الفصل التاسع

البيانات الزمنية

دار الجزاء

البيانات الزمنية

تقدم لغة PHP العديد من الادوات التى تمكننا من عمل حسابات زمنية لمختلف المجلات وفيما يلى اهم هذه الادوات.

الدالة **Time()**:

تقوم الدالة time() بارجاع التاريخ والوقت فى صورة رقم صحيح Integer لايعنى الكثير بالنسبة الينا فمثلا اذا كتبت الامر القادم:

```
print time();
```

سيتم ارجاع رقم مثل هذا الرقم 1105177857 و هذا الرقم هو عبارة عن عدد الثوانى التى انقضت منذ منتصف الليل سنة 1970/1/1 وهو ما يسمى ببداية عصر اليونيكس .

ولا تندهش عندما تعرف انه يمكن استخراج جميع المعلومات المتعلقة بتاريخ و وقت معين من الرقم السابق فهناك ادوات اخرى تؤدى هذه الوظيفة و التعامل مع الزمن كرقم واحد يعطينا امكانية القيام بعمليات بحسابية معقدة مثلا اضافة يوم واحد الى تاريخ و ارجاع الناتج فأذا لم تكن هذه الدوال موجودة فيجب ان تقوم بحساب نوع السنة (كبيسة / بسيطة) و ايام الشهور (31/30/29/28) و فيما يلى سنرى بساطة اداء ذلك

استخدام الدالة **getdate()**:

الان لديك الرقم السابق وهو مايسمى timestamp و ستقوم الان بتحويل الرقم السابق الى تاريخ له معنى قبل ان تقوم بعرضة على الزائر و تقوم الدالة getdate() باستخدام معامل واحد قيمتة هى timestap الذى تريد تحويله و تقوم بارجاع مصفوفة مترابطة تحتوى على جميع البيانات الخاصة بهذا الزمن كما يتضح من الجدول الاتى :

المفتاح	***الشرح***	***مثال***

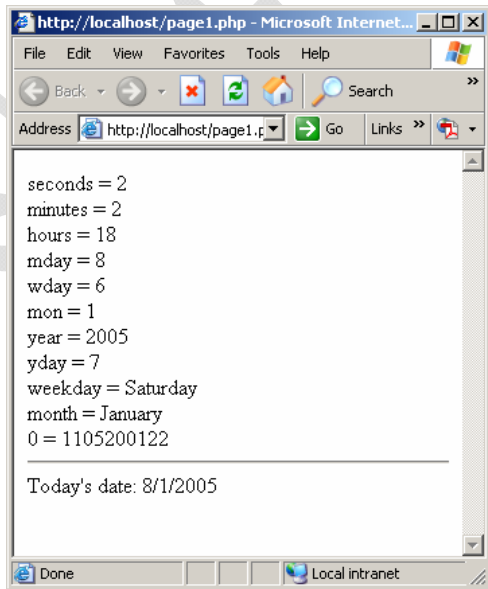
seconds	عدد الثوانى من 0 الى 59	28
minutes	عدد الدقائق من 0 الى 59	7
hours	الساعات من 0 الى 23	12
mday	أيام الشهر من 1 الى 31	20
wday	أيام الاسبوع من 0 الى 6	4
mon	شهور السنة من 1 الى 12	1
year	السنة مرقم ذو اربع حدود	2000
yday	أيام السنة من 0 الى 365	19
weekday	أسم اليوم	Thursday
month	أسم الشهر	January
0	الزمن Timestamp	948370048

وبما ان المخرجات مصفوفة مترابط فيمكننا استخراجها كلها عن طريق الحلقة foreach كما يلى :

```
1: <html dir="rtl">
2: <head>
3: <title>الدالة getdate() </title>
4: </head>
5: <body>
6: <?php
```

```
7: $date_array = getdate(); // لا يتم وضع معامل للدالة لذلك فأن تاريخ اليوم سيتم استخدامه
8: foreach ( $date_array as $key => $val )
9: {
10: print "$key = $val<br>";
11: }
12: ?>
13: <hr>
14: <?
15: print "Today's date:
$date_array[mday]/$date_array[mon]/
$date_array[year]<p>";
16: ?>
17: </body>
18: </html>
```

وتكون نتيجة الكود السابق كمابالشكل:



يتضح من الشكل السابق كيفية اظهار عناصر الزمن المختلفة فى المصفوفة و يمكنك بالطبع الوصول الى اى عنصر لعرضه فى المتصفح لاحظ انك اذا قمت بتحديث الصفحة عن طريق الامر refresh (F5) سترى تغير ارقام الثوانى و ارقام عنصر timestamp كل ثانية .

الدالة **()Date**:

تقتوم هذه الدالة بعرض التاريخ فقط اذا كنت تريد مجرد عرض التاريخ كنص حرفى وتقبل معامل حرفى هو التشكيل المراد اظهاره عند النداء على هذه الدالة و تقبل ايضا اختياريا المعامل timestamp.

التشكيل	*الشرح*	*مثال*
a	للتعبير عن صباحا او مساء بحروف صغيرة	pm
A	للتعبير عن صباحا او مساء بحروف كبيرة	PM
d	رقم اليوم بالنسبة للشهر مع صفر فى البداية	07
D	أسم اليوم بثلاثة حروف	Thu
F	أسم الشهر	January
h	الساعة بالارقام حتى 12 مع صفر فى البداية	12
H	الساعة بالنظام 24 مع صفر فى البداية	12
g	الساعات بالنظام 12 مع عد وجود صفر فى البداية	12
G	الساعة بالنظام 24 مع عدم وجود صفر فى البداية	17
i	الدقائق	47
j	رقم اليوم بالنسبة للشهر مع عد وجود صفر	4
1	أسم اليوم بالكامل	Sunday

L	السنة الكبيسة 1 اذا كانت كبيسة و 0 اذا لم تكن	1
m	رقم الشهر مع صفر فى البداية	01
M	اسم الشهر ثلاثة حروف فقط	Jun
n	رقم الشهر مع عدم وجود صفر فى البداية	6
s	الثوانى	24
U	الزمن timestamp	948372444
y	السنه حدين فقط	95
Y	السنة اربع حدود	2005
z	اليوم من 0 الى 365	40

مثال:

```
1: <html dir="rtl">
2: <head>
3: <title>الدالة  date() </title>
4: </head>
5: <body>
6: <?php
7: print date("m/d/y G.i:s<br>", time());
8: // 1/9/2005 13.27:55
9: print "اليوم هو ";
10: print date("j of F Y, \a\\t g.i a", time());
11: // اليوم هو 9 من يناير 2005
12: ?>
13: </body>
14: </html>
```

الدالة mktime():

تقوم الدوال السابقة بارجاع الزمن الحالى ولكن ماذا اذا كنت تريد العمل مع زمن افتراضى اخر هنا يمكنك استخدام هذه الدالة التى تقوم بارجاع timestamp الذى يمكن وضعه كمعامل للدوال السابقة .

وتستقبل الدالة mktime() 6 معاملات عددية هم على الترتيب : الساعة – الدقائق – الثوانى – الشهر – اليوم – السنة .

مثال:
سنقوم فى هذا المثال باستخدام الدالة mktime() لتقوم بارجاع زمن لتاريخ معين يمكن استخدامه مع الدالة date() كمايلى:

```
1: <html dir="rtl">
2: <head>
3: <title>الدالة mktime</title>
4: </head>
5: <body>
6: <?php
7: // 99/5/1 ارجاع الزمن للتاريخ at 2.30 am
8: $ts = mktime( 2, 30, 0, 5, 1, 1999 );
9: print date("m/d/y G.i:s<br>", $ts);
10: // 05/01/99 2.30:00
11: print "التاريخ هو ";
12: print date("j of F Y, \a\\t g.i a", $ts );
13: // 1 التاريخ هو of May 1999, at 2.30 am
14: ?>
15: </body>
16: </html>
```

وفى هذه الدالة يمكنك الاستغناء عن احد معاملاتها و سيتم استخدام الوقت او التاريخ الحالى و يمكنك ايضا زيادة عدد الساعات مثلا الساعة 25 ليتم اضافة ساعة على التاريخ المحدد.

أختبار تاريخ **()checkdate**:

احيانا نحتاج ان نقوم باستقبال قيمة التاريخ من المستخدم فيمكننا التأكد عن طريق هذه الدالة من ان المستخدم قام بادخال التاريخ الصحيح بالنسبة للسنة المحددة.

مثال عام :

سنقوم الان بتجميع المعلومات السابقة لانشاء نتيجة يستطيع المستخدم من خلالها اختيار السنة و الشهر من خلال قوائم منسدلة بحيث تكون هذه النتيجة اولها سنة 1980 واخرها 2010 و سنقوم باستخدام المتغير العالمى $year و المتغير $month لنقوم بتسجيل اختيار المستخدم و اذا لم يقم المستخدم بكتابة اليوم سنأخذ اول يوم من ايام الشهر .

- سنقوم فى اول خطوة بفحص مدخلات المستخدم بحيث نتأكد من المتغيران $year و المتغير $month يحملان قيمة و يمكن التأكد من ذلك باستخدام الدالة isset() ولكننا سنقوم فى هذا المثال باستخدام الدالة checkdate() كمايلى:

```
1: <?php
2: if ( ! checkdate( $month, 1, $year ) )
3: {
4: $nowArray = getdate();
5: $month = $nowArray[mon];
6: $year = $nowArray[year];
7: }
8: $start = mktime ( 0, 0, 0, $month, 1, $year );
9: $firstDayArray = getdate($start);
10: ?>
```

الكود السابق هو جزء من المثال الكبير الذى سيتم تنفيذه ونقوم فيه اولا باختبار وجود قيم فى المتغييرين المستخدمين فأذا قامت الدالة checkdate() بارجاع القيمة false فيتم الحصول على التاريخ الحالى باستخدام الدالة getdate() التى تقوم بارجاع التاريخ الحالى فى مصفوفة مترابطة وبذلك يمكن استخدام

الدالة mktime() و التى ستقوم بارجاع اول يوم فى الشهر و هو ماسنحتاج اليه لاحقا عن طري المتغير $firstdayarray

بناء نموذج الادخال:

سنقوم الان باستخدام الخاصية select حتى نستطيع ديناميكيا عرض الشهر الحالى او المختار .

```
1: <?php
2: if ( ! checkdate( $month, 1, $year ) )
3: {
4: $nowArray = getdate();
5: $month = $nowArray[mon];
6: $year = $nowArray[year];
7: }
8: $start = mktime ( 0, 0, 0, $month, 1, $year );
9: $firstDayArray = getdate($start);
10: ?>
11: <html dir="rtl">
12: <head>
13: <title><?php print "النتيجة:
$firstDayArray[month]
14: $firstDayArray[year]" ?></title>
15: <head>
16: <body>
17: <form method="post" action="<? print
$PHP_self ?>">
18: <select name="month">
19: <?php
20: $months = Array("يناير", "فبراير", "مارس", "ابريل",
21: "مايو", "يونيه", "يوليو", "أغسطس", "سبتمبر",
22: "أكتوبر", "نوفمبر", "ديسمبر");
23: for ( $x=1; $x <= count( $months ); $x++ )
24: {
```

```
25: print "\t<option value=\"$x\"";
26: print ($x == $month)?" SELECTED":"";
27: print ">".$months[$x– 1]."\n";
28: }
29: ?>
30: </select>
31: <select name="year">
32: <?php
33: for ( $x=1980; $x<2010; $x++ )
34: {
35: print "\t<option";
36: print ($x == $year)?" SELECTED":"";
37: print ">$x\n";
38: }
39: ?>
40: </select>
41: <input type="submit" value="أرسال">
42: </form>
43: </body>
44: </html>
```

يجب بعد ان يختار المستخدم التاريخ ان نقوم بعرض ايام الشهر المختار وفيما يلى الكود الكامل لهذا المثال:

```
1: <?php
2: define("ADAY", (60*60*24) );
3: if (! checkdate( $month, 1, $year ) )
4: {
5: $nowArray = getdate();
6: $month = $nowArray[mon];
7: $year = $nowArray[year];
8: }
9: $start = mktime ( 0, 0, 0, $month, 1, $year );
10: $firstDayArray = getdate($start);
```

```
11: ?>
12: <html dir= "rtl">
13: <head>
14: <title><?php print "النتيجة:
$firstDayArray[month]
15: $firstDayArray[year]" ?></title>
16: <head>
17: <body>
18: <form action="<? print $PHP_SELF ?>"
method="post">
19: <select name="month">
20: <?php
21: $months = Array("يناير", "فبراير", "مارس", "ابريل",
22: "مايو", "يونيه", "يوليو", "أغسطس", "سبتمبر",
23: "أكتوبر", "نوفمبر", "ديسمبر");
24: for ( $x=1; $x <= count( $months ); $x++ )
25: {
26: print "\t<option value=\"$x\"";
27: print ($x == $month)?" SELECTED":"";
28: print ">".$months[$x– 1]."\n";
29: }
30: ?>
31: </select>
32: <select name="year">
33: <?php
34: for ( $x=1980; $x<2010; $x++ )
35: {
36: print "\t<option";
37: print ($x == $year)?" SELECTED":"";
38: print ">$x\n";
39: }
40: ?>
41: </select>
42: <input type="submit" value="أرسال">
```

```
43: </form>
44: <p>
45: <?php
46: $days = Array("الأحد", "الأثنين", "الثلاثاء", "الأربعاء",
47: "الخميس", "الجمعة", "السبت");
48: print "<TABLE BORDER = 1
CELLPADDING=5>\n";
49: foreach ( $days as $day )
50: print "\t<td><b>$day</b></td>\n";
51: for ( $count=0; $count < (6*7); $count++ )
52: {
53: $dayArray = getdate( $start );
54: if ( (($count) % 7) == 0 )
55: {
56: if ( $dayArray[mon] != $month )
57: break;
58: print "</tr><tr>\n";
59: }
60: if ( $count < $firstDayArray[wday] ||
$dayArray[mon] != $month )
61: {
62: print "\t<td><br></td>\n";
63: }
64: else
65: {
66: print "\t<td>$dayArray[mday] </td>\n";
67: $start += ADAY;
68: }
69: }
70: print "</tr></table>";
71: ?>
72: </body>
73: </html>
```

وتكون النتيجة كما بالشكل :

ملاحظات:

- اخر حلقة For هى المسئولة فعليا عن طباعة أرقام الايام فى الجدول (أنظر الشكل) وهى بالطبع مرتبة .

- لقد قمنا بتمهيد متغير $count يحمل عدد مرات التكرار داخل الحلقة و قد حددنا ان تكون نهايته 42 (6*7) حتى تكون الخلايا المطبوعة كافية لكل ايام الشهر .

- فى السطر 54 تم اختبار قيمة المتغير $count اذا كانت ناتج بقية القسمة على 7 تساوى صفر هذا معناه ان الاختبار سيتحقق فقط اذا كان فقط المتغير $count صفر او مضاعفات 7 وبهذه الطريقة يمكننا معرفة متى نقوم بانهاء التكرار او بداية صف جديد .

- فى السطر 56 قمنا بعمل اختبار للشهر الحالى $month مع اول يوم فى الشهر مخزن فى المتغير $dayarray[mon] و بالوصول الى بيان الشهر ومقارنته نعرف متى نصل الى نهاية الشهر وبذلك ننهى الحلقة .

- فى السطر 60 تم معرفة اول يوم من ايام الشهر بحيث يتم الطبع فى الخلية بالجدول عند اول يوم يوافق 1 من الشهر لذلك نقوم بمقارنة المتغير $count مع متغير يحمل اول يوم من الشهر $firstdayarray[wday] فأذا كان أقل منه نعرف اننا لايجب ان نقوم بطبع رقم اليوم فى هذه الخلية .

- فى السطر 64 العبارة else تتحقق عندما يكون المتغير $count يساوى المتغير $firstdayarray[wday] و نبدء فى طباعة ارقام الايام فى الخلية المناسبة من ايام الاسبوع عن طريق المصفوفة المترابطة $dayarray

- واخيرا نقوم بزيادة المتغير $start يوم كامل عن طريق اضافة عدد ثوانى يوم كامل (60*60*24) عن طريق المتغير الذى قمنا بتعريفه فى اول البرنامج ADAY.