

BAB 10

BEKERJA DENGAN FUNGSI WAKTU

PHP memiliki banyak sekali fungsi yang digunakan untuk mengolah data waktu, antara lain untuk memeriksa kebenaran data tanggal, menciptakan data tanggal dan mengkonversi data tanggal. Pada bab ini akan membahas fungsi-fungsi yang digunakan untuk mengolah data tanggal dan waktu dalam sebuah aplikasi program PHP.

10.1 Memeriksa Kebenaran Data Tanggal

Untuk memeriksa kebenaran data tanggal, misalkan data tanggal 29 Februari 2005 itu benar atau tidak, Anda dapat menggunakan fungsi `checkdate`. Bentuk sintaksnya adalah :

`Checkdate(bulan, tanggal, tahun);`

Keterangan :

Bulan, tanggal, tahun adalah format data yang menetapkan data tanggal.

Untuk lebih jelasnya, silakan cermati contoh di bawah ini:

```
<html>
<head>
```

```

<title>Periksa kebenaran data tanggal</title>
</head>
<body>
<p>&nbsp;</p>
<form method="POST" action="periksa.php">
  <p align="center">Pilih Tanggal :<br>&nbsp;<select
size="1" name="tanggal">
  <option value="-">Tanggal</option>
  <?
  for($i=1;$i<32;$i++){
  echo"<option value=\"\$i\">\$i</option>";
  }
  ?>
</select> Bulan&nbsp;<select size="1" name="bulan"
tabindex="1">
  <option value="-">Bulan</option>
  <option value="1">Januari</option>
  <option value="2">Pebruari</option>
  . . .
  <option value="12">Desember</option>
</select> Tahun
  <input type="text" name="tahun" size="8" maxlength="4"
value="2006"><br>
  <br>
  <input type="submit" value="Periksa Tanggal"
name="periksa"></p>
</form>
</body>
</html>

```

Kemudian buatlah sebuah script untuk memeriksa kebenaran format data tanggal pada file tersendiri, yaitu file `periksa.php`.

```

<?
if(isset($tahun)){
if($tanggal=="-"||$bulan=="-"||$tahun==""){
echo"<br><br>Anda belum memilih Tanggal, Bulan dan Tahun.";
}elseif(checkdate($tanggal,$bulan,$tahun)){
echo"<br><br> Tanggal <b>$tanggal - $bulan - $tahun</b>
adalah valid";
}else{
echo"<br><br> Tanggal <b>$tanggal - $bulan - $tahun</b>
adalah tidak valid";
}
}
?>

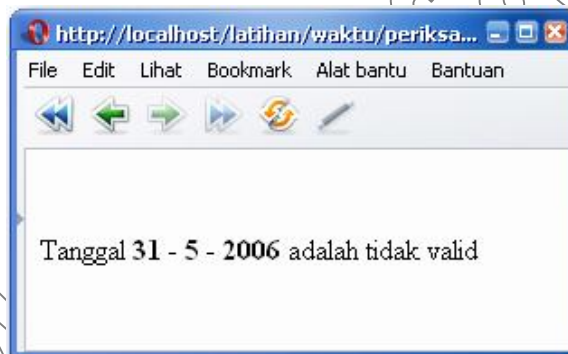
```

Jalankan file **periksa_tanggal.php** di browser dan pilih tanggal 31 bulan Mei dan tahun 2006, seperti pada Gambar 10.1. berikut:



Gambar 10.1 Tampilan formulir periksa tanggal

Untuk memeriksa apakah tanggal tersebut valid atau tidak, akan dilakukan oleh file **periksa.php**. Tekan tombol Periksa Tanggal, maka akan tampil halaman validasi tanggal seperti pada Gambar 10.2. berikut :



Gambar 10.2 Tampilan hasil data tanggal yang telah diperiksa

Coba Lihat, Hasilnya akan ditampilkan bahwa tanggal 31 bulan Mei Tahun 2006 itu tidak ada.

Program yang terdapat dalam file `periksa.php` mempunyai beberapa fungsi pendukung untuk menguji validasi tanggal, yaitu :

isset(\$tahun)

Keterangan :

Pada fungsi di atas adalah untuk memeriksa apakah variabel **\$tahun** sudah terkirim.

Penulisan program di atas berfungsi untuk memeriksa apakah ada variabel tahun atau tidak. Kalau ada, maka pernyataan selanjutnya akan dilaksanakan / dikerjakan.

Berikut ini adalah program untuk menguji variabel Tanggal, Bulan dan Tahun, apakah data yang telah dipilih oleh user bernilai benar atau belum.

```
if($tanggal=="-"||$bulan=="-"||$tahun==""){  
echo"<br><br>Anda belum memilih Tanggal, Bulan dan Tahun."  
}
```

Kemudian apabila data tanggal, bulan dan tahun telah dipilih, maka data tersebut akan diperiksa apakah valid atau tidak dengan menggunakan fungsi **checkdate()** berikut :

```
elseif(checkdate($tanggal, $bulan, $tahun)){  
echo"<br><br> Tanggal <b>$tanggal - $bulan - $tahun</b>  
adalah valid";  
}else{  
echo"<br><br> Tanggal <b>$tanggal - $bulan - $tahun</b>  
adalah TIDAK  
valid";  
}
```

Apabila bernilai **True** maka akan dinyatakan bahwa data tanggal valid dan apabila bernilai **False** maka data tanggal maka dinyatakan bahwa tanggal tidak valid dengan menjalankan fungsi echo untuk mencetaknya pada browser.

10.2 Menciptakan Data Tanggal dan Waktu

Banyak sekali fungsi yang dimiliki oleh PHP untuk mendeklarasikan data tanggal dalam sebuah halaman website. Salah satu fungsi yang sering digunakan untuk menciptakan data tanggal adalah fungsi date.

10.2.1 Fungsi date().

Fungsi date ini digunakan untuk memperoleh tanggal atau jam pada system atau webserver. Bentuk sintaksnya adalah:

Date(String_Format)

102 Oleh : Yeni Cahyono- www.nicdesain.net

Keterangan :

String_format adalah string yang berisikan karakter tertentu yang menjadi format waktu.

Hasil yang diberikan oleh fungsi date ini adalah sebuah string tanggal / waktu sesuai dengan argumennya.

Berikut ini adalah tabel yang memperlihatkan hasil dan string format tanggal sebagai argumennya:

Argumen	Hasil
a	“am” atau “pm”
A	“AM” atau “PM”
d	Tanggal dengan panjang 2 digit (“01”-“31”)
D	Nama hari dengan panjang 3 karakter (“Sun”, “Mon”...)
F	Nama bulan dengan format lengkap. Misal: “January”
g	Jam dengan format 12 jam (“1” -“12”)
G	Jam dengan format 24 jam (“0” -“24”)
H	Jam dengan format dua digit (“00”-“23”)
h	Jam dengan format 12 jam dan panjang 2 digit (“01”-“12”)
i	Menit dengan format 2 digit (“00”-“59”)
J	Tanggal sekarang (“1”-“31”)
l	Nama hari sekarang dengan format panjang lengkap. Misal : “Monday”
L	“0” jika tahun sekarang bukan tahun kabisat “1” jika tahun sekarang tahun kabisat
m	Nama bulan dengan format 2 digit (“01”-“12”)
M	Nama bulan dengan format 3 karakter (“Jan”-“Dec”)
n	Nama bulan dengan tanpa 0 di depan (“1”-“12”)
R	Tanggal dengan format RFC 822

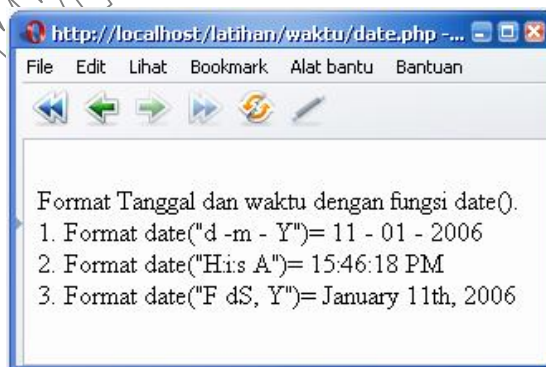
Argumen	Hasil
s	Detik dengan format "00"- "59"
S	Akhiran "th" atau "nd"
t	Jumlah hari dalam sebulan ("28" – "31")
T	Pengaturan timezone pada mesin. Misalnya "MDT"
w	Kode hari dalam angka, misal: "0" = Minggu "1" = Senin, dst.
Y	Tahun dalam format 4 digit. Misal "2003"
y	Tahun dalam format 2 digit. Misal "03"
z	Tanggal dalam tahun (dari "0" sampai "365")

Contohnya coba lihat script di bawah ini:

Berikut adalah contoh penggunaan fungsi date() untuk mendapatkan beberapa bentuk format tanggal dalam sebuah program PHP :

```
<?
// mencetak tanggal dan waktu menggunakan fungsi date()
// date.php
echo"<br>Format Tanggal dan waktu dengan fungsi
date().<br>";
echo"1. Format date(\"d -m - Y\")= ".date("d - m - Y");
echo"<br>2. Format date(\"H:i:s A\")= ".date("H:i:s A");
echo"<br>3. Format date(\"F dS, Y\")= ".date("F dS, Y");
?>
```

Coba dijalankan pada browser, kalau memang penulisan script Anda benar akan tampak seperti pada Gambar 10.3 berikut:



Gambar 10.3 Tampilan data tanggal dengan fungsi date

10.2.2 Fungsi Date dengan Dua Argumen

Fungsi Date ini juga dapat dituliskan dengan menggunakan 2 argumen. Argumen kedua adalah berupa waktu yang akan diformat oleh fungsi Date. Jika argumen ini tidak dideklarasikan, maka waktu yang digunakan adalah waktu yang berlaku sekarang. Bentuk syntaksnya adalah:

Date(String_format, waktu);

Keterangan :

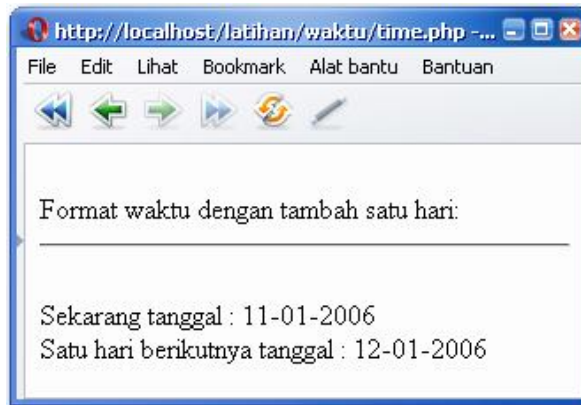
String_format adalah string yang berisikan karakter tertentu yang menjadi format waktu.

Waktu adalah data waktu dalam bentuk angka yang menunjukkan detik.

Berikut ini adalah contoh cara memperlihatkan tanggal berdasarkan data yang dihasilkan oleh fungsi Time. Kemudian data waktu tersebut ditambah dengan jumlah detik dalam satu hari. Jadi data tanggal Anda akan menjadi data tanggal besok. Praktis bukan...

```
<?
// Fungsi time dengan menggunakan dua argumen.
// time.php
echo"<br>Format waktu dengan tambah satu hari:<br><hr>";
$sekarang = time();
$besok = $sekarang + 24 * 60 * 60;
echo"<br>Sekarang tanggal : ". Date("d-m-Y", $sekarang);
$nanti = date("d-m-Y", $besok);
echo"<br>Satu hari berikutnya tanggal : $nanti";
?>
```

Hasil dari penulisan program di atas apabila dijalankan di dalam browser adalah seperti gambar 10.4 berikut ini:



Gambar 10.4 Tampilan data tanggal dengan fungsi date dua argumen

10.2.3 Menciptakan Data Tanggal dan Waktu Dengan Fungsi Getdate

Fungsi **getdate** hampir sama kok dengan fungsi **date**, perbedaannya adalah pada data waktu yang diciptakan. Apabila fungsi **date**, Anda harus menentukan format waktu yang akan diciptakan. Namun pada fungsi **getdate** akan menghasilkan data waktu lengkap dan akan disimpan pada data array dengan index yang sudah ditetapkan oleh script PHP. Fungsi **getdate** ini tidak diikuti dengan format yang diciptakan oleh programmer. Bentuk syntaksnya adalah:

Getdate();

Keterangan :

Fungsi **getdate** tidak diikuti dengan format tertentu, namun akan menghasilkan data waktu lengkap dan dimasukkan pada data array dengan index unik. Berikut adalah index yang digunakan untuk menyimpan data waktu tersebut:

Index	Keterangan	Data yang dihasil
"seconds"	Menghasilkan data detik	0 sampai 59
"minutes"	Menghasilkan data menit	0 sampai 59

Index	Keterangan	Data yang dihasil
"hours"	Menghasilkan data jam	0 sampai 23
"mday"	Menghasilkan data tanggal	1 sampai 31
"wday"	Menghasilkan nama hari dengan format angka	0 = minggu s/d 6 = senin
"mon"	Menghasilkan data bulan	1 sampai 12
"year"	Menghasilkan data tahun	Contoh: 1999 atau 2006
"yday"	Menghasilkan jumlah hari dalam satu tahun	0 sampai 366
"weekday"	Menghasilkan nama hari lengkap	Sunday sampai Saturday
"month"	Nama bulan dengan format nama lengkap	January sampai December
0	Menghasilkan data angka unik	Menghasilkan jumlah detik dari waktu 0:0:0 1-1-1970 sampai sekarang.

Lihat dan cobalah tulis contoh script di bawah ini, Agar Anda lebih mengerti:

```

<?php
// fungsi getdate
// getdate.php
$data=getdate();
echo "Hasil data waktu menggunakan fungsi getdate:<br>";
echo "Data array \$$data[seconds] / detik adalah
$data[seconds]<br>";
echo "Data array \$$data[minutes] / menit adalah
$data[minutes]<br>";
echo "Data array \$$data[hours] / jam adalah
$data[hours]<br>";
echo "Data array \$$data[mday] / tanggal adalah
$data[mday]<br>";
echo "Data array \$$data[wday] / hari angka adalah
$data[wday]<br>";
echo "Data array \$$data[mon] / bulan adalah $data[mon]<br>";
echo "Data array \$$data[year] / tahun adalah

```

```

$data[year]<br>;
echo"Data array \$data[yday] / jumlah hari adalah
$data[yday]<br>;
echo"Data array \$data[weekday] / nama hari adalah
$data[weekday]<br>;
echo"Data array \$data[month] / nama bulan adalah
$data[month]<br>;
echo"Data array \$data[0] / angka unik adalah
$data[0]<br>;
?>

```

Kemudian coba jalankan panggil file untuk menyimpan script di atas pada browser, hasilnya pasti seperti di bawah ini:



Gambar 10.5 Tampilan data tanggal dengan fungsi getdate

10.2.4 Mengambil Data Waktu dengan Mktime

Fungsi **mktime** digunakan untuk menghasilkan data waktu dalam hitungan detik berdasarkan format waktu tertentu. Apabila format tidak dideklarasikan, maka data waktu dalam detik ini dihitung mulai dari jam 00:00:00 tanggal 1 Januari 1970. Bentuk syntaksnya adalah:

Mktime(detik,menit,jam,bulan,tanggal,tahun);

108 Oleh : *Yeni Cahyono- www.nicdesain.net*

Keterangan :

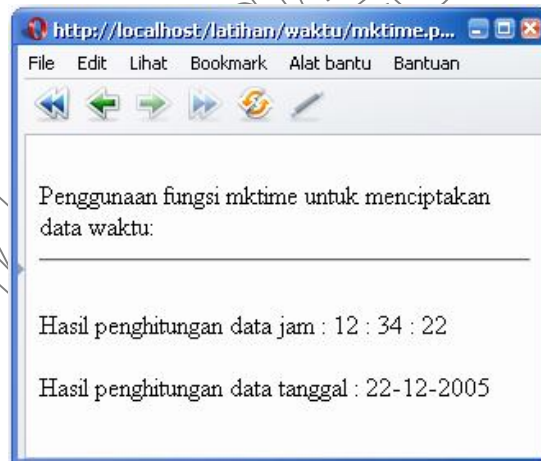
Format data waktu dinyatakan dengan angka, untuk angka tahun dapat ditulis dengan dua angka atau empat angka. Data tahun yang dapat dijalankan pada fungsi ini adalah antara 1901 sampai 2038. Range-nya cukup jauh bukan.

Apabila format tidak ditulis, maka data hasilnya akan menghitung jumlah detik dari jam 00:00:00 tanggal 1 Januari 1970 sampai script ini dijalankan.

Berikut adalah contoh penggunaan fungsi mktime untuk Anda coba:

```
<?
// Fungsi mktime.
// mktime.php
echo"<br>Penggunaan fungsi mktime untuk menciptakan data
waktu:<br><hr>";
$data = mktime(12,34,22,12,22,2005);
$jam = date("H : i : s", $data);
$tanggal = date("d-m-Y", $data);
echo"<br>Hasil penghitungan data jam : $jam<br>";
echo"<br>Hasil penghitungan data tanggal : $tanggal";
?>
```

Hasil dari penulisan program di atas apabila dijalankan di dalam browser adalah:



Gambar 10.6 Tampilan data tanggal dengan fungsi mktime

10.2.5 Menciptakan Data Waktu dengan Strtotime

Untuk menciptakan data waktu berdasarkan data string yang menyatakan data tanggal atau waktu, PHP menyediakan sebuah fungsi yang disebut dengan `strtotime()`;

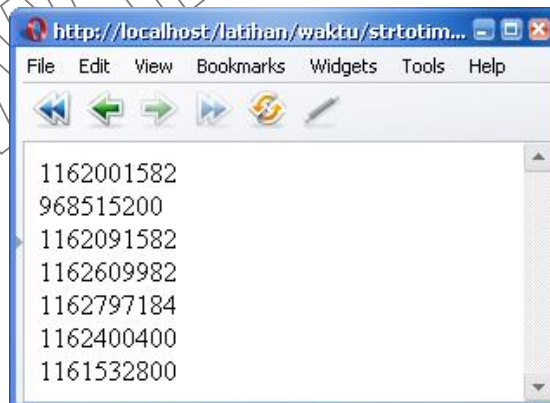
Strtotime("string");

Keterangan :

String adalah data string, namun menyatakan sebuah data waktu atau tanggal. Misalkan "10 January 2004", "13 July 2005", "now". Yang perlu diperhatikan adalah string yang digunakan adalah menyatakan data waktu atau tanggal dalam format Inggris. Berikut adalah contoh script yang menggunakan fungsi `strtotime`:

```
<?php
// strtotime.php
echo strtotime("now"), "<br>";
echo strtotime("10 September 2000"), "<br>";
echo strtotime("+1 day"), "<br>";
echo strtotime("+1 week"), "<br>";
echo strtotime("+1 week 2 (days 4 hours 2 seconds)"), "<br>";
echo strtotime("next Thursday"), "<br>";
echo strtotime("last Monday"), "<br>";
?>
```

Kalau Anda jalankan, maka masing-masing baris perintah dengan fungsi `strtotime` di atas adalah sejumlah angka yang merujuk pada jumlah detik dari jam 00:00:00 tanggal 1 Januari 1970. Mudah bukan.



Gambar 10.7 Tampilan data dengan fungsi strtotime

www.nicdesain.net