

MODUL III

DATE DAN FILE

A. TUJUAN

- Memahami konsep dasar date dan time serta cara penggunaannya.
- Mampu melakukan operasi-operasi dasar pada file dan direktori.
- Mampu memanfaatkan date dan file untuk membuat aplikasi yang sederhana.

B. ALOKASI WAKTU

3 JS (3 x 50 menit)

C. PETUNJUK

- Awali setiap aktivitas dengan do'a, semoga berkah dan mendapat kemudahan.
- Pahami tujuan, dasar teori, dan latihan-latihan praktikum dengan baik dan benar.
- Kerjakan tugas-tugas praktikum dengan baik, sabar, dan jujur.
- Tanyakan kepada asisten/dosen apabila ada hal-hal yang kurang jelas.

D. DASAR TEORI

1. Date dan Time

Informasi mengenai tanggal (*date*) dan waktu (*time*) acapkali diperlukan ketika membangun aplikasi web. Bagaimanapun, terkadang kita perlu melakukan perhitungan tanggal maupun waktu. Sehubungan dengan hal ini, PHP telah menyediakan ekstensi *built-in* yang dapat dimanfaatkan tanpa perlu melakukan konfigurasi tambahan.

2. File System

Aplikasi-aplikasi PHP yang bekerja dengan data permanen akan menggunakan salah satu dari dua pendekatan yang tersedia: file system dan basis data relasional.

Dalam konteks penyimpanan data, file dapat dikatakan sebagai tempat penyimpanan alternatif (selain database). Meskipun demikian, operasi-operasi terhadap file mutlak diperlukan dalam pembuatan aplikasi web. Urgensi pembahasan ini semakin terasa karena PHP memungkinkan URL dan sumber daya jaringan lainnya diperlakukan layaknya file lokal.

E. LATIHAN

1. Operasi Dasar Date dan Time

Pada saat bekerja dengan date dan time, kita tidak akan lepas dari operasi-operasi dasar terkait. Salah satunya yang utama adalah mendapatkan tanggal dan waktu sekarang (*current date* dan *time*).

- Current Date dan Time

Operasi ini sebenarnya bisa dilakukan melalui beberapa pendekatan, namun yang paling umum adalah dengan menggunakan fungsi `date()`.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
lang="en">
<head>
  <title>Info Datetime</title>
</head>

<body>

<?php

// Current date
echo "date('d-m-Y') = ", date('d-m-Y'), '<br />';
// Output (misal): date('d-m-Y') = 31-03-2010

// Current time
echo "date('H:i:s') = ", date('H:i:s'), '<br />';
// Output (misal): date('H:i:s') = 08:10:27

?>

</body>
</html>
```

Selain fungsi `date()`, juga tersedia fungsi-fungsi lain untuk mendapatkan current date dan time, seperti `strftime()`, `idate()`, `getdate()`, dan `localtime()`.

```
echo "strftime('%d-%m-%Y')= ", strftime('%d-%m-%Y'), '<br />';
echo "idate('d') = ", idate('d'), '<br />';
echo "idate('m') = ", idate('m'), '<br />';
echo "idate('Y') = ", idate('Y'), '<br />';

// Mengembalikan array date dan time
print_r(getdate());

// Argumen true mengembalikan array asosiatif
print_r(localtime(time()), true);
```



Lihat dokumentasi/manual PHP untuk mengetahui lebih detail mengenai karakter-karakter yang bisa digunakan pada fungsi `date` dan `time`.

- Informasi Date dan Time Lengkap

Di samping menyediakan informasi-informasi standar, fungsi `date()` juga menyediakan karakter-karakter spesifik untuk mendapatkan gabungan elemen sekaligus.

```
// Format date dan time lengkap (RFC 2822)
echo date('r');
// Output (misal): Wed, 31 Mar 2010 22:27:05 +0800
```

Sejak versi 5, PHP menambahkan karakter `c` yang juga berfungsi untuk merepresentasikan date dan time lengkap.

```
echo date('c');
// Output (misal): 2010-03-31T22:27:05+08:00
```

2. Kalkulasi Date dan Time

Operasi-operasi date dan time umumnya tidak hanya mengacu pada current date atau time. Dengan kata lain, kita mungkin ingin melakukan operasi untuk date dan time yang telah atau belum lewat. Selain itu, tak jarang juga diperlukan kalkulasi date dan time.

- Pembentukan Date dan Time

Setidaknya ada dua pendekatan yang bisa digunakan untuk melakukan pembentukan datetime, yaitu fungsi `time()` dan `mktime()`.

```
// Jam sekarang
echo date('H:i:s'), '<br />';

// Satu jam kemudian
echo date('H:i:s', time() + 3600), '<br />';

// Tanggal sekarang
echo date('d/m/Y'), '<br />';

// Satu hari kemudian
echo date('d/m/Y', time() + 1 * 24 * 3600);
```

Meskipun sama-sama mengembalikan timestamp Unix, namun `mktime()` menyediakan argumen-argumen opsional untuk memudahkan penetapan date dan time.

```
// Menetapkan jam 11, 30 menit, 15 detik
echo date('H:i:s', mktime(11, 30, 15, 0, 0, 0));

// Menetapkan tanggal 29, bulan 3, tahun 2010
echo date('d/m/Y', mktime(0, 0, 0, 3, 29, 2010));
```

- Penambahan dan Pengurangan

Selain fungsi `time()` dan `mktime()`, PHP juga memiliki fungsi `strtotime()` yang dapat digunakan untuk melakukan kalkulasi secara mudah.

```
$kemarin = strtotime('-1 day');
echo date('d/m/Y', $kemarin), '<br />';

$esok = strtotime('+1 day');
echo date('d/m/Y', $esok), '<br />';

// Satu jam kemudian
$tm = strtotime('+1 hour');
echo date('H:i:s', $tm), '<br />';

// Satu minggu kemudian
$mg = strtotime('+1 week');
echo date('d/m/Y', $mg), '<br />';
```

3. Operasi File Dasar

Sehubungan dengan penggunaan file di dalam program, ada beberapa operasi dasar yang perlu sekali diperhatikan.

- Memeriksa Eksistensi File

Ekstensi Filesystem menyediakan fungsi `file_exists()` yang akan mengembalikan nilai `true` jika file ditemukan; sebaliknya, fungsi ini mengembalikan nilai `false`.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Eksistensi File</title>
</head>

<body>

<?php

$fl = 'test.txt';

if (file_exists($fl)) {
  echo 'File ' . $fl . ' ditemukan';
} else {
  echo 'File ' . $fl . ' tidak ditemukan';
}

?>

</body>
</html>
```

Sekarang coba buat file bernama `test.txt` dan letakkan dalam satu direktori dengan file program di atas. Selanjutnya, jalankan file program dan amati hasilnya.

- Mendapatkan Nama dan Ekstensi File

Meskipun ekstensi Filesystem tidak menyediakan fungsi khusus untuk mendapatkan nama atau ekstensi file, namun terdapat sejumlah fungsi yang mencakup informasi tersebut, diantaranya adalah `pathinfo()`.

```
<?php

$fl = 'test.txt';

$info = pathinfo($fl);

echo 'File: ', $info['basename'], '<br />';
// Output: test.txt

echo 'Nama: ', $info['filename'], '<br />';
// Output: test

echo 'Ekstensi: ', $info['extension'];
// Output: txt

?>
```

4. Operasi File Lanjut

Sehubungan dengan fungsionalitas utamanya, ada beberapa operasi file yang disediakan oleh ekstensi Filesystem. Operasi-operasi yang lazim dilakukan meliputi pembuatan, penghapusan, penulisan, dan pembacaan.

- Menulis ke File

Secara garis besar, ada tiga langkah yang diperlukan sehubungan dengan operasi penulisan: membuka file, melakukan penulisan, dan menutup file.

```
<?php

$fl = 'test.txt';

// String yang akan diisikan
$str = "Pemrograman PHP\n";

// Membuka file dengan mode a+
if (!$handle = fopen($fl, 'a+')) {
    echo 'Gagal membuka file ' . $fl;
    exit;
}

// Melakukan penulisan
$wr = fwrite($handle, $str);
if ($wr === FALSE) {
    echo 'Gagal menulisi file';
    exit;
}

echo 'Penulisan berhasil';
```

```
// Menutup resource
fclose($handle);

?>
```

Operasi pembukaan file di atas menggunakan mode a+, yang berarti bahwa baris file akan disisipkan dibelakang dan operasi akan sekaligus menciptakan file baru jika belum eksis. Mode lain yang bisa digunakan antara lain adalah r (read-only) dan w (write-only).

Di PHP versi 5 ke atas, operasi penulisan file dapat dilakukan lebih praktis, yakni dengan memanfaatkan fungsi `file_put_contents()`. Fungsi ini akan melakukan tiga pekerjaan sekaligus: membuka, menulis, dan menutup file.

```
<?php

$f1 = 'test2.txt';

// String yang akan diisikan
$str = "Pemrograman PHP\n";

$write = file_put_contents($f1, $str);

if ($write) {
    echo 'Berhasil, ', $write, ' byte tersimpan';
}

?>
```

▪ Membaca Isi File

Sama seperti penulisan, operasi ini juga dapat dilakukan melalui beragam pendekatan. Apabila Anda ingin membaca seluruh isi file secara praktis gunakan fungsi `file_get_contents()`. Tidak seperti `file_put_contents()` fungsi ini sudah tersedia sejak PHP 4.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
    <title>Membaca File</title>
</head>

<body>

<?php

$f1 = 'test2.txt';

// Membaca seluruh isi file
echo file_get_contents($f1);

?>

</body>
</html>
```

Pembacaan praktis seluruh file juga dapat dilakukan dengan memanfaatkan fungsi `file()`. Fungsi ini identik dengan `file_get_contents()`, kecuali nilai kembaliannya adalah suatu array.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Membaca File</title>
</head>

<body>

<?php

$f1 = 'test.txt';

// Mencetak array baris file
print_r(file($f1));

?>

</body>
</html>
```

5. Operasi Direktori

Seperti halnya operasi file, ada beragam operasi direktori yang lazim dilakukan.

▪ Traversal File

Berbekal fungsi-fungsi yang telah tersedia, kita tidak akan kesulitan melakukan operasi ini. Bagi pengguna PHP 5 ke atas, telah tersedia fungsi `scandir()` untuk memindai file-file di suatu direktori.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en" lang="en">
<head>
  <title>Traversal File</title>
</head>

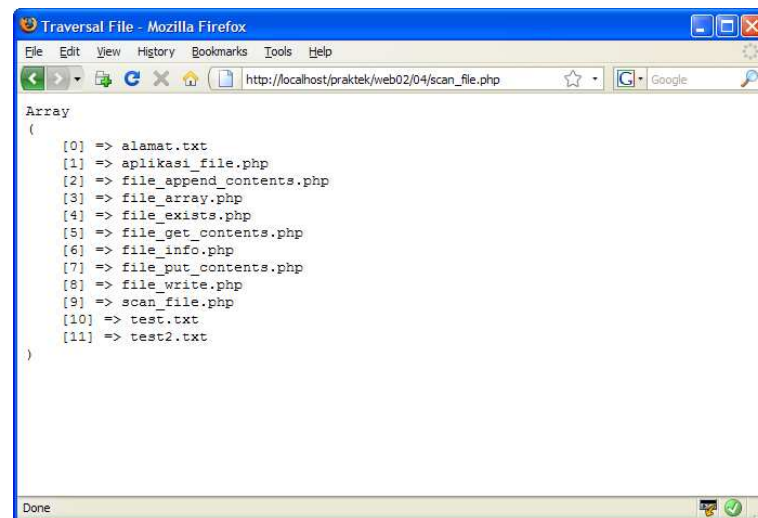
<body>

<?php

echo '<pre>';
// Menguji fungsi traversal
print_r(scanFile('.'));
echo '</pre>';

function scanFile($dir) {
  $arr = array();
  if (is_dir($dir)) {
    $fls = scandir($dir);
    foreach ($fls as $fl) {
      $path = $dir . '/' . $fl;
      // Hanya mengambil file
      if (is_file($path)) {
        $arr[] = $fl;
      }
    }
  }
}
```

```
    }  
    return $arr;  
  } else {  
    return false;  
  }  
}  
  
?>  
  
</body>  
</html>
```



Gambar 1. Traversal file

6. Include/Require File

Mengacu pada struktur aplikasi web, yang lazimnya terdiri dari beberapa file, tidak mungkin kita dapat menghasilkan aplikasi tanpa meng-include file lain. Adapun dalam implementasinya, include bisa dilakukan melalui fungsi `include()` atau `require()`.

Sebagai contoh, kita akan membuat file PHP yang meng-include program `scan_file.php` dan menggunakan fungsi `scanFile()` yang telah terdefinisi.

```
<?php  
  
include 'scan_file.php';  
  
// memanggil fungsi di scan_file.php  
print_r(scanFile('..'));  
  
?>
```


Untuk mencegah include/require ulang, gunakan fungsi include_once() atau require_once(). Kedua fungsi ini memastikan bahwa file yang sama hanya akan di-include sekali.

7. Upload File

Seperti halnya memproses field form, gunakan superglobal ketika ingin memproses file yang di-upload oleh user. Khusus untuk menangani file-file yang di-upload melalui method POST, PHP menyediakan superglobal \$_FILES.

```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html xmlns="http://www.w3.org/1999/xhtml" xml:lang="en"
lang="en">
<head>
  <title>Upload File</title>
</head>

<body>

<form action="<?php $_SERVER['PHP_SELF'];?>"
method="post" enctype="multipart/form-data">

  File: <input type="file" name="fl" />
  <input type="submit" name="submit" value="Upload" />
</form>

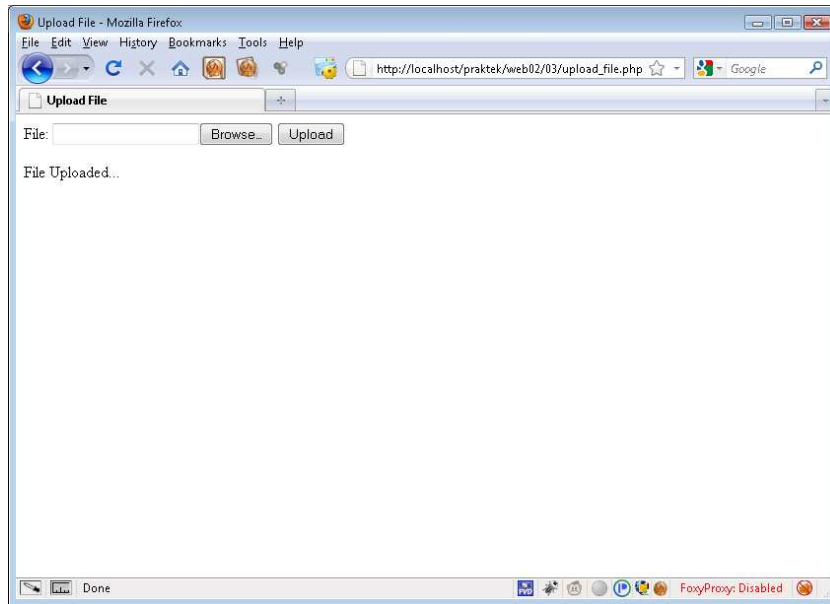
<?php

if (isset($_POST['submit'])
&& isset($_FILES['fl'])) {
  // Lokasi upload, misal di subdir upload
  $dir = './upload/';
  $file = $_FILES['fl']['tmp_name'];
  $name = $_FILES['fl']['name'];

  // Memastikan file tidak kosong
  if (!is_uploaded_file($file)) {
    exit('No file selected...');
  }

  // Upload ke server
  if (!move_uploaded_file($file, $dir.$name)) {
    echo 'Unable to upload file';
  } else {
    echo 'File Uploaded...';
  }
}
?>

</body>
</html>
```



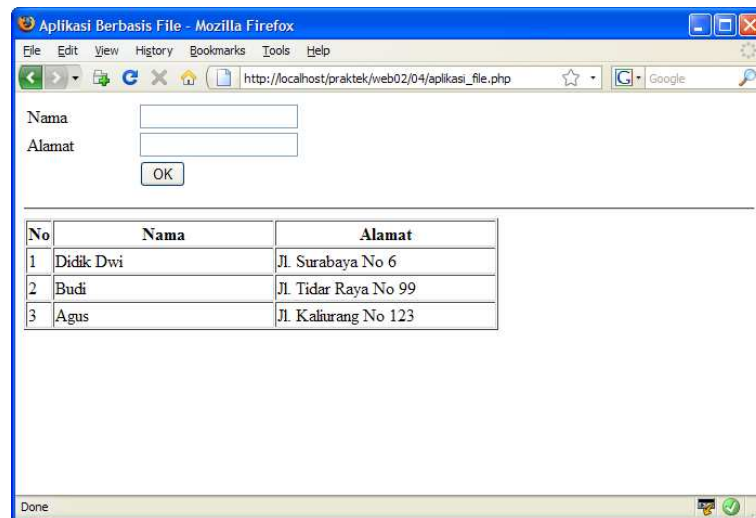
Gambar 2. Upload file

F. STUDI KASUS

1. Modifikasi program upload file dengan menambahkan fitur penyaringan jenis file (hanya mengizinkan file pdf dan zip) dan limitasi ukuran (maks 2MB).

G. TUGAS PRAKTIKUM

1. Buat fungsi countdown untuk menghitung mundur tanggal dan mengembalikan nilai integer selisih hari. Argumen dari fungsi ini adalah string tanggal. Misal tanggal sekarang adalah 10/04/2010 (dari sistem) dan argumen yang diberikan 15/04/2010 maka fungsi akan mengembalikan nilai 5 (selisih tanggal).
2. Buat aplikasi pengolahan data berbasis file seperti diperlihatkan pada Gambar 3. Spesifikasinya: aplikasi mampu mengambil masukan data dan menyimpan ke file, serta menampilkan hasilnya ke browser.



Gambar 3. Aplikasi berbasis file