

File dan direktori

Penanganan File

Dalam management file dan direktori, PHP menyediakan lebih dari 70 fungsi. Beberapa fungsi utama yang berhubungan dengan management file (create, write, append, dan delete)

Membuka dan Membuat File

```
fopen ($namafile, $mode);
```

Keterangan :

\$namafile merupakan nama file yang akan dibuat, sedangkan \$mode merupakan mode akses file.

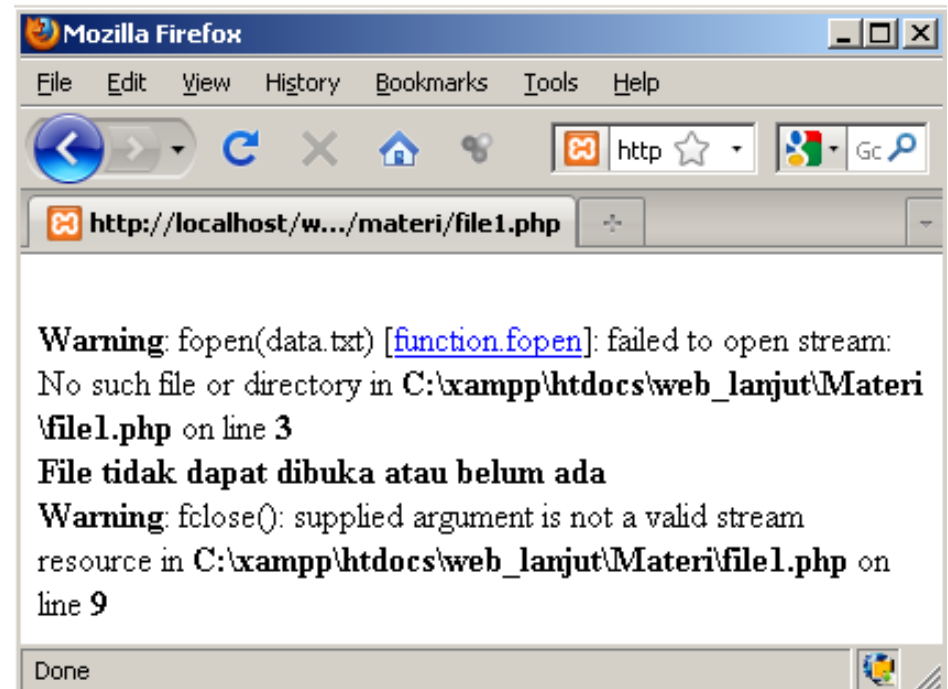
Mode Akses File :

Mode	Keterangan
r	Hanya untuk baca file, pointer berada di awal file
r+	Untuk baca dan tulis file, pointer berada di awal file
w	Hanya untuk tulis file, isi file lama dihapus, jika file belum ada maka akan di-create
w+	Untuk baca dan tulis file, isi file lama dihapus, jika file belum ada maka akan di-create
a	Hanya untuk menambahkan isi file, pointer berada di akhir file, jika file belum ada maka di-create
a+	Untuk membaca dan menambahkan isi file, pointer berada di akhir file, jika file belum ada maka di-create

Mengakses (membuka) file dengan mode "r"

Nama File : File1.php

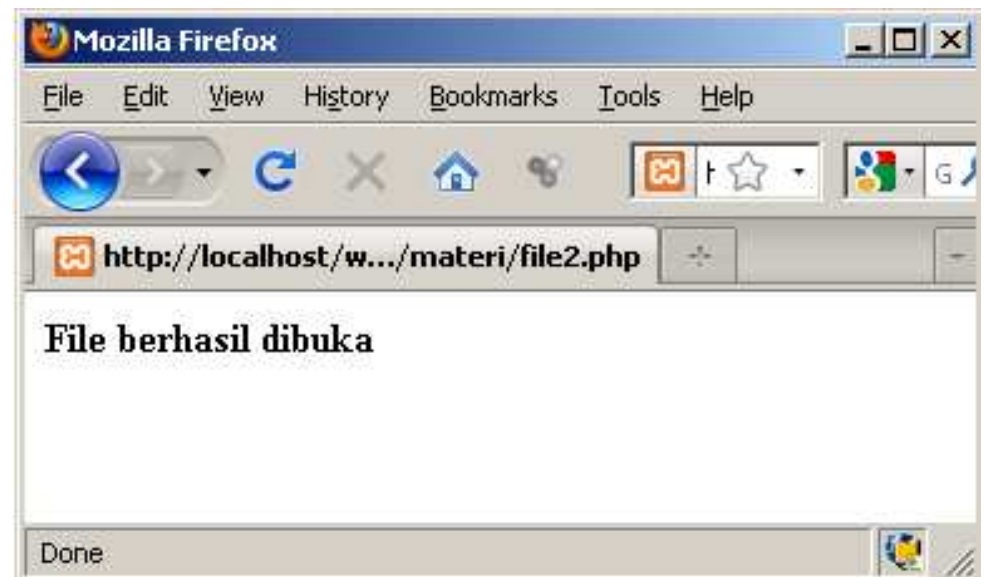
```
<?php
$namafile = "data.txt";
$handle = fopen ($namafile, "r");
if (!$handle) {
echo "<b>File tidak dapat dibuka atau belum ada</b>";
} else {
echo "<b>File berhasil dibuka</b>";
}
fclose($handle);
?>
```



Mengakses (membuka) file dengan mode “w”

Nama File : File2.php

```
<?php
$namafile = "data.txt";
$handle = fopen ($namafile, "w");
if (!$handle) {
echo "<b>File tidak dapat dibuka atau belum ada</b>";
} else {
echo "<b>File berhasil dibuka</b>";
}
fclose($handle);
?>
```



Menulis ke File

```
fwrite ($handle, $isi);  
fputs ($handle, $isi);
```

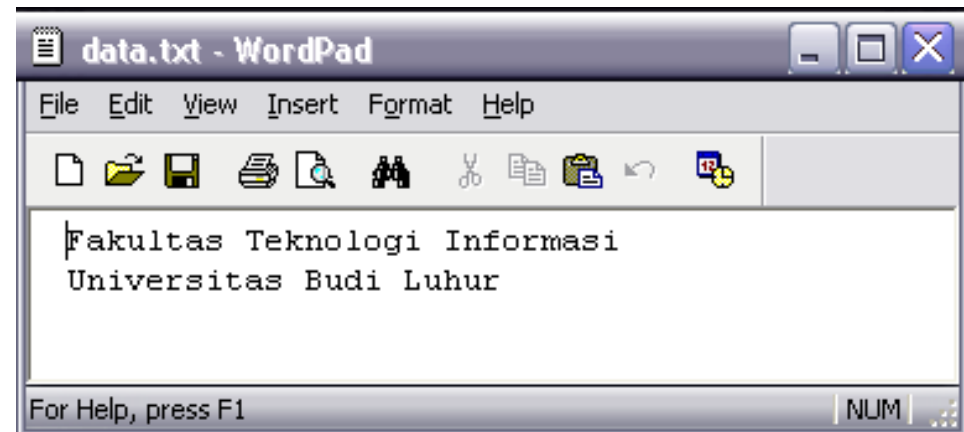
Keterangan :

\$handle merupakan nama file yang akan ditulis, sedangkan \$isi merupakan isi yang akan ditambahkan ke file

Contoh : File3.php

Nama File : file3.php

```
<?php
$namafile = "data.txt";
$handle = fopen ($namafile, "w");
if (!$handle) {
echo "<b>File tidak dapat dibuka atau belum ada</b>";
} else {
fwrite ($handle, "Fakultas Teknologi Informasi\n");
fputs ($handle, "Universitas Budi Luhur\n");
//file_put_contents ($namafile, "Jakarta");
echo "<b>File berhasil ditulis</b>";
}
fclose($handle);
?>
```



Membaca Isi File

```
fread ($handle [, $panjang]);  
fgets ($handle [, $panjang]);
```

Keterangan :

\$handle merupakan nama file yang akan ditulis, sedangkan \$panjang merupakan panjang isi file yang akan dibaca

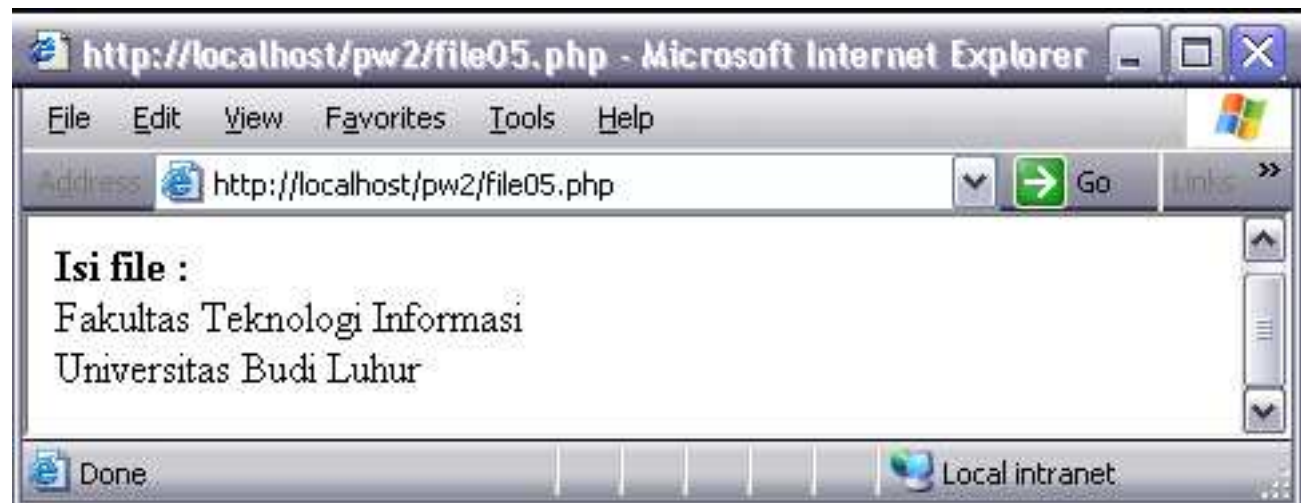
Contoh : file4.php

```
<?php
$namafile = "data.txt";
$handle = fopen ($namafile, "r");
if (!$handle) {
echo "<b>File tidak dapat dibuka atau belum ada</b>";
} else {
$isi = fgets ($handle, 2048);
$isi2 = fread ($handle, 20);
echo "Isi 1 : $isi<br>";
echo "Isi 2 : $isi2<br>";
}
fclose($handle);
?>
```

Membaca isi file baris tiap baris

Nama File : file5.php

```
<?php
$namafile = "data.txt";
$handle = fopen ($namafile, "r");
if (!$handle) {
echo "<b>File tidak dapat dibuka atau belum ada</b>";
} else {
echo "<b>Isi file : </b><br>";
while ($isi = fgets ($handle, 2048)) {
echo "$isi<br>";
}
}
fclose($handle);
?>
```



Membaca isi file baris tiap baris

Nama File : file6.php

```
<?php
$namafile = "data.txt";
$handle = @fopen($namafile, "r");
if ($handle) {
while (!feof($handle)) {
$buffer = fgets($handle, 4096);
echo $buffer."<br>";
}
fclose($handle);
}
?>
```

Menutup File

```
fclose ($handle);
```

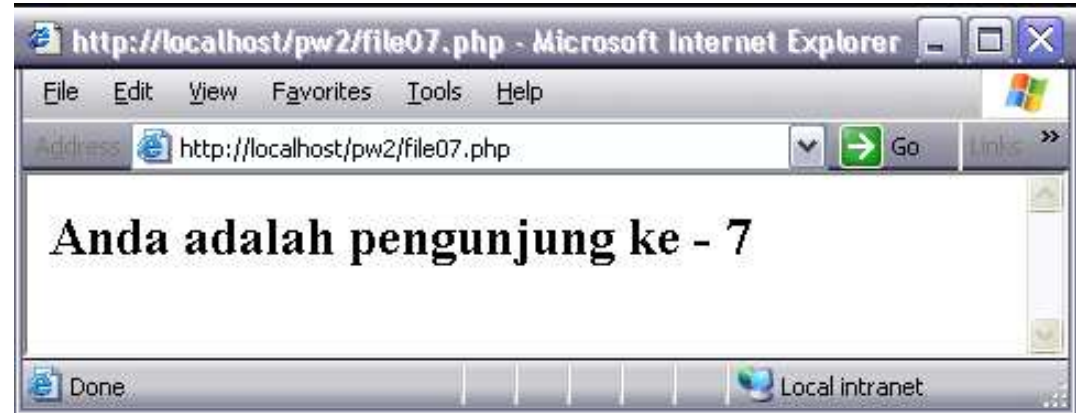
Keterangan :

\$handle merupakan nama file yang akan ditutup.

Aplikasi user counter dengan file

Nama File : file7.php

```
<?php
$counter_file="counter.txt";
if (!file_exists ($counter_file)) {
fopen ($counter_file, "w");
}
$file = fopen($counter_file,"r");
$counter = fread($file,10);
fclose($file);
$counter++;
echo "<h2>Anda adalah pengunjung ke - $counter</h2>";
$file = fopen($counter_file, "w");
fwrite($file,$counter);
fclose($file);
?>
```



Membuat dan Menghapus Direktori

- *membuat direktori baru*

```
mkdir ($dir);
```

Nama File : file8.php

```
<?php
$dir = "images"; //nama direktori
$cek = mkdir ($dir);
if ($cek) {
echo "Direktori <b>$dir</b> berhasil dibuat";
} else {
echo "Direktori <b>$dir</b> gagal dibuat";
}
?>
```

Membuat dan Menghapus Direktori

- *menghapus direktori*

```
rmmdir ($dir);
```

Nama File : file9.php

```
<?php
//buat direktori
$dir = "images"; //nama direktori
//hapus direktori
$del = rmmdir ($dir);
if ($del) {
echo "<br>Direktori <b>$dir</b> berhasil dihapus";
} else {
echo "<br>Direktori <b>$dir</b> gagal dihapus";
}
?>
```


Menampilkan Isi Direktori

- Menampilkan Isi Direktori

`readdir($handle)`

`$handle = opendir($dir)`

Nama File : file10.php

```
<?php
$dir = "images";
if ($handle = opendir($dir)) {
while (false !== ($file = readdir($handle))) {
if ($file != "." && $file != "..") {
echo "$file<br>";
}
}
closedir($handle);
}
?>
```

Mendapatkan Informasi File

- Beberapa Fungsi yang berhubungan dengan informasi dan manipulasi file, terdapat dalam tabel sebagai berikut :

Fungsi	Keterangan
<code>file_exists(\$file)</code>	Memeriksa apakah \$file ada atau tidak
<code>is_file(\$file)</code>	Memeriksa apakah \$file adalah file atau bukan.
<code>is_dir(\$file)</code>	Memeriksa apakah \$file adalah direktori atau bukan.
<code>is_executable(\$file)</code>	Memeriksa apakah \$file termasuk file yang bisa dijalankan secara langsung atau bukan.
<code>is_writable(\$file)</code>	Memeriksa apakah \$file termasuk file yang bisa ditulis atau diedit.
<code>is_readable(\$file)</code>	Memeriksa apakah \$file termasuk file yang bisa dibaca atau tidak.
<code>fileatime(\$file)</code>	Menghasilkan waktu akses terakhir file (unix timestamp).
<code>filectime(\$file)</code>	Menghasilkan waktu pembuatan file (unix timestamp).
<code>filemtime(\$file)</code>	Menghasilkan waktu modifikasi terakhir file (dalam unix timestamp).
<code>filesize(\$file)</code>	Menghasilkan ukuran besar file (dalam byte).
<code>filetype(\$file)</code>	Menghasilkan jenis file.

Memeriksa keberadaan File

- File_exists()

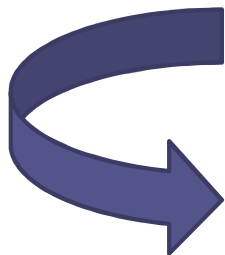
nama file : File11.php

```
<?php
$filename = "data.txt";
if (file_exists($filename)) {
echo "File $filename ADA";
} else {
echo "File $filename TIDAK ADA";
}
?>
```

Mendapatkan beberapa informasi file

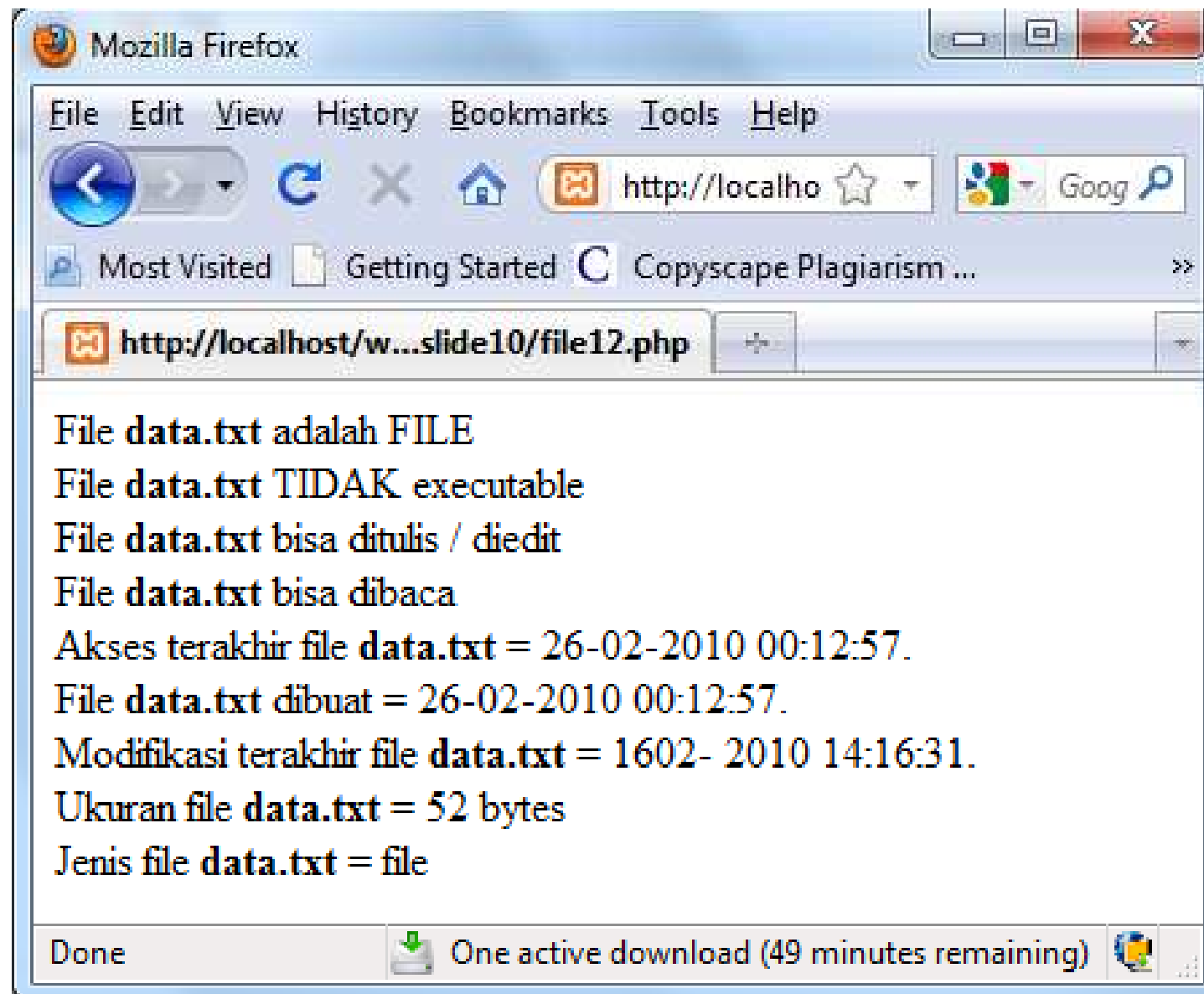
Nama File : File12.php

```
<?php
$file = "data.txt";
if (is_file($file)) {
    echo "File <b>$file</b> adalah FILE<br>";
    //
    if (is_executable ($file)) {
        echo "File <b>$file</b> bisa dijalankan secara langsung
        (executable)<br>";
    } else {
        echo "File <b>$file</b> TIDAK executable<br>";
    }
    //
    if (is_writable ($file)) {
        echo "File <b>$file</b> bisa ditulis / diedit<br>";
    } else {
        echo "File <b>$file</b> TIDAK bisa ditulis /diedit<br>";
    }
}
```



```
//
if (is_readable ($file)) {
echo "File <b>$file</b> bisa dibaca<br>";
} else {
echo "File <b>$file</b> TIDAK bisa dibaca<br>";
}
//
echo "Akses terakhir file <b>$file</b> = ". date("d-m-Y
H:i:s.", filetime($file)). "<br>";
echo "File <b>$file</b> dibuat = ". date("d-m-Y H:i:s.",
filectime($file)). "<br>";
echo "Modifikasi terakhir file <b>$file</b> = ". date("dm-
Y H:i:s.", filemtime($file)). "<br>";
echo "Ukuran file <b>$file</b> = ". filesize($file). "
bytes<br>";
echo "Jenis file <b>$file</b> = ". filetype($file). "<br>";
} else if (is_dir($file)){
echo "File <b>$file</b> adalah DIREKTORI<br>";
} else {
echo "File <b>$file</b> tidak dikenal<br>";
}
?>
```

Hasil : file12.php

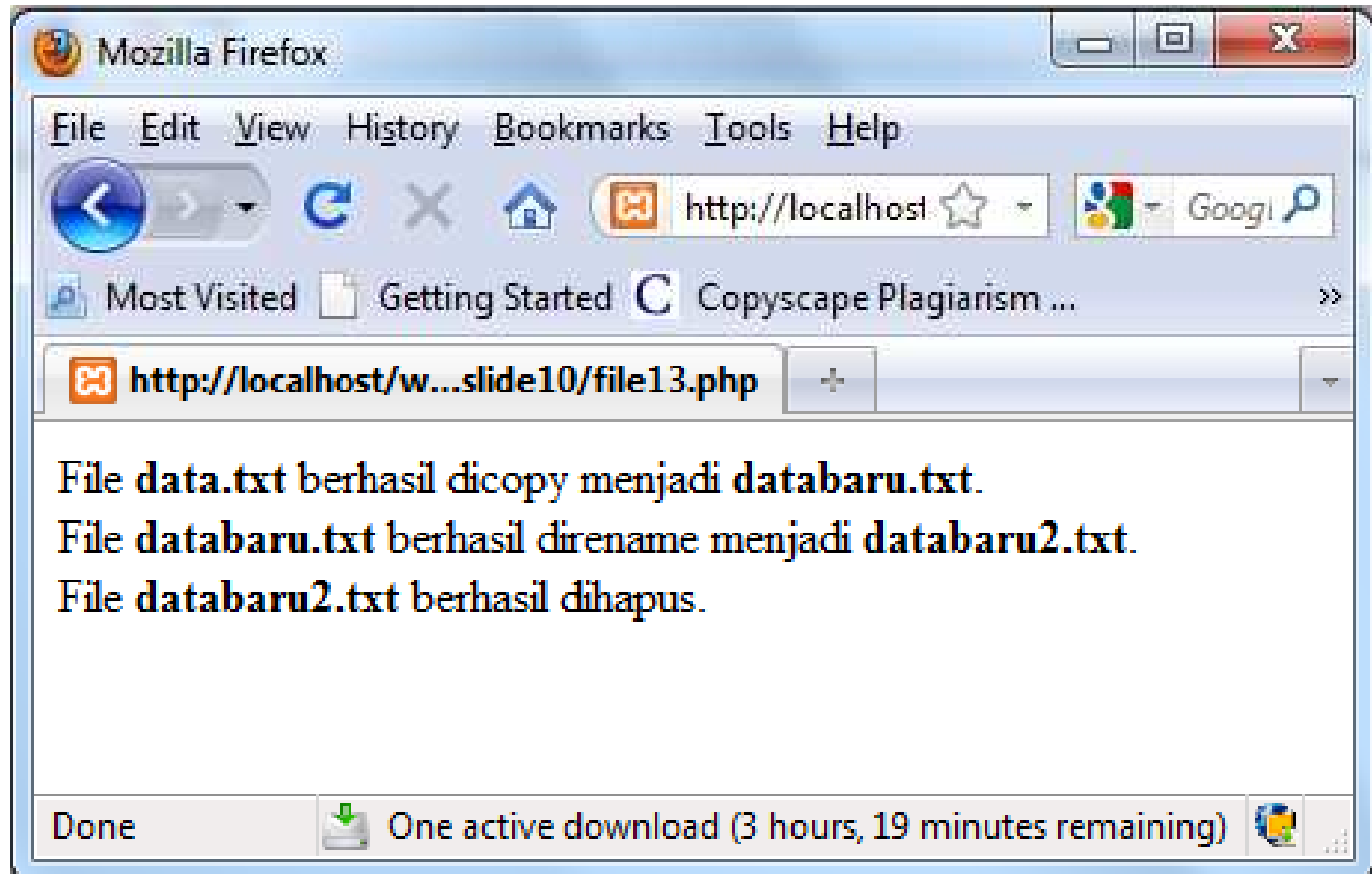


Copy, Rename dan Delete File

Nama file : file13.php

```
<?php
$file = "data.txt";
$filebaru = "databaru.txt";
$filebaru2 = "databaru2.txt";
if (copy ($file, $filebaru)) {
    echo "File <b>$file</b> berhasil dicopy menjadi<b>$filebaru</b>. <br>";
}
if (rename ($filebaru, $filebaru2)) {
    echo "File <b>$filebaru</b> berhasil direname menjadi<b>$filebaru2</b>.
<br>";
}
if (unlink ($filebaru2)) {
    echo "File <b>$filebaru2</b> berhasil dihapus. <br>";
}
?>
```

Hasil: File13.php



Upload File

- PHP, sebagai bahasa server-side-scripting, sangat memungkinkan untuk menangani upload file ke server. Ada beberapa hal yang perlu diperhatikan dalam upload file ini, yaitu :
 1. Pada Form HTML harus ditambahkan atribut : ENCTYPE="multipart/form-data"
 2. Form inputan upload file dapat menggunakan tag <input> dengan value atribut TYPE="FILE".
 3. Untuk menangani inputan, PHP menyediakan sebuah variabel array global yaitu \$_FILES. Index dari variabel ini antara lain :
 - \$_FILES['file']['name'] : Nama asli dari file yang diupload
 - \$_FILES['file']['tmp_name'] : Nama temporary file yang diupload
 - \$_FILES['file']['size'] : Ukuran file asli (dalam byte)
 - \$_FILES['file']['type'] : MIME type file yang diupload
 4. *Destination folder file upload harus bisa writable (bisa diakses), biasanya dengan permission 777 atau 775.*

Menampilkan form upload file.

Nama File : Form Upload.php

```
<html>
<head><title>Upload File</title></head>
<body>
<FORM ACTION="upload.php" METHOD="POST"
ENCTYPE="multipart/form-data">
Upload File : <input type="file" name="file"><br>
<input type="submit" name="Upload" value="Upload">
</FORM>
</body>
</html>
```

Proses upload file

Nama File : upload.php

```
<?php
if (isset($_POST['Upload'])) {
    $dir_upload = "data/";
    $nama_file = $_FILES['file']['name'];
    //
    if (is_uploaded_file($_FILES['file']['tmp_name'])) {
        $cek = move_uploaded_file ($_FILES['file']['tmp_name'],
        $dir_upload.$nama_file);
        if ($cek) {
            die ("File berhasil diupload");
        } else {
            die ("File gagal diupload");
        }
    }
}
?>
```

Hasil :

