

動態網頁設計

第三週

老師：蔡承洋

大綱

- php 資料型態 (字串、數字、布林值、陣列)
- php 流程控制 (判別)
- php 迴圈控制
- 課堂小作業

php 資料型態 (字串、數字、布林值、陣列)

- 字串

做為一般文字，不做運算，通常以單引號或雙引號一前一後包覆。

“雙引號” 的字串，裡面可以夾帶變數，夾帶的變數會呈現變數內容。

‘單引號’ 的字串，不會呈現變數內容，會將該變數名稱當成字串輸出。

```
<?php
    $my_name = "蔡承洋";
    $my_age = 18;

    //使用雙引號
    echo "我叫{$my_name}，今年{$my_age}歲"; //會輸出 我叫蔡承洋，今年18歲
    echo "<br>";

    //使用單引號
    echo '我叫{$my_name}，今年$my_age歲'; // 會輸出 我叫{$my_name}，今年$my_age歲
    echo "<br>";

?>
```

- 字串

輸出html時，同時會用到雙引號及單引號

正常來說，有些標籤會有屬性，若使用php印出包含屬性的標籤，就得要注意。

```

<br>
<?php
    //使用php 列印的時候，要注意 單/雙引號的使用
    echo '';
    echo "<hr>";
    echo "<img src='images/my_photo.jpg'>";
?>
```

- 數字

數字包含正負整數、浮點數(有小數點的數值)等，數字可以做運算。

```
<?php
$year = 2015;
$taiwan_year = $year - 1911; //為何要減1911? 請 google 民國紀年

echo "現在是民國{$taiwan_year}年"; //輸出 現在是民國104年

echo "<hr>";

$service = 1.1; //一成服務費， 會寫1.1 是因為 100%(原價) + 10%(一成) = 110%，換成可計算的數字就是 1.1，
$price = 599; //晚餐價位
$man = 3; //三個人吃
$discount = 0.9; //打九折
$total = ($price * $man * $service) * $discount;
echo "晚餐時段{$price}/人，共{$man}位客官，消費金額" . ($price * $man * $service) . "(已加一成服務費)，打九折後是NT:{$total}。請問刷卡或付現?";
?>
```


- 布林值

只有true跟false兩個值，代表對(true)或錯(false)。

布林值，大多用在於「做判別時」使用。

```
<?php
    $good_to_go = true;    //一個變數 $good_to_go 放了一個 true 的布林變數，若用 echo 輸出，會是 1
    $is_good_drink = false; //一個變數 $is_good_drink 放了一個 false 的布林變數，若用 echo 輸出，會是空白

    echo '$good_to_go echo 結果是: ';
    echo $good_to_go;
    echo "<hr>";
    echo '$is_good_drink echo 結果是: ';
    echo $is_good_drink;
?>
```

- 陣列

它讓你可以儲存多個值在同一個變數中。

好比像血拼時有個購物清單一樣，不用分很多張紙寫，全寫在一張紙中就好。

陣列裡可以儲存（字串、數字、布林值、陣列）

```
<?php
//一個 party_need 陣列變數，裡面放了七個字串
$party_need = array("可口可樂", "雪碧", "洋芋片", "衛生杯", "啤酒", "PS4", "XBOX");

//使用 print_r(); 方法，將陣列輸出
echo 'party 需要買的項目';
print_r($party_need);
```

```
?>
```


- 陣列-索引值

陣列中的每個值，都會有自己的索引值，通常是從0開始的數字。

(比如一棟公寓，每個房間都有自己的門牌號碼，而房間裡就是住著字串、數字、布林值、陣列)

```
<?php
//紅酒茄汁肉醬義大利麵需要的食材
$pasta = array(
    "豬絞肉", //索引值 0
    "橄欖油", //索引值 1
    "洋蔥",   //索引值 2
    "蒜頭",   //索引值 3
    "牛番茄", //索引值 4
    "番茄醬", //索引值 5
    "起司粉", //索引值 6
    "紅酒",   //索引值 7
    "月桂葉", //索引值 8
    "義大利麵", //索引值 9
    "鹽巴"   //索引值 10
);

//印出蒜頭
echo $pasta[2];
?>
```

php 流程控制 (判別)

● 判別表示式

判別式

`==` 兩個等於，在程式裡面意思是判別前後兩個數值是否「等於」。

`===` 三個等於，除了值要一樣，型態也要一樣 `'1' === 1` 會是 `false`，因為 字串1 跟 數字1 嚴格來說是不一樣的

`!=` 一個驚嘆號一個等於，是「不等於」

`!==` 一個驚嘆號兩個等於，是「不等於，同時也比較型態」

`>` 左邊是否大於右邊

`<` 左邊是否小於右邊

`>=` 左邊是否大於等於右邊

`<=` 左邊是否小於等於右邊

```
$a = 10;  
$b = 10;  
$c = "老婆";  
$d = "小三";  
  
echo $a == $b; //輸出1  
echo "<hr>";  
  
echo $c == $d; //輸出空白  
echo "<hr>";
```


- if/else

如果是/不然就是

```
$a = 10;
$b = 10;
$c = "老婆";
$d = "小三";

// if / else 判別式

if($a == $b)
{
    //當判別結果為 true 時
    echo "{$a} 等於 {$b}";
}
else
{
    echo "{$a} 不等於 {$b}";
}

echo "<hr>";

if($c == $d)
{
    //當判別結果為 true 時
    echo "{$c} 等於 {$d}";
}
else
{
    echo "{$c} 不等於 {$d}";
}
```

```
/**
 * 另一種寫法，去掉大括弧，用「冒號」以及 endif; 結束
 */
if($a == $b):
    echo "{$a} 等於 {$b}";
else:
    echo "{$a} 不等於 {$b}";
endif;
```

- if/elseif/else

如果是/要不然如果是/不然就是

```
//多重判別 if / elseif / else
if($a < 18)
{
    //當判別結果為 true 時
    echo "{$a}歲，未成年";
}
elseif($a == 18)
{
    //或另外再判斷，且結果為 true 時
    echo "{$a}歲，剛好成年";
}
else
{
    echo "{$a}歲，超過18歲以上了";
}
```

```
/**
 * 另一種寫法，去掉大括弧，用「分號」以及 endif; 結束
 */

if($a < 18):
    //當判別結果為 true 時
    echo "{$a}歲，未成年";
elseif($a == 18):
    //或另外再判斷，且結果為 true 時
    echo "{$a}歲，剛好成年";
else:
    echo "{$a}歲，超過18歲以上了";
endif;
```

- switch

有符合哪些情況，就執行。

```
$i_love = "女朋友";
//switch 判別式
//符合 case 定的條件時，就執行
//遇到 break; 就跳出判別，若忘了寫 break; 就會繼續跑下去唷。
switch ($i_love){
    case "女朋友":
        echo "忠心耿耿";

        break;
    case "小三":
        echo "恩～～要小心...這樣不好唷";

        break;
    case "隔壁帥哥":
        echo "我支持多元成家方案";

        break;
    default:
        echo "太博愛了";

        break;
}
```

```
/**
 * 另一種寫法，去掉括弧用「冒號」以及 endswitch; 結束
 */
switch ($i_love):
    case "女朋友":
        echo "忠心耿耿";

        break;
    case "小三":
        echo "恩～～要小心...這樣不好唷";

        break;
    case "隔壁帥哥":
        echo "我支持多元成家方案";

        break;
    default:
        echo "太博愛了";

        break;
endswitch;
```


php 迴圈控制

- 迴圈-for

我們要重複一直做一些事情的時候，就可以使用迴圈來處理。

而for迴圈讓我們可以自訂一個重複的範圍

比如說，重複輸出年份。

```
<?php
// $start_year 初始內容為 2000 數字，當 $start_year 小於等於 2030的時候執行迴圈內容，
// 完成後把 $start_year + 1 並存入 $start_year 中
// 然後再繼續判別 $start_year 有沒有小於等於 2030 如有就繼續
for($start_year = 2000; $start_year <= 2030; $start_year += 1)
{
    echo "<p>{$start_year}</p>";
}
?>
```

```
for(變數; 條件判別; 重複執行的事情完成後，要做的事情，通常都是把變數遞增或遞減){
    //重複執行的事情
}
```

● 運算簡寫

```
//一般算法
$a = 10;
echo "<p>{$a}</p>"; //輸出 10
$a = $a + 1;        // $a = 10 + 1
echo "<p>{$a}</p>"; //輸出 11

echo "<hr>";

// ++ 變數自己+1 簡寫
$b = 10;
echo "<p>{$b}</p>"; //輸出 10
$b++;              // $b + 1 後存入 $b 中
echo "<p>{$b}</p>"; //輸出 11

echo "<hr>";

// -- 變數自己-1 簡寫
$b = 10;
echo "<p>{$b}</p>"; //輸出 10
$b--;              // $b - 1 後存入 $b 中
echo "<p>{$b}</p>"; //輸出 9

echo "<hr>";

// += 變數+後面的數值，在存回變數 簡寫
$c = 10;
echo "<p>{$c}</p>"; //輸出 10
$c += 1;           //等於 $c = $c + 1
echo "<p>{$c}</p>"; //輸出 11

echo "<hr>";
```

```
$d = 10;
echo "<p>{$d}</p>"; //輸出 10
$d += 4;           //等於 $d = $d + 4
echo "<p>{$d}</p>"; //輸出 14

echo "<hr>";

// -= 變數-後面的數值，在存回變數 簡寫
$d = 10;
echo "<p>{$d}</p>"; //輸出 10
$d -= 4;           //等於 $d = $d - 4
echo "<p>{$d}</p>"; //輸出 6

echo "<hr>";

// *= 變數*後面的數值，在存回變數 簡寫
$e = 10;
echo "<p>{$e}</p>"; //輸出 10
$e *= 4;           //等於 $d = $d * 4
echo "<p>{$e}</p>"; //輸出 40

echo "<hr>";

// /= 變數/後面的數值，在存回變數 簡寫
$f = 10;
echo "<p>{$f}</p>"; //輸出 10
$f /= 4;           //等於 $d = $d / 4
echo "<p>{$f}</p>"; //輸出 2.5
```


- 迴圈-while

while迴圈的用途，只有做一個判別，若條件正確就跑迴圈內容，直到不正確才停止。

```
<?php
//while迴圈
$a = 0;           // $a 的內容為 0
echo "<ol>";       // 使用 ol 列表的標籤
while($a < 100)   // 當 $a < 100 時，就做迴圈內容，直到 $a < 100 是錯的時候，才停止回圈，也就是 $a 為 100 或以上的時候
{
    echo "<li>a變數是:{$a}</li>";           // 每個列表內容使用 <li> </li> 包起來
    $a += 1;                               // 執行輸出後，最後再把 $a + 1 並存回 $a 中
}
echo "</ol>";    // ol 列表標籤的結束
?>
```

```
while(條件判別){
    //重複執行的事情
}
```

● 迴圈-foreach

用foreach重複造訪array陣列的內容，便可根據陣列內容進行處理。

```
<?php
    //一個 party_need 陣列變數，裡面放了七個字串
    $party_need = array("可口可樂", "雪碧", "洋芋片", "衛生杯", "啤酒", "PS4", "XBOX");
?>
<!DOCTYPE html>
<html lang="zh-TW">
    <head>
        <meta charset="utf-8">
        <meta http-equiv="X-UA-Compatible" content="IE=edge,chrome=1">

        <title>party 準備物品</title>
    </head>

    <body>
        <h2>請購買以下商品，以供party使用</h2>
        <ul>
            <?php
                //foreach迴圈，使用它來根據 $party_need 裡面每個項目進行處理，有幾個就處理幾次
                foreach ($party_need as $key => $value) {
                    echo "<li>{$value}</li>";
                }
            ?>
        </ul>
    </body>
</html>
```

```
foreach(陣列變數 as 陣列索引值 => 該索引裡的值){
    //重複執行的事情
}
```

課堂小練習

- 請使用php輸出 10, 20, 30, 40, 50, 60, 80, 100
要輸出的結果如紅色字體，要一模一樣唷，100後面沒有, 唷
提示：可用for或者while迴圈，並搭配if/else判別輸出
- 有一個新聞標題的陣列，請將該陣列輸出成可以點擊的新聞連結，連結暫時用#代替，輸出的結果，可用 li 或者 ol 列表標籤來呈現。
`$news_title = array("契約書耍詐 惡房東涉坑房客", "國道4車撞全指飛鳥肇逃", "賓士失控撞破門 家具金童棄醉友落跑", "機車逆向撞單車 高中生撞死高中生", "外交部網頁涉侵權 攝影家擬告");`
提示：用foreach迴圈，輸出 `新聞標題`