格是：\$

# 函數 – 內建函數

- PHP與其他程式語言比較起來的強項之一就是極為豐富又便利的內建函數
- 其實一些平時會遇到的問題，PHP的內建函數大都有解決之道，善用PHP的內建函數，可迅速、有效率的解決問題
- 該怎麼知道有哪些內建函數？
  - PHP官方網站有函數列表可查：  
<http://www.php.net/manual/en/funcref.php>
  - 建議可以下載一份 PHP manual 在電腦中，這樣離線狀態下也可以查函數

# 內建函數 – 陣列相關

- array: 建立一個新的陣列
- sort / rsort: 排序陣列的值
- asort / arsort: 排序陣列的值
- ksort / krsort: 排序陣列索引的值
- count / sizeof: 計算變數或陣列中的元素個數
- range: 建立一個整數範圍的陣列
- shuffle: 將陣列的順序隨機排列
- list + each: 一個一個取得陣列的索引和值
- rand(): 亂數隨機取樣
- 還有更多：  
<http://www.php.net/manual/en/book.array.php>

# 陣列排序相關函數

- 基本排序函數：
  - `sort($array)` - 由小到大
  - `rsort($array)` - 由大到小
  - 例：`sort($name)` · `rsort($score);`
- 當陣列是 `$array['key'] = 'value';` 時：
  - 用 `key` 來排序：
    - `ksort($array)` - 依 `key` 值由小到大
    - `krsort($array)` - 依 `key` 值由大到小
  - 用 `value` 來排序：
    - `asort($array)` - 依 `value` 值由小到大
    - `arsort($array)` - 依 `value` 值由大到小

# 陣列排序的範例 – sort/rsort

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>內建函數 - sort 範例</title>
5 </head>
6 <body>
7   <?php
8     $score = array(85, 95, 57, 83, 64, 92, 50, 75);
9
10    echo '排序前全班同學的成績:<br>';
11    for ($i = 0; $i < sizeof($score); $i++)
12    {
13      echo $score[$i] . ' ';
14    }
15    echo '<hr>';
16
17    sort($score);
18    echo '排序後全班同學的成績 (由小到大):<br>';
19    for ($i = 0; $i < sizeof($score); $i++)
20    {
21      echo $score[$i] . ' ';
22    }
23    echo '<hr>';
24
25    rsort($score);
26    echo '排序後全班同學的成績 (由大到小):<br>';
27    for ($i = 0; $i < sizeof($score); $i++)
28    {
29      echo $score[$i] . ' ';
30    }
31    echo '<hr>';
32    ?>
33 </body>
34 </html>
```

排序前全班同學的成績:  
85 95 57 83 64 92 50 75

排序後全班同學的成績 (由小到大):  
50 57 64 75 83 85 92 95

排序後全班同學的成績 (由大到小):  
95 92 85 83 75 64 57 50

# 陣列排序的範例 – ksort/arsort

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>內建函數 - ksort/arsort 範例</title>
5 </head>
6 <body>
7   <?php
8   $i = 0;
9   $score = array(
10     'cindy' => 85, 'evelyn' => 95, 'gary' => 57,
11     'david' => 83, 'bill' => 64, 'harry' => 92,
12     'allan' => 50, 'frank' => 75
13   );
14
15   ksort($score);
16   echo '使用 ksort 排序: <br>';
17   while (list($index, $key) = each($score))
18   {
19     echo $index . ' 考了 ' . $key . ' 分<br>';
20   }
21   echo '<hr>';
22
23   arsort($score);
24   echo '使用 arsort 排序: <br>';
25   foreach ($score as $index => $key)
26   {
27     echo '第' . ++$i . '名是 ' . $index . ' 考了 ' . $key . ' 分<br>';
28   }
29   echo '<hr>';
30
31 ?>
32 </body>
33 </html>
```

使用 ksort 排序:  
allan 考了 50 分  
bill 考了 64 分  
cindy 考了 85 分  
david 考了 83 分  
evelyn 考了 95 分  
frank 考了 75 分  
gary 考了 57 分  
harry 考了 92 分

使用 arsort 排序:  
第1名是 evelyn 考了 95 分  
第2名是 harry 考了 92 分  
第3名是 cindy 考了 85 分  
第4名是 david 考了 83 分  
第5名是 frank 考了 75 分  
第6名是 bill 考了 64 分  
第7名是 gary 考了 57 分  
第8名是 allan 考了 50 分

# 建立一個範圍值的函數

- 函數：**range(start, end)**
- 回傳值：一個範圍的陣列
- 舉例：
  - PHP 官方網站的例子

```
<?php
// array(0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12)
foreach (range(0, 12) as $number) {
    echo $number;
}

// The step parameter was introduced in 5.0.0
// array(0, 10, 20, 30, 40, 50, 60, 70, 80, 90, 100)
foreach (range(0, 100, 10) as $number) {
    echo $number;
}

// Use of character sequences introduced in 4.1.0
// array('a', 'b', 'c', 'd', 'e', 'f', 'g', 'h', 'i')
foreach (range('a', 'i') as $letter) {
    echo $letter;
}
// array('c', 'b', 'a');
foreach (range('c', 'a') as $letter) {
    echo $letter;
}
?>
```

# 把陣列隨機的重新排列 (洗牌)

- 函數：**shuffle**(陣列)
- 回傳值：一個洗牌過後的的陣列
- 舉例：

```
<?php
$array = range(1, 10);

echo '一開始的陣列: <br>';
for ($i = 0; $i < sizeof($array); $i++)
{
    echo $array[$i] . ' ';
}
echo '<hr>';

shuffle($array);
echo 'shuffle 過後的陣列: <br>';
for ($i = 0; $i < sizeof($array); $i++)
{
    echo $array[$i] . ' ';
}
echo '<hr>';

?>
```

一開始的陣列:  
1 2 3 4 5 6 7 8 9 10

shuffle 過後的陣列:  
2 10 8 4 6 5 7 3 9 1

每次重新整理的結果都不一樣！

# 亂數隨機取樣

- 函數：**rand(start,end)**
- 回傳值：取樣的結果(資料型態通常為整數)
- 舉例：
  - \$number = rand(1, 10);
  - 會傳回 1 到 10 中的任何一個數字

# 亂數隨機取樣的範例

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>內建函數 - rand 範例</title>
5 </head>
6 <body>
7   <?php
8   echo '～歡迎使用網路吃角子老虎簡易版～<br><br>';
9
10  $first  = rand(1,7);
11  $second = rand(1,7);
12  $third  = rand(1,7);
13
14  echo '第一個數是：' . $first . '<br><br>';
15  echo '第二個數是：' . $second . '<br><br>';
16  echo '第三個數是：' . $third . '<br><br>';
17
18  if($first == $second && $first == $third)
19  {
20    echo '恭喜你！三個數字皆相同，獨得 100 萬！<br><br>';
21  }
22  else
23  {
24    echo '下一次手氣也許會好些...<br><br>';
25  }
26  ?>
27 </body>
28 </html>
```

～歡迎使用網路吃角子老虎簡易版～

第一個數是：4

第二個數是：4

第三個數是：4

恭喜你！三個數字皆相同，獨得 100 萬！

OR

～歡迎使用網路吃角子老虎簡易版～

第一個數是：7

第二個數是：7

第三個數是：4

下一次手氣也許會好些...

# 課堂練習 (4-2)

- 設計一個程式從rand(1,6)函數取得一萬個1至6的整數，分別代表擲一個骰子一萬次，統計一下，1點至6點出現的次數及出現的機率
  - 機率 = 出現的次數/總次數



# 內建函數 – 字串相關

- `strtoupper()` / `strtolower()`: 轉換英文的大小寫
- `trim()`: 刪除無效空白字元
- `strlen()`: 計算字串裡的字數
- `substr()`: 摷取部分字串
- `explode()`: 字串分割
- `str_replace()`: 字串取代
- PHP對於字串的處理能力非常強，內建的函數庫非常豐富，詳可見：  
<http://www.php.net/manual/en/book.strings.php>

# 轉換英文的大小寫

- 函數：
  - `strtoupper($str)`: 將字串通通變成大寫
  - `strtolower($str)`: 將字串通通變成小寫
- 回傳的值：轉換之後的結果，資料型態為「字串」
- 舉例：
  - `strtoupper($name);`
  - `strtolower($name);`
- 「Mary」與「MARY」這兩個字串在系統中，會被認為是兩個不一樣的字串，在進行系統登入時，會因此而導致登入失敗，密碼也是如此，其他像資料欄位的取得，若與資料表內的資料大小寫不符，也完全派不上用場，此時，便可使用這兩個函數，將大小寫統一

# 轉換英文大小寫的範例

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>內建函數 - strtoupper/strtolower 範例</title>
5 </head>
6 <body>
7   <?php
8   $name = 'Mary';
9   echo '原來的字串：';
10  echo $name . '<br><br>';
11  echo '這是全部大寫的瑪麗：';
12  echo strtoupper($name) . '<br><br>';
13  echo '這是全部小寫的瑪麗：';
14  echo strtolower($name);
15  ?>
16 </body>
17 </html>
```

原來的字串：Mary

這是全部大寫的瑪麗：MARY

這是全部小寫的瑪麗：mary

# 刪除無效的空白字元

- 函數：**trim(\$var)**
- 回傳的值：刪除無效空白字元的結果，資料型態為「字串」
- 舉例：**trim(\$name)**
- 使用此函數，可避免因使用者多打了一些空格，而導致無法辨認的後果

# 多了空白字元造成的問題

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>內建函數 - without trim 範例</title>
5 </head>
6 <body>
7   <?php
8     $loginName = ' Allan'; // 假設為使用者輸入的
9     echo 'Hi: ' . $loginName . '<br><br>';
10    if($loginName == 'Allan')
11    {
12      echo '偉大的站長，歡迎登入！';
13    }
14    else
15    {
16      echo '資料錯誤！';
17    }
18  ?>
19 </body>
20 </html>
```

Hi: Allan

資料錯誤！

# 去除空白字元之後

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>內建函數 - with trim 範例</title>
5 </head>
6 <body>
7   <?php
8   $loginName = ' Allan'; // 假設為使用者輸入的
9   echo 'Hi: ' . $loginName . '<br><br>';
10  if(trim($loginName) == 'Allan')
11  {
12    echo '偉大的站長，歡迎登入！';
13  }
14  else
15  {
16    echo '資料錯誤！';
17  }
18 ?>
19 </body>
20 </html>
```

Hi: Allan

偉大的站長，歡迎登入！

# 字串取代 – str\_replace()

- 函數：`str_replace($from, $to, $src)`
- 用途：從 `$src` 中找出 `$from` 換成 `$to`
- 範例：

```
<?php  
$src = 'an apple a day, keeps the doctor away.';  
$replaced = str_replace('apple', 'orange', $src);  
  
echo $replaced;  
?>
```

an orange a day, keeps the doctor away.

- 另有一個 `str_ireplace()`
  - 差別在取代時忽略 (ignore) 英文的大小寫

# 課堂練習 (4-3)

- 設計一個輸入帳號、密碼的頁面，利用 trim函數、strtolower函數，過濾使用者輸入的帳號部分，以避免使用者不小心打錯大小寫或多打空白鍵而造成無法登入

# 計算字串裡的字數

- 函數：**strlen(\$var)**
- 回傳的值：字串裡包含的字元數，資料型態為「整數」
- 一個英文字佔一個字元，一個中文字佔2個字元

```
$name1='David';
echo strlen($name1); // 輸出5
$name2='David Wang';
echo strlen($name2); // 輸出10
```

# 擷取部分字符串

- 函數：**substr**(參數1,參數2[,參數3])
  - 參數1為原始字符串
  - 參數2為擷取的啟始位元數，從0起算
  - 參數3為擷取的長度，省略時，表啟始位元後面全部擷取
- 範例：
  - `substr('abcde', 0, 3)` → 得到 abc
  - `substr('abcde', 2)` → 得到 cde

# 字串分割

- 函數：**explode('以什麼區隔', \$source)**
- 回傳值：分割後的字串陣列
- 舉例：
  - \$url = explode('.', 'www.ntu.edu.tw');
  - 回傳：
    - \$url[0] 裡面為 www
    - \$url[1] 裡面為 ntu
    - \$url[2] 裡面為 edu
    - \$url[3] 裡面為 tw

# 課堂練習 (4-4)

- 加強課堂練習(4-3)的程式
- 利用strlen函數，規定使用者密碼欄位長度必須在6~12個字元之間，否則會出現錯誤訊息

# 內建函數 – 數學相關

- `abs()`: 取絕對值
- `round()`: 四捨五入
- `ceil()`: 無條件進位
- `floor()`: 無條件捨去
- 其他的數學函數也很豐富：  
<http://www.php.net/manual/en/book.math.php>

# 取絕對值

- 函數：**abs(\$var)**
- 回傳的值：轉換之後的絕對值
- 舉例：**abs(\$num);**
  - PHP 官方網站的例子：

## Example #1 abs() example

```
<?php
$abs = abs(-4.2); // $abs = 4.2; (double/float)
$abs2 = abs(5);   // $abs2 = 5; (integer)
$abs3 = abs(-5); // $abs3 = 5; (integer)
?>
```

# 四捨五入

- 函數：**round(\$var)**
- 回傳的值：計算四捨五入後的數值
- 舉例：**round(\$num);**
  - PHP 官方網站的例子：

## Example #1 round() examples

```
<?php
echo round(3.4);           // 3
echo round(3.5);           // 4
echo round(3.6);           // 4
echo round(3.6, 0);         // 4
echo round(1.95583, 2);    // 1.96
echo round(1241757, -3);   // 1242000
echo round(5.045, 2);       // 5.05
echo round(5.055, 2);      // 5.06
?>
```

# 無條件進位

- 函數：**ceil(\$var)**
- 回傳的值：無條件進位後的數值
- 舉例：**ceil(\$num);**
  - PHP 官方網站的例子：

## Example #1 ceil() example

```
<?php
echo ceil(4.3);    // 5
echo ceil(9.999);  // 10
echo ceil(-3.14);  // -3
?>
```

# 無條件捨去

- 函數：**floor(\$var)**
- 回傳的值：無條件捨去後的數值
- 舉例：**floor(\$num);**
  - PHP 官方網站的例子：

## Example #1 floor() example

```
<?php
echo floor(4.3); // 4
echo floor(9.999); // 9
echo floor(-3.14); // -4
?>
```

# 內建函數 – 型態相關

- `is_???( )`: 判斷變數的資料型態函數
- `gettype( )`: 取得變數資料型態的函數
- `settype( )`: 指定變數資料型態的函數
  - 會影響原變數
- `intval( )`: 轉換成整數
  - 不會影響原變數
- `doubleval( )`: 轉換成浮點數
  - 不會影響原變數
- `number_format( )`: 格式化數值
- `unset( )`: 刪除變數釋放記憶體的函數

# 判斷變數的資料型態

- 函數：`is_???( $var )`
- 回傳的值：若資料型態與判斷函數相符，則回傳「`true`」，反之則回傳「`false`」
- 舉例：`is_int( $num );`  
*//判斷\$num是否為整數*

函數名稱	判斷的資料型態
<code>is_int()</code> <code>is_integer()</code>	整數判斷
<code>is_float()</code>	浮點數判斷
<code>is_numeric()</code>	數字判斷
<code>is_string()</code>	字串判斷
<code>is_array()</code>	陣列判斷

# 型態錯誤造成的計算錯誤問題

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>型態錯誤 範例</title>
5 </head>
6 <body>
7   <?php
8   $north = 1000;
9   $east = 2000;
10  $west = 3000;
11  $south = 4000;
12  $south = '文字';
13
14 echo '北區的總得票數為：' . $north . '<br>';
15 echo '東區的總得票數為：' . $east . '<br>';
16 echo '西區的總得票數為：' . $west . '<br>';
17 echo '南區的總得票數為：' . $south . '<br><br>';
18 $sum = $north + $east + $west + $south;
19 echo '全國總得票數為：' . $sum . '票';
20 ?>
21 </body>
22 </html>
```

北區的總得票數為：1000  
東區的總得票數為：2000  
西區的總得票數為：3000  
南區的總得票數為：文字

全國總得票數為：6000票

# 先經過確認為數值後再計算

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>型態檢範例</title>
5 </head>
6 <body>
7   <?php
8   $north = 1000;
9   $east = 2000;
10  $west = 3000;
11  $south = 4000;
12 // $south = '文字';
13
14 if (is_integer($north) && is_integer($east))
15 {
16   if(is_integer($south) && is_integer($west))
17   {
18     $sum = $north + $east + $west + $south;
19     echo '全國總得票數為：' . $sum . '票';
20   }
21   else
22   {
23     echo '西區或南區的資料型態錯誤！';
24   }
25 }
26 else
27 {
28   echo '北區或東區的資料型態錯誤！';
29 }
30 ?>
31 </body>
32 </html>
```

全國總得票數為 10000 票

西區或南區的資料型態錯誤！

# 取得變數資料型態的函數

- 函數名稱：**gettype(\$var)**
- 回傳的值：「integer」、「string」、「double」、「NULL」，其資料型態為「字串」
- 舉例：`gettype($name);`

# 指定變數資料型態的函數

- 函數：`settype($var, 'type')`，其中type可為`integer`、`double`、`string`、`array`
- 回傳的值：轉換成功回傳「true」；失敗回傳「false」
- 舉例：`settype($num, 'string')` 將整數轉成字串資料型態
- 程式裡最好不要任意變更資料型態，與其在執行過程中變更，不如一開始就宣告好變數的資料型態，以避免不必要的錯誤發生

# 字符串轉換成整數

- 字串中不包含數字：

```
$v1 = 'Hello';  
settype($v1, 'integer');  
echo $v1; // $v1=0
```

- 字串中包含數字：

```
$v2 = '100Hello';  
settype($v2, 'integer');  
echo $v2; // $V2=100
```

# 整數轉換成字串

- \$v3 = 100;  
settype(\$v3, 'string');  
echo \$v3; // \$v3='100'

# 浮點數轉成整數

- \$v4 = 50.5;  
settype(\$v4, 'integer');  
echo \$v4; // \$v4 = 50

# 整數轉換為浮點數

```
$v5 = 50;  
settype($v5, 'double');  
echo $v5; // $v5 = 50
```

# 浮點數轉為字串

- \$v6 = 70.5;  
settype(\$v6, 'string');  
echo \$v6; // \$v6 = '70.5'

# 將數值轉換成整數

- 函數：`intval($var)`
- 回傳的值：轉換之後的數值，資料型態為「整數」
- 舉例：`intval($num);`
- 利用`intval()`函數，可將小數、浮點數或字串等變數，全部轉換成「整數」

# 將數值轉成浮點數

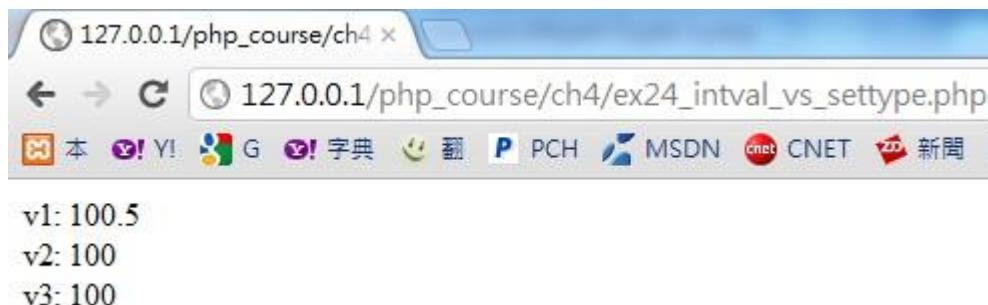
- 函數：**doubleval(\$var)**
- 回傳的值：轉換之後的數值，其資料型態為浮點數
- 舉例：**doubleval(\$num);**

# settype vs. intval/doubleval

- settype 會將變數數的類型真的改變
  - \$v = 100.1;
  - settype(\$v, 'integer');
  - echo \$v; // 印出 100
- intval 或 doubleval 只會將變數的值改變型態，並沒有影響到變數本身
  - \$v1 = 100.1;
  - \$v2 = intval(\$v1);
  - echo \$v1; // 印出 100.1
  - echo \$v2; // 印出 100

# settype vs. intval/doubleval

```
1 <?php
2 $v1 = 100.5;
3
4 $v2 = intval($v1);
5
6 $v3 = 100.5;
7
8 settype($v3, 'integer');
9
10 echo 'v1: ' . $v1 . '<br>; // v1: 100.5
11 echo 'v2: ' . $v2 . '<br>; // v2: 100
12 echo 'v3: ' . $v3;           // v3: 100
13 ?>
```



# 將數值格式化的函數

- 函數：`number_format($var, $dec)`
- 回傳的值：格式化後的數值
- 用法1：
  - `$price = 10000;`
  - `echo number_format($price);`
    - 印出 10,000 // 每三位數會自動加一個逗點
- 用法2：
  - `$score = 123.45678;`
  - `echo number_format($score, 2);`
    - 印出 123.46 // 取到小數下第2位，四捨五入！

# 刪除變數釋放記憶體的函數

- 函數：**unset(\$var)**
- 回傳的值：沒有
- 舉例：`unset($money); //將$money  
釋放`
- `unset()`只能用來釋放變數，無法釋放常數，若是嘗試釋放常數，會發生執行錯誤的警告

# unset() 的範例

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>unset 變數 範例</title>
5 </head>
6 <body>
7   <?php
8   $var = 'Hello';
9   echo '釋放前：' . $var . '<br>';
10  unset($var);
11  echo '釋放後：' . $var;
12  ?>
13 </body>
14 </html>
```

釋放前：Hello  
釋放後：

# 如果 unset() 常數的錯誤

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>unset 常數 範例</title>
5 </head>
6 <body>
7   <?php
8   define('PI', 3.1415926);
9   unset(PI);
10  echo PI;
11  ?>
12 </body>
13 </html>
```

**Parse error:** syntax error, unexpected ')', expecting T\_PAAMAYIM\_NEKUDOTAYIM in  
C:\xampp\apache\htdocs\php\_course\ch4\ex18\_unsetConst.php on line 9

# 課堂練習 (4-5)

- 設計一個表單，讓使用者可以填寫一個數字格式的文字欄位
- 若使用者輸入的內容不是數字則印出錯誤訊息
- 若是數字則計算出：
  - 絕對值
  - 四捨五入後的值
  - 無條件進位的值
  - 無條件捨去的值

# 內建函數 – 日期時間相關

- `date()`: 取得系統日期/時間
- `mktime()`: 取得系統的timestamp
- PHP裡的日期時間函數非常多，但絕大部份時我們只需要用到上述的兩個
- 其他的日期時間函數：  
<http://www.php.net/manual/en/book.datetime.php>

# 先調整一下系統時間

- 用 notepad++ 打開 php.ini 的位址  
(預設在 : C:\Windows\php.ini)

```
625 | ;:::::::::::::::::::  
626 | ; Module Settings ;  
627 | ;:::::::::::::::::::  
628 |  
629 | [Date]  
630 | ; Defines the default timezone used by the date functions  
631 | date.timezone = Asia/Taipei  
632 |
```

- 將 631 行改成
  - date.timezone = Asia/Taipei
- 存檔

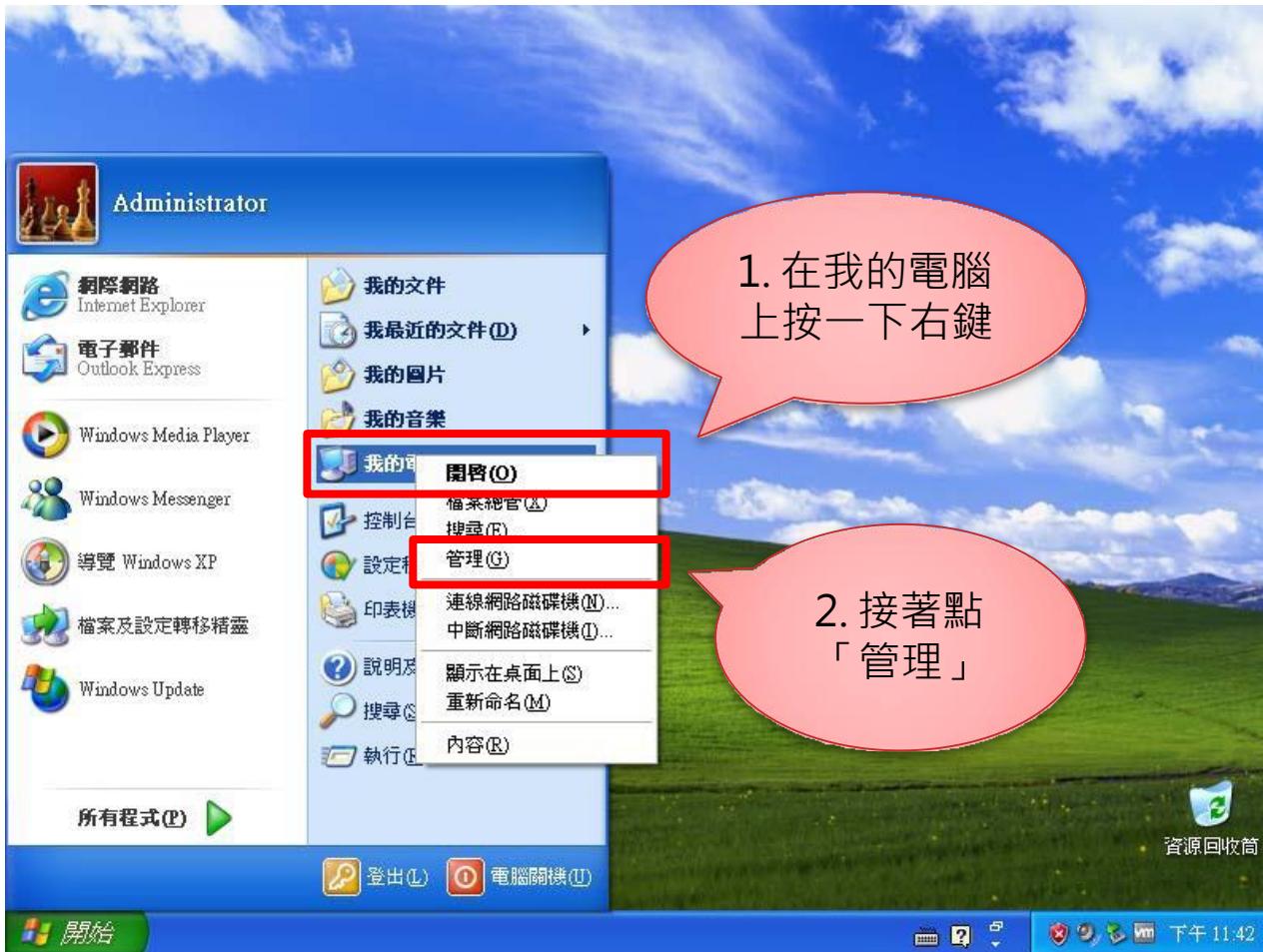
# 調整完時間後需重啟 Apache

- Windows XP 版本 (1)



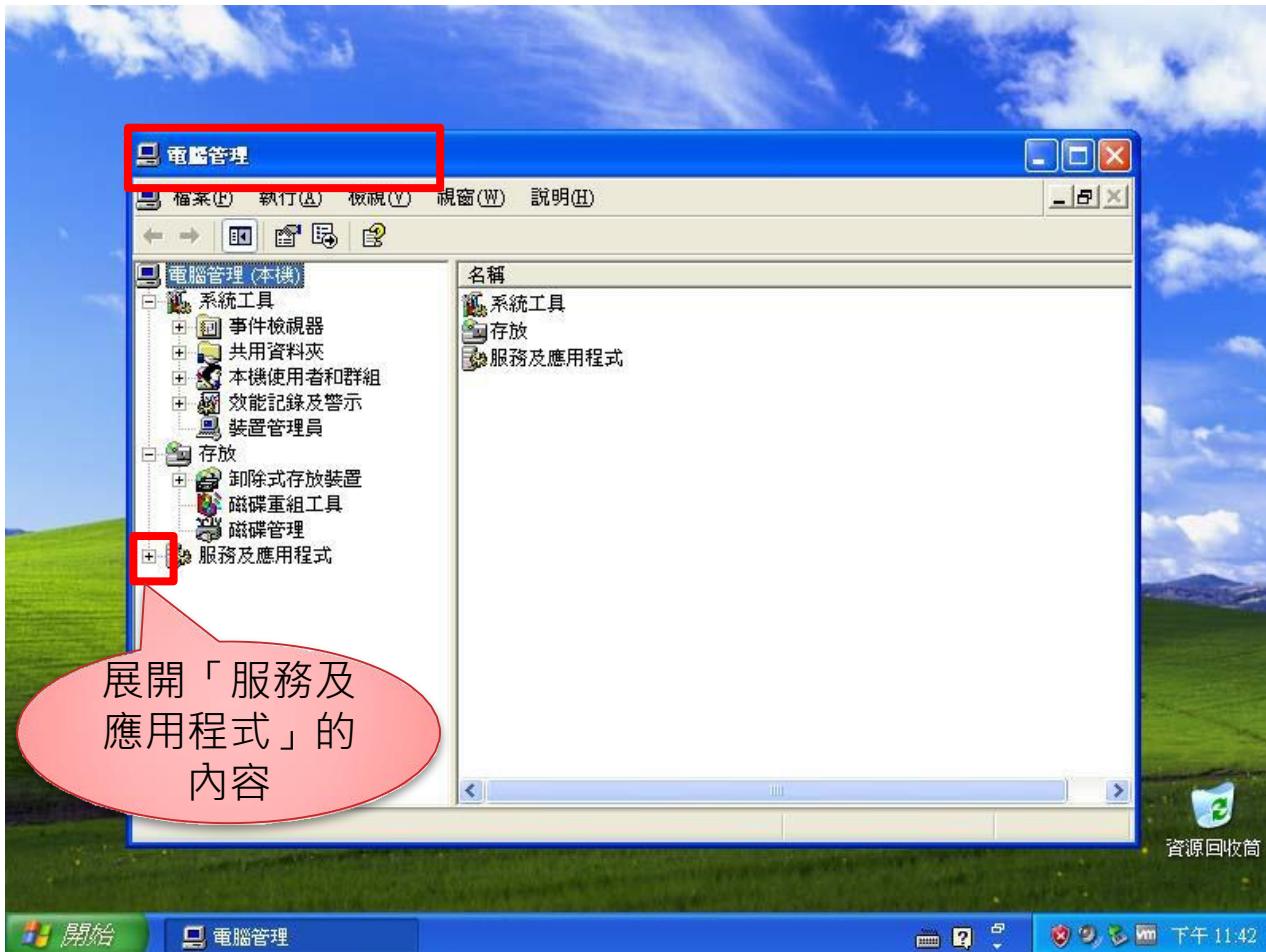
# 調整完時間後需重啟 Apache

- Windows XP 版本 (2)



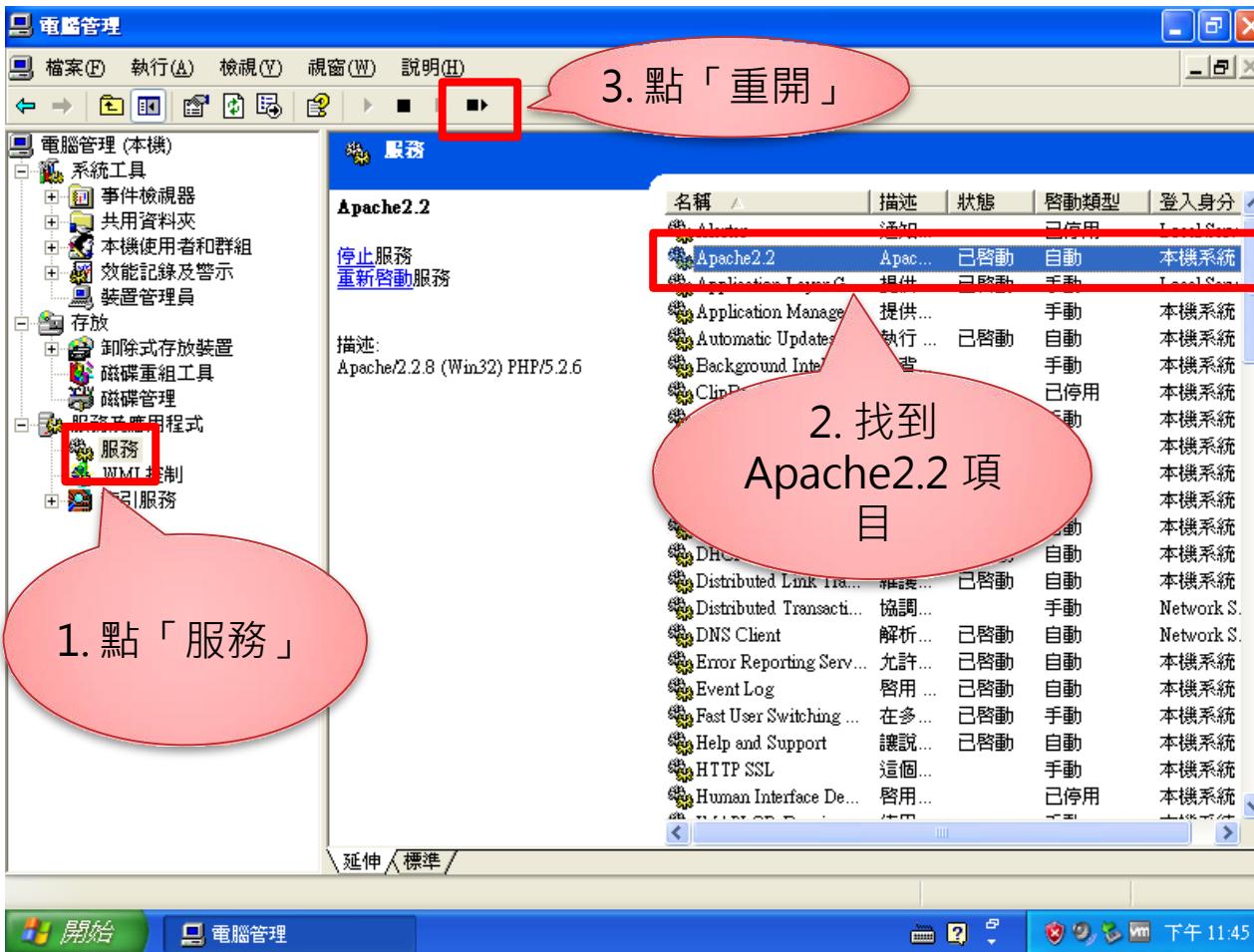
# 調整完時間後需重啟 Apache

- Windows XP 版本 (3)



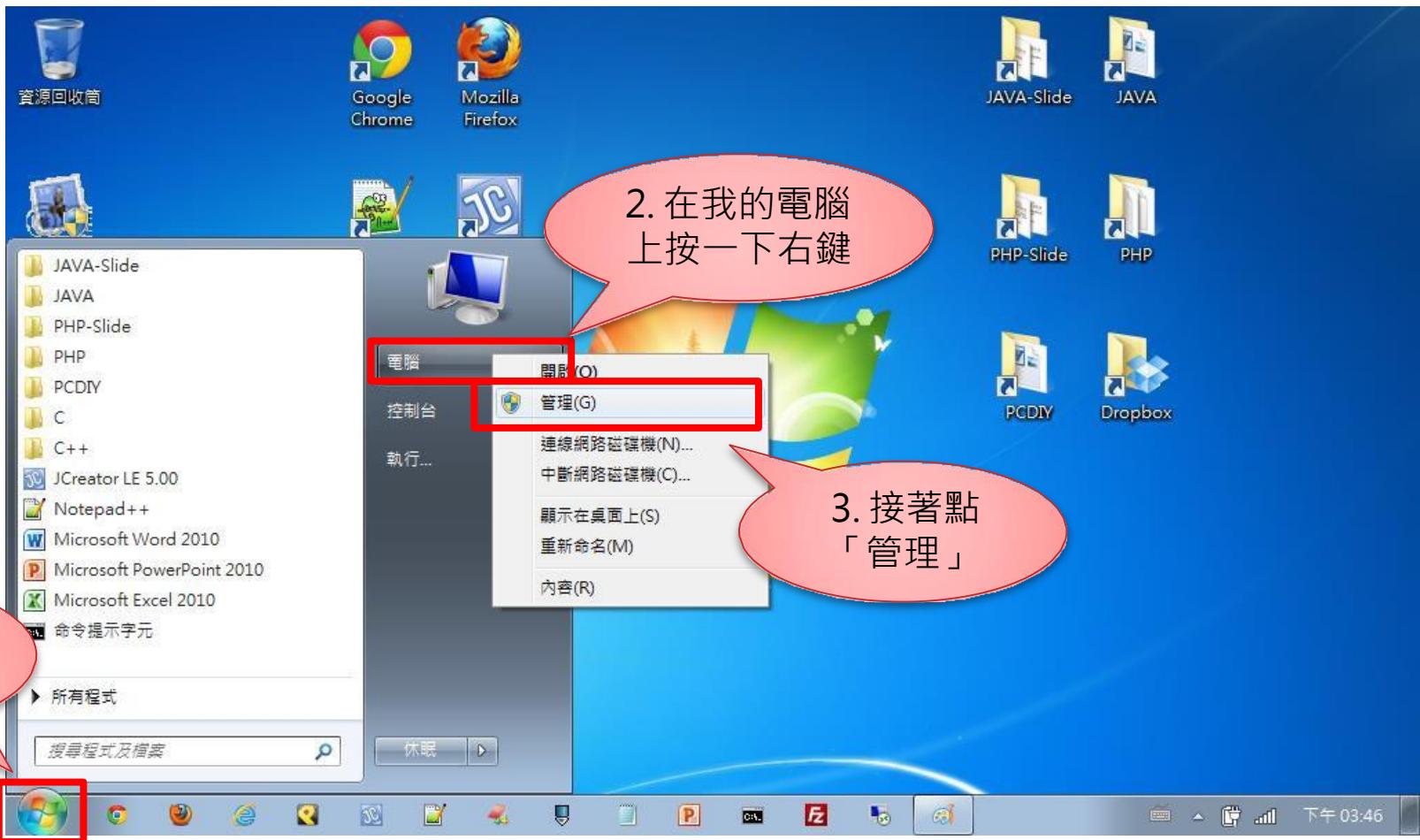
# 調整完時間後需重啟 Apache

- Windows XP 版本 (4)



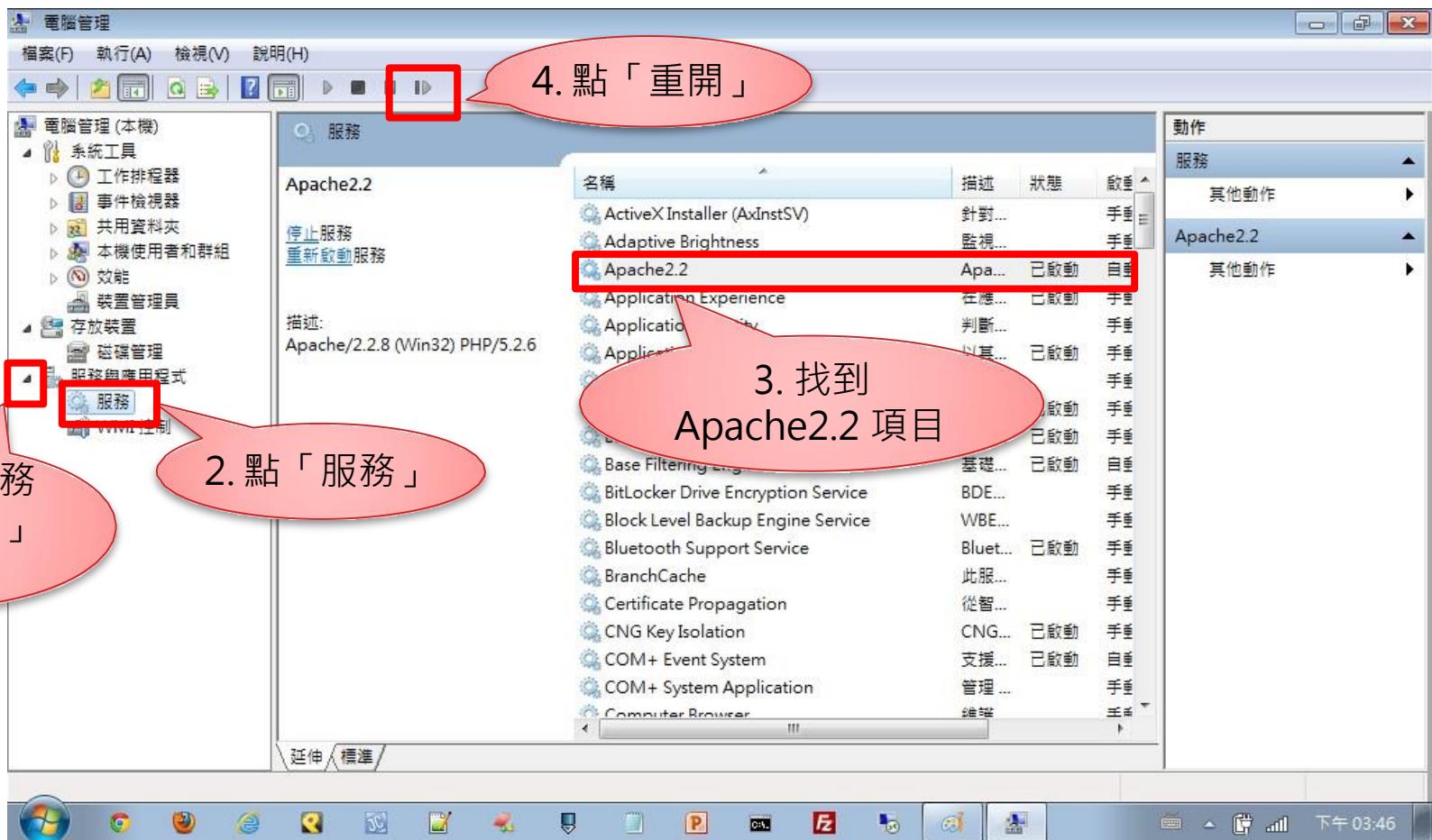
# 調整完時間後需重啟 Apache

- Vista / Windows7 版本 (1)



# 調整完時間後需重啟 Apache

- Vista / Windows7 版本 (2)



# 取得系統日期時間

- 函數：**date()**
- 回傳的值：日期或時間，其資料型態為「字串」
- 舉例：`date('Y-m-d H:i:s');`  
會輸出：2010-08-05 11:00:00
- `date()`的輸入參數眾多，請參考下頁
- 最新的`date()`參數可參考：  
<http://www.php.net/manual/en/function.date.php>

# date() 參數詳解

a	am 或是 pm
A	AM 或是 PM
d	幾日 · 二位數字 · 若不足二位則前面補零; 如: 01 至 31
D	星期幾 · 三個英文字母; 如: Fri
F	月份 · 英文全名; 如: January
h	12 小時制的小時; 如: 01 至 12
H	24 小時制的小時; 如: 00 至 23
g	12 小時制的小時 · 不足二位不補零; 如: 1 至 12
G	24 小時制的小時 · 不足二位不補零; 如: 0 至 23
i	分鐘; 如: 00 至 59
j	幾日 · 二位數字 · 若不足二位不補零; 如: 1 至 31
l	星期幾 · 英文全名; 如: Friday
m	月份 · 二位數字 · 若不足二位則在前面補零; 如: 01 至 12
n	月份 · 二位數字 · 若不足二位則不補零; 如: 1 至 12
M	月份 · 三個英文字母; 如: Jan
s	秒; 如: 00 至 59
S	字尾加英文序數 · 二個英文字母; 如: th、nd
t	指定月份的天數; 如: 28 至 31
U	從總 1970/1/1 00:00:00 到現在經過的秒數
w	數字型的星期幾 · 如: 0 (星期日) 至 6 (星期六)
Y	年 · 四位數字; 如: 1999
y	年 · 二位數字; 如: 99
z	一年中的第幾天; 如: 0 至 365

# 取得系統時間的範例

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>datetime 範例</title>
5 </head>
6 <body>
7   <?php
8   $hour = date('H');
9   $min = date('i');
10  $sec = date('s');
11  echo '現在時刻是 ' . $hour . ' 時 ';
12  echo $min . ' 分 ' . $sec . ' 秒 ';
13  ?>
14 </body>
15 </html>
```

現在時刻是 17 時 37 分 51 秒

# 課堂練習 (4-6)

- 寫一個PHP程式可以輸出：

今天是民國 99 年 8 月 5 日  
現在是早上 xx 點 xx 分 xx 秒

# 從絕對日期到某日的時間差

- 函數：  
**mkttime([hour],[minute],[second],[month],[day],[year])**
- 回傳的值：從絕對日期到該天的秒數，資料型態為整數
- 舉例：**mkttime(0,0,0,1,1,2020)**

# 取得時間差的範例

```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>mktime 範例</title>
5 </head>
6 <body>
7   <?php
8   // 假設下一代 iPhone 是 2015年6月1日00:00:00 上市
9   $next_iPhone = mktime(0,0,0,6,1,2015);
10
11  // 取得目前系統絕對時間
12  $currentTime = date('U');
13
14  // 計算剩幾天
15  $leftTime = ($next_iPhone - $currentTime);
16  $leftTime = intval($leftTime / 60 / 60 /24);
17
18  echo '距離下一代 iPhone 上市還有 ' . $leftTime . ' 天';
19 ?>
20 </body>
21 </html>
```

距離下一代 iPhone 上市還有 1500 天

# 檔案上傳

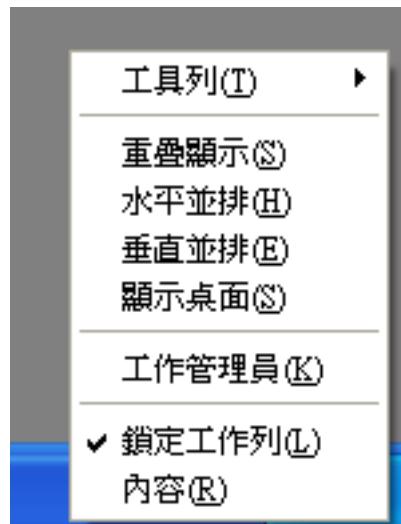
- 利用 PHP 做檔案上傳時的幾個重點：
  - 檔案該上傳到哪裡？
  - <form> 標籤的改寫
  - <input type="file"> 的檔案選擇鈕
  - 怎麼讓 PHP 取得檔案資訊

# 檔案該上傳到哪裡

- 檔案既然是從使用者的電腦裡上傳到伺服器上的話，表示伺服器必須開一個門讓使用者能把檔案上傳
- 也就是說：
  - 要開放上傳的權限給使用者
  - 必須做好管理，除了提供上傳的路徑外，路徑仍不應該給使用者上傳

# 確定使用者的身份 (1)

- 請開啟工作管理員
  - 在windows下方工具列上空白處按一下右鍵→工作管理員
  - 或按  $ctrl + alt + del$  選工作管理員

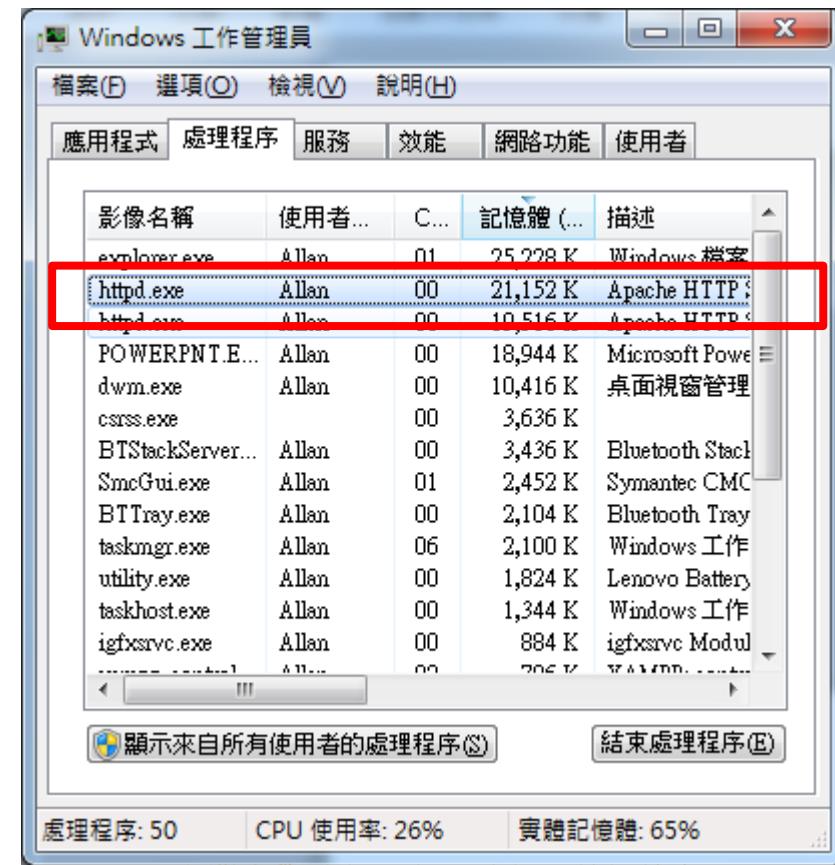
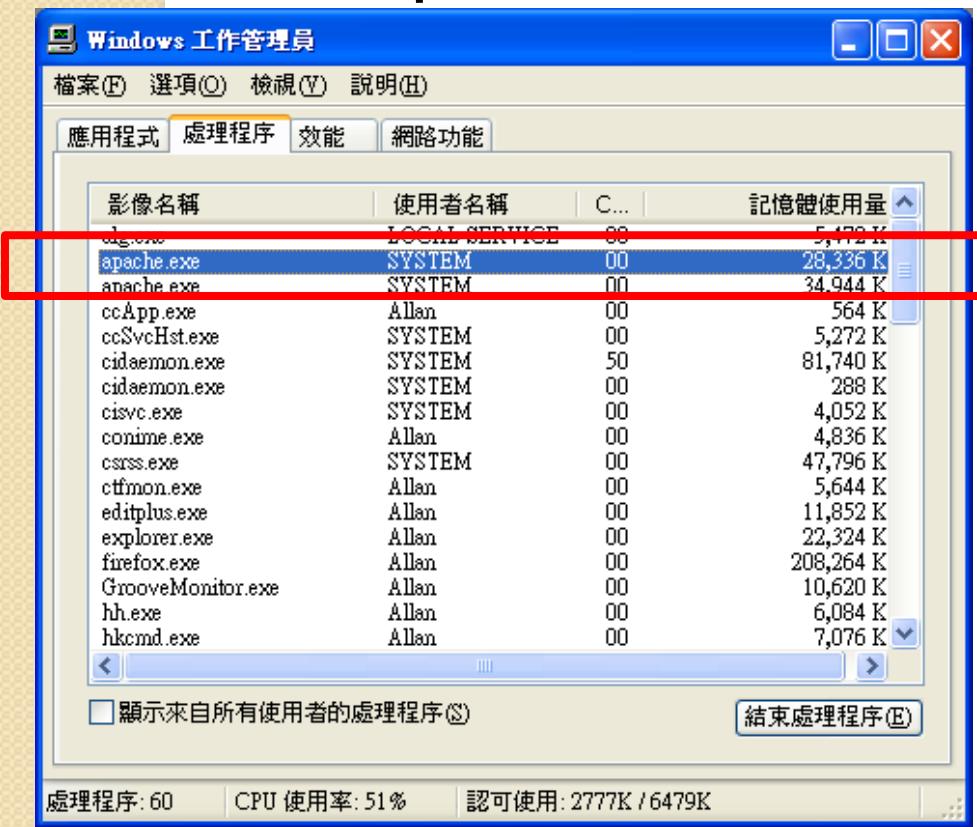


# 確定使用者的身份 (2)

- 切換到「處理程序」的頁籤尋找 Apache 的處理程序是誰在用
  - Apache 的程序可能是：
    - apache.exe
    - httpd.exe
  - 使用者可能是：
    - SYSTEM
    - 你

# 確定使用者的身份 (3)

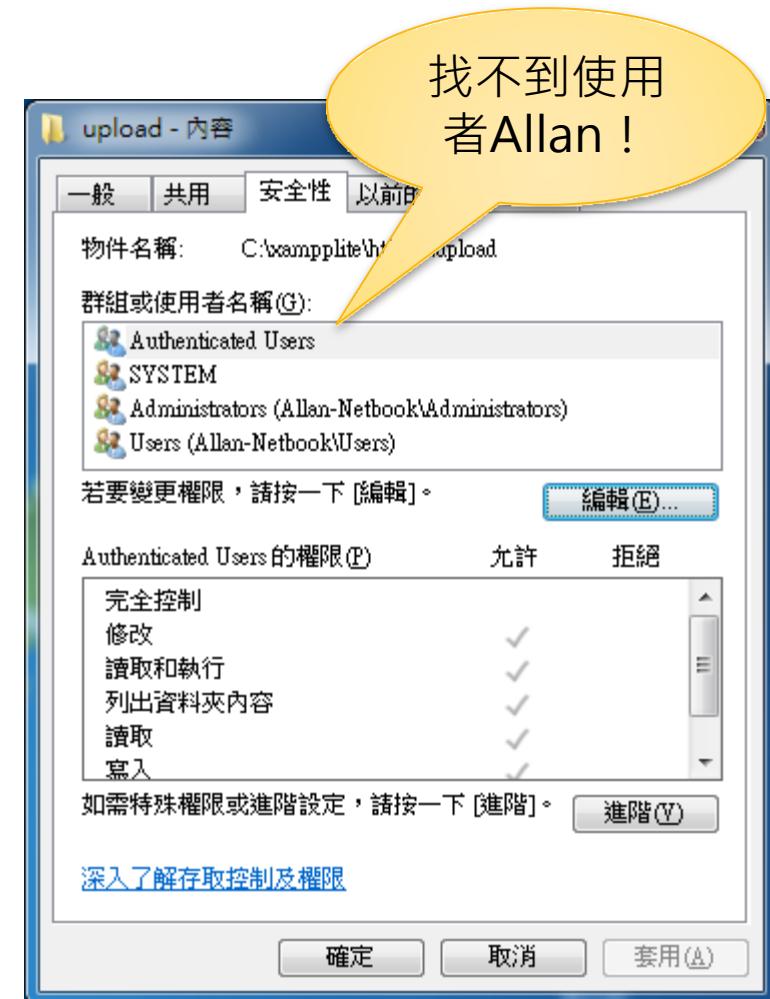
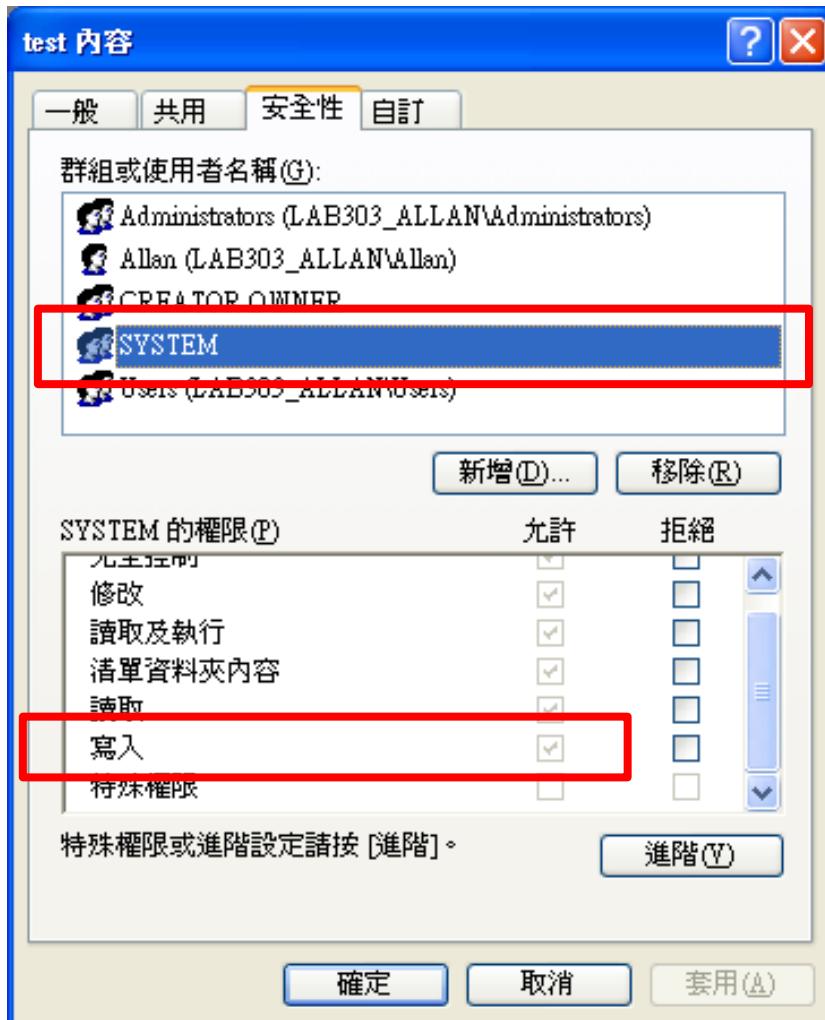
- 請記得現在是哪個使用者在使用你的 Apache



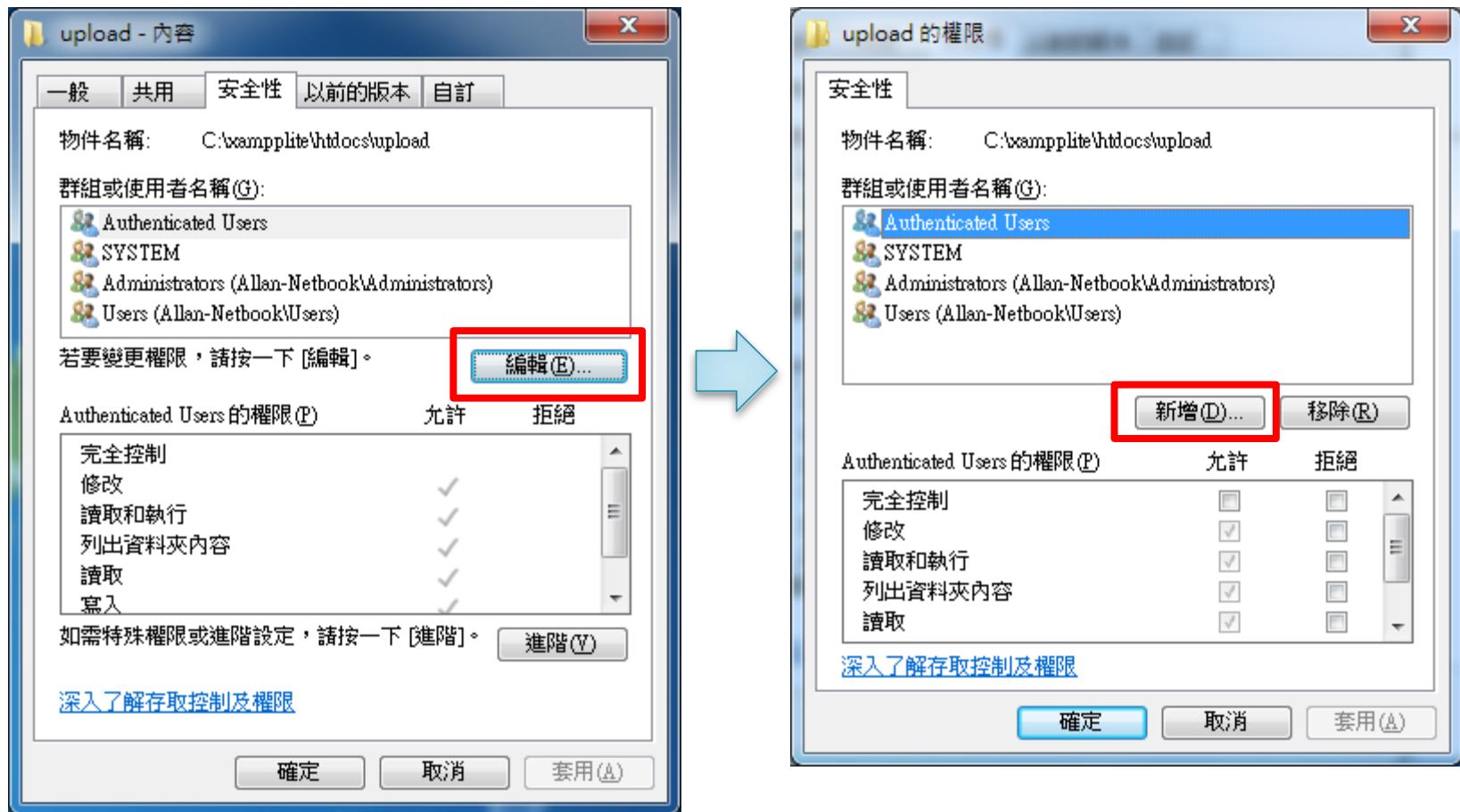
# 開放使用者上傳的目錄 (1)

- 在 C:\AppServ\www 下新增一個資料夾，命名 upload
- 新增完之後在資料夾上點右鍵 → 選內容
- 內容頁出來後選到「安全性」的頁籤
- 看看剛剛在使用 Apache 的使用者是不是有「寫入」的權限

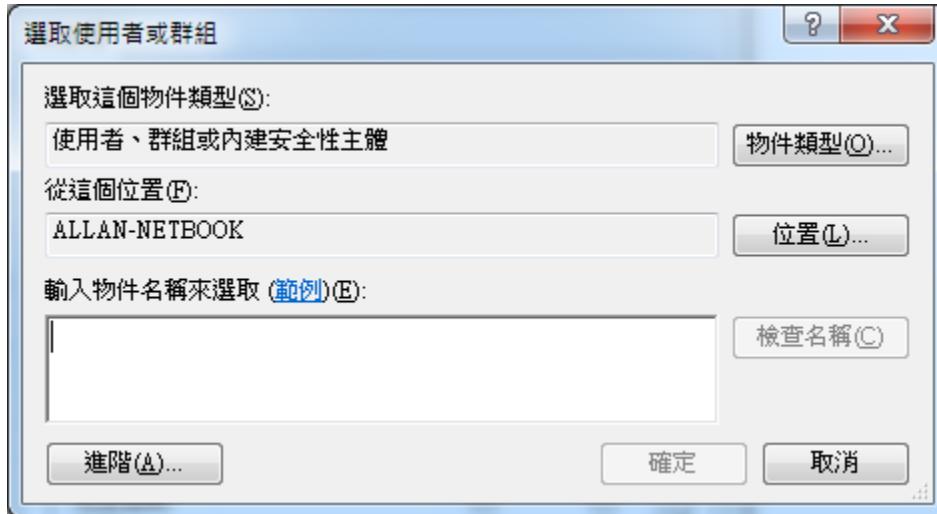
# 開放使用者上傳的目錄 (2)



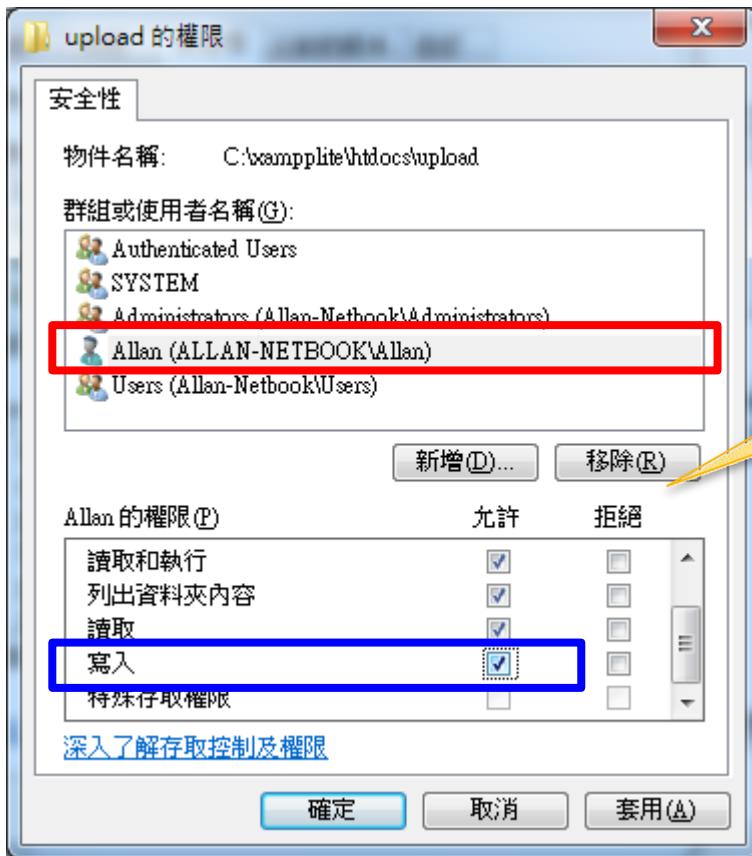
# 開放使用者上傳的目錄 (3)



# 開放使用者上傳的目錄 (4)



# 開放使用者上傳的目錄 (5)



確定使用者 Allan  
有寫入權限！

# <form> 標籤的改寫

- 無論原先你的<form>已經寫了多少屬性，必須再加上 enctype 屬性：
  - enctype="multipart/form-data"
  - 例如：
    - <form method="POST" action="test.php" enctype="multipart/form-data">

# 檔案選擇鈕

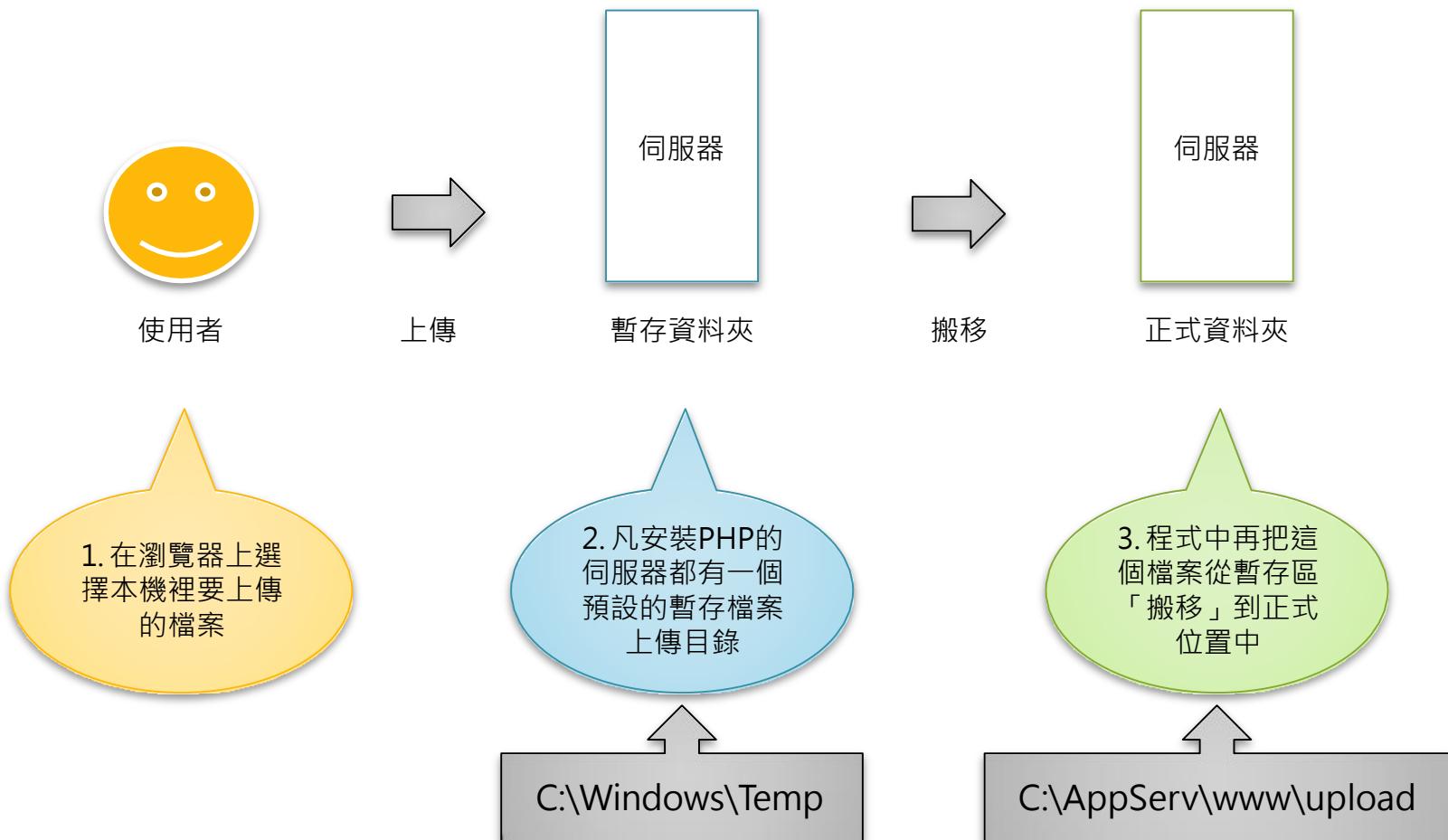
- HTML 表單中除了之前介紹的元件外，還有一個檔案選擇鈕的元件：
  - <input type="file" name="自訂名稱">
  - 會長成這樣：



# 怎麼讓PHP取得檔案資訊

- 利用 `$_FILES` 這個陣列，可以取得使用者上傳的資料，如：
  - `$_FILES['自訂名稱']['name']`
    - 客戶端機器文件的原名稱
  - `$_FILES['自訂名稱']['type']`
    - 文件的 MIME 型別，需要瀏覽器提供該訊息的支援，例如"image/gif"
  - `$_FILES['自訂名稱']['size']`
    - 已上傳文件的大小，單位為 Byte(s)
  - `$_FILES['自訂名稱']['tmp_name']`
    - 文件被上傳後在伺服端儲存的臨時文件名
  - 其他的 MIME type：MIME\_Type.pdf

# PHP中檔案上傳的架構



# 檔案上傳相關函數

- move\_uploaded\_file(**tmp**, **path**)
  - **tmp**: 暫存檔的檔名
    - 通常是：
      - `$_FILES['自訂名稱']['tmp_name']`
  - **path**: 公開資料夾路徑，含完整路徑與檔名
    - 通常是：
      - `C:\AppServ\www\upload\$_FILES['自訂名稱']['name']`
      - `C:\AppServ\www\upload\?$_FILES['自訂名稱']['type']`
- basename(**path**)
  - 過濾 Windows 中的 \ 與 UNIX / 等路徑符號，只留下單純的**檔案名稱.副檔名**
    - 例：`basename('C:\test\test.jpg');` → 得到 `test.jpg`

# 檔案上傳的衍生問題 (1)

- 問題1. 若有相同檔名的檔案該怎麼辦？
  - 使用者A 上傳了一張他的個人照片 me.jpg 到伺服器上
  - 使用者B 也一樣上傳了一張照片 me.jpg
  - 則使用者B 會把使用者A 的照片給覆蓋掉！
- 解決方式：
  - 將檔案重新命名
    - 在每個檔案前面加上上傳的日期時間
    - 例：me.jpg → 上傳後 **100223125019\_me.jpg**

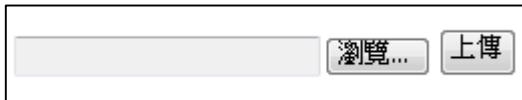
# 檔案上傳的衍生問題 (2)

- 問題2. 中文檔名
  - 如前面章節所述，編碼問題、中文並非所有伺服器系統本身所支援，因此當使用者上傳中文檔名的檔案時有機會造成亂碼！
- 解決方式：
  - 幫使用者重新把檔案命名，如：
    - 直接使用 `$_FILES['自訂名稱']['tmp_name']` 當新檔名
    - 使用亂數產生的一組檔名當新檔名

# 練習一下

- upload.html

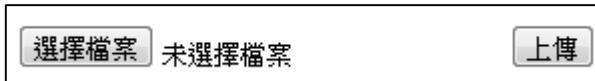
```
1 <html>
2 <head>
3   <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
4   <title>檔案上傳 範例</title>
5 </head>
6 <body>
7   <form method="POST" action="upload.php" enctype="multipart/form-data">
8     <input type="file" name="filename">
9     <input type="submit" value="上傳">
10  </form>
11 </body>
12 </html>
```



IE



FF



GC

各家瀏覽器  
的顯示樣子  
有異



```
1 <?php
2 // 檢查使用者是否有輸入要上傳的檔案
3 // 若使用者有輸入檔案，$_FILES['filename']['name'] 中會有檔案名稱
4 // 若使用者沒有輸入，則$_FILES['filename']['name'] 中會是空白（即 false）
5 if ($_FILES['filename']['name'])
6 {
7     // 公開目錄的本機位址
8     $path = 'C:\\AppServ\\www\\upload\\';
9
10    // 公開目錄的網址
11    $url = 'http://127.0.0.1:8080/upload/';
12
13    // 於 c:\\Windows\\Temp\\ 暫存區中的檔案名稱
14    $tmp = $_FILES['filename']['tmp_name'];
15
16    // 利用 basename() 確保檔名沒有包含其他不該有的路徑分割符號
17    $filename = basename($_FILES['filename']['name']);
18
19    // 加上絕對時間的檔名
20    #$currentTime = date('U');
21    #$filename = $currentTime . basename($_FILES['filename']['name']);
22
23    // 加上日期時間的檔名
24    #$currentDateTime = date('YmdHis');
25    #$filename = $currentDateTime . basename($_FILES['filename']['name']);
26
27    // 直接用 $_FILES['自訂名稱']['tmp_name'] 當新檔名，避免撞名及中文檔名
28    #$filename = basename($_FILES['filename']['tmp_name']);
29
30    // 用 move_uploaded_file() 把檔案從暫存區中搬到正式目錄中
31    // 若搬移失敗，大部份是因為檔名關係，如：使用了非英文的檔名
32    if (!move_uploaded_file($tmp, $path . $filename))
33    {
34        echo '上傳圖檔時發生錯誤，系統找不到檔案，請避免使用中文檔名！';
35        exit;
36    }
37    else
38    {
39        // 上傳成功！
40        echo '上傳成功，看看<a href="'. $url . $filename .'>你上傳的檔案</a>吧！';
41    }
42}
43?>
```

# 上傳檔案的另一種方式

- 若搭配資料庫系統，則有另外一種檔案上傳的方式
- 將上傳後的檔案轉成二進位編碼後儲存在某一個資料庫的欄位中
- 要瀏覽檔案時將該筆資料讀出後再轉回瀏覽器可解讀的格式
- 如此一來便沒有檔名衝突、中文檔名等問題了
- 相關範例會在第7章時介紹

# 回家作業 – 加強版水果字典

- 加強 hw3 的簡易水果字典
- 加強點：
  - 改用二維陣列儲存資料，第一維存該單字的起始字母，如：\$fruit['a']['apple'] = '蘋果'；
  - 輸入中文也能反查
  - 輸入英文時大小寫都能查到資料
  - 增加單字瀏覽列表，A-Z 一字排開供使用者點選瀏覽該字母下的所有單字 (Hint: range())
    - 該字母下沒有水果資料時不顯示超連結

# 範例畫面

## 簡易字典查詢系統

請輸入要查詢的中/英文單字：

送出查詢

或直接瀏覽單字列表 [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](#)

A:

apple: 蘋果



# 繳交回家作業

- 把「加強版水果字典」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：「[PHP265][hw][4]姓名」
  - 收件人：[m10112101@mail.ntust.edu.tw](mailto:m10112101@mail.ntust.edu.tw)
- Deadline: 下周同一時間

# 繳交課堂作業

- 把課堂練習(4-1)~課堂練習(4-n) 分別儲存在：
  - pt4-1 ~ pt4-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：「[PHP265][pt][4]姓名」
    - 收件人：[m10112101@mail.ntust.edu.tw](mailto:m10112101@mail.ntust.edu.tw)
    - Deadline: 下周同一時間