

الفصل العاشر

التعامل مع النصوص

التعامل مع النصوص

كل ماتراه امامك عند تصفح الانترنت من أشكال تفاعليه و قوائم رسومية هو فى الواقع عبارة عن نصوص بلغة HTML لذلك فالتعامل مع النصوص من الامور الهامة فى لغة PHP .

تشكيل النصوص:

حتى الان قمنا باستخدام العبارة print لطباعة مخرجات بسيطة و لكن تأتى الدالة printf() و الدالة sprintf() لتعطى اكثر من الامكانيات كما يتضح من الفقرات القادمة.

الدالة printf():

تستقبل الدالة عدة معاملات الاول دائما هو نص التشكيل و المعاملات الاخرى هى البيانات المراد طبعتها فمثلا اذا اردت طباعة عدد و التعامل معه كرقم عشرى يمكن كتابة الامر:

```
printf("my number: %d",76);
```

و يتم طباعة الجملة:

my number: 76

ونلاحظ استخدام علامة التشكيل المعرفة % و التى تأخذ أشكال عديدة نسردها فى الجدول الاتى :

المعامل	الشرح
d	يقوم بعض المعامل النصى كرقم عشرى
b	يقوم بعرض العدد الصحيح Integer كرقم ثنائى
c	طباعة الحرف المقابل لكود ascii
f	يقوم بعرض العدد الصحيح Integer كرقم ذو دقة مضاعفة Float
o	يقوم بعرض العدد الصحيح كرقم ثمانى (اساسه 8)

التعامل مع النصوص

يقوم بعرض المعامل (سواء أرقام او نصوص) كنص حرفي	S
يقوم بعرض العدد الصحيح Integer كرقم سداسي عشر باحرف صغيره (اساسه 16)	x
يقوم بعرض العدد الصحيح Integer كرقم سداسي عشر باحرف كبيرة	X

و لا نقوم بادراج المعاملات الخاص بتحويل النصوص (%) فقط و لكن يمكننا طباعة حتى وسوم HTML مثل
 كما يتضح من المثال :

```

1: <html dir="rtl">
2: <head>
3: <title>التعامل مع النصوص</title>
4: </head>
5: <body>
6: <?php
7: $number = 543;
8: printf("رقم عشري: %d<br>", $number );
9: printf("ثنائي: %b<br>", $number );
10: printf("رقم ذو دقة مضاعفة: %f<br>", $number );
11: printf("ثماني: %o<br>", $number );
12: printf("حرفي: %s<br>", $number );
13: printf("سداسي عشر (صغير): %x<br>", $number );
14: printf("سداسي عشر (كبير): %X<br>", $number );
15: ?>
16: </body>
17: </html>

```



مثال 2:

في صفحات الويب يتم التعامل مع الالوان كأرقام بالنظام السداسي عشر و يمكن تحويل الارقام العادية بالنظام RGB الى رقم يعبر عن نفس اللون بالنظام السداسي كما يلي:

```
$red = 204;
$green = 204;
$blue = 204;
printf( "#%X%X%X", $red, $green, $blue );
// يقوم بطباعة النص "#CCCCCC"
```

حشو النص:

احيانا نحتاج الى ان يتم طباعة الرقم دائما كحدين فمثلا اذا كان 5 يتم طباعته هكذا 05 و يمكن تنفيذ ذلك كمايلي:

```
printf( "%04d", 36 );
// يتم طباعة "0036"
```

التعامل مع النصوص

ويمكنك ان تقوم بطبع الحشو كحرف اخر غير الصفر عن طريق الحرف ^T كمايلي:

```
printf ( "%'x4d", 36 );  
// يتم طباعة "xx36"
```

بالمثل اذا قمت بادخال المعاملات (1,1,1) في المثال السابق الخاص بالالوان فان القيمة 111 سيتم طباعتها و هذا خطأ لن يقبله المتصفح اذا حولت استخدام هذا اللون فيجب كتابة عبارة printf () كمايلي:

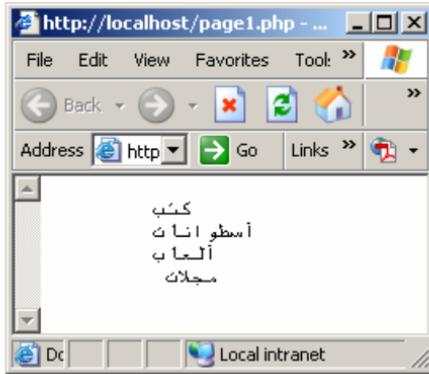
```
$red = 1;  
$green = 1;  
$blue = 1;  
printf( "#%02X%02X%02X", $red, $green, $blue  
);  
// يتم طباعة "#010101"
```

أدراج مسافات:

يمكنك أدراج مسافات خالية في النص المراد عرضه عن طريق حقل يسمى width او "عرض النص" ويتم تحديده بعد علامة "%" مباشرة و يمكن أداء ذلك كمايلي:

```
<html dir="rtl">  
<?  
print "<pre>";  
printf("%20s\n", "كتب");  
printf("%20s\n", "أسطوانات");  
printf("%20s\n", "العباب");  
printf("%20s\n", "مجلات");  
print "</pre>";  
?>  
</html>
```

قمنا هنا بوضع الوسم <pre> حتى يتمكن المتصفح من عرض المسافات كما يتضح من الشكل:



أفترضيا يتم وضع المسافات جهة اليمين و يمكن عكس ذلك الى جهة اليسار اذا قمت بوضع علامة - قبل الرقم هكذا .

```
printf("%-20s\n", "محازاة للييسار");
```

تحديد دقة الرقم:

يمكن للأرقام ذو الدقة المضاعفة أن تكون طويلة ولا يكون من المناسب عرض كل الرقم في المتصفح لذلك يمكن تحديد الدقة لكي يتم تقريب الرقم الى حدود أقل كما يلي:

```
printf( "%.2f", 5.333333 );  
// يتم طباعة الرقم "5.33"
```

لاحظ العلامة العشرية ثم رقم قبل العلامة "%" و يمكن تحديد حقل العرض في عبارة واحدة هكذا "%f4.2" و لا يمكن استخدام معامل الحشو مع حقل العرض في آن واحد و فيما يلي جدول يلخص الترتيب الذي يجب استخدامه عند كتابة نص التشكيل .

مثال	الشرح	أسم التحويل
------	-------	-------------

التعامل مع النصوص

"4"	يحدد عدد الحروف الواجب أدرجها	معامل الحشو
"20"	يحدد عدد المسافات التي يجب أدرجها ضمن مخرجات النص	حقل العرض
"4."	يحدد عدد الحدود الواجب التقريب إليها	معامل الدقة
"d"	يحدد نوع البيانات التي يتم أخراجها	معامل النوع

مثال:

سنقوم الآن بعرض تقرير في صفحة عبارة عن أسعار بعض المنتجات

```

1: <html dir="rtl">
2: <head>
3: <title>لعرض أسعار المنتجات printf() استخدام الدالة</title>
4: </head>
5: <body>
6: <?php
7: $products = Array( "كرسي سفره"=>"222.4",
8: "شمعة"=>"4",
9: منضدة"=>"80.6"
10: );
11: print "<pre>";
12: printf("%-20s%23s\n", "المنتج", "السعر");
13: printf("%'-43s\n", "");
14: foreach ( $products as $key=>$val )
15: printf( "%-20s%20.2f\n", $key, $val );
16: printf("</pre>");
17: ?>
18: </body>
19: </html>

```

قمنا في هذا المثال أولاً بتعريف مصفوفة مترابطة لعرض أسماء المنتجات و أسعارها و أيضاً قمنا بتضمين الوسم <pre> حتى يتعرف المتصفح على المسافات التي نريد أظهارها في الصفحة

في السطر 12 استخدمنا التشكيل :

```
"%-20s%23s\n"
```

ويعبر أول جزء منه "%-20s" عن تشكيل لبيانات حرفية بحيث يتم طباعة البيان في عرض 20 حرف مع محاذاة الى اليسار أما بقية التشكيل "%s23" يستخدم حقل عرضه 23 مع محاذاة جهة اليمين و هذه العبارة هي المسئولة عن طباعة رأس التقرير .

في السطر 13 قمنا بطباعة سطر من الحرف "-" 43 مرة أسفل رأس البيان وتم ذلك عن طريق معامل الحشو الى يقوم بطباعة الحرف "-" الى نص فارغ .

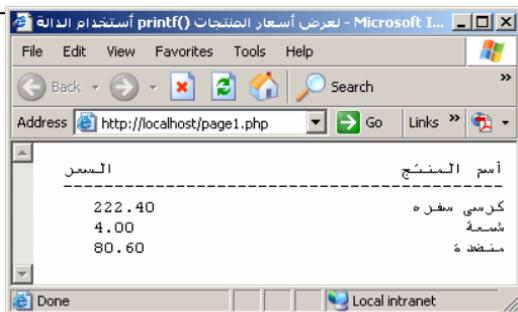
```
printf("%'- 43s\n", "");
```

آخر عبارة printf() في السطر 15 وهي تقع داخل الحلقة foreach التي تقوم بطبع محتويات المصفوفة .

```
printf( "%-20s%20.2f\n", $key, $val );
```

نلاحظ انه يتم طبع أسم المنتج في حقل عرضه 20 مع محاذاة الى اليسار ثم سعره في حقل عرضه 20 مع دقه حدين و محاذاة الى اليمين .

التعامل مع النصوص



الدالة sprintf():

تعمل هذه الدالة مثل الدالة printf() تماما مع فرق واحد وهو انها لا تقوم بعرض المخرجات مباشرة على المتصفح بل تقوم بأرجاع القيمة الى متغير حرفي و يمكن بعد ذلك طبع محتويات هذا المتغير .
مثال:

```
$dosh = sprintf("%.2f", 2.334454);  
print "جنبيه متاحة للصرف $dosh أنت معك";
```

التعرف على النصوص:

تعطى لغة PHP العديد من الأدوات التي تمكننا من معرفة مصدر النص الذي نحصل عليه فقط يكوم من مدخلات مستخدم عن طريق نموذج أو من قاعدة بيانات او حتى من صفحة أخرى .

ملحوظة هامة:

النص الحرفي String هو عبارة عن مصفوفة تتكون من حروف فمثلا العبارة الاتية يمكن الوصول الى الحرف 3 و 8 كمايلي:

```
$test="النص الخاص بي";  
print $test[2];  
print $test[7];
```

معرفة طول النص:

من أهم المعلومات التي يحتاج إليها المبرمج كثيرا عند تعامله مع النصوص هو طول هذا النص ويمكن الوصول الى ذلك عن طريق الدالة (strlen) وهي دالة تقبل معامل حرفي و تقوم بأرجاع عدد صحيح هو عدد الحروف في هذا النص.

ومن الاستخدامات النموذجية لهذه الدالة اختبار أمدخال بيانات الى حقل معين من قبل المستخدم كما يمكن أيضا حه من المثال الأتي :

```
if ( strlen( $membership ) == 4 )
    print "مرحب بك";
else
    print "<P>رقم الدخول يجب أن يتكون من 4 أرقام";
```

البحث عن نص داخل نص:

يمكن الوصول الى أو البحث عن نص عبارة عن جزء من نص أكبر منه عن طريق الدالة (strstr) التي تقبل معاملين هما النص الأساسي أو المصدر و النص المراد البحث عنه و تقوم الدالة بأرجاع true إذا وجدت النص أو false إذا لم تجده

مثال:

```
$membership = "pAB7";
if ( strstr( $membership, "AB" ) )
    print "مرحب بك – نرجو تذكر تجديد الأشتراك";
else
    print "مرحب بك";
```

من المثال السابق يتضح كيف تمكنا من طباعة نص مختلف بناء على قيمة في متغير العضوية

الوصول الى نص داخل نص آخر:

التعامل مع النصوص

بعكس الطريقة السابقة تقدم لنا الدالة `strpos()` إمكانية الوصول الى فهرس (رقم الحرف في المصفوفة الحرفية) أول حرف من النص المراد البحث عنه إذا وجدته أما إذا لم يجده فيقوم بأرجاع القيمة `false` و تقبل الدالة معاملين النص المصدر و النص المراد البحث عنه و معامل ثالث اختياري هو فهرس أول حرف المراد البدء بالبحث عنه .

```
$membership = "mz00xyz";  
if ( strpos($membership, "mz") === 0 )  
    print "hello mz";
```

لاحظ من المثال السابق أننا نريد البحث عن النص "mz" وهو ماستجده الدالة في الموقع 0 أى أول حرف مما يؤدي الى تطوير عبارة IF الى `false` وهذا ما ليس نريده لذلك لكي نقوم بتفادي هذا الاجراء نقوم باستخدام المعامل "===" الذى يتحقق فقط اذا كان طرفى التساوى متساويان فى القيمة و فى النوع أيضا

استخراج جزء من نص:

تقوم الدالة `substr()` باستقبال معاملين هما النص الاصلى أو المصدر و الفهرس الذى يتم عنده البدء فى قطع النص من أوله كما يوجد معامل ثالث اذا تم تحديده يمكن التحكم فى عدد الحروف المقطوعة و هذا المعامل اختياريًا

مثال:

```
$test = "نص حرفى";  
print substr($test,6); // يتم طباعة "فى"  
print substr($test,6,1) // يتم طباعة "ف"
```

أما إذا قمت بتحديد الفهرس الخاص ببداية القطع (المعامل الثانى) برقم سالب فإن بداية القطع تبدأ تبداً من آخر النص بدلاً من أوله

مثال:

```
$test = "embekheet@yahoo.com";  
if ( $test = substr( $test, -9 ) == "yahoo.com" )  
    print "أنت مستخدم بريد الكترونى فى الياهو";
```

else

print "أنت لا تستخدم بريد الياهو";

النصوص المرزمة:

يمكن الحصول على أجزاء من نص واحد كبير عن طريق الدالة strtok() التي تقبل معاملين الأول هو النص الأصلي المراد تقطيعه و الثاني يسمى النص الفاصل و قد يكون أى عدد من الحروف و يتم استخدام هذا النص الفاصل للفصل بين أجزاء النصوص وبعضها و حتى تتمكن من الوصول الى بقية الأجزاء يجب أن تتادى على هذه الدالة فى المرات التالية بدون تحديد النص الأصلي و لكن النص الفاصل فقط و يتم استخدام هذه الدالة بطريقه نموذجيه داخل حلقة تكرار كما يتضح من المثال القادم .

مثال:

```

1: <html dir="rtl">
2: <head>
3: <title>الدالة strtok()</title>
5: </head>
6: <body>
7: <?php
8: $test =
"http://www.egyptbooks.net/qs.xp?OP=dnquery.x
p&ST=MS&DBS=2&QRY=developer+php";
9: $delims = "?&";

10: $word = strtok( $test, $delims );
11: while (is_string( $word ) )
12: {
13: if ( $word )
14: print "$word<br>";
15: $word = strtok( $delims);
16: }
17: ?>
18: </body>
19: </html>

```

هذه الدالة الى حد ما أمكانيّتها ضعيفه لذلك علينا أن نقوم بأستخدامها بطريقه معينه لذلك قمنا أولاً بتخزين النص الفاصل في المتغير `delims$`

ثم نقوم بأستخدام الدالة `()strtok` أول مرة و تخزين الناتج في المتغير `word$` السطر 10

ثم نقوم بأختبار قيمة المتغير `word$` داخل الحلقة `while` إذا كانت `string` فإذا لم تكن نعرف أنها نهاية النص و يجب الخروج من الحلقة

لاحظ أننا في السطر 15 لا نقوم بتحديد النص الأصلي فقط النص الفاصل و ألا سندخل الى حلقة تكرارية لانهاية

الدالة `trim()` و `ltrim()`

عندما تقوم بأستقبال قيمة نصية من مستخدم فأنت لا تعرف إذا كان النص يوجد به مسافات خالية أم لا لذلك يجب أستخدام الدالة `trim()` التي تقوم بحذف المسافات الخالية و `ltrim()` من أول و آخر النص و تقوم هذه الدالة بأستقبال النص المراد تعديله و تقوم بأرجاع النسخة الجديدة منه .

مثال:

```
$text = "\t\t\t نص حرفي به مسافات كثيرة";  
$text = trim( $text );  
print $text;  
// يتم طباعة "نص حرفي به مسافات كثيرة"
```

أما إذا كنت تريد الاحتفاظ بالمسافات في أول النص و ألغاؤها في آخر النص فيمكنك أستعمال الدالة `()chop` تماماً مثل الدالة `()trim`

مثال:

```
$text = "\t\t\t نص حرفي به مسافات كثيرة";  
$text = chop( $text );
```

```
print $text;
// يتم طباعة "نص حرفى به مسافات كثيرة ";
```

و عكس عمل الدالة chop () فإن الدالة ltrim () تقوم بالأحتفاظ بالمسافات الخالية فى أخر النص و تحذفها فى اخره كما يتضح من المثال

مثال:

```
$text = "\t\t\tنص حرفى به مسافات كثيرة ";
$text = ltrim($text);
print $text;
// prints "نص حرفى به مسافات كثيرة";
```

أستبدال نص الدالة substr_replace()

تشابه هذه الدالة عمل الدالة substr () ولكنها تقوم بتعديل النص التى تقوم بأستخراجه و تقبل هذه الدالة ثلاثة معاملات الأول هو النص المراد تعديله و الثانى هو النص المراد أضافته و الأخير هو الفهرس الذى تبدء عنده عملية الأستبدال و تستطيع أن تدرج معامل رابع أختياري و هو طول النص المراد أستبداله

و بذلك تستطيع هذه الدالة أن تقوم بأستبدال النص الأصلي بالنص المعدل بدء من الحرف الذى يحدده معامل الفهرس و نهاية الى أخر النص أو الى عدد الحروف معين إذا تم تحديد المعامل الرابع

مثال:

فى هذا المثال نفترض أن المستخدم عضويته أنتهت لانه يوجد الرقمين 99 فى كود العضوية الخاص به و نريد تجديد العضوية بالنسبة له الى 00 فنستخدم الدالة كمايلى

```
<?
$membership = "mz99xyz";
```

التعامل مع النصوص

```
$membership = substr_replace($membership,  
"00", 2, 2 );  
print "رقم العضوية الجديد: $membership<p>";  
?>
```

استبدال النص بأستخدام الدالة (str_replace):

تقبل هذه الدالة ثلاثة معاملات الأول هو النص المراد البحث عنه لأستبداله والثاني هو النص الجديد الذى سيتم أستخدامه و الأخير وهو النص الأصيل الذى ستم عمل التعديلات به .

مثال:

سنقوم فى النص القادم بتغيير النص "05" الى "2005"

```
$string = "جميع الحقوق محفوظة لسنة 05";  
$string = "الموقع الجديد لدار البراء سنة 05";  
print str_replace("05", "2005", $string);
```

لاحظ ان المتغير \$string يوجد بعده علامة النقطة التى تقوم بأضافة النص الجديد اليه و عدم ألغاء القديم.

تحويل حالة الأحرف:

عندما تريد مقارنة نص يتم أدخاله بواسطة مستخدم و تريد مقارنته بقيمة حرفية فأنت لا تعرف اذا كان المستخدم سيقوم بأدخال حروف كبيرة أو صغيرة لذلك حتى تتم المقارنة بطريقة صحيحة يجب تحويل النصين المراد مقارنتهم الى حروف صغيرة أو حروف كبيرة.

و تستخدم الدالة (strtoupper) لتقوم بتحويل النص الى حروف كبيرة و تقبل معامل واحد وهو النص المراد تعديله و ترجع النص المعدل

مثال:

```
$membership = "mz00xyz";  
$membership = strtoupper($membership);
```

```
print "$membership<P>"; // prints "MZ00XYZ"
```

وبالمثل يمكن تحويل النص الى حروف صغيرة باستخدام الدالة `strtolower()` كما يلي:

```
$my_address = "WWW.EGYPTBOOKS.NET";
$my_address = strtolower( $my_address );
print "$my_address <P>"; // تطبع
"www.egyptbooks.net"
```

هناك دالة أخرى مفيدة و هي الدالة `ucwords()` التي تقوم بتحويل أول حرف من كل كلمة الى حرف كبير .

```
$my_msg = "windows control panel";
$my_msg = ucwords( $my_msg);
print "$my_msg <P>"; // يطبع "Windows Control
Panel"
```

إذا قمت بكتابة نص على الشكل "winDOWs conTRoL paNeL" فإن الدالة السابقة بمفردها لن تحسن شكل النص و لكن يجب استخدام الدالة `strtolower()` أولاً لتحويل كل الحروف الى حروف صغيرة ثم نقوم باستخدام الدالة `ucwords()`

تقسيم النص الى مصفوفة:

يمكنك تقسيم النص الى مصفوفة حرفية عن طريق الدالة `explode()` و هي مشابه لعمل الدالة `strtok()` فهي تقبل معاملين الأول هو النص الفاصل و الثاني هو النص الأصلي الذي تريد تقسيمه و تقوم الدالة بأرجاع مصفوفة حرفية و يمكنك استعمال اي عدد من الحروف كنص فاصل فهذه الدالة تتعامل معهم كلهم كوحدة واحدة عكس النص الفاصل في الدالة `strtok()` فإن كل حرف يحدد كنص فاصل يعامل كوحدة منفصلة

```
$start_date = "1976-06-22";  
$date_array = explode("-", $start_date);  
// $date[0] == "1976"  
// $date[1] == "06"  
// $date[2] == "22"
```

دار الأمل