PHP程式基本概念

在網頁中要加入PHP程式碼，有4種方式，如下:

請盡量使用第一種方法就好其他的很常會有問題。

1. <? Php

…..(PHP程式碼)

?>

1. <?

PHP

…..(PHP程式碼)

?>

1. <?

…..(PHP程式碼)

?>

4. <script language=”PHP”>

…..(PHP程式碼)

</script>

但是有時因為版本的因素，除了第1種外，並不是全部都能通行，所以基本上還是建議直接採用第1種型態。

還記得前面提到的「程式運作的基本流程」嗎?三階段處理流程：

　　　　輸入　→　處理　→　輸出

　　我們先介紹「輸出」最常用(95%以上)的指令：

「echo」：在網頁顯示資料的指令

格式：

數字資料

"文字資料"(註:html語法以本型態執行)

echo 日期資料

變數資料

運算式

各種資料型態，我們在後續各節(或單元)再陸續介紹，下面我們先以例子導入概略的觀念。

例：ch08-01.php (記得存在 c:/xampp/htdocs/110-all/ 資料夾中)

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>echo指令的應用</title>

</head>

<body>

<?php

echo " echo指令的應用";

echo "<hr>";

echo "也可以包括<font color='#000ff'> HTML </font>標籤";

?>

</body>

</html>

註：1.記得在瀏覽器中的網址列要輸入 「<http://localhost/>程式名稱(http://localhost/classlevel01/level01php.php)」，才能觀看  
程式執行結果。

2.HTML的標籤指令在PHP中，被視為「"文字資料"」，且在瀏覽器上，也  
 是依HTML的功能定義來顯示。

3.php每一個指令必須以「；」做結束。

※HTML與PHP的連結運用

1.在HTML中，以<?php (PHP程式碼) ?> 置入PHP程式

2.在PHP程式的 echo 指令中，以 "文字資料"(文字型態) 置入HTML程  
 式碼，例：echo "也可以包括<font color='#000ff'> HTML </font>標籤";

在程式碼中加入註解

HTML程式的註解指令<!—註解文字 -->，我們可能比較少用，但在PHP程式中，善用註解指令是非常重要，因為在做程式的維護/修改階段，他能提供很大的幫助。他是程式設計SOP的一項重要指標。

註解指令格式有二種，如下：

格式1：(單行註解)

// 註解文字

格式2：(多行註解)

/\*

註解文字

\*/

例：ch08-02.php (應用HTML和PHP的註解指令)

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>註解指令的應用</title>

</head>

<body bgcolor="#FFCCFF">

<!-- HTML註解指令 -->

<h1>網頁內容</h1>

<?php

// PHP單行註解程式碼

echo "<h2>這是顯示的文字</h2>";

/\*

PHP多行註解程式碼

\*/

echo "PHP網頁以 Dreamweaver 來編輯很方便!";

?>

</body>

</html>

PHP連續多行資料列印(輸出／顯示)指令

　　當在程式中有連續多行的資料需要列印時，如果每一行都要先寫「echo」，也許會讓人覺得囉嗦，因此PHP提供[echo<<<end]….[end;]指令，格式如下：

格式：(列印多行資料)

echo <<< end

要顯示的資料

End;

例:ch08-03.php

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>輸出指令的應用</title>

</head>

<body>

<?php

$sna="PHP程式班";

//輸出變數$sna所儲存的資料並指定以H2標籤字顯示

echo "<h2>".$sna."</h2>";

echo "60人"; // 也可用 echo (60)."人";

echo $sna."<br>"; //輸出變數值並且加入換行標籤

print ("<font color=red>也可以print來輸出資料</font><p>");

echo <<< end

顯示相關資料 <br>

show multiline data

end; //end; 一定要從第1 欄開始

echo<<<  
end 中 可以使用定義後的變數與常數，但她不能夠做運算喔。

?>

</body>

</html>

註：1. 程式中有一個 print指令，用法和 echo指令一樣，格式如下：

格式：(應用print指令)

Print（”輸出資料”）;

或 Print ”輸出資料” ;

要用 echo 或 print就看個人喜好。

2. 程式中 $sna 是一個 字串變數名稱，在後面再做解說。

3. //也可用 echo (60)."人";中的「.」是連續二組列印資料的連接符號。

PHP基本語法

**變數：**

在上例中有一個指令「$sna="PHP程式班";」，另外在前一單元中三角形面積計算程式也有三個指令「$hi=3;」、「$low=3;」及「$area=$hi\*$low/2;」，其中的「$sna、$hi、$low及$area」我們稱為「變數」。

「變數」是輸入階段所必需應用的一個重要元素。我們所輸入的資料必須存放在電腦的記憶體中，但是這個資料是存放在那一個位置(址)呢?要如何才能使用到這個位置(址)的資料呢?

因此，記憶體中每一個存放資料的位置都要有一個對應的「門牌號碼(稱為:變數)」。所以我們對「變數」的定義如下：

「變數」是指　記憶體中【存放資料】的位置名稱（這個位置，由電腦自動設定），因為這個位置中所存放的資料，可視需要而隨時變更，因此，我們就將這一個記憶體中存放「資料的位置的名稱」，叫做　變數名稱　，簡稱「變數」。

而變數指令的格式如下：

格式：

數字資料

"文字資料"

變數＝ 日期資料

變數資料

運算式

在　echo（輸出指令）及 上面的 變數資料設定指令中，我們看到了「文字、數字、日期、…」多種的資料型態，先介紹如下：

PHP資料型態 ： 有八種介紹如下

|  |  |  |
| --- | --- | --- |
| 資料型態 | 識別字 | 說明 |
| 整數 | integer、int |  |
| 雙精確數 | double、float、real | 雙精確數、浮點數、實數 |
| (文)字串 | string | 必須用 " " 或 ' ' 包含 |
| 布林值 | Boolean | 真（T/1）、假（F/0） |
| 陣列 | array | 存放整組資料 |
| 物件 | object |  |
| 資源 | resource | 用來保存對外部資料的引用 |
| 空值 | NULL | 未設資料的變數值 |

特殊字串資料(又稱跳脫資料，以反斜線 \ 引導)

|  |  |  |  |
| --- | --- | --- | --- |
| 跳脫字元 | 說明 | 跳脫字元 | 說明 |
| \' | 顯示 ' 符號 | \r | Return |
| \" | 顯示 " 符號 | \t | Tab鍵 |
| \b | Backspace鍵 | \$ | 顯示 $ 符號 |
| \n | 換行符號 | \\ | 顯示 \ 符號 |

數值資料的種類及表示法

|  |  |
| --- | --- |
| 表示法 | 說明 |
| 123 | 整數 ( -2147483647 ~ +2147483647 ) |
| 123.34 | 雙精確數 |
| 0123 | 八進位數 |
| 0x123 | 十六進位數 |
| 1.23E+5 | 科學記號表示法 ( 123000) |

在變數中存放何種型態的資料，並不需要事先宣告，這是PHP語言與其他語言不同的地方，PHP在執行程式時會自動判斷，不過，

變數名稱開頭要加上 $ 符號做區別，變數名稱的第1個字元必須是英文字母。

再看一下變數的格式定義：

格式：

數字資料

"文字資料"

變數＝ 日期資料

變數資料

運算式

下面，我們以簡單的敍述指令來加深讀者對「變數」的印象。

例：ch08-04.php (變數應用)

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>資料類型</title>

</head>

<body>

<?php

$var1=566;

$var2=5.66E+5;

$var3="This a string.";

$var4=566.0;

//以gettype()函數,輸出對應的資料類型

echo "566 的資料型態：";

echo "<font color=blue>".gettype($var1)."</font><br>";

echo "5.66E+5 的資料型態：";

echo "<font color=blue>".gettype($var2)."</font><br>";

echo '"This a string."的資料型態：';

echo "<font color=blue>".gettype($var3)."</font><br>";

echo "566.0 的資料型態：";

echo "<font color=blue>".gettype($var4)."</font><br>";

?>

</body>

</html>

※相互參考另一個變數資料

格式：

$變數名稱1 = &$變數名稱2

註：變數名稱1 與 變數名稱2 之資料會互相參考，當其中一個內容改變時，另一個的內容也會改變。

例；ch08-05.php(變數應用)

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>使用變數</title>

</head>

<body>

<?php

$text\_string="這是字串資料"; //字串資料要以"或'括起來

$AA=10;

$aa=(2+3)\*12; // $AA和$aa是不同的變數

$BB=168;

數字轉字串的方法 $ff=(string)$ff; 需要把要改的數字前面使用(字串) 才可

echo $text\_string."<br>";

//要顯示$所以使用跳脫字元 \$

echo "\$AA =".$AA."<br>";

echo "\$aa = ".$aa."<br>";

echo "\$BB = ".$BB."<br>";

echo "<hr>";

//設定變數之間的參考,$AA和$BB兩個變數會互相影響

$AA=&$BB;

echo "變數參考後......<br>";

echo "\$AA = ".$AA."<br>";

echo "\$BB = ".$BB."<br>";

$AA=1000;

echo "<b>\$AA變數值</b>改變後......<br>";

echo "\$AA = ".$AA."<br>";

echo "\$BB = ".$BB."<br>";

$BB=666;

echo "<b>\$BB變數值</b>改變後......<br>";

echo "\$AA = ".$AA."<br>";

echo "\$BB = ".$BB."<br>";

?>

</body>

</html>

**常數**

相對於變數的資料可以變更，常數是指固定不變的資料，也就是說  
常數雖然也是存放資料的位置名稱，但這個位置的資料內容必須透過定義函數來宣告，無法像變數一樣透過直接設定或運算來變更。

常數定義指令

格式：

define(“常數名稱”,”資料內容”);

例：ch08-06.php(常數值指令設定的應用)

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>定義常數</title>

</head>

<body bgcolor="#FFFFCC">

<?php

define("PI",3.14159);

$name="張三丰";

echo "<h2>".$name."</h2>";

$rad=10;

echo "圓面積=半徑\*半徑\*圓周率=";

echo PI\*$rad\*$rad;

echo "<p>";

define("PI",6);

$name="陳大同";

echo "<h2>".$name."</h2>";

$rad=10;

echo "圓面積=半徑\*半徑\*圓周率=";

echo PI\*$rad\*$rad;

?>

</body>

</html>

註：看程式執行結果，可以瞭解，變數中的資料可以視需要隨時變更，  
　　　　　　　　　　　　　而　常數資料一旦宣告就不可以更改。

**轉換變數的資料型態**

前面提到，PHP會自動判斷變數的資料型態，但我們也可以用下列的指令來指定變數的資料型態

格式：

$變數名稱 = (資料型態識別字)$變數名稱;

格式：

~~settype($變數名稱,資料類型);~~

格式：(取得變數的資料型態)

gettype($變數名稱)

例：ch08-07.php(轉換資料型態的應用)

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>轉換資料類型</title>

</head>

<body>

<?php

$str="轉換資料類型";

echo "<h2>".$str."</h2>";

$data="566.34"; //原先是字串

echo "<font color='purple'>".$data."</font>";

echo "==>".gettype($data)."<br>";

settype($data,float); //轉成浮點 $data=(float) $data;

// 或 @settype($data,float);

echo "<font color='purple'>".$data."</font>";

echo "==>".gettype($data)."<br>";

$data=(int)$data; //轉成整數

echo "<font color='purple'>".$data."</font>";

echo "==>".gettype($data)."<br>";

?>

</body>

</html>

註：php7.0版本已經不用settype($data,float);的格式，

可改用$data=(float) $data;方式來變更資料型態

程式執行時會出現如下的 Warrning訊息，但指令還是會執行

**轉換資料類型**

566.34==>string  
  
**Warning**: Use of undefined constant float - assumed 'float' (this will throw an Error in a future version of PHP) in **C:\xampp\htdocs\110-all\ch08-07.php** on line **16**  
566.34==>double  
566==>integer

如不想出現上述Warrning訊息，可在程式前加上一個「@」符號

PHP的運算符號

變數的格式中有一項，「變數=運算式 」，運算式是指資料的處理方法，在三角形面積計算程式中，「$area=$hi\*$low/2;」的 ($hi\*$low/2)就是一種方法。

運算式的組成為因子有二種：

1. 運算元素： $hi、$low及2都是運算元素，指被用來運算的資料。
2. 運算符號：\*、/ 就是運算符號，指資料的處理方式。

之所以稱為「運算式」而不稱為「計算式」，是因為，針對PHP語言的各種資料型態，也各有其相對應的運算符號，分別表列如下：

算術運算符號

|  |  |  |
| --- | --- | --- |
| 運算符號 | 功能 | 說明 |
| + | 加 | $A+$B |
| - | 減 | $A-$B |
| \* | 乘 | $A\*$B |
| / | 除 | $A/$B |
| % | 取餘數 | $A%7 |
| ++ | 遞加 | $A++ 或 ++$A |
| -- | 遞減 | $A-- 或 --$A |
| . | 資料連接 | 將( . )左右兩邊的資料連接在一起。 |

註：$A++ 先將$A的值傳回後，再將$A的值加1

echo "<br>".++$c;

echo$c;

要使用++記得要使用echo幫忙否則是不會改變的，此外使用++之後的數字會永久改變 ++也分成$c++與++$c兩種

++$A 先將$A的值加1，再將$A的值傳回

例：ch08-08.php(算術運算的運用)

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>算術運算子</title>

</head>

<body>

<?php

$a=5\*4+3; //$a的值經運算後是23

$b=20%3; //$b的值經運算後是2

echo "原始 \$a 的值 = <b>".$a."</b><br>"; // . 為資料連接符號

//$a的值先加1再傳回

echo "++\$a 的值 = <b>".++$a."</b><br>";

echo "<p>原始 \$b 的值 = <b>".$b."</b><br>";

//先傳回$b的值再做加1的動作

echo "\$b++ 的值 = <b>".$b++."</b><br>";

echo "最後 \$b 的值 = <b>".$b."</b>";

?>

</body>

</html>

判斷(比較)運算符號

|  |  |  |
| --- | --- | --- |
| 運算符號 | 功能 | 說明 |
| = = | 資料值相等 | $a = = $b |
| = = = | 資料值及型態相同 | $a = = = $b |
| != | 資料值不相等 | $a != $b |
| <> | 資料值不相等 | $a <> $b |
| != = | 資料值或型態不相同 | $a != = $b |
| > | 大於 | $a > $b |
| < | 小於 | $a < $b |
| >= | 大於或等於 | $a >= $b |
| <= | 小於或等於 | $a <= $b |

例：ch08-09.php(比較運算符號的運用)

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>比較運算子</title>

</head>

<body>

<h2>比較運算子</h2>

<?php

$a=12.33;

$b=566; //整數變數

$c="566"; //字串變數

echo "\$a = ".$a."<br>";

echo "\$b = ".$b."<br>";

echo "\$c = ".$c."<br><hr>";

echo "<font color='purple'>";

if($a<$b){

echo "\$a 小於 \$b <br>";

}else{

echo "\$a 大於 \$b <br>";

}

if($b==$c){

echo "\$b 等於 \$c <br>";

}else{

echo "\$b 不等於 \$c <br>";

}

if($b===$c){

echo "\$b 等於 \$c 且資料型態相同 <br>";

}else{

echo "\$b 不等於 \$c 或資料型態不相同 <br>";

}

?>

</body>

</html>

註：本例中所用到 if … then …else … 是一個條件判斷指令，在下一單元中介紹

邏輯運算符號(兩個條件的組和判斷)

|  |  |
| --- | --- |
| 運算符號 | 說明 |
| && | 而且 (兩者都對才算對) |
| and | 同上 |
| or | 或者 (兩者中只要有一個對就算對) |
| xor | 互斥或 (兩者相反才算對) |
| ! | 相反 |

例：ch08-10.php (邏輯運算符號的運用)

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>邏輯運算子</title>

</head>

<body>

<h2>邏輯運算子</h2>

<?php

$a=true;

$b=false;

echo "<font color='blue'>";

echo "\$a = true<br>";

echo "\$b = false</font><br><hr>";

echo "\$a && \$b 運算結果 ";

if($a && $b){

echo "true<p>";

}else{

echo "false<p>";

}

echo "\$a xor \$b 運算結果 ";

if($a xor $b){

echo "true<p>";

}else{

echo "false<p>";

}

?>

</body>

</html>

註：if指令在下一單元會講解

複合運算符號

|  |  |  |  |
| --- | --- | --- | --- |
| 運算符號 | 功能 | 範例 | 說明 |
| += | 相加後再指定 | $a +=5; | $a=$a+5; |
| -= | 相減後再指定 | $a -=5; | $a=$a-5; |
| \*= | 相乘後再指定 | $a \*=5; | $a=$a\*5; |
| /= | 相除後再指定 | $a /=5; | $a=$a/5; |
| %= | 取餘數後再指定 | $a %=5; | $a=$a%5; |

例：ch08-11.php(複合運算符號的運用)

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>複合運算子</title>

</head>

<body>

<h2>複合運算子</h2>

<?php

$a=566;

$b=20;

echo "<font color='blue'>";

echo "\$a=566<br>";

echo "\$b=20</font><br><hr>";

$a += $b;

echo "\$a += \$b 的結果是".$a."<p>";

$b \*=5;

echo "\$b \*=5 的結果是".$b."<p>";

?>

</body>

</html>

註：「複合運算符號」在下一單元會講解

位元運算符號

位元運算符號是針對資料做二進位式的運算，有「邏輯運算、位移運算、反相運算」三種方式

|  |  |  |
| --- | --- | --- |
| 位元運算符號 | 範例 | 說明 |
| & | $a & $b | 左右兩變數and運算 |
| ∣ | $a∣$b | 左右兩變數or運算 |
| ^ | $a ^ $b | 左右兩變數xor運算 |
| ~ | ~$a | 變數資料做反相運算 |
| >> | $a>>2 | 資料往右移動2位元 |
| << | $a<<2 | 資料往左移動2位元 |

例：ch08-12.php( $a=6,$b=10,計算$a & $b 和 ~$a 的值

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>位元運算子</title>

</head>

<body>

<?php

//指定變數值

$a=6;

$b=10;

$result1= ~$a;

$result2=$a & $b;

//顯示原變數值

echo "\$a = ".$a."<br>";

echo "\$b = ".$b."<hr>";

//顯示位元運算後的值

echo "~\$a = ".$result1."<br>";

echo "\$a&\$b = ".$result2."<br>";

?>

</body>

</html>

註：位元運算在商業應用上比較少見，較常用於工程應用或資料加密應用。

PHP表單資料傳送

網頁間的資料傳送方式，最常見的就是透過表單元件功能傳送，PHP程式語言本身並没有提供表單指令，它是藉由HTML所提供的<FORM>標籤指令為媒介來做資料傳送的處理工作。

<FORM>標籤指令的屬性設定中有兩項：

　 method 是用來設定「資料傳送方式」

而 action 是指到伺服器端負責處理傳送過來的資料的程式。

Method 的設定值有兩種如下:

1. method= "post ": 這種方式會先將表單內的資料編碼在一個封包內傳送。  
 傳送的資料可以多512byte，資料的保密性也較佳。

2. method= "get ": 表示採用URL網址參數傳送(附在網址列後面)，這種方式所  
能傳送的資料不能超過512byte。傳送的資料没有保密功能。

對於用戶端（client）所傳送過來的資料，php程式語言是透過內建陣列變數【 $\_GET 】、【 $\_POST 】或 【 $\_REQUEST 】來接收。

|  |  |
| --- | --- |
| 內建陣列變數 | 說明 |
| $\_GET["欄位名稱"] | 接收GET方法所傳送的資料 |
| $\_POST["欄位名稱"] | 接收POST方法所傳送的資料 |
| $\_REQUEST["欄位名稱"] | 兩種方法所傳送的資料都可接收 |

下面我們直接以範例來做說明:

例.設計一個HTML網頁表單程式 ( Ch08-13.htm )，傳送會員姓名和密碼資料；  
伺服器端的處理程式為 ( Ch08-13.php ) ，再回傳所收到的資料。

Ch08-13.htm

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>設計表單</title>

</head>

<body>

<form name="form1" method="post" action=" Ch08-13.php ">

<p>請輸入會員資料:</p>

<p>姓名:

<input type="text" name="uname"></p>

<p>密碼:

<input type="text" name="upass"></p>

<p>

<input type="submit" name="Submit" value="送出">

<input name="Reset" type="reset" id="Reset" value="重設"> </p>

</form>

</body>

</html>

Ch08-13.php

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>伺服端的PHP程式</title>

</head>

<body>

<?php

echo "<h2><font color='purple'>PHP程式所回應的資料</font></h2>";

echo "你的姓名是 : ".$\_POST["uname"]."<p>";

echo "你的密碼是 : ".$\_POST["upass"]."<p>";

?>

</body>

</html>

在上例中，我們將接收資料的指令，放在echo 指令後面，直接印出，主要在印證PHP程式具有「接收表單資料」的功能。

實際上，依照 「輸入　→　處理　→　輸出」的SOP，我們通常會先將「表單資料」接收並存放在 對應的 變數名稱(輸入)中，待程式的所有處理步驟皆完成後(處理)，才會一併進行結果報表的列印(輸出)。

以下例說明：「輸入　→　處理　→　輸出」的SOP流程

Ch08-14.htm

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>設計表單</title>

</head>

<body>

<form name="form1" method="post" action=" Ch08-14.php ">

<p>請輸入矩形的長寬資料:</p>

<p>長:

<input type="text" name="len"></p>

<p>寬:

<input type="text" name="wid"></p>

<p>

<input type="submit" name="Submit" value="送出">

<input name="Reset" type="reset" id="Reset" value="重設"> </p>

</form>

</body>

</html>

Ch08-14.php

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>伺服端的PHP程式</title>

</head>

<body>

<?php

//////// 接收表單傳送資料（輸入）/////////////////

$len=$\_POST["len"];

$wid=$\_POST["wid"];

/////// 進行面積計算 (處理) ///////////////////////

$area=$len\*$wid;

/////// 列印報表 (輸出) ////////////////////////////

echo "<h2><font color='purple'>矩形的面積計算</font></h2>";

echo "矩形的長為 : ".$len."<p>";

echo "矩形的寬為 : ".$wid."<p><hr>";

echo "矩形的面積為 : ".$area."<p>";

?>

</body>

</html>

※同學在演練範例的時候，希望能夠親自按鍵輸入程式碼，不要用複製的方式，這樣有二個好處：

　1.加速記憶指令單字及格式語法

2.訓練除錯的能力。親自按鍵輸入資料，當然有輸入錯誤資料的可能，如此在程式執行時就會產生錯誤。

程式執行錯誤有兩種情形：

1. 實體錯誤：指令單字(關鍵字)錯誤或語法錯誤，導致程式無法執行。此種錯誤，電腦會出現 錯誤訊息 ，設計者可以依據 訊息內容進行程式修改。
2. 邏輯錯誤：邏輯錯誤通常是 運算式錯誤(運算資料或運算符號錯誤)，此種錯誤則需運用幾組資料進行測試，稱為trace。我們在下一單元再來說明。

例. 設計一個HTML網頁表單程式 ( Ch08-15.htm )，傳送會員姓名、密碼及備註資料；伺服器端的處理程式為 ( Ch08-15.php ) ，再回傳所收到的資料。

Ch08-15.htm

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>輸入會員資料</title>

</head>

<body>

<form name="form1" method="post" action=" Ch08-15.php">

<p>姓名:

<input type="text" name="uname">

</p>

<p>密碼:

<input type="password" name="upass">

</p>

<p>備註:

<textarea name="note"></textarea>

</p>

<p>

<input type="submit" name="Submit" value="送出">

<input name="reset" type="reset" id="reset" value="重設">

</p>

</form>

</body>

</html>

Ch08-15.php

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>顯示會員資料</title>

</head>

<body>

<?php

/////// 接收資料 (取得表單中所輸入對應欄位中的資料) /////////////////

$uname=$\_POST["uname"];

$upass=$\_POST["upass"];

$note=$\_POST["note"];

///////////////// 列印資料 /////////////////

echo "會員名稱: ".$uname."<p>";

echo "會員密碼: ".$upass."<p>";

echo "會員備註: ".$note ."<p>";

?>

</body>

</html>

選項鈕的應用

例:設計一個商品送方式的選項按鈕功能程式( Ch08-16.htm )，伺服器端的處理程式為 (Ch08-16.php ) ，再回傳所收到的資料。

Ch08-16.htm

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>訂購確認單</title>

</head>

<body>

<h2>商品訂購確認單</h2>

<form name="form1" method="post" action=" Ch08-16.php">

<p>姓名:

<input type="text" name="uname">

</p>

<p>電話:

<input type="text" name="telephone">

</p>

<p>地址:

<input type="text" name="address" size="50">

</p>

<p>請選取郵寄方式 :</p>

<p>

<input type="radio" name="delivery" value="掛號" checked> 掛號

<input type="radio" name="delivery" value="普通"> 普遍

<input type="radio" name="delivery" value="快捷"> 快捷</p>

<p>

<input type="submit" name="Submit" value="送出">

<input name="reset" type="reset" id="reset" value="重設">

</p>

</form>

</body>

</html>

Ch08-16.php

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>Radio選項鈕</title>

</head>

<body>

<?php

/////// 接收資料 (取得表單中所輸入對應欄位中的資料) /////////////////

$uname=$\_POST["uname"];

$telephone=$\_POST["telephone"];

$delivery=$\_POST["delivery"];

$address=$\_POST["address"];

///////////////// 列印資料 /////////////////

echo "<b>".$uname." </b>的連絡電話是: <b>".$telephone."</b><p>";

//取得選項鈕所選取的項目

echo "您訂購產品會依照 <b>".$delivery." </b>郵寄方式寄到下列地址:</br>";

//顯示一條水平線

echo "<hr><p>";

//取得地址資料

echo $address;

?>

</body>

</html>

核取方塊的應用

例:設計一個新聞頻道票選網頁(可複選)（Ch08-17.htm），伺服器端的處理程式為 ( Ch08-17.php ) ，再回傳所收到的資料。

Ch08-17.htm 有異常 已經修正

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>新聞頻道票選</title>

</head>

<body>

<h2>喜歡的新聞頻道</h2>

<form name="form1" method="post" action=" Ch08-17.php">

<p>會員姓名:

<input type="text" name="uname">

</p>

<p>選取喜歡的新聞頻道:</p>

<p>

<input type="checkbox" name="box1" value="年代" checked>年代

<input type="checkbox" name="box2" value="東森">東森

<input type="checkbox" name="box3" value="中天">中天

<input type="checkbox" name="box4" value="民視">民視

<input type="checkbox" name="box5" value="三立">三立

<input type="checkbox" name="box6" value="TVBS"> TVBS </p>

<p>

<input type="submit" name="Submit" value="送出">

<input name="reset" type="reset" id="reset" value="重設">

</p>

</form>

<p>&nbsp;</p>

</body>

</html>

Ch08-17.php

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>Checkbox核取方塊</title>

</head>

<body>

<?php

/////// 接收資料 (取得表單中所輸入對應欄位中的資料) /////////////////

$uname=$\_POST["uname"];

for($x=0;$x<7;$x++)

{

$str="box".$x; //box與索引值合併成新字串

//利用isset函數判定核取方塊是否被選取

if(isset($\_POST[$str]))

{//true表示有被選取

$box[]=$\_POST[$str];

}

}

///////////////// 列印資料 /////////////////

echo $uname." 會員:<p>";

echo "你喜歡的新聞頻道有:<p>";

//for重覆結構

for($x=1;$x<count($box);$x++)

{

echo $box[$x]." , ";

}

?>

</body>

</html>

說明：本範例中有用到for指令，在重覆結構單元中再做說明

下拉式清單的應用

例:設計一個會員基本資料的程式如下(Ch08-18.htm )，伺服器端的處理程式為 ( Ch08-18.php ) ，再回傳所收到的資料。

Ch08-18.htm

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>會員喜好的水果</title>

</head>

<body>

<form name="form1" method="post" action=" Ch08-18.php">

<p>姓名:

<input type="text" name="uname">

</p>

<p>職稱:

<select name="select1">

<option value="總經理">總經理</option>

<option value="經理">經理</option>

<option value="業務">業務</option>

<option value="行政助理">行政助理</option>

<option value="工程師">工程師</option>

<option value="資訊人員" selected>資訊人員</option>

</select>

</p>

<p>喜歡的水果:

<select name="select2[]" size="4" multiple>

<option value="蘋果">蘋果</option>

<option value="香蕉">香蕉</option>

<option value="蓮霧">蓮霧</option>

<option value="西瓜">西瓜</option>

<option value="文旦">文旦</option>

<option value="水梨">水梨</option>

<option value="桃子">桃子</option>

<option value="鳳梨">鳳梨</option>

<option value="龍眼">龍眼</option>

<option value="楊桃">楊桃</option>

<option value="芭樂" selected>芭樂</option>

</select>

</p>

<p>

<input type="submit" name="Submit" value="送出">

<input name="reset" type="reset" id="reset" value="重設">

</p>

</form>

</body>

</html>

Ch08-18.php

<html>

<head>

<meta http-equiv="Content-Type" content="text/html; charset=big5">

<title>清單/選單控制項</title>

</head>

<body>

<?php

/////// 接收資料 (取得表單中所輸入對應欄位中的資料) /////////////////

$uname=$\_POST["uname"];

$select1=$\_POST["select1"];

//將陣列資料指定給$arr變數

$arr=$\_POST["select2"];

///////////////// 列印資料 /////////////////

echo $uname."你好:<p>";

echo "你的職稱是 : ";

echo $select1."<p>";

echo "喜歡的水果有以下這些 : <p>";

//以print\_r函數顯示陣列資料

print\_r($arr);

?>

</body>

</html>

**補充說明**

※一個程式中，可以存在多組的表單

例: ch08-19.htm 設計一個可以計算「三角形、方形、梯形及圓形」面積的程式

<html>

<head>

<title>面積計算</title>

</head>

<body>

<h1>面積計算</h1>

<table border="1" cellspacing="0">

<tr>

<td>三角形<br>

<form method="post" action="ch08-19-1.php">

請輸入高：<input type="text" name="hi" size="4"><br>

請輸入底：<input type="text" name="low" size="4"><br><br>

<input type="submit" value="送出">

</form>

</td>

<td>矩形<br>

<form method="post" action="ch08-19-2.php">

請輸入長：<input type="text" name="len" size="4"><br>

請輸入寬：<input type="text" name="wid" size="4"><br><br>

<input type="submit" value="送出">

</form>

</td>

<td>梯形<br>

<form method="post" action="ch08-19-3.php">

請輸入　高：<input type="text" name="hi" size="4"><br>

請輸入上底：<input type="text" name="flow" size="4"><br>

請輸入下底：<input type="text" name="slow" size="4"><br>

<input type="submit" value="送出">

</form>

</td>

<td>

<form method="post" action="ch08-19-4.php">

請輸入半徑：<input type="text" name="rad" size="4"><br>

<br><br>

<input type="submit" value="送出">

</form>

</td>

</tr>

<tr>

<td>面積為：</td>

<td>面積為：</td>

<td>面積為：</td>

<td>面積為：</td>

</tr>

</table>

</body>

</html>

ch08-19-1.php

<html>

<head>

<title>面積計算</title>

</head>

<body>

<?php

///////// 接收資料(輸入) /////////////

$hi=$\_POST["hi"];

$low=$\_POST["low"];

///////// 計算面積(處理) /////////////

$area=$hi\*$low/2;

///////// 列印表單 /////////////

?>

<h1>面積計算</h1>

<table border="1" cellspacing="0">

<tr>

<td>三角形<br>

<form method="post" action="ch08-19-1.php">

<?php

echo '請輸入高：<input type="text" name="hi" size="4" value="'.$hi.'"><br>';

echo '請輸入底：<input type="text" name="low" size="4" value="'.$low.'"><br><br>';

?>

<input type="submit" value="送出">

</form>

</td>

<td>矩形<br>

<form method="post" action="ch08-19-2.php">

請輸入長：<input type="text" name="len" size="4"><br>

請輸入寬：<input type="text" name="wid" size="4"><br><br>

<input type="submit" value="送出">

</form>

</td>

<td>梯形<br>

<form method="post" action="ch08-19-3.php">

請輸入高　：<input type="text" name="hi" size="4"><br>

請輸入上底：<input type="text" name="flow" size="4"><br>

請輸入下底：<input type="text" name="slow" size="4"><br>

<input type="submit" value="送出">

</form>

</td>

<td>圓形<br>

<form method="post" action="ch08-19-4.php">

請輸入半徑：<input type="text" name="rad" size="4"><br>

<br><br>

<input type="submit" value="送出">

</form>

</td>

</tr>

<tr>

<td>面積為：

<?php echo $area; ?> <!-- php程式碼 -->

</td>

<td>面積為：</td>

<td>面積為：</td>

<td>面積為：</td>

</tr>

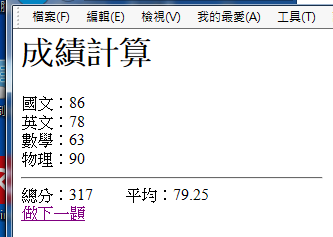
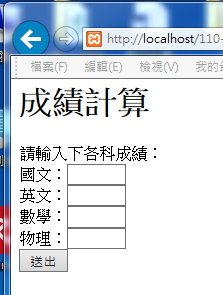
</table>

</body>

</html>

單元作業１：請仿照 ch08-19-1.php程式架構，完成 ch08-19-2.php 、   
 ch08-19-3.php 及 ch08-19-4.php

單元作業2：設計表單資料傳送程式，畫面如下：(ex08-02.htm，ex08-02.php)



點 做下一題 會跳回 ex08-02.htm