

第四章 PHP 語言

4-1 PHP 語法彙集

4-1-1 PHP 簡介

PHP (Hypertext Preprocessor) (超文字預先處理器) 是附屬於網頁伺服器(如：**Apache Server**) 的程式語言。基本上 Web Server 上僅儲存 HTML 文件，當瀏覽器要求文件時，則將 HTML 文件下載到瀏覽器並顯示其上。然而文件內容由 Makeup 標示如何顯示，譬如文字的字體、大小、顏色等等，瀏覽器依照這些標示將文字顯示出多采多姿的型態。然而利用 HTML 所產生的文件大多是固定死板的，並無法隨環境或需求改變文件內容。PHP 就是用來改變此狀態，讓文件可以隨環境或需求改變，它被嵌入於網頁伺服器(如 Apache)內。吾人將利用 PHP 程式編寫網頁欲顯示內容，網頁伺服器執行 PHP 程式，並依照程式內容製作出 HTML 文件，再下載給瀏覽器顯示，就可以依照程式改變文件內容。譬如：吾人要在網頁上顯示 10 行『我愛您 !!』，利用 HTML 語言標示則須連續寫 10 句，但利用 PHP 程式製作只要寫一句『我愛您 !!』，用一個迴圈繞 10 次即可。簡單的說，PHP 功能是 HTML 文件內容產生器，又稱為『**PHP 引擎**』(**PHP Engine**)。

由此可見，PHP 程式功能很大，隨時環境需求的進化與改變，版本升級也非常快速，本書並打算詳細介紹 PHP 語法，僅介紹較常用或與本書有關的語句，如 PHP-MySQL 函數引用方面。

4-1-2 PHP 常用語句

吾人將 PHP 程式常用語句歸納如下：

功能	基本語句
PHP 嵌入語法	<?php PHP 程式碼; ... ?> or <? PHP 程式碼;... ?>
單行注解	// 這一行是注解
多行注解	/* 這是注解的範圍 ... */
基本輸出函數(換行)	echo "Hello, Word"; print "Hello word"; echo " ";
整數變數與輸出	\$number = 50; \$number = \$number + 1 echo \$number;
浮點數變數與輸出	\$number1 = 45.78; print \$number1;
字串變數與輸出	\$name = "Mary"; \$str1 = "Hello!, \$name"; echo \$str1;

布林變數	<code>\$boolean1 = true; \$boolean2 = false;</code>
變數空值	<code>\$data = null;</code> 或 <code>unset(\$data);</code>
強制轉換	<code>\$a1=45.6; \$a2 = (int)\$a1;</code>
函數內引用全域變數	<code>global \$number1</code>
基本運算子	<code>+, -, *, /, %, a++, a--,</code>
指定與複合運算子	<code>=, +=, -=, *=, /=, %=,</code>
左右移位元	<code>\$a<<=1; \$a = \$a <<1; \$a>>=2; \$a = \$a >>1</code>
比較運算子	<code>==, ===, !=, <>, !==, <, >, <=, >=</code>
邏輯運算子	<code>&&(and), (or), !(not), xor</code>
位元運算子	<code>\$(AND), (OR), ^(XOR), ~(NOT), <<(左移), >>(右移)</code>
字串連接	<code>(.)</code> “AB”.”CD” => ABCD
if 條件敘述	<code>if(\$number == “是”) { ... PHP 敘述句... }</code>
if/else 條件敘述	<code>if(\$number == “是”) { ... PHP 敘述句... } else { ... PHP 敘述句... }</code>
switch(變數或運算式) 敘述	<code>switch (\$n1/\$n2) { case “1”: echo “結果是 1”; break; default: “以上皆非”; }</code>
for 迴圈	<code>for (\$i=1; \$i<=100; \$i++) { \$sum +=\$i; }</code>
while 迴圈	<code>While(\$i<=100) { \$sum = \$sum +\$i; \$i++; }</code>
do while 迴圈	<code>do { \$sum = \$sum + \$i; \$i++; }while(\$i <= 100);</code>
迴圈中斷與繼續	<code>break; continue;</code>
自訂函數宣告	<code>function sum(\$a, \$b) { return \$a+\$b; }</code>
自訂函數呼叫	<code>\$total = sum(45, 70);</code>

4-1-3 PHP 陣列語法

陣列宣告與指定	範 例
陣列元素指定與存取	<code>\$name[] = "張有名
"; \$name[] = "劉先生"; echo \$name[0]; echo \$name[1];</code>
一維陣列宣告	<code>\$score = array(56, 78, 90, 94, 88); echo \$score[2];</code>
陣列變數指定	<code>\$data[] =23; \$data[]=45; 則 data[0]=23, data[1]=45</code>
foreach 陣列迴圈	<pre>foreach (\$score as \$element) { echo \$element."\t"; \$sum += \$element; }</pre>
二維陣列宣告	<code>\$row0 = array(1, 2); \$row1 = array(3, 4); \$row2 = array(5, 6); \$array = array(\$row0, \$row1, \$row2);</code>

4-2 PHP 程式架構

4-2-1 PHP 與 HTML 的關係

(A) HTML 與 PHP 兩者關係如下：

- (1) HTML 是製作網頁文件的原始稿，利用標籤符號(<...>)告訴瀏覽器如何顯示文件內容，它的檔案格式是 html 或 htm，譬如 index.html 或 index.htm。
- (2) PHP 是 HTML 的嵌入程式，具有一般程式設計的能力，它將程式執行結果寫入 HTML 文件內，瀏覽器再由 HTML 文件內顯示 PHP 的執行結果。具有 PHP 文件稿檔案的副檔名必須是 php，譬如 index.php。
- (3) PHP 嵌入程式片段的開始是 `<? php`，並以 `?>` 標籤作為結束。
- (4) PHP 的每一個敘述句必須以分號 (;) 代表結束，譬如 `echo "
;`;

圖 4-1 為 PHP 程式嵌入 HTML，並執行 PHP 程式後結果的範例：

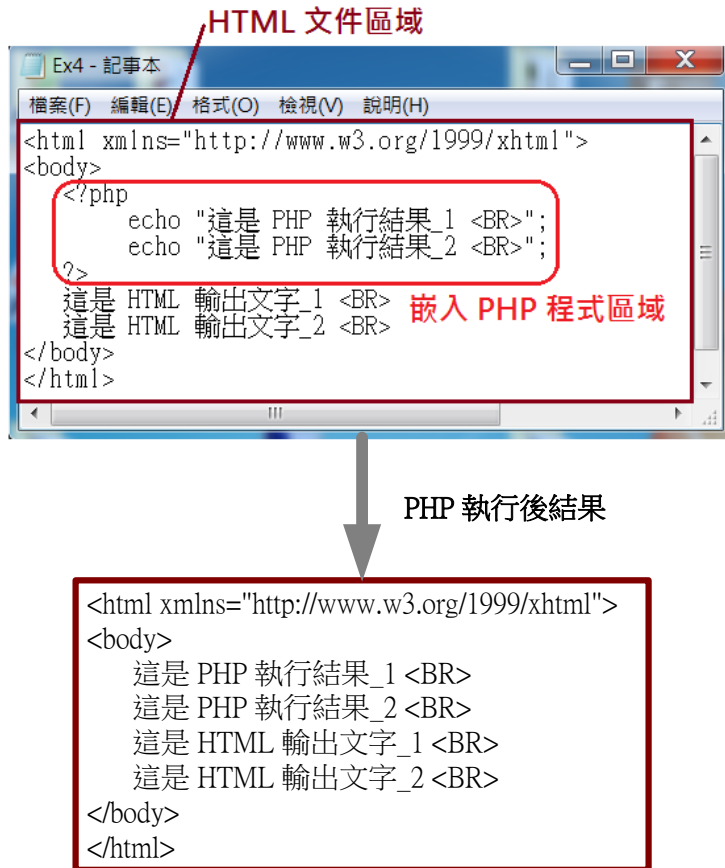


圖 4-1 PHP 與 HTML 之關係

瀏覽器執行如下：(圖 4-2)

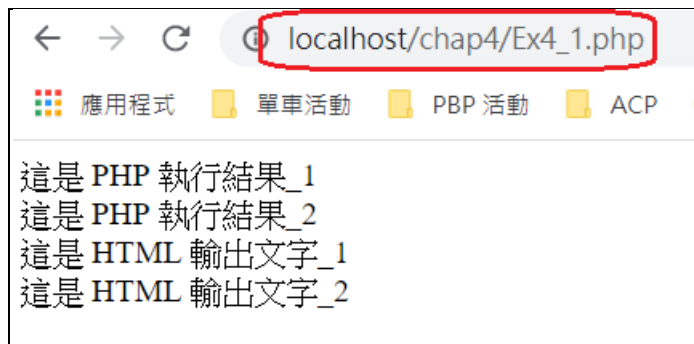


圖 4-2 PHP&HTML 關係二

(B) 程式範例(Ex4-1.php)如下：

```

1 <html xmlns="http://www.w3.org/1999/xhtml">
2 <body>
3   <p> PHP 與 HTML 關係之範例 </p>
4   <?php
5     echo "這是 PHP 執行結果_1<BR>";
6     echo "這是 PHP 執行結果_2<BR>";
7   ?>
8   這是 HTML 輸出文字_1<BR>
9
10
11

```

```

12     這是 HTML 輸出文字_2<BR>
13
14     </body>
        </html>
    
```

4-2-2 PHP 變數宣告與範例

凡是程式語言一定會有變數，PHP 屬於直譯程式 與一般草稿語言(如 javascript) 一樣，不需要預先宣告變數，變數也沒有型態之分，變數的前面一定要有錢記號(\$)，譬如 \$data。我們用下表歸納 PHP 變數的使用方法。

變數操作	說 明
\$value= 20;	產生變數 value 並給予內容 20。
\$value = \$value + 10;	將 value 內容拿出來，加 10 後再存回 value。
\$a = \$value;	產生變數 a，並將變數 value 內容複製到 a。
\$value = \$a + \$value;	變數 a 與 value 內容相加後，再存入 value 內。
\$b = function_1();	將函數 function_1 執行結果存入變數 b。
\$a++;	相當於 \$a = \$a + 1;
\$a--;	相當於 \$a = \$a - 1;
\$data = '我的資料';	存入字串資料，以單引號包起來('...')。
\$data = "您的資料";	同上，用雙引號包起來(" ... ")。
+、-、*、/、%	變數的加、減、乘、除、餘數運算符號。

● 程式範例(Ex4-2.php)如下：

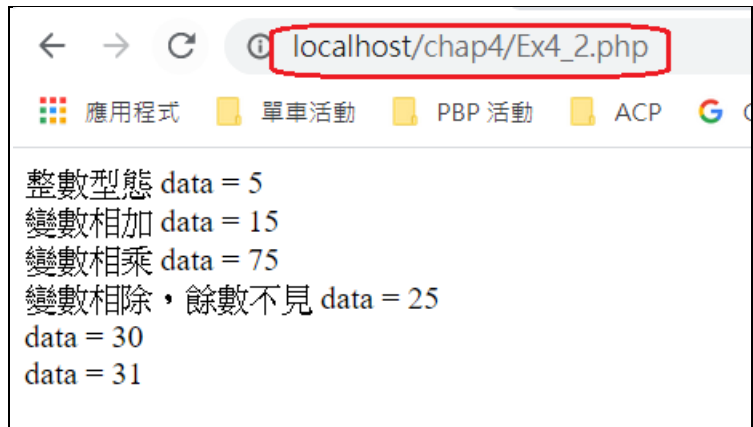
```

1     <?php
2         $data = 5;
3         echo "整數型態 data = $data <BR>";
4         $data = $data + 10;
5         echo "變數相加 data = $data <BR>";
6         $data = $data * 5;
7         echo "變數相乘 data = $data <BR>";
8         $prime = 3;
9         $data = $data / $prime;
10
11
12
    
```

```

13     echo “變數相除，餘數不見 data = $data <BR>”;
14
15     $data = 20;
16     $data += 10;
17     echo “data = $data <BR>”;
18     $data++;
19     echo “data = $data <BR>”;
20     ?>
    
```

● 執行結果：



4-2-3 PHP 輸出&範例

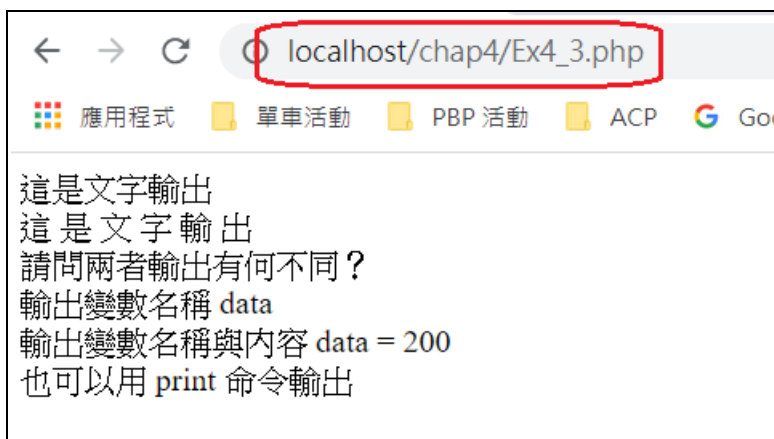
在 PHP 裡，任何資料的輸出，都被視為字串，一般都用 echo 語句來達成。比較特殊的地方是，一般語言輸出一串訊息，大多會用雙引號包起來。在 PHP 的 echo 命令單引號與雙引號同樣功能，但雙引號內不可以再用雙引號，一定要用單引號。也就是說，雙引號內可以再出現單引號。

(A) 輸出格式如下：

字串輸出	說明
echo data;	輸出 data 字串
echo \$data	輸出 data 變數的內容
echo “Good Luck To You”;	輸出一段訊息(Good Luck To You)。
echo “data = \$data”;	輸出 data = 20，假設 \$data=20。
echo ‘data = \$data’;	單引號與雙引號功能相同。
print Hello;	輸出 Hello 字串。
Print “how are you”	輸出一段訊息。

(B) 程式範例(Ex4-3.php)如下：

```
1 <?php
2     echo 這是文字輸出;
3
4     echo "<BR>";
5     echo “這 是 文 字 輸 出<BR>”;
6
7     echo “請問兩者輸出有何不同？<BR>”;
8     $data = 200;
9
10    echo “輸出變數名稱 data <BR>”;
11
12    echo “輸出變數名稱與內容 data = $data <BR>”;
13
14    print “也可以用 print 命令輸出<BR>”;
15 ?>
```

(C) 執行結果：**4-2-4 範例研討：PHP 列印方法****(A) 網頁功能：Ex4_4.php**

吾人希望製作一個網頁來說明 PHP 內文製作方法，譬如：

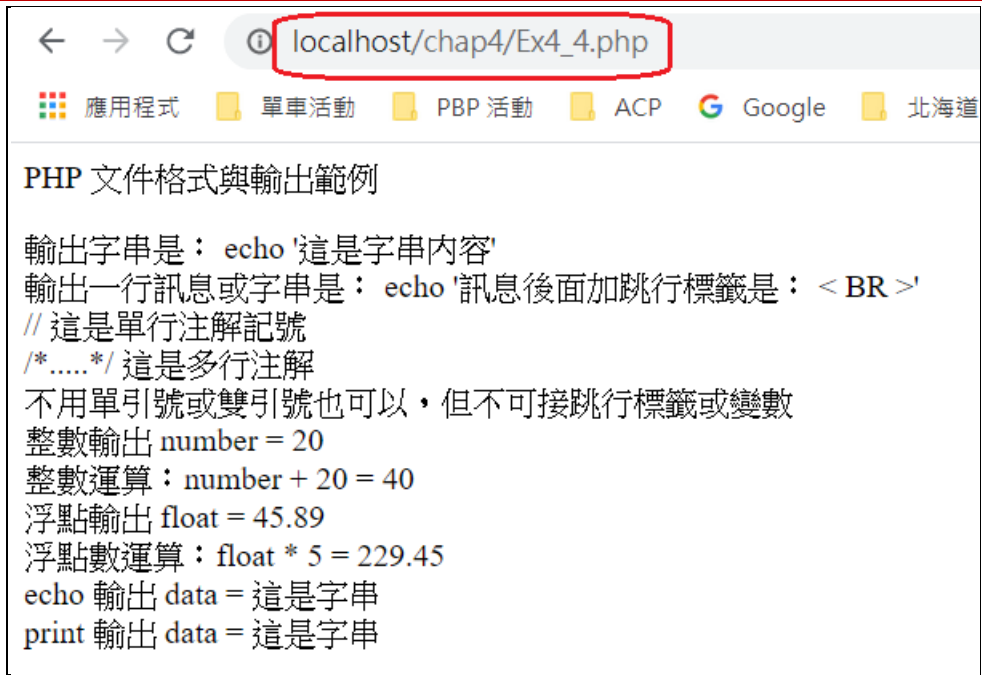
- (1) PHP 輸出命令的使用方法。
- (2) PHP 輸出整行的格式如何。
- (3) 單行注解如何標記。
- (4) 多行注解如何標記
- (5) 整數與浮點數的輸出。

(B) 程式範例：

我們希望製作一個網頁來說明上述製作技巧，如下：

```
1 <p> PHP 文件格式與輸出範例 </p>
2 <?php
3 echo "輸出字串是： echo '這是字串內容' <BR>";
4
5 echo "輸出一行訊息或字串是： echo '訊息後面加跳行標籤是： < BR >' <BR>";
6
7 // 單行注解
8
9 echo "// 這是單行注解記號<BR>";
10
11 /* 多行注解 */
12
13 echo "/*.....*/ 這是多行注解 <BR>";
14
15 echo 不用單引號或雙引號也可以，但不可接跳行標籤或變數;
16
17 echo "<BR>";
18 $number = 20;
19
20 echo "整數輸出 number = $number<BR>";
21
22 $number = $number + 20;
23
24 echo "整數運算： number + 20 = $number <BR>";
25
26 $float = 45.89;
27
28 echo "浮點輸出 float = $float <BR>";
29
30 $float = $float * 5;
31
32 echo "浮點數運算： float * 5 = $float <BR>";
33
34 $data = "這是字串";
35
36 echo "echo 輸出 data = $data <BR>";
37
38 print "print 輸出 data = $data <BR>";
39
40 ?>
```

(B) 執行結果：



4-2-5 範例研討：字串連結

字串連結是 PHP 最佳的應用工具之一，很簡單就是一個逗點記號『.』。功能是可以將多個變數內的字串，連結成一組字串，說明如下：

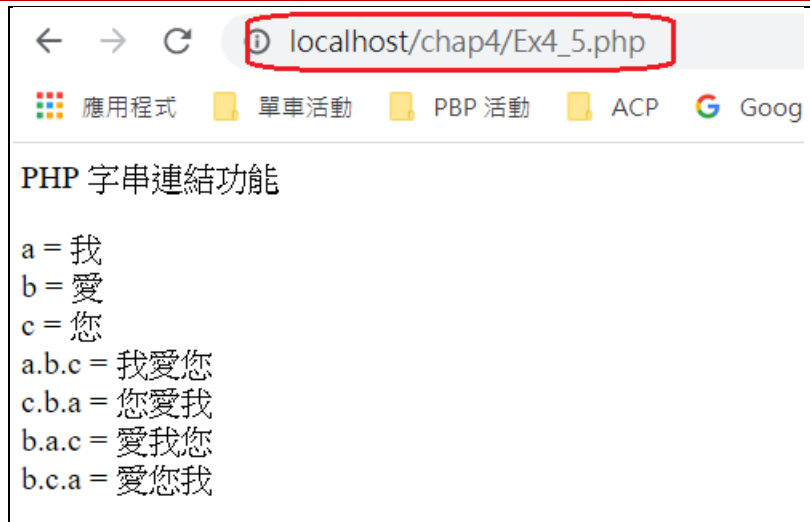
```

$d1 = "大家";           // d1 變數內容放：大家
$d2 = "恭喜";          // d2 內容放：恭喜
$d3 = "新年";
$d4 = "快樂";
$d5 = $d1.$d2           //d5 內容是：大家恭喜
$d6 = $d3.$d4           //d6 內容是：新年快樂
$d7 = $d5." ".$d6       // d7 內容是『大家恭喜 新年快樂』

```

(A) 網頁功能：Ex4_5.php

年輕人喜歡排字遊戲，請建立一個網頁，裡面有 3 個字，再出現三種排列組合結果，如下：

**(B) 程式範例：**

```

1 <html xmlns="http://www.w3.org/1999/xhtml">
2 <body>
3   <p> PHP 字串連結功能 </p>
4   <?php
5     $a = "我";
6     $b = "愛";
7     $c = "您";
8     echo "a = $a <BR>";
9     echo "b = $b <BR>";
10    echo "c = $c <BR>";
11    $string = $a.$b.$c;
12    echo "a.b.c = $string <BR>";
13    $string = $c.$b.$a;
14    echo "c.b.a = $string <BR>";
15    $string = $b.$a.$c;
16    echo "b.a.c = $string <BR>";
17    $string = $b.$c.$a;
18    echo "b.c.a = $string <BR>";
19    ?>
20 </body>
21 </html>

```

4-2-6 範例研討 – 列印字型樣式**(A) HTML 字體設定：Ex4_6.html**

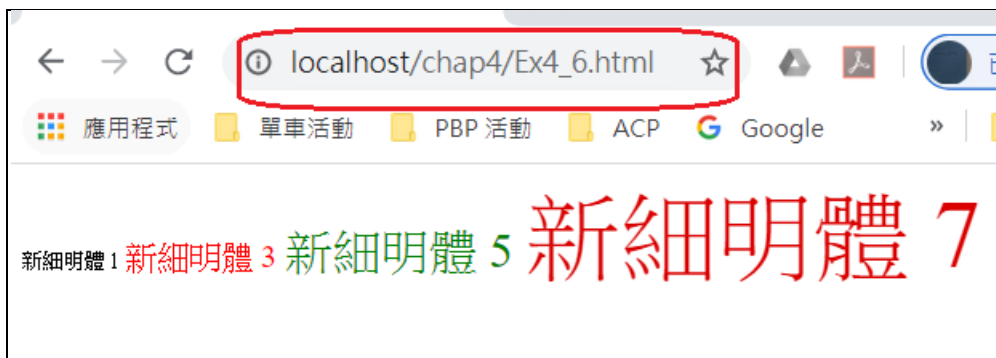
HTML 的字體標示記號是 ` ... `，其中可利用 `face`、`size`、`color` 等變數來指定字形、字體大小與字體顏色，但指定某一變數後，沒有重新變更其內容，則將繼續沿用。譬如第 1 行 `face="新細明體"`，則未更改 `face` 參數之前，它都是新細明體，因此，在 4, 5, 6 行不需要再設定 `face` 參數，只要變更 `size` 和 `color` 參數內容即可。

```

1 <html>
2 <body>
3   <font face='新細明體' size='1' color='black'>新細明體 1 </font>
4   <font size='3' color='red'>新細明體 3 </font>
5   <font size='5' color='green'>新細明體 5 </font>
6   <font size='7' color='dark'>新細明體 7 </font>
7   <BR>
8   </body>
9 </html>

```

(B) HTML 執行結果



(C) PHP 字體設定：Ex4_6.php

吾人將上述 HTML 文件改由 PHP 程式來執行，只要將標示符號以 `echo` 語句輸出即可。希望製作一個網頁來測試各種字型、顏色、大小顯示效果如何，測試規範如下：

- (1) 新細明體、標楷體、微軟正黑體。
- (2) 大小：1、3、5、7。
- (3) 顏色：黑色、紅色、藍色與暗紅色。

PHP 程式如下：

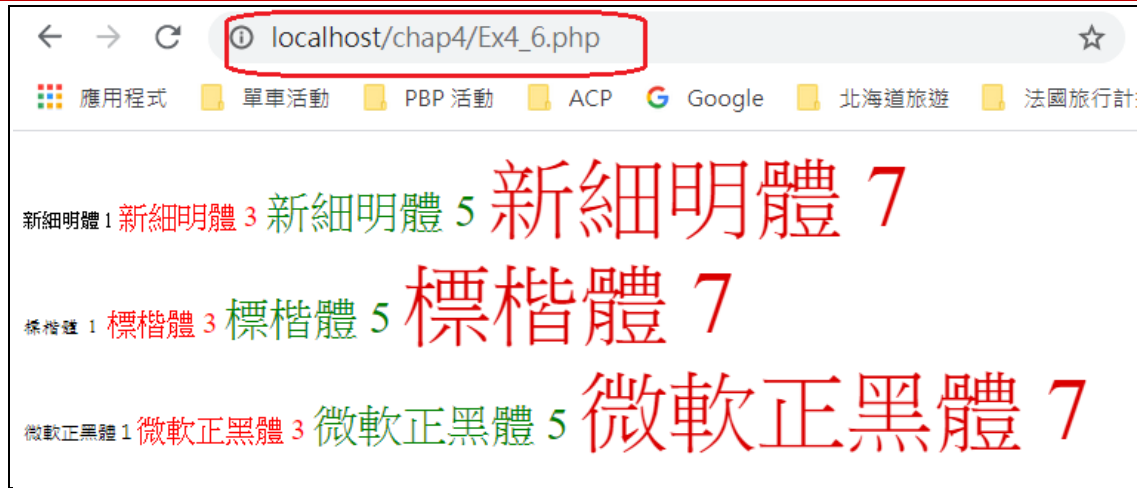
```

1 <?php

```

```
2 // 列印細明體字型
3
4 echo "<font face='新細明體' size='1' color='black'>";
5 echo "新細明體 1 </font>";
6
7 echo "<font size='3' color='red'>";
8 echo "新細明體 3 </font>";
9
10 echo "<font size='5' color='green'>";
11 echo "新細明體 5 </font>";
12
13 echo "<font size='7' color='dark'>";
14 echo "新細明體 7 </font>";
15
16 echo "<BR>";
17
18 // 列印標楷體字型
19
20 echo "<font face='標楷體' size='1' color='black'>";
21 echo "標楷體 1 </font>";
22
23 echo "<font size='3' color='red'>";
24 echo "標楷體 3 </font>";
25
26 echo "<font size='5' color='green'>";
27 echo "標楷體 5 </font>";
28
29 echo "<font size='7' color='dark'>";
30 echo "標楷體 7 </font>";
31
32 echo "<BR>";
33
34 // 列印標楷體字型
35
36 echo "<font face='微軟正黑體' size='1' color='black'>";
37 echo "微軟正黑體 1 </font>";
38
39 echo "<font size='3' color='red'>";
40 echo "微軟正黑體 3 </font>";
41
42 echo "<font size='5' color='green'>";
43 echo "微軟正黑體 5 </font>";
44
45 echo "<font size='7' color='dark'>";
46 echo "微軟正黑體 7 </font>";
47
48 ?>
```

(D) PHP 執行結果：



4-3 PHP 迴圈敘述

4-3-1 PHP 敘述&範例

與一般程式一樣，PHP 迴圈敘述有：

- (1) for 迴圈：如 `for($a=1; $a<=10; $a++) { $sum = $sum + $a; }`。
- (2) while 迴圈：如 `while ($a <=10) { $sum = $sum + $a; }`。
- (3) do/while 迴圈：如 `do { $sum = $sum + $a } while($a <10);`。
- (4) foreach 迴圈：如 `foreach ($arr as $key) { echo "$arr[$key]
"; }` 繼續敘述句。
- (5) continue：迴圈繼續敘述句、break：迴圈中斷敘述句。

(A) 程式範例如下：Ex4_7.php

迴圈最容顯現出 PHP Engine 的功能，如欲利用 HTML 語言顯示十句『我愛您』，則必須連續寫十句，改用 PHP 迴圈來敘述只要寫一句就可以，要顯示 10 次、100 次都可以，範例如下：

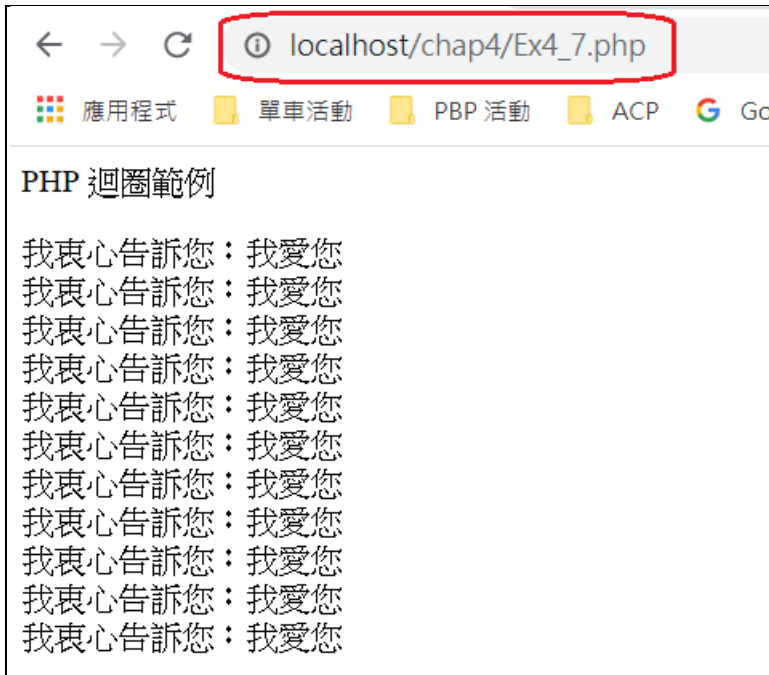
```

1  <?php
2      echo " <p> PHP 迴圈範例 </p>";
3      for($i=0; $i<=10; $i=$i+1) {
4          echo "我衷心告訴您：我愛您 <BR>";
5      }
6
7

```

8	?>
---	----

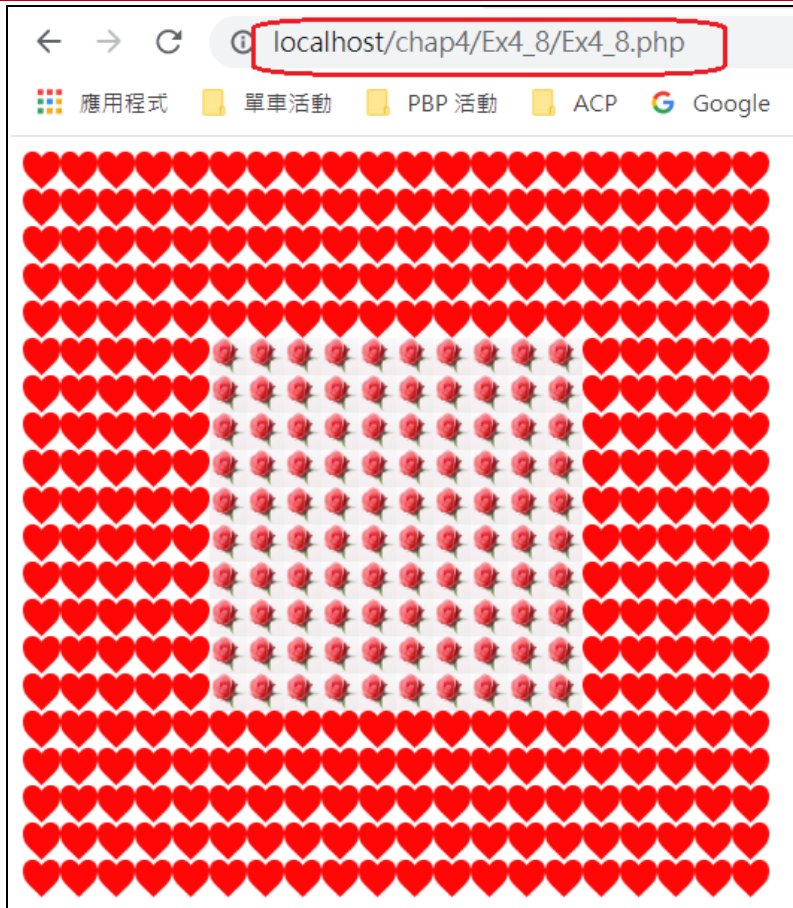
(B) 執行結果如下：



4-3-2 範例研討 - 列印中空矩形

(A) 網頁功能 – Ex4_8.php

公司希望製作一個有愛心的 logo，期望顯示如下：



(B) 製作技巧

這張圖如果僅利用 HTML 來製作可能非常費時，不滿意要更改也很困難，如果用 PHP 就簡單許多了。首先，我們必須到 google 搜尋愛心與玫瑰圖形的素材，並將它下載到範例的資料夾內。這張圖寬與高大略由 20 個愛心排列而成，因電腦是一行接一行輸出，則輸出步驟如下：

- (1) 輸出 5 行，每行 20 個愛心圖。
- (2) 再輸出 5 行，每行先 5 個愛心、再 10 個玫瑰花、最後再 5 個愛心。
- (3) 最後再輸出 5 行，每行 20 個愛心圖。
- (4) 注意每一個圖形輸出都需指定相同大小(`width=20`、`height=20`)

我們將上述的步驟寫成程式即可。

(C) 程式範例：

首先需要網路上下載愛心圖片(`love.png`)與玫瑰花圖片(`flower.png`)，兩者皆要放在與此程式同一目錄下。

```
1 <?php
2     for ($i=1; $i<=5; $i++) {
3         for($j=1; $j<=20; $j++) {
4             echo "<img src='love.png' width='20' height='20'>";
5         }
6         echo "<BR>";
7     }
8     for($i=1; $i<=10; $i++) {
9         for($j=1; $j<=5; $j++){
10            echo "<img src='love.png' width='20' height='20'>";
11        }
12        for($j=1; $j<=10; $j++){
13            echo "<img src='flower.jpg' width='20' height='20'>";
14        }
15        for($j=1; $j<=5; $j++){
16            echo "<img src='love.png' width='20' height='20'>";
17        }
18        echo "<BR>";
19    }
20    for ($i=1; $i<=5; $i++) {
21        for($j=1; $j<=20; $j++) {
22            echo "<img src='love.png' width='20' height='20'>";
23        }
24        echo "<BR>";
25    }
26 ?>
```

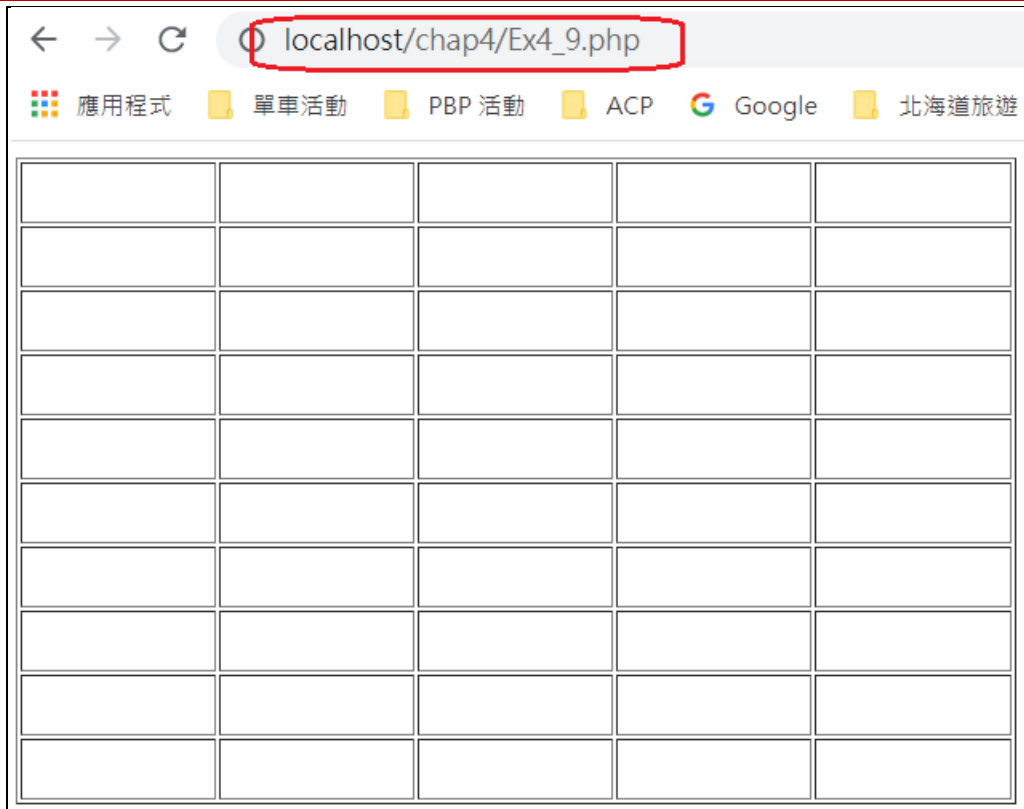
4-3-3 範例研討：製作表格

吾人可以發現利用 HTML 語言建立表格實在很麻煩，需要一行、接著一個欄位順序規劃，如利用 PHP 迴圈可方便許多。

(A) 網頁功能 – Ex4_9.php

公司期望設計一個通用表格讓各單位使用，規格是：

- (1) 10 行、每行 5 個欄位的表格。
- (2) 寬 500 pix、每行高度為 30 pix。



(B) 製作技巧

如僅利用 HTML 來製作這張圖可能滿繁複的，不滿意要修改也很難。如利用 PHP 可就容易許多了。首先，先觀察 HTML 製作表格的方法，如下：

```

<table wide="寬" height="高" border ="線粗">
  <tr> <td> 欄位_1</td><td> 欄位_2</td></tr>      (備注：第一行)
  <tr> <td> 欄位_1</td><td> 欄位_2</td></tr>      (備注：第二行)
  <tr> <td> 欄位_1</td><td> 欄位_2</td></tr>      (備注：第三行)
  ...
</table>
  
```

到底需要幾行(<tr> ... </tr>)、每行需要多少欄位(<td> ... </td>)，這可利用 PHP 迴圈來達成，如此就可以簡化許多，萬一要改變行數或欄位數，只要變更迴圈的參數即可。

(C) 程式範例

```

1  <?php
2      echo " <table width='500' height='120' border='1'>";
3      for ($i=1; $i<=10; $i++) {                // 表格有 10 行
4          echo "<tr>";
5      }
  
```

```
6         for ($j=1; $j<=5; $j++) {           //每行有 5 個欄位
7
8             echo "<td height='30'></td>"; // 欄位高 30 pix
9         }
10        echo "</tr>";
11    }
?>
```

(D) 程式分析：

- (1) 第 5 行到第 11 行 for 迴圈是規劃表格的行數。
- (2) 第 7 行到第 9 行，for 迴圈是規劃每行的欄位數。

4-3-4 範例研討：製作報價單

(A) 網頁要求 – Ex4_10.php

希望將公司報價單格式統一化，製作一個範本公佈在網路上以供同仁參考，期望格式如下圖，請您幫它製作出來。



(B) 製作技巧

首先分析這張報價單的結構，它是由 1 個表頭與 3 個表格所構成，說明如下：

- (1) 表格表頭：表格標頭是(<caption> 很堅固公司報價單 </caption>)。
- (2) 報價單表頭：是由兩行、每行 4 個欄位所構成。第一行第一個欄位是『客戶名稱：』、第二欄位空白、第三欄位是『報價日期：』、第四欄位空白；第二行也是如此。另外，將此表格的線隱藏 (border="0")，就看不到表格線了。
- (3) 報價單內容：計有 11 行、每行 6 個欄位，只有第一行有輸入文字(項目、品名、)、其他 10 行都是空白可利用 PHP 的迴圈製作。另外，須規範每行高度、與各個欄位的寬度。
- (4) 報價單結尾：只有一行、2 個欄位，第 1 欄位內容是『總計：』，第 2 欄位是空白，高度與欄位寬度自行調整。

(5) HTML 與 PHP 語言混合：此範例吾人特地將 HTML 與 PHP 語言混合使用，給讀者了解一般網頁製作技巧大多如此。

(C) 程式範例：

```

1 <html xmlns="http://www.w3.org/1999/xhtml">
2 <body>
3 <table width="1000" height="100" border="0">
4   <caption>
5     <font size="6"><u><p>很堅固工程公司 報價單</p></u></font>
6   </caption>
7   <tr height="50" align="left">
8     <td>客戶名稱：</td><td width="400"></td>
9     <td>報價日期：</td><td width="400"></td>
10  </tr>
11  <tr height="50" align="left">
12    <td>客戶地址：</td><td width="400"></td>
13    <td>客戶電話：</td><td width="400"></td>
14  </tr>
15 </table>
16 <table width="1000" border="1">
17   <tr height="50" align="center">
18     <td width="50">項目</td>
19     <td width="200">品 名</td>
20     <td width="450">規 格</td>
21     <td width="100">數 量</td>
22     <td width="100">單 價</td>
23     <td width="100">小 計</td>
24   </tr>
25   <?php
26     for ($i=1; $i<=10; $i++) {
27       echo "<tr height='50'>";
28       for ($j=1; $j<=6; $j++) {
29         echo "<td></td>";
30       }
31       echo "</tr>";
32     }
33   ?>
34 </table>
35 <table width="1000" border="1">
36   <tr height="50" align="left">

```

```
43      <td width="200">總計</td>
        <td width="800"></td>
    </tr>
</font>
</table>
</body>
</html>
```

4-4 PHP 陣列

4-4-1 PHP 陣列相關敘述

(A) 一維陣列宣告

一維陣列的宣告語法如下：

```
$name[] = "張大名"; //name[0] 內容是張大名
$name[] = "劉友德"; //name[1] 內容是劉友德
$data = array(90, 80, 40); //data[0]=90、data[1]=80、data[2]=40
```

(B) 二維陣列宣告

二維陣列的語法如下：

```
$score = array(
    array(80, 90, 70, 78, 97),
    array(90, 56, 89, 76, 99),
    array(89, 76, 78, 72, 91));
或
$score1 = array(80, 90, 70, 78, 97);
$score2 = array(90, 56, 89, 76, 99);
$score3 = array(89, 76, 78, 72, 91);
$score = array($score1, $score2, $score3);
```

(C) 陣列迴圈

PHP 有專屬處理迴圈敘述句：foreach，其語法如下：

```
$score = array(80, 90, 70, 78, 97); //一維陣列範例
```

```

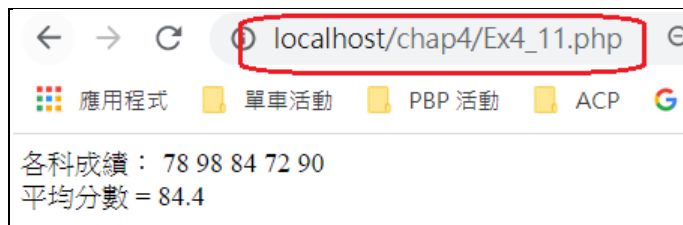
$sum = 0;
foreach($score as $element){
    $sum = $sum + $element;    //sum = 80+90+70+78+97
}

```

4-4-2 範例研討：計算單筆成績

(A) 網頁功能 – Ex4_11.php

我們用一個陣列 `$score = array(78, 98, 84, 72, 90)` 存放學生的資料，順序是：程式、計概、網頁、資料庫、系統分析。請製作一網頁可顯示各科成績，以及平均分數，格式如下：



(B) 製作技巧

我們利用 `$sum` 變數作為累加器，累加各科成績，再除以 5 就是平均分數。

(C) 程式範例

```

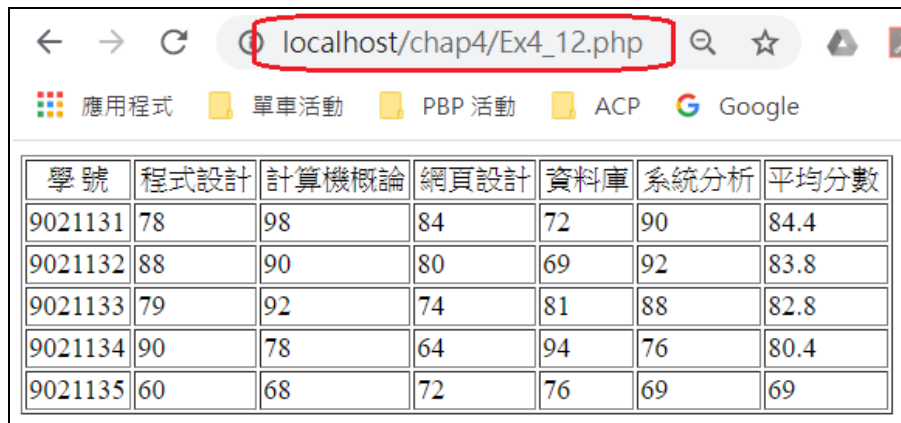
1  <?php
2      $score = array(78, 98, 84, 72, 90);
3      $sum = 0;
4      echo "各科成績：\t";
5      foreach($score as $item){
6          $sum = $sum + $item;
7          echo "$item \t";
8      }
9      echo "<BR>";
10     $ave = $sum / 5;
11     echo "平均分數 = $ave";
12
13  ?>

```

4-4-3 範例研討：列印班級成績單

(A) 網頁功能 – Ex4_12.php

張老師希望有一套工具可以列印全班同學各科成績，以及其平均分數，格式如下圖，請您幫她設計出來，並使它可以顯示在網頁上。



學號	程式設計	計算機概論	網頁設計	資料庫	系統分析	平均分數
9021131	78	98	84	72	90	84.4
9021132	88	90	80	69	92	83.8
9021133	79	92	74	81	88	82.8
9021134	90	78	64	94	76	80.4
9021135	60	68	72	76	69	69

(B) 製作技巧

吾人將這張圖歸納如下：

- (1) 有 5 位同學，每筆資料內記錄學號、程式設計、計算機概論、資料庫、系統分析等資料。因此需要一個 $5 * 6$ 的二維陣列來存放。我們用 `$score1 ~ $score5` 等 5 個一維陣列，分別存放每位學生的資料。再將這 5 個陣列結合成一個二維陣列，並命名為 `score`。
- (2) 利用一個陣列存放表格的欄位名稱，命名為 `name`。
- (3) 表格需要 6 行、每行 7 個欄位，前 6 個欄位是存放每位同學的學號與各科成績，最後欄位是平均分數。利用 PHP 繪製表格，填入資料時，順便計算它的平均值，將它填入最後欄位。

(C) 程式範例

```

1  <?php
2      // 一維陣列宣告
3
4      $name = array("學 號", "程式設計", "計算機概論", "網頁設計", "資料庫", "系統
5  分析");
6
7      $score1 = array(9021131, 78, 98, 84, 72, 90);
8      $score2 = array(9021132, 88, 90, 80, 69, 92);
9      $score3 = array(9021133, 79, 92, 74, 81, 88);
10     $score4 = array(9021134, 90, 78, 64, 94, 76);
11     $score5 = array(9021135, 60, 68, 72, 76, 69);
12     // 二維陣列宣告
13     $score = array($score1,$score2,$score3,$score4,$score5);

```

```
14 // 宣告表格寬度與線條大小
15 echo "<table width='500' border='1'>";
16 // 產生表格標題
17 echo "<tr>";
18 for($i=0; $i<6; $i++){
19     echo "<td align='center'>$name[$i]</td>";
20 }
21 echo "<td> 平均分數</td>";
22 echo "</tr>";
23 // 產生表格內容
24 for($i=0; $i<5; $i++) {
25     echo "<tr>";
26     $sum = 0;
27     $item = $score[$i][0];
28     echo "<td> $item </td>";
29     for($j=1; $j<=5; $j++){
30         $sum = $sum + $score[$i][$j];
31         $item = $score[$i][$j];
32         echo "<td> $item </td>";
33     }
34     $ave = $sum/5;
35     echo "<td>$ave</td>";
36     echo "</tr>";
37 }
38 // 表格結束
39 echo "</table>";
40
41 ?>
42
```

4-4-4 自我挑戰：顯示及格成績單

(A) 網頁功能 – Ex4_13.php

張老師對 Ex4_12 的成績單並不滿意，希望成績不及格(低於 60 分)能夠顯示紅色數字，這樣對學生比較有警惕的功能，期望如下圖這樣。

學號	程式設計	計算機概論	網頁設計	資料庫	系統分析	平均分數
9021131	78	98	45	72	90	76.6
9021132	88	90	80	40	92	78
9021133	45	92	74	50	45	61.2
9021134	90	78	64	94	76	80.4
9021135	60	68	42	50	56	55.2

(B) 製作技巧

很簡單，我們修改 Ex4_12.php 程式，將各項成績填入欄位之前，先判斷其內容是否低於 60，如是的話，則將字型顏色設定為 red (紅色)。

(C) 程式提示

```

1      .....
2          $item = $score[$i][$j];
3          $sum = $sum + $item;
4          if ($item >= 60)
5              echo "<td> $item </td>";
6          else //成績不及格顯示紅色
7              echo "<td><font color='red'>$item</font></td>";
8          }
9      .....
    
```

4-4-5 範例研討：列印月曆

(A) 網頁功能 – Ex4_14.php

我們期望在網頁上顯示月曆，先做一個起頭的練習，假設該月的第一天是星期三，而那個月有 31 天，期望印出來如下圖所示。

localhost/chap4/Ex4_14.php

應用程式 單車活動 PBP 活動 ACP Go

假設當月第一天是星期三，該月有 31 天

星期一	星期二	星期三	星期四	星期五	星期六	星期日
		1	2	3	4	5
6	7	8	9	10	11	12
13	14	15	16	17	18	19
20	21	22	23	24	25	26
27	28	29	30	31		

(B)製作技巧

吾人將列印的關鍵性技巧歸納如下：

- (1) 排列方式：它是新型格式，由星期一排到星期日。
- (2) 當月的第一天：如果是星期三，則用 3 來取代，第一行須先入 2 個(3-1)空白欄位，第 3 欄位才由 1 開始排。
- (3) 每次排 7 個欄位後須再開下一行(<tr>)。
- (4) 所有日期(假設 31 天)排列完後，看排到星期幾(假設星期五，5)，則需在排兩個空白欄位。

(C)程式範例：

```

1  <?php
2      echo "<p> 假設當月第一天是星期三，該月有 31 天 </p>";
3
4      $week = array("星期一","星期二","星期三","星期四",
5                  "星期五","星期六","星期日");
6
7      $first=3;      // 第一天是星期三
8      $number=31;   // 這個月有 31 天
9
10     echo "<table width='400' border='1'>";
11     echo "<tr>";
12     for($i=0; $i<7; $i++){          // 列印星期一 ~ 日
13         echo "<td align='center'>$week[$i]</td>";
14     }
15     echo "</tr>";
16     echo "<tr>";
17
18     $now = 0;          // 目前印到星期幾

```

```

19         for($i=1; $i<$first; $i++){           // 列印第一週前面空白
20             echo "<td>    </td>";
21             $now = $now + 1;                 //目前印到星期幾
22         }
23     $day = 1;
24     while ($day <= 31) {                     //從 1 日開始輸出
25         echo "<td> $day </td>";
26         $now = $now + 1;
27         if (($now % 7) == 0){               // 是否已到星期日
28             echo "</tr>";                     // 如果是這行結束再開新行
29             echo "<tr>";
30             $now = 0;                       //又歸 0 開始計算
31         }
32         $day = $day + 1;
33     }                                       // 所有日數填寫完
34     while ($now <7){                         //印到星期幾，後面印空格
35         echo "<td>    </td>";
36         $now = $now + 1;
37     }
38     echo "</tr>";
39     echo "</table>";
40     ?>
41
42
43
44
45

```

4-4-6 PHP 陣列迴圈 -foreach

(A) foreach 迴圈語法

PHP 陣列宣告時，並非陣列元素一定是 0, 1, 2, 3,4, ...，整數排列，陣列元素單元可以任意取名或任意數值排列，如此的話，一般程式語言內的 for、while、do-while 迴圈便無法執行，因此有一專屬迴圈 - foreach，範例如下：

```

1  <?php
2  $arr = array(41, 2, 3, 17);
3  foreach ($arr as $value) {
4      echo "$value <BR>";
5  }
6  foreach ($arr as $key => $value) {

```

```

7      each "arr[$key] = $value <BR>";
8      }
9  ?>

```

執行結果如下：

```

1
2
3
17
arr[0] = 1
arr[1] = 2
arr[2] = 3
arr[3] = 17

```

語法說明：

1. 第 3 行：`foreach($arr as $value) {...}`。迴圈自動由陣列 \$arr 的第 0 元素取到最後元素的內容 \$value。
2. 第 6 行：`foreach($arr as $key => $value) { ...}`。迴圈自動陣列的 \$arr[\$key] 元素取出內容 \$value，到最後元素。

(B) PHP 指定元素宣告

PHP 陣列元素可以任意，如下：

```
arr['sunday'] = '星期日'、arr['monday'] = '星期一'、arr['tuesday'] = '星期二'、...。
```

宣告語法如下：

```
$arr = array('sunday' => '星期日', 'monday' => '星期一', 'tuesday' => '星期二', ...);
```

(C) 程式範例：Ex4-14-1.php

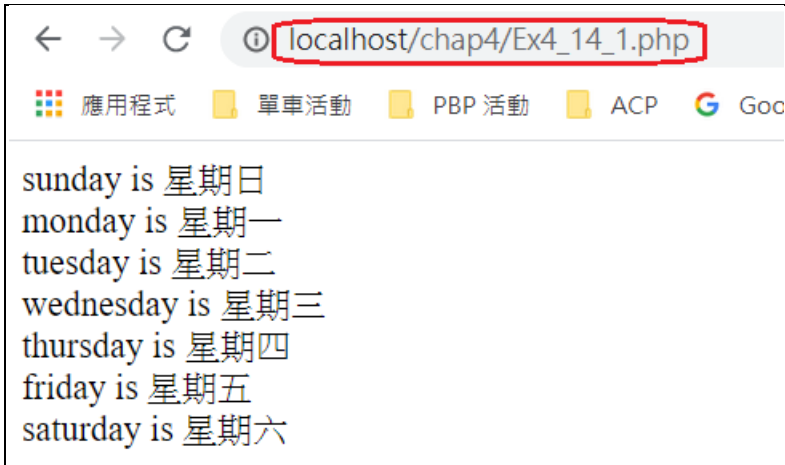
```

1  <?php
2      $arr = array ('sunday'=>'星期日', 'monday'=>'星期一',
3
4          'tuesday'=>'星期二', 'wednesday'=>'星期三',
5
6          'thursday'=>'星期四', 'friday'=>'星期五',

```

```
6         'saturday'=>'星期六');
7     foreach ($arr as $key => $value) {
8         echo "$key is $value <BR>";
9     }
10    ?>
```

執行結果：



4-5 Cookie 與 Session

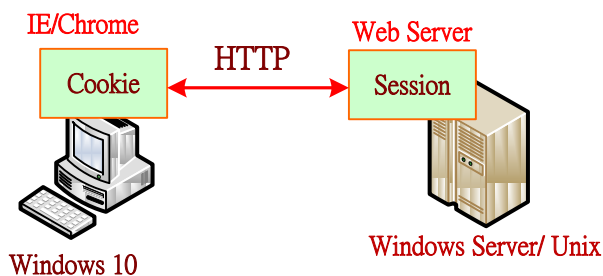
4-5-1 Cookie 與 Session 區別

瀏覽器與網頁伺服器之間是以『要求/回應』(Request/Response)方式操作，當客戶端瀏覽器有需求時，提出網頁要求(Request)，則伺服器端回應(Response)網頁給客戶端。基本上，每一個要求/回應動作都是獨立性的，完成之後不會保留任何訊息，又稱為『HTTP 無狀態』(HTTP stateless)。譬如，瀏覽器執行會員登入網頁時，要求使用者登入帳號/密碼，並經由伺服器端確認，登入網頁結束之後，不再保留使用者輸入訊息，下次再要求登入時，必須重新輸入帳號/密碼。

許多情況需要保留一些通用訊息，譬如登入帳號、密碼、IP 位址、計數器等等。而且訊息大多以文字格式存取。以確保不同網頁之間可以共同存取。但這些訊息要存在甚麼地方呢？如儲存於瀏覽器(客戶端硬碟)則稱為『Cookie』，如儲存於伺服器端(Apache)則稱為『Session』，兩者差異如下：

- (1) Cookie 與 Session 都是變數型態，以供儲存網頁存取訊息使用。
- (2) Cookie 變數儲存於瀏覽器(客戶端)，Session 變數儲存於網頁伺服器(伺服器端)。

- (3) Cookie 變數存活時間大多是宣告時設定，超過時間或刻意刪除才會被消除。Session 變數大多依照『會議連線』(Session)存活時間而定。當產生一條 Session 連線時，伺服器會給予一個 Session ID (識別碼)，之間也許會有多個不同網頁的存取，所產生的 Session 變數也都共同使用，當 Session 連線結束時，所有該 Session ID 下的變數也隨之被刪除。
- (4) 伺服器端大多會同時處理多個客戶端要求，伺服器端利用 Session ID 來辨識哪一個客戶端，不至於產生混擾。
- (5) Cookie 變數儲存於瀏覽器上，一般 JavaScript、Jquery、等前端發展工具都可以存取，因此安全較低。Session 變數儲存於伺服器端，一般較難存取到，相對的安全性較高。



4-5-2 Cookie 宣告與讀取

產生 Cookie 變數很簡單，它只有一個 PHP 函數，名為：

```
setcookie(name, value, expire, path, domain, secure)
```

呼叫函數執行後，回應 "True" 表示成功，否則 "False" 表示失敗，個個參數說明如下：

參數名稱	說明
name	必須，Cookie 變數名稱。
value	必須，Cookie 變數的內容(字串格式)。
expire	選項，Cookie 變數的有效期。
path	選項，Cookie 變數儲存路徑。
domain	選項，Cookie 共用網域。
secure	選項，可否通過 HTTPS 連線傳輸 cookie。

宣告產生後，Cookie 變數名稱為：『`$_COOKIE["name"]`』，變數型態又分為兩種：

1. 單一變數：如同一般變數宣告方法，如：`setcookie("var1", "300");`，則表示宣告一個名為 `var1` 的 Cookie 變數，其內容為 300。如下：
 - 宣告：`setcookie("var1", "300");`
 - 讀取：`echo $_COOKIE["var1"]; => 300`
2. 陣列變數：如同一般陣列宣告方法，如：`setcookie("arry1[three]", "good")`，其中變數名稱 (`arry1`) 與 元素名稱 (`three`) 可以任意指定，並沒有先後順序關係，如下：
 - 宣告：`setcookie("arry1[one]", "Luck");`
 - 讀取：`echo $_COOKIE["arry1"]["one"]; => Luck`

4-5-3 Cookie 簡單操作

(A) 程式範例：Ex4_15.php

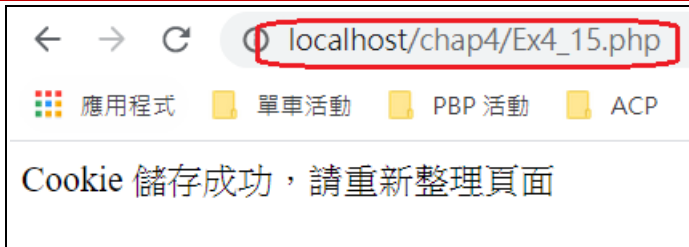
吾人用一個簡單範例來說明 Cookie 操作方法。如下：

```
1  <?php
2      $setVar = setcookie("var1", "php-cookie"); // 宣告 cookie 變數
3
4      if ($setVar) {                               // 是否宣告成功
5          if (isset($_COOKIE["var1"]))           // 是否以設定內容
6              echo "Cookie 變數 var1= ".$_COOKIE["var1"]."<br>";
7          else
8              echo "Cookie 儲存成功，請重新整理頁面 <BR>";
9      }
10
11  ?>
12
```

程式說明：

- 2：利用 `setcookie()` 函數宣告 Cookie 變數，成功傳回 "True"，否則 "False"。
- 5：測試變數內容是否設定。
- 第一次執行時，雖然已產生成功，但未擷取到內容，網頁重新整理後，即可取得內容。

(B) 執行結果：

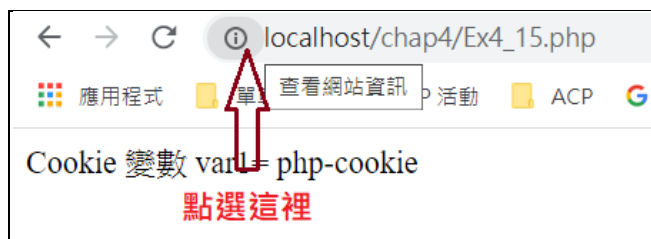


重新整理頁面後，出現：



(C) 由網頁查詢 Cookie 變數

1. 點選網址前位置，如下：



2. 可以找到該網頁的 Cookie，並且可以將它刪除。



4-5-4 Cookie 陣列操作

(A) 程式範例：Ex4_16.php

吾人用一個簡單範例來說明 Cookie 陣列操作方法。如下：

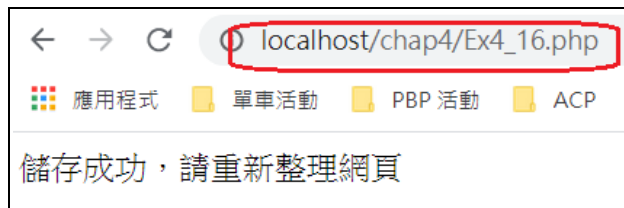
```

1  <?php
2      // 宣告 Cookie 陣列
3      setcookie("var1[1]", "Cookie-1");
4      setcookie("var1[one]", "Cookie-one");
5      setcookie("var1[three]", "Cookie-three");
6
7      // 測試 Cookie 變數是否存在
8      if (isset($_COOKIE["var1"])) {
9          foreach($_COOKIE["var1"] as $name => $value)
10             echo "$name : $value <br>";
11     }else {
12         echo "儲存成功，請重新整理網頁 <br>";
13     }
14 ?>

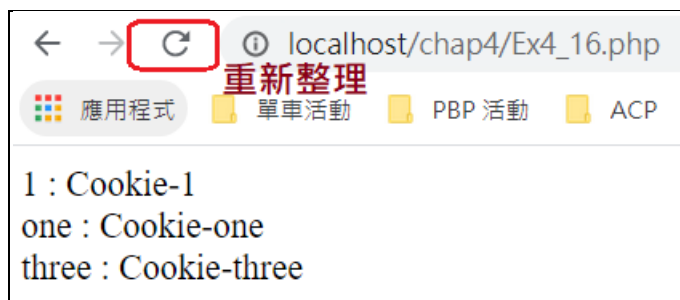
```

1. 程式說明：

- 3~5: 利用 `setcookie()` 產生 3 個陣列 Cookie 變數, 名稱為 `var1[1]`、`var1[one]`、`var1[three]`, 其中元素可以任意指定。
 - 8: 測試變數是否設定成功。
 - 9~10: 利用 `foreach()` 迴圈列印出陣列內容。
2. 第一次執行結果: 如沒出現此畫面, 請由網頁刪除 Cookie 再重新執行就會出現。



3. 重新整理後結果:



4-5-5 Cookie 有效時間

宣告產生 Cookie 變數時沒有指定它的有效時間, 則該變數可以存留多久? 大多依照瀏覽器設定, 或者關閉瀏覽器就自動刪除, 但還是要指定存活時間較安全。存活時間大多以『秒』為單位計算, 宣告格式如下:

```
setcookie(name, value, expire)
```

譬如: 希望它存活時間為 1 小時 (3600 秒), 則存活時間為 `expire = time()+3600`, 其中 `time()` 為產生時的時間, 再加上 3600 秒(1 小時)。

(A) 程式範例: Ex4_17.php

吾人希望網頁上記錄『本人』當天瀏覽次數, 但過了今日該紀錄就歸零重新計算。如下:

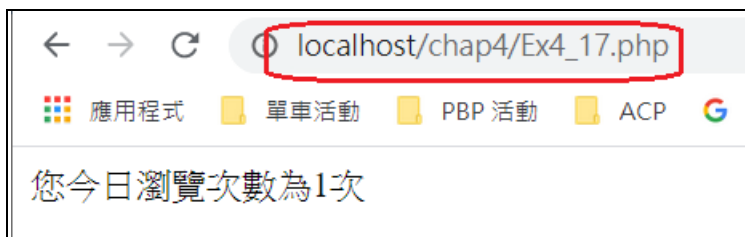
```
1 <?php
2     $Today = date("y-m-d 23:59:59");
3     echo "計數器截止時間是".$Today."<br>";
```

```
4     if (isset($_COOKIE["counter"])) {
5         $count = $_COOKIE["counter"];
6         $count = $count + 1;
7         setcookie("counter", $count, strtotime($Today));
8         echo "您今日瀏覽次數為".$count."次 <br>";
9     } else {
10        setcookie("counter", "0", strtotime($Today));
11        header("Location: Ex4_17.php");
12    }
13    ?>
```

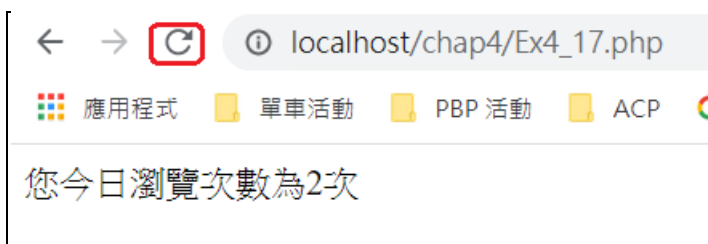
1. 程式說明：

- 2：利用 `date("y-m-d 23:59:59")` 函數設定截止時間。
- 7：利用 `strtotime($Today)` 將其轉換成 Unix 時間格式。
- 11：利用 `header("Location: Ex4_17.php")`函數傳回客戶端網頁位址，再重新整理一次。

2. 第一次執行結果：如沒出現此畫面，請由網頁刪除 Cookie 再重新執行就會出現。



3. 重新整理後：(或重新開啟瀏覽器，再瀏覽該網址)



4-5-6 Session 宣告與讀取

Session 變數記錄著使用者與網站之間『會議連結』(Session)，可能會重複使用的訊息，譬如：登入帳號/密碼、登入 IP 位址、或在電商平台紀錄著客戶訂購貨品清單(購物車)、等等。有了 Session 變數可以減少許多客戶重複輸入的次數，但當『會議連結』終止時，這些變數也隨之消失。

基本上，Session 變數是紀錄於伺服器端 (Apache Server)，每一個客戶的『會議連結』都有一個唯一識別的 Session ID。雖然伺服器端同時處理多條客戶連線，但都有唯一的 Session ID，之間的 Session 變數也不會產生混擾。

常用的 Session 相關函數有：

函數名稱	說明
session_start()	宣告 session 開始並產生 session ID，此函數之前不可有任何輸出。
session_id()	取得 session ID 識別值。
session_unset()	釋放所有 session 變數。
session_destroy()	註銷 session ID，刪除伺服器上所有 session 變數。
Session 變數	\$_SESSION["變數名稱"]

(A) Session 變數的宣告與讀取

說明如下：

- **啟動連線**：session_start();
- **變數名稱**：\$_SESSION["變數名稱"]，如 \$_SESSION["name"] = "tsnien";
- **陣列名稱**：\$_SESSION["陣列名稱"]["元素名稱"]，如：
 - \$_SESSION ["teachers"]["name"] = "粘添壽";
 - \$_SESSION ["teachers"]["no"] = "235";
- **讀取變數**：echo \$_SESSION["變數名稱"]。
- **刪除變數**：unset(\$_SESSION["變數名稱"])，如 unset(\$_SESSION["name"]);
- **刪除所有變數**：session_unset();

(B) Session 的有效期限

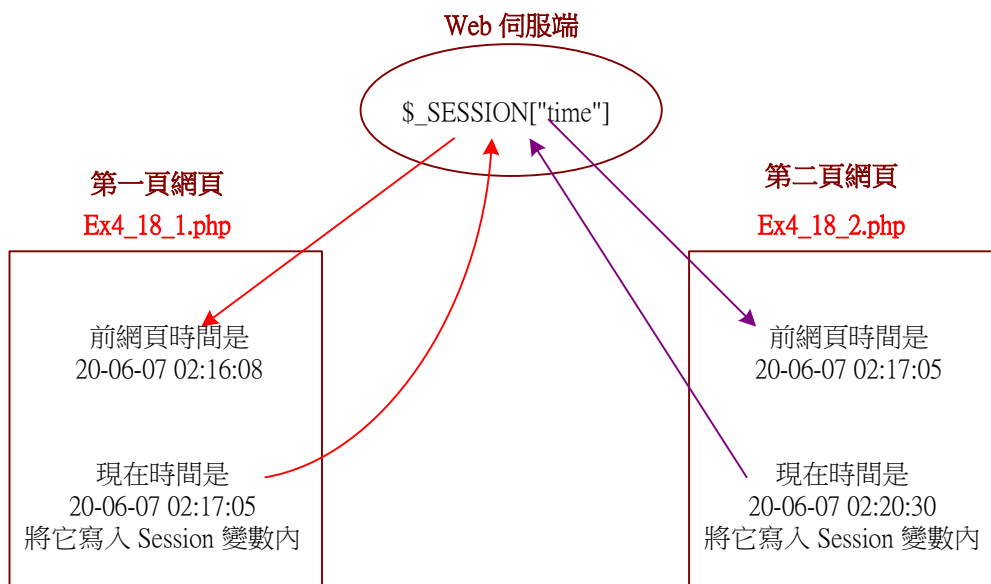
發生以下四種情況之一時，Session 與其中資料會消失：

- 客戶端關閉目前使用的瀏覽器。
- 關閉網頁伺服器。
- 客戶端超過預設時間未向伺服器提出請求，一般預設時間為 30 分鐘。
- 所執行程式結束 Session。

4-5-7 範例研討：Session 變數共用

(A) 程式範例：Ex4_18.php

接下用一個簡單範例來說明 Session 連結的啟動與操作方法。為了驗證 Session 變數儲存於伺服器端，吾人用兩個網頁 (Ex4_18_1.php 與 Ex4_18_2.php) 共同存取 Session 變數，如下：



(B) 第一頁網頁：Ex4_18_1.php

程式功能是：進入網頁時立即啟動 Session，接著測試 \$_SESSION["time"] 變數是否以產生，如果以產生，則讀取並顯示『前網頁的時間』，再產生目前時間 (\$dateTime)，並將其寫入 \$_SESSION["time"] 變數內。最後請求連結下一個網頁，重點說明如下。

- 1：啟動 Session 連結：session_start()。
- 9：讀取 Session 變數並將轉換成文字化格式 (\$timeFormat)。
- 17：將目前時間寫入 Session 變數內。

```
1 <?php session_start(); ?>
```

```
2 <html><head>
3 <title> 第一頁網頁 </title></head>
4 <body>
5 <?php
6     echo "第一頁網頁 <BR>";
7
8     if (isset($_SESSION["time"])) {
9         $timeFormat = $_SESSION["time"] -> format ("y-m-d h:i:s");
10        echo "前網頁時間是".$timeFormat." <br>";
11    }
12    $dateTime = new DateTime("now");
13    $timeFormat = $dateTime -> format ("y-m-d h:i:s");
14    echo "現在的時間是".$timeFormat." · 並將它寫入 Session 變數內 <br>";
15    $_SESSION["time"] = $dateTime;
16
17    echo "<a href='Ex4_18_2.php'> 回第二頁 </a> <br>";
18 ?>
19 </body>
20 </html>
21
```

(C) 第二頁網頁：Ex4_18_2.php

程式功能幾乎與第一頁網頁 (Ex4_18_1.php) 相同，不再另外說明。

```
1 <?php session_start(); ?>
2 <html><head>
3 <title> 第二頁網頁 </title></head>
4 <body>
5 <?php
6     echo "第二頁網頁 <BR>";
7
8     if (isset($_SESSION["time"])) {
9         $timeFormat = $_SESSION["time"] -> format ("y-m-d h:i:s");
10        echo "前網頁時間是".$timeFormat." <br>";
11    }
12    $dateTime = new DateTime("now");
13    $timeFormat = $dateTime -> format ("y-m-d h:i:s");
14    echo "現在的時間是".$timeFormat." · 並將它寫入 Session 變數內 <br>";
15    $_SESSION["time"] = $dateTime;
16
17    echo "<a href='Ex4_18_1.php'> 回第一頁 </a> <br>";
18 ?>
19 </body>
20
```

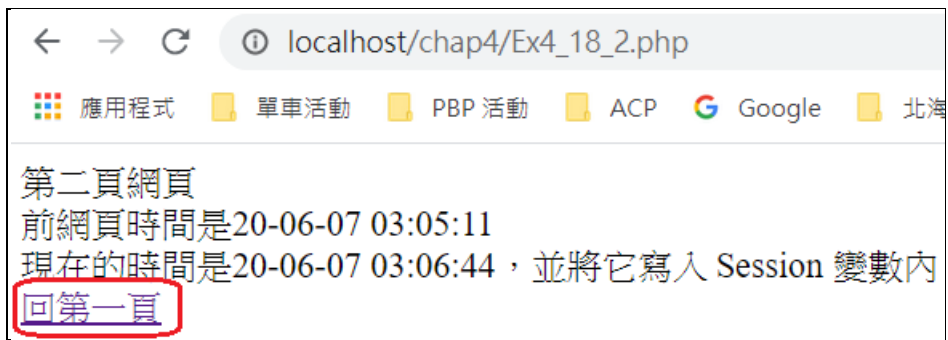
```
21 </html>
```

(D) 執行結果

- 執行第一頁網頁：



- 執行第二頁網頁



- 查閱網頁儲存變數：吾人可發現在網頁上找不到該變數，它是儲存在伺服器端。

