**函數的應用**

當某一程式設計如下：

<?php

定義 函數(A)

{

/////////////////////////////////////

工作A

/////////////////////////////////////

}

呼叫 函數(A)

呼叫 函數(A)

呼叫 函數(A)

?>

<?php

////////////////////////////////////////

工作A

////////////////////////////////////////

////////////////////////////////////////

工作A

////////////////////////////////////////

////////////////////////////////////////

工作A

////////////////////////////////////////

?>

圖一 圖二

如上圖一，在程式中我們可以發現「工作A」一再出現，若每一次都要重覆編寫「工作A」的程式步驟，實屬煩瑣，因此，PHP程式語言提供「函數」功能，來簡化程式設計的篇幅。

如上圖二，我們先在程式開頭(習慣用法)定義 一個函數(A)，來包裹「工作A的程式步驟」。在後面的程式執行步驟中，當要執行「工作A」的步驟時，只要下 「呼叫 函數(A)」的指令即可。

* 有些工作項目，例如：「排序、隨機檢取、資料分割、合併或比對、…」常見於不同的作業領域中。對於這些工作項目，PHP程式主動提供相關的函數，稱為「內建函數」。本單元中要討論的就是一些常用的 「內建函數」。
* 另外，也許對於某些特定的工作領域內，可能常見的工作項目，而PHP卻未提供相關函數；對此部份，設計者可自行 設定(計)函數，稱為「自訂函數」，將在下一單元中探討。

數值函數

我們也經常會碰到一些數學計算問題，同樣的PHP也提供了一些常見計算函數，列表如下：

|  |  |
| --- | --- |
| 函數指令 | 功能說明 |
| abs(數值) | 絕對值函數 |
| max(數值1，數值2,…) | 最大值值函數 |
| max(陣列) | 傳回陣列中資料的最大值 |
| min(數值1，數值2,…) | 最小值函數 |
| min(陣列) | 傳回陣列中資料的最小值 |
| pi( ) | 圓周率 |
| rand(n,m) | 隨機傳回介於 n~m中的數值(n,m預設0~32768) |
| round(數值[,小數位數]) | 於指定的小數位置，執行四捨五入 |
| sqrt(數值) | 平方根函數 |

例：設定一個含有負數的浮點數字，取其絕對值，並取小數兩位4捨5入；再以隨機(亂數)方式產生10筆介於0~1000的資料，並將資料存於陣列中，再找出這10 筆資料中的最大值及最小值。

ch12-01.php

<html>

<head>

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

<title>常用數學函數</title>

</head>

<body>

<?php

//原始數值

$nm=-123.456;

echo "原數值 = ".$nm."<br/>";

$nm=abs($nm);

$nm=round($nm,2);

echo "取絕對值和四捨五入小數取二位 => ".$nm;

echo "<p>亂數產生陣列值 : <br/>";

for($a=0;$a<10;$a++){

//以亂數產生10筆陣列值(0~1000)

$arr[$a]=rand(0,1000);

//顯示陣列元素

echo "\$arr[$a] => ".$arr[$a]."<br/>";

}

echo "</p><p>陣列最大值 : ".max($arr)."<br/>";

echo "陣列最小值 : ".min($arr);

?>

</body>

</html>

**抽籤問題(亂數)**

1.可重覆抽取

例：從1~10(n)中，隨機產生6個號碼(可重覆)

ch12-02.php

<?php

For($x=0;$x<6;$x++)

{

$ans[$x]=rand(1,10); //隨機產生 1~10的數字

}

Echo "抽出的號碼如下(1~10)(可重覆)：<br>";

For($x=0;$x<6;$x++)

{

Echo $ans[$x]." 、 ";

}

?>

2.不重覆抽取(抽出之籤可放回)

例：從1~10(n)中，隨機產生6個號碼(不可重覆)

ch12-03.php

<?php

$nu=0; //計算已抽出的籤數

$cp=0; //判斷是否有重覆抽取，0 為没有，1 為有

do

{

$pro=rand(1,10); //隨機產生 1~10的數字

//與先前已抽出的號碼比對看是否有重覆抽取

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

{

if($pro==$ans[$x]) //如果有相同

{

$cp=1; //$cp值設為1

}

}

if($cp==0) //比對完後，進行判斷，如果 $cp=0 表示没有重覆

{

$ans[$nu]=$pro; //將產生的號碼，放入答案陣列

$nu++; //已抽出的籤數 +1

}

else

{

$cp=0;

}

}while($nu<=6); //若抽出籤號數量小於6，則繼續抽

Echo "抽出的號碼如下(1~10)(不可重覆)：<br>";

For($x=0;$x<6;$x++)

{

Echo $ans[$x]." 、 ";

}

?>

3. 不重覆抽取(抽出之籤不再放回)

觀念：1.先設定一個號碼陣列(索引值1~10，對應號碼 1~10)

2.第1次，隨機從10個號碼陣列位置(1~10)中抽出1個(假設抽出位置  
 為3)，再將最後一個位置(10)的資料，放到位置3中(原位置3

的資料已被抽出)

3.第2次，隨機從剩餘9個號碼陣列位置(1~9)中抽出1個(假設抽出位置  
又為3，則實際抽出的號碼為10)，再將本次的最後一個位置(9)的資料，放到位置3中(原位置3的資料已被抽出)

以此類推，如下圖示：

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 位置 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| 號碼 | 1 | 2 | ~~3~~ | 4 | 5 | 6 | 7 | 8 | 9 | 10 |

10

第1次:抽位置3 ，號碼3

第2次:抽位置3 ，號碼已變為10

ch12-04.php

<?php

//設定號碼陣列

$nu=array(0,1,2,3,4,5,6,7,8,9,10);

//進行 6 次抽籤

for($x=1;$x<=6;$x++)

{

$k=rand(1,11-$x);

$ans[]=$nu[$k]; //將抽出陣列位置中的號碼，依序放到答案陣列中

$nu[$k]=$nu[11-$x]; //將最後一個位置的資料，移到剛被抽出的位置中

}

抽到的數字將會被最後一個數字取代，接著每一次做就減少抓取範圍1也就是說第一次會抓到陣列10的位置但二次就只會抓到陣列9的位置了。

Echo "抽出的號碼如下(1~10)(不可重覆)：<br>";

For($x=0;$x<6;$x++)

{

Echo $ans[$x]." 、 ";

}

?>

單元作業1：設計一個樂透電腦選號網站，如下圖示：(lerto1.php)



　　　　　　　　　　　　　　點選項目後

不同數字方式轉換函數

|  |  |
| --- | --- |
| 函數指令 | 功能說明 |
| bass\_convert(字串,來源,目地) | 由一種進位方式改成另一種進位方式 |
| bindec(字串) | 二進位換成十進位 |
| decbin(數值) | 十進位換成二進位 |
| decoct(數值) | 十進位換成八進位 |
| dechex(數值) | 十進位換成十六進位 |
| hexdec(數值) | 十六進位換成十進位 |
| octdec(數值) | 八進位換成十進位 |

例：設計一個程式，輸入十進位數，送出後會在網頁上出現轉換成二、、十六進位後的數值。

ch12-05.htm

<html>

<head>

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

<title>輸入十進位數值</title>

</head>

<body>

<h2>請輸入整數 :</h2>

<form name="form1" method="post" action="5-2-2.php">

<p>

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

</p>

<p>

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

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

</p>

</form>

</body>

</html>

ch12-05.php

<html>

<head>

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

<title>不同進位方式的轉換</title></title>

</head>

<body>

<h2>不同進位方式的轉換</h2>

<?php

//取得表單傳遞過來的數值

$numb=$\_POST[number];

echo "輸入的十進位資料 => ".$numb."<p>";

//判斷輸入的是否是數值資料

if(is\_numeric($numb)){

echo "轉成二進位表示 => ".base\_convert($numb,10,2)."<p>";

echo "轉成八進位表示 => ".decoct($numb)."<p>";

echo "轉成十六進位表示 => ".dechex($numb)."<p>";

}else{

echo "<font color='red'>所輸入的不是數值資料請重新輸入!</font>";

}

?>

</body>

</html>

**字串函數**

常用字串函數

|  |  |
| --- | --- |
| 字串函數指數 | 功能說明 |
| strlen(“字串”) | 傳回字串的長度 |
| strtoupper(“字串”) | 將字串中所有的英文字母改為大寫 |
| strtolower(“字串”) | 將字串中所有的英文字母改為小寫 |
| ucfirst(“字串”) | 字串中的第一個字如為英文，改為大寫 |
| ucwords(“字串”) | 字串中的所有英文單字的第一個字改為大寫 |
| str\_pad(“字串”,長度[,”新字串”[,方式]]) | 以指定的字串和長度，將字串加長。 方式可以右側（STR\_PAD\_RIGHT）  左側（STR\_PAD\_LEFT）  兩側（STR\_PAD\_BOTH） |
| str\_repeat(“字串”,次數) | 重覆顯示字串多次 |
| str\_shuffle(“字串”) | 將字串隨機重新排列 |
| str\_word\_count(“字串”[,1或2]) | ( )傳回字串中所用的單字數目  (1 )會同時以陣列方式顯示每一個單字  (2)同1，陣列索引是單字在原字串的對應位置 |
| strcmp(“字串1”,”字串2”) | 比較兩字串的大小 |
| trim(“字串”) | 去除字串前後的空白字元 |

※ 上面所列之指令，大部份適用在英文環境，只有最後二項在中文資料環境中也有可能用到。

企業可能會因為這些空格而多寄送簡訊等

我們先以下面的例子做說明：

　　$na1=”陳大同＂

　　$na2=” 陳 大 同 “

上面這二個字串資料對電腦而言是不相同的，因為「空白」也是一項資料。但有時資料輸入人員再key資料時，常會在無意中按了「空白鍵」，進而造成電腦在比對資料時，產生誤判；因此，去除資料前後或中間的空白符號指令是必須的。

※去除中間的空白可用「尋找/取代」指令功能來完成。

例:：先設定一個含有很多單字的字，再利用strlen函數顯示字串的長度，再用strtouppre及strtolower函數將字串改成大寫及小寫，最後以str\_word\_count函數來顯示陣列每一個單字。

ch12-06a.php

<html>

<head>

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

<title>常用字串函數(一)</title>

</head>

<body>

<?php

//原始字串

$info="Returns string with all alphabetic characters converted to uppercase.";

echo "原字串 : ";

echo $info."<br/>";

echo "此字串長度 = ".strlen($info)."<br/>";

echo "<p>全部改成大寫:<br/>";

echo strtoupper($info);

echo "</p><p>全部改成小寫:<br/>";

echo strtolower($info)."</p><p>";

//將字串依單字轉存成陣列

$result=str\_word\_count($info,1);

//列印陣列

print\_r($result);

?>

</body>

</html>

例：設定一個字串，以str\_pad函數來擴充字串長度並以str\_shuffle函數隨機排列字，最後以str\_repeat函數建立一個新字串。

ch12-06b.php

<html>

<head>

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

<title>常用字串函數(二)</title>

</head>

<body>

<?php

//原始字串

$info="Holiday";

echo "原字串 : ".$info;

//在前後加入"!",使字串長度延伸到10

echo "<p>以!符號延伸字串長度到10 : ";

echo str\_pad($info,10,"!",STR\_PAD\_BOTH);

//隨機重排字串

echo "</p><p>隨機重排字串 : ";

echo str\_shuffle($info);

//以Day重覆三次建立新字串

echo "</p><p>以Day重覆三次建立新字串 : ";

echo str\_repeat("Day",3);

?>

</body>

</html>

例：設定一前後填入空白之字串，再用trim函數將字串前後空白去除，用strlen函數比較前後長度變化。

ch12-06c.php

<html>

<head>

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

<title>常用字串函數(二)</title>

</head>

<body>

<?php

//原始字串

$info=" Holiday ";

echo "原字串 : ".$info;

//傳回字串長度

echo "<p>字串長度為 : ";

echo strlen($info);

//去除字串前後空白字元

$info1=trim($info);

echo "</p><p>原字串去除前後空白 : ".$info1;

echo "<p>字串長度為 : ";

echo strlen($info1);

?>

</body>

</html>

尋找與取代字串內容

|  |  |
| --- | --- |
| 函數指數 | 功能說明 |
| strchr(“字串”,”字元”) | 傳回字串中，從設定字元之後的內容 |
| strstr(“字串”,”字元”) | 同上 |
| stristr(“字串”,”字元”) | 同上，但本函數會區分大小寫之不同 |
| str\_replace(“尋找”,”替代”,”目標字串”) | 在目標字串中，將尋找字串換成替代字串 |
| str\_ireplace(“尋找”,”替代”,”目標字串”) | 同上，但本函數會區分大小寫之不同 |

例：設定兩個字串，一個將Valentine取代成Father， 另一個是擷取email帳號中@和之後的所有資料。

ch12-07.php

<html>

<head>

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

<title>尋找與取代字串</title>

</head>

<body>

<?php

//原字串

$info="Happy Valentine's Day!";

echo "原字串 : ".$info;

echo "<p>取代後的字串 : ";

echo str\_replace("Valentine","Father",$info);

//另一字串

$mail="service@hotmail.com";

echo "</p><p>原email : ".$mail;

echo "</p><p>擷取@之後的字串 : ";

echo strstr($mail,"@");

?>

</body>

</html>

字串分割與合併

|  |  |
| --- | --- |
| 函數指數 | 功能說明 |
| str\_split(“字串”[,長度]) | 依指定長度將字串分割後再存入陣列，長度預設為1 |
| explode(“分隔符號”,“字串”[,數量]) | 依指定的分隔符號分割字串後存入陣列，數量是定義陣列的大小 |
| implode(“分隔符號”,“字串陣列”) | 以指定的分隔符號將來源字串陣列中的所有資料，合併成一個新字串 |
| join( ) | 同上 |

例：設定一個字串資料，分別以explode函數及str\_split函數，將資料分割後儲存到陣列中。另外設定一個陣列，以implode函數將陣列的資料合併成一個新字串。

ch12-08.php

<html>

<head>

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

<title>字串分割與合併</title>

</head>

<body>

<?php

//原始字串

$info="4.0(compatible; MSIE 6.0; Windows NT 5.1; SV1; .NET CLR 1.1.4322)";

echo "原始字串 : ".$info;

//以;為分隔,將資料分割放入陣列

$arr\_info=explode(";",$info);

echo "<p>explode分割後的陣列資料 : <br/>";

print\_r($arr\_info);

//以大小為10字元來分割資料

$arr\_info=str\_split($info,10);

echo "</p><p>str\_split分割後的陣列資料 : <br/>";

print\_r($arr\_info);

//原始陣列

$info2=array("Microsoft","Internet","Explorer","6.0");

echo "<p>原始的陣列資料 : ";

print\_r($info2);

//以空格合併陣列中的資料

echo "<p>合併後的字串資料 : ".implode(" ",$info2);

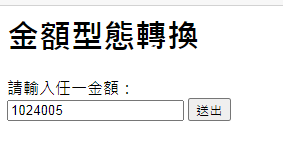
?>

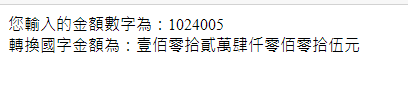
</body>

</html>

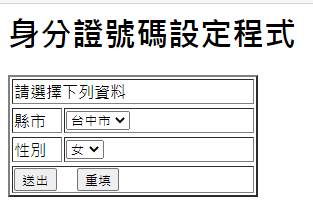
資料分割與合併是工作中常見的工作項目，我們以下列練習來讓同學更加熟練此一指令功能：

單元作業2：輸入數字金額，輸出大寫的國字金額，如下圖：(ex12-02.htm,ex12-02.php)





單元作業3：身分證號碼產生程式：輸入出生地(縣市)及性別，後台自動產生一  
 組身分證號碼，如下圖：（Idask.php，idcrea.php）





**提示：**

**身分證號碼產生規則如下：**

第一碼英文字代表該組號碼配賦時所屬的縣市﹝非出生地﹞，  
第二碼數字1為男生2為女生，  
第3碼至第10碼為任意一串數字，  
但含英文字之全部號碼可用公式加以核算該組號碼是否正確，

例如R123456783，R=25，

檢查公式是：2+(5\*9)+(1\*8)+(2\*7)+(3\*6)+(4\*5)+(5\*4)+(6\*)3+(7\*2)+(8\*1)=167，

上面的公式老師寫錯了需要相反的看\，由右邊到左邊。

其167再除以10求餘數結果。

其餘數結果的個位數為7以10減去得3(檢查碼)。

其餘數結果的個位數為0時，則得0(檢查碼)。  
  
**A=10 B=11 C=12 D=13 E=14 F=15 G=16 H=17 J=18 K=19 L=20 M=21 N=22**

**P=23 Q=24 R=25 S=26 T=27 U=28 V=29 W=32 X=30 Y=31 Z=33 I=34 O=35**

A=10 台北市 J=18 新竹縣 S=26 高雄縣B=11 台中市 K=19 苗栗縣  
T=27 屏東縣 C=12 基隆市 L=20 台中縣 U=28 花蓮縣D=13 台南市  
M=21 南投縣 V=29 台東縣E=14 高雄市 N=22 彰化縣 \* W=32 金門縣  
F=15 台北縣 \* O=35 新竹市 X=30 澎湖縣G=16 宜蘭縣 P=23 雲林縣  
Y=31 陽明山 H=17 桃園縣 Q=24 嘉義縣 \* Z=33 連江縣 \* I=34 嘉義市 R=25 台南縣

輸出格式化字串

用echo或print所顯示的資料是以「資料的原有格式」來顯示。而如果要用指定的格式來顯示資料的話，就必須用printf( )函數，指令格式如下：

可以更改所要印出的資料

printf(“轉換格式參數”,字串1[,字串2…]);

sprintf( )與printf( )相反，是傳回一個格式化的字串，其指令格式與printf( )相同，如下：

sprintf(“轉換格式參數”,字串1[,字串2…]);

有點看不懂的地方

說明：1.轉換格式參數是以百分比符號『％』當起始字元，若兩個字串有不同的格式要求時，轉換格式也要有兩個以百分比符號「％」所引導的字串。

2.百分比符號後面，依序(若需用到)可以輸入下面這些選項參數：

|  |  |  |
| --- | --- | --- |
| 參數 | | 功能說明 |
| + 或 - | | 代表正數或負數。(省略表正數) |
| 單引號「’」 | | 表示要擴充字串。「’」後接要擴充的字元 |
| - | | 表示資料要靠左對齊。(省略表示靠右) |
| 指定擴充字串大小 | | 接在「’」之後的數字 |
| 「.n」 | | 指定浮點表示法時要顯示的小數位數 或 科學記號表示法總共要顯示的位數 |
| 指定要顯示的格式參數 | b | 將數值轉換成二進位格式顯示。 |
| c | 將數值轉換成ASCII碼。 |
| d | 將數值轉換成有正負號的十進位格式。 |
| u | 將數值轉換成無正負號的十進位格式。 |
| Ee | 將數值轉換成科學記號表示法。 |
| f | 將數值轉換成浮點數字型態表示法。 |
| o | 將數值轉換成八進位格式顯示。 |
| s | 將數值轉換成字串格式顯示。 |
| x | 將數值轉換成十六進位小寫格式顯示。 |
| X | 將數值轉換成十六進位大寫格式顯示。 |

例：設定一個含有數字和文字的字串$info，利用sprintf函數，將字串轉換成 科學記號 和 浮點數值型態 顯示。再設定一個字串$message，利用print函數以指定的格式擴充$message，再以字串格式輸出$info字串。

ch12-09.php

<html>

<head>

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

<title>輸出格式化字串</title>

</head>

<body>

<?php

//原始字串

$info="5600 ml";

echo "原始字串 : ".$info;

//轉換成數值並以科學記號表示3位數值來顯示

$num=sprintf("%.3e",$info);

echo "<p>轉換成科學記號表示 : ".$num;

//轉換成數值並以浮點小數兩位來顯示

$num=sprintf("%.2f",$info);

echo "<p>轉換浮點數表示 : ".$num;

//另一個訊息字串

$message="每天需要的水份大於";

echo "<p>";

//%'.-25s設定message靠左,以 ..擴充到25位元

//%s設定info以字串顯示

printf("%'.-35s%s",$message,$info);

?>

</body>

</html>

正規表示法

正規表示法最主要在用於比對字串資料是否符合規定，如判斷「身分證號、E\_MAIL帳號、密碼…」等。

PHP提供ereg( )及eregi( )兩個正規化比對字串函數指令來進行字串比對的工作，指令格式如下：

Ereg(“樣本”,”來源字串”[,陣列]);

Eregi(“樣本”,”來源字串”[,陣列]);

說明：1.兩者差別在於eregi函數，不會去分辨英文字母的大小寫。(差一個i差很多)

2.格式中如果加入陣列，系統會依據樣本的每一個子字串比對來源資  
　　　　料，每符合一個，就將其放入陣列中，

如果不包含陣列，系統僅做單純比對，再傳回比對結果(True或False)

3.樣本的表示方法：是用一些控制變化字元來表達，相關控制字元列表於下：

中括號外的字元

|  |  |
| --- | --- |
| 字元 | 說明 |
| ^ | 比對字串的開頭 |
| $ | 比對字串的結尾 |
| **.** | 符合任何字元，但不括新行 **\n** |
| | | 或(二選一) |
| ? | 代表比對成功是0 或1 次 |
| \* | 代表比對成功是0 或多 次 |
| + | 代表比對成功是1 或多 次 |
| [ ] | 定義字元集 |
| ( ) | 定義子樣本 |
| { } | 指定出現的次數 |

中括號內的字元

|  |  |
| --- | --- |
| 字元 | 說明 |
| ^ | 否定類別 |
| - | 指定範圍 |
| **[0-9]** | 一個介於0-9的數字 |
| [^0-9] | 一個非數字的字元 |
| [a-z] | 一個介於a-z的小寫字元 |
| [A-Z] | 一個介於A-Z的大寫字元 |
| [a-z,A-Z] | 英文字元 |
| {n} | 剛好出現n次 |
| {n,} | 出現n次以上 |
| {n,m} | 至少n 次，最多m次 |

常用樣本

|  |  |
| --- | --- |
| 字元 | 說明 |
| [[:alpha:]] | 任意的英文字母 |
| [[:alnum:]] | 任意的英文字母或數字 |
| **[[:digit:]]** | 數字 |
| [[;space:]] | 任何空白字字元 |
| [[:lower:]] | 小寫英文字母 |
| [[:upper:]] | 大寫英文字母 |
| [[:punct:]] | 半形的英文標點符號 |
| [[:xdigit:]] | 十六進位數字 |

例：設定一個身份證字號陣列和一個E-mail帳號陣列，用正規表示式來判定每一個陣列元素的資料格式是否正確。

ch12-10.php

<html>

<head>

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

<title>正規表示式</title>

</head>

<body>

<?php

//二個原始字串陣列

$id=array("f123456789","p561098789",1288);

$mail=array("service@hotmail.com.tw","linda@123454","1234@hinet.com");

//定義樣式

$ptn1="[a-z]{1}[1-2]{1}[0-9]{8}";

for($a=0;$a<count($id);$a++){

//每個元素一一比較身分證字號

if(ereg($ptn1,$id[$a])){

echo $id[$a]." => 格式正確<p>";

}else{

echo $id[$a]." => 格式不正確<p>";

}

}

//找尋開頭可以是任何英文或數值,接著有一個@其後還有一個點

$ptn2="^[[:alnum:]]+@[[:alnum:]]+\.";

for($a=0;$a<count($mail);$a++){

//每個元素一一比較email帳號

if(ereg($ptn2,$mail[$a])){

echo $mail[$a]." => 格式正確<p>";

}else{

echo $mail[$a]." => 格式不正確<p>";

}

}

?>

</body>

</html>

※在 PHP 5.3 以後，使用 ereg 、 eregi 函式會出現這個函式即將被廢棄的提示。

而 PHP 官網也告知： eregi() is deprecated as of PHP 5.3.0. preg\_match() with the i (PCRE\_CASELESS) modifier is the suggested alternative.

建議用 preg\_match 函式來取代 eregi 函式的使用。

那麼如果想使用 ereg 、 eregi 函式，又不想在新版 PHP 上修改已寫好的程式，有什麼方法呢？

加入以下程式，可在新版 PHP 沒有 ereg 、 eregi 函式的狀況下，自動以 preg\_match 來取代。

//////////////////////重新定義 ereg 與 eregi 函數 ////////////////////////////////////////

if (!function\_exists('ereg'))

{

function ereg($pattern, $string, $regs)

{

return preg\_match('/'.$pattern.'/', $string, $regs);

}

}

if (!function\_exists('eregi'))

{

function eregi($pattern, $string, $regs)

{

return preg\_match('/'.$pattern.'/i', $string, $regs);

}

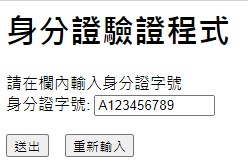
}

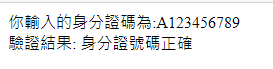
/////////////////////////////////////////////////////////////////////////////////////////

單元作業4：設計一身分證號碼驗證程式，輸入身分證號，

i.驗證格式是否正確 ii.驗證號碼是否符合身分證號產生規則

如下圖：（Idin.php，Idche.php）





單元作業5：中職商城現有會員帳號如後「cjk1029,taj3345,a98uto,lin10win,….」，  
 設計一會員「登入/註冊」程式，功能如下：(畫面及資料內容可自行設計)

i. 點選會員註冊，進入註冊頁面

ii. 如不符合，則出現訊息「您還未註冊，請先註冊加入成為會員」

進入註冊 回上畫面

iii. 會員登入帳號符合則進入商城購物畫面

中職商城 中職商城購物中心

帳號：

密碼：

登入 重設

會員註冊

歡迎加入中職商城

請填寫申請帳號密碼

帳號： （1~10位英數字）

密碼： （1~10位英數字，第1個字為英文）

登入 重設

XXXX(帳號)會員您好：

歡迎加入XX俱樂部

請詳實填寫您的基本資料如下

姓 名：

生 日 年 月 日

性 別：○男 ○女

身分證號：

教育程度：

電 話：

E-mail ：

住 址：

登入 重設

單元作業5-1：註冊畫面分二階段，第一階段：輸入「帳號／密碼」，帳號須與現  
 有會員帳號逐一比對，若有重覆，則須重新設一組「帳號/密碼」重新比  
 對。若没有重覆，則進入第二階段：填寫個人基本資料。(參考上圖)

身分證驗證步驟

|  |  |
| --- | --- |
| 正  規  表  示  法 | 1. 驗證長度   If(strlen(字串)!=10){長度不符} |
| 2拆解字串 A123456789  $da[0]=”A”,$da[1]=1,$da[2]=2,….. |
| 1. 驗證第一碼   If((($da[0]>=”a”)and($da[0]<=”z”))or(($da[0]>=”A”)and($da[0]<=”Z”)))  {符合} |
| 4驗證第2碼 邏輯同上 |
| 5驗證3~10碼 邏輯同上 |
|  | 6.驗證運算規則 |

日期與時間函數

日期和時間也是程式常見問題，同樣的PHP也提供了一些常見日期時間函數，列表如下：

|  |  |
| --- | --- |
| 函數指令 | 功能說明 |
| checkdate(m,d,y) | 驗證是否為日期格式(傳回true或false) |
| date(format[,timestamp]) | (時間資料)以指定的格式傳回日期資料  (若省略，系統會以本機日期時間為資料) |
| getdate(timestamp) | (時間資料)以指定的格式傳回日期陣列資料(舊式下方資料喔!) |
| gmdate(format[,timestamp]) | 同上，但傳回的為格林威治時間 |

.

說明：1.format是字串格式，必須包含在「”」中，相關格式字元表列如下

|  |  |
| --- | --- |
| 字元 | 功能說明 |
| a | 表示以am或pm顯示上下午 |
| A | 表示以AM或PM顯示上下午 |
| d | 日期以2位數顯示，個位數前補0 |
| D | 星期資料以英文前三字母表示 |
| F | 以完整的月分英文名稱顯示 |
| g | 以12小時顯示鐘點 |
| G | 以24小時顯示鐘點 |
| h | 以12小時且用2位數顯示鐘點 |
| H | 以24小時且用2位數顯示鐘點 |
| i | 以00到59顯示分鐘 |
| l | 星期資料以完整英文字母顯示 |
| m | 以01~12顯示月份 |
| M | 以三個英文字母顯示月份 |
| n | 以1~12顯示月份 |
| s | 以00~59顯示秒數 |
| S | 月份的日數加上st(nd、rd、th…) |
| t | 給定月份應用的天數 |
| w | 以數字0~6顯示星期 |
| y | 以兩位數字顯示年份 |
| Y | 以四位數字顯示年份 |
| z | 顯示一年中的第幾天 |

說明:2.getdate( )傳回陣列資料的索引編號值為

|  |  |
| --- | --- |
| 索引編號值 | 儲存資料 |
| “seconds” | 秒數 |
| “minutes” | 分鐘數 |
| “hours” | 小時數 |
| “mday” | 月份中的第幾天 |
| “wday” | 星期(0~6) |
| “mon” | 月份(1~12) |
| “year” | 年份(數字) |
| “yday” | 年份中的第幾天 |
| “weekday” | 星期(以英文) |
| “month” | 月份(以英文) |

例：以getdate( )函數取得系統，並由其中找出年份、月份和天的資料，並以設定的日期格顯示系統日期。

ch12-11.php

<html>

<head>

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

<title>顯示日期</title>

</head>

<body>

<?php

//取得系統的日期資料

$info=getdate();

//取得陣列中的對應資料

$m=$info['mon'];

$d=$info['mday'];

$y=$info['year'];

//驗證是否為合法日期格式

if(checkdate($m,$d,$y)){

echo "這是合法的日期格式!<br/>";

echo "年份是 => ".$y."<br/>";

echo "月份是 => ".$m."<br/>";

echo "日期是 => ".$d."<br/>";

}else{

echo "<font color='red'>這不是合法的日期格式!</font>";

}

//顯示當天的月份英文簡稱,日期後加(st,nd),

//四位數年份和星期的英文全名

echo "<p>".date ("M. dS Y l ");

?>

</body>

</html>

時間函數

相關時間函數表列如下：

|  |  |
| --- | --- |
| 函數指令 | 功能說明 |
| time( ) | 傳回自1970/1/1零時到目前的秒數 |
| localtime( ) | 傳回一個包括本地日期與時間的陣列資料 |
| mktime( ) | 取得一個日期的UNIX時間戳記 |
| strtotime( ) | 以英文描述的日期時間資料轉成UNIX時間戳記 |

說明：1. localtime([timestamp[,is\_associative]];

timestamp省略，會以系統時間為傳回資料

is\_associative為一布林值，若設定False，會傳回以數值為陣列索引編號的陣列資料

若設定True，會傳回以字串為陣列索引編號的陣列資料

|  |  |
| --- | --- |
| 陣列索引編號 | 說明 |
| “tm\_sec” | 秒數 |
| “tm\_min” | 分鐘數 |
| “tm\_hour” | 時數 |
| “tm\_mday” | 每月的天數 |
| “tm\_mon” | 月數(0~11) 需要設定+1 |
| “tm\_year” | 年數(由1901年開始計數)/(y-1900) |
| “tm\_wday” | 星期幾 |
| “tm\_yday” | 一年中的天數 |
| “tm\_isdst” | 是否為日光節約時間 |

2. mktime(hour,minute,second,month,day,year,is\_dst);

本指令通常會和date()函數搭配使用

例：以time( )函數取得目前時間，並以localtime( )函數將對應的時間資料存到陣列中，列出陣列的各項索引和陣列值，並以指定的格式顯示資料。再以mktime( )函數設定一個日期和時間，最後以date( )函數顯示所要的格式。

ch12-12.php

<html>

<head>

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

<title>顯示時間</title>

</head>

<body>

<?php

//取得localtime資料

$info=localtime(time(),true);

echo "取得Local時間資料:<br/>";

//列印陣列中的資料

foreach($info as $index => $value)

{

echo "\$info[$index] => ".$value."<br/>";

}

//轉成日期格式

echo "<p>系統日期和時間 : <br/>";

echo ($info["tm\_year"]+1900)."年".($info["tm\_mon"]+1)."月".$info["tm\_mday"]."日<br/>";

echo "星期".$info["tm\_wday"]." => ".$info["tm\_hour"]."點";

//取得UNIX時間戳記並指定顯示格式

$tim=mktime(11,30,20,8,16,2005);

echo "<p>另一筆時間資料 : <br/>";

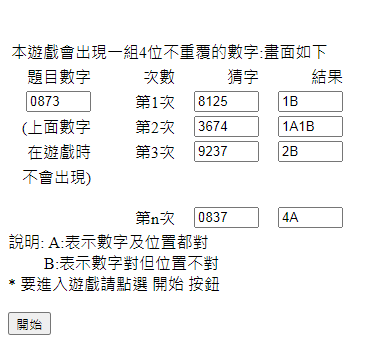
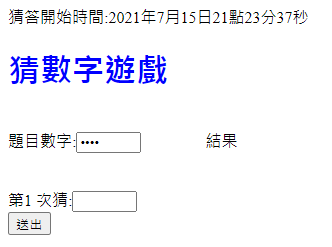
echo date("Y-M-d h:i:s A",$tim);

?>

</body>

</html>

單元作業6：設計一個猜數字遊戲，說明如下：（guess.htm,guess1.php,guess2.php）



提示:可以將3個程式， 

組合成一個程式



**資訊安全概述**

1. 防毀損

電磁破壞、地震、火災、…….

解決方案（研究結論）：備份３份

　　　　且 異地存放：距離３０公里以上

1. 防盜竊

1.加強防盜措施：保險箱、監視器、禁帶隨身碟、notobook、智慧型手機  
 禁止列印文件攜出、……..

2.資料加密：

例: (原始)1234 → (第1段反相) 8765 → (第2段移位) 6587

※資料加密的方法很多，同學可自行上網探索相關資訊

但不管用任何方式，重點為 加密資料 可解密還原 為原始資料