

# Bab 03

## Mengenal Smarty Lebih Jauh

### 3.1 Koneksi Database

Untuk menghubungkan smarty dengan database, tidak ada konfigurasi yang berarti. Yah.. koneksi database seperti biasa kita melakukan koneksi database menggunakan php biasa. Yaitu menggunakan perintah:

`mysql_connect`, jika kita ingin menghubungkan dengan database mysql.

`mysql_select_db`, jika kita ingin menghubungkan atau menggunakan dengan database tertentu.

Caranya:

1. Anda buat terlebih dahulu sebuah database di dalam phpmyadmin Anda (<http://localhost/phpmyadmin>), dengan nama database **dbsmarty**.
2. Buat 1 tabel dengan nama **tmahasiswa** dengan spesifikasi field sebagai berikut.

Field	Type	Length	PrimaryKey	Autoincrement
Nim	varchar	10	*	
Nama	varchar	50		
alamat	text			
id_jurusan	Int	11		

3. Buat 1 file php dengan nama **koneksi.php** kemudian simpan di dalam folder kerja Anda (folder smarty). Adapun skripnya sebagai berikut:

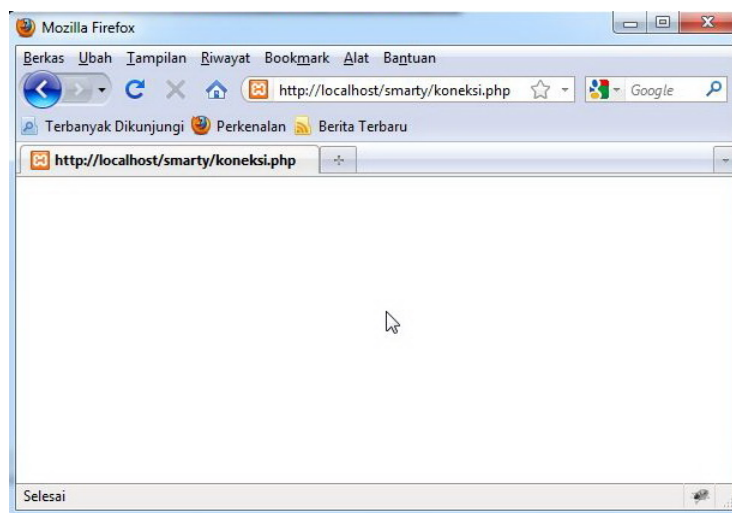
```

<?php
$hostName      = "localhost";
$username      = "root";
$password      = "";
$database      = "dbsmarty";

mysql_connect($hostName,$username,$password) or
die("Koneksi Gagal");
mysql_select_db($database) or die("Database tidak
ditemukan.");
?>

```

4. Buka dan jalankan skrip yang telah dibuat melalui web browser Anda dengan mengetikkan <http://localhost/smarty/koneksi.php>. Jika pada jendela web browser Anda tidak menampilkan apa-apa, maka SELAMAT! koneksi database telah berhasil Anda lakukan.



Gambar 3.1 Hasil skrip koneksi.php

### 3.2 Mengenal Section

Section dalam smarty digunakan untuk mengulang suatu lembaran array yang dihasilkan dari file logika (php). Ibarat jika di file logika kita menggunakan foreach, maka di file layout kita menggunakan section.

Sebagai contoh, penulis akan buat studi kasusnya.

Buat file dengan nama **section.php** dan letakkan dalam folder smarty Anda. Adapun skripnya sebagai berikut:

```
<?php
require('libs/Smarty.class.php');
$smarty = new Smarty;

for ($i=1; $i <= 10; $i++){
    $data_array[] = array( 'no' => $i,
        'nama' => "Agus Saputra",
        'position' => "Lead Programmer",
        'company' => "Asfa Solution");
}

$smarty->assign("data", $data_array);
$smarty->display("section.tpl");
?>
```

#### Penjelasan kode:

- `require('libs/Smarty.class.php');` artinya kita membutuhkan atau memanggil file library smarty, yaitu `Smarty.class.php`.
- `$smarty = new Smarty;` kita buat objek baru dari class smarty. Ini merupakan konsep oop php.
- `for ($i=1; $i <= 10; $i++){` kita lakukan perulangan sebagai ganti dari menampilkan data melalui database. Kita set for sebanyak 10 kali perulangan.
- `$data_array[]` kita buat format array untuk penyimpanan value. Pada `$data_array`, kita berikan fungsi [], karena data yang ditampilkan lebih daripada 1. Adapun value data array itu ada pada nama, position, dan company.
- `$smarty->assign("data", $data_array);` kita masukkan assign hasil dari `$data_array`, ke dalam variabel assign data. Variabel assign inilah nanti yang akan digunakan pada file layout.
- `$smarty->display("section.tpl");` kita akan tampilkan semua skrip yang telah dibuat ke dalam **section.tpl**.

Buat file layout dengan nama **section.tpl** dan simpan ke dalam folder template. Adapun skripnya sebagai berikut:

```
{section name=ct loop=$data}
    {$data[ct].no}. {$data[ct].nama} - {$data[ct].position}
<br>
{/section}
```

#### Penjelasan kode:

- `{section name=ct loop=$data}` kita buat section untuk mengulang suatu data.

Formatnya adalah:

```
{section
  name=ct
  loop=$data
}
```

Section → perintah smarty tpl untuk memulai suatu sesi.

Name → nama section.

Loop → variabel data yang akan di-looping, dalam hal ini kita sudah mempunyai variabel assign dari file php (`$data`).

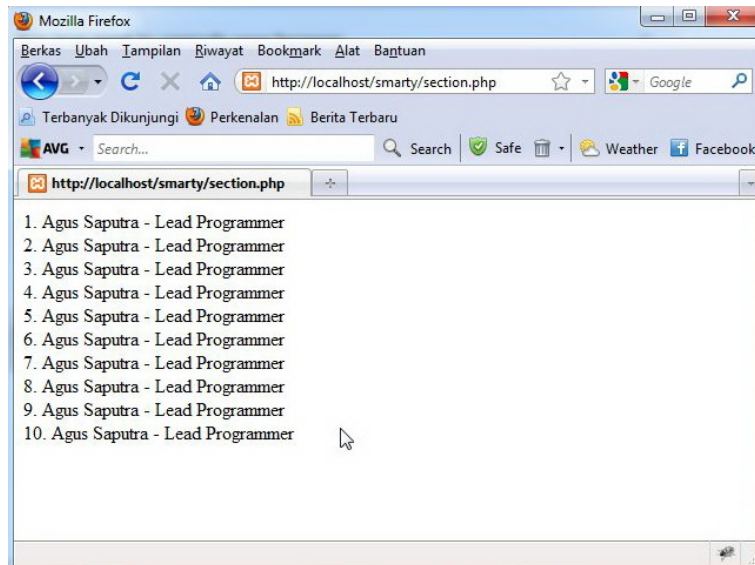
- `{$data[ct].no}` perintah untuk menampilkan data ke halaman web.

Formatnya adalah:

```
{loop[nama section].nama array}
```

- `{/section}` perintah untuk menutup perintah section.

Jalankan skrip yang telah Anda buat melalui web browser dengan mengetikkan <http://localhost/smarty/section.php>, maka akan ditampilkan hasil seperti pada Gambar 3.2.



Gambar 3.2 Hasil skrip section.php

### ***3.3 Section dalam Section***

Seperti dalam subjudulnya saja sudah jelas. Section digunakan untuk memulai suatu sesi perulangan. Di dalam suatu sesi tersebut, ada sesi perulangan lagi. Jadi ganda deh. Penulis ambil contoh, misalnya suatu kali kita dihadapkan oleh suatu data dengan format berikut.

#### **Komputer**

- **Sistem Informasi Nilai Akademik untuk Panduan Skripsi**
- **Web Tips: PHP, HTML5, dan CSS3**

#### **Novel**

- **Poconggg juga Pocong**
- **Ada Apa dengan Cinta**

#### **Majalah**

- **Pulsa**
- **Info Komputer**

Artinya, kita mengulang data kategori, sedangkan di dalam kategori tersebut juga diulang suatu data buku yang merupakan isi kategori. Biasanya ada pada situs seperti kompas.com, detik.com, dan lain-lain, menampilkan list kategori berita dengan isinya yang dibatasi hanya beberapa berita.

Sebagai sampel saja, penulis akan memberikan contoh tanpa menggunakan database.

Buat file dengan nama **section2.php** dan simpan ke dalam folder smarty Anda. Adapun skripnya sebagai berikut:

```
<?php
require('libs/Smarty.class.php');
$smarty = new Smarty;

for ($i=1; $i <= 2; $i++){
    $isi[] = array('spesifikasi' => "PHP");
}

for ($i=1; $i <= 10; $i++){
    $data_array[] = array( 'no' => $i,
        'nama' => "Agus Saputra",
        'position' => "Lead Programmer",
        'company' => "Asfa Solution",
        'spesifikasi' => $isi
    );
}

$smarty->assign("data", $data_array);
$smarty->display("section2.tpl");
?>
```

Untuk penjelasan kodenya hampir sama dengan yang subbab sebelumnya. Hanya saja di sini kita membuat variable array menjadi 2 bagian. Yang pertama untuk menampilkan kategori, yang ke-2 (\$isi) digunakan untuk menampilkan data dari bagian kategori tersebut. Kemudian \$isi kita masukkan jadi 1 ke dalam variabel \$data\_array.

Selanjutnya bisa Anda buat file layout dengan nama **section2.tpl** kemudian simpan dalam folder template. Adapun skripnya sebagai berikut:

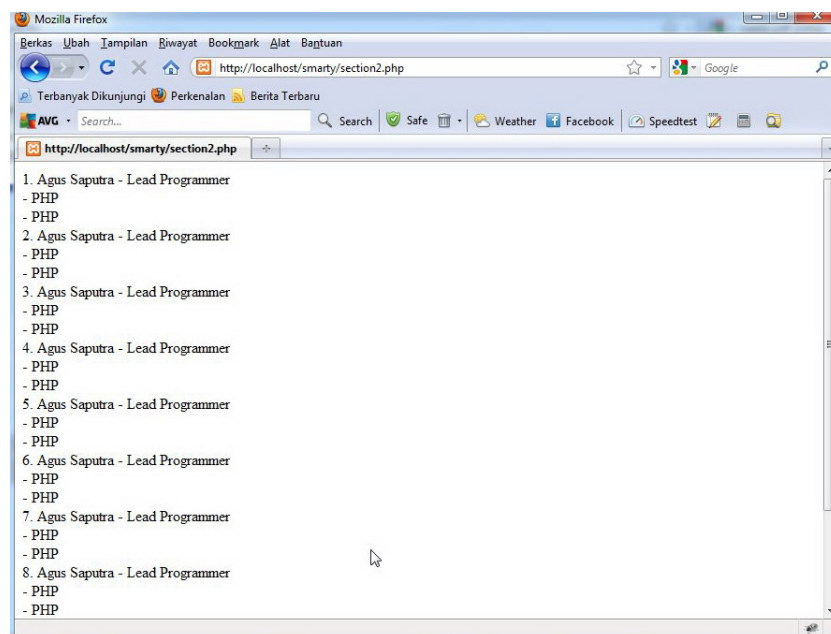
```
{section name=ct loop=$data}
    {$data[ct].no}. {$data[ct].nama} - {$data[ct].position}

    {section name=ct2 loop=$data[ct].spesifikasi}<br>
    - {$data[ct].spesifikasi[ct2].spesifikasi}
{/section}
```

```
<br>
{/section}
```

Mungkin pada skrip `{section name=ct2 loop=$data[ct].spesifikasi}` inilah yang akan membuat Anda bingung. Benar tidak? 😊 nggak usah bingung ... kan data\_array kolom spesifikasi itu kan bersifat array lagi. Jadi, mau nggak mau, ya kita ulang lagi. Coba deh, perhatikan lagi lebih lanjut dan lebih teliti. Nanti juga paham. Di sinilah maksud dari subbab “Array Adalah Hal yang Wajib dalam Smarty”.. 😊

Hasil skrip di atas, akan menampilkan hasil seperti pada Gambar 3.3.



Gambar 3.3 Hasil skrip section2.php

### 3.4 Operasi PHP dalam Smarty

Dalam php, kita mengenal 4 macam operasi, yaitu create, read, update, dan delete. Di mana ke-4 operasi tersebut mutlak harus dikuasai.

- **Create**, yaitu operasi yang digunakan untuk menambah suatu data ke dalam database.
- **Read**, yaitu operasi yang digunakan untuk mengambil data dari database.
- **Edit**, yaitu operasi yang digunakan untuk mengubah data dari database.
- **Delete**, yaitu operasi yang digunakan untuk menghapus data dari database.

### 3.4.1 Tambah Data

Sudah penulis jelaskan masing-masing dari pengertian operasi. Sebaiknya tak usah panjang lebar, penulis akan berikan contoh menambah data menggunakan smarty.

Buat file dengan nama **tambah.php**, dan simpan dalam folder smarty Anda. Selanjutnya ketikkan skrip berikut:

```
<?php
error_reporting(0);
require('libs/Smarty.class.php');
$smarty = new Smarty;

include "koneksi.php";

if ($_POST['simpan'] == 'Save'){
    mysql_query("INSERT INTO tmahasiswa
(nim,nama,alamat,id_jurusan)

VALUES('$_POST[nim]','$_POST[nama]','$_POST[alamat]','$_POST[id_jurusan]')");
    $smarty->assign("success", 1);
}

$smarty->display("tambah.tpl");
?>
```

Untuk masing-masing penjelasan kode, dapat Anda lihat pada penjelasan sebelumnya. Hanya saja pada skrip:

```
if ($_POST['simpan'] == 'Save')
```

artinya jika file php mendapat kiriman parameter POST, maka akan terjadi proses penyimpanan data. Kemudian kita set assign terhadap variabel success. Ini nanti akan digunakan pada file layout-nya.



Cukup jelas bukan? ☺

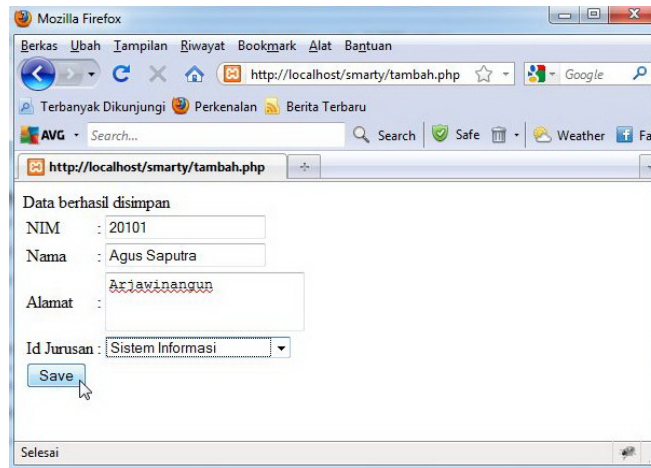
Selanjutnya berikut skrip untuk file layout (**tambah.tpl**):

```
{if $success eq '1'}
    Data berhasil disimpan
{/if}

<form method="POST" action="">
<table>
<tr>
    <td>NIM</td>
    <td>:</td>
    <td><input type="text" name="nim"></td>
</tr>
<tr>
    <td>Nama</td>
    <td>:</td>
    <td><input type="text" name="nama"></td>
</tr>
<tr>
    <td>Alamat</td>
    <td>:</td>
    <td><textarea name="alamat"></textarea></td>
</tr>
<tr>
    <td>Id Jurusan</td>
    <td>:</td>
    <td>
        <select name="id_jurusan">
            <option value="1">Teknik Informatika</option>
            <option value="2">Sistem Informasi</option>
            <option value="3">Manajemen Informatika</option>
            <option value="4">komputerisasi Akuntansi</option>
        </select>
    </td>
</tr>
<tr>
    <td><input type="submit" name="simpan" value="Save"></td>
</tr>
</table>
</form>
```

Pada skrip yang tercetak tebal, merupakan kiriman dari file php. Jika `$success` - nya terdeteksi 1, maka akan ditampilkan “Data berhasil disimpan”.

Jalankan skrip yang telah dibuat melalui web browser Anda dengan mengetikkan url <http://localhost/smarty/tambah.php>. Silakan tambah data sebanyak-banyaknya. Lihat Gambar 3.4.



*Gambar 3.4 Hasil skrip tambah.php*

### 3.4.2 Tampil Data

Setelah data disimpan, kali ini kita buat action untuk menampilkan data.

Buat file dengan nama **tampil.php** dan simpan ke dalam folder smarty Anda. Adapun skripnya sebagai berikut:

```
<?php
error_reporting(0);
require('libs/Smarty.class.php');
$smarty = new Smarty;

include "koneksi.php";

$no = 1;
$sql = mysql_query("SELECT * FROM tmahasiswa ORDER BY nama ASC");
while ($data = mysql_fetch_array($sql)){
    $dataresult[] = array( 'no' => $no,
                          'nim' => $data[nim],
                          'nama' => $data[nama],
                          'alamat' => $data[alamat],
                          'id_jurusan' => $data[id_jurusan]
    );
    $no++;
}

$smarty->assign("data", $dataresult);
$smarty->display("tampil.tpl");
?>
```

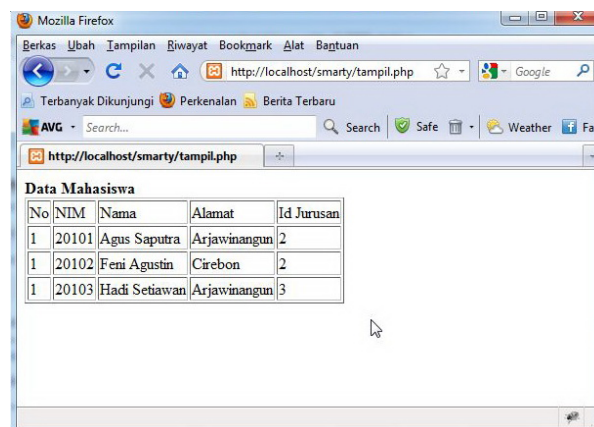
Penulis tidak jelaskan kembali untuk skrip di atas, bisa Anda lihat pada bab-bab sebelumnya.. ^^ . Sayang kertas kalo bahasanya diulang-ulang mulu, kecuali kalo ada skrip baru, baru penulis jelaskan lagi.

Buat file layout dan beri nama **tampil.tpl** dan simpan ke dalam folder template. Adapun skripnya sebagai berikut:

```
<h4>Data Mahasiswa</td>
<table border=1>
<tr>
    <td>No</td>
    <td>NIM</td>
    <td>Nama</td>
    <td>Alamat</td>
    <td>Id Jurusan</td>
</tr>

{section name=hasil loop=$data}
<tr>
    <td>{$data[hasil].no}</td>
    <td>{$data[hasil].nim}</td>
    <td>{$data[hasil].nama}</td>
    <td>{$data[hasil].alamat}</td>
    <td>{$data[hasil].id_jurusan}</td>
</tr>
{/section}
</table>
```

Hasil skrip di atas akan menampilkan hasil seperti pada Gambar 3.5. (<http://localhost/smarty/tampil.php>)



No	NIM	Nama	Alamat	Id Jurusan
1	20101	Agus Saputra	Arjawinangun	2
1	20102	Feni Agustin	Cirebon	2
1	20103	Hadi Setiawan	Arjawinangun	3

**Gambar 3.5 Hasil skrip tampil.php**

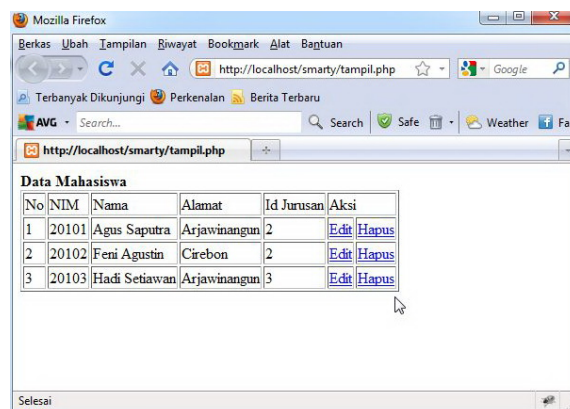
### 3.4.3 Ubah Data

Untuk proses ubah data, sebaiknya pada file **tampil.tpl** tambahkan skrip yang tercetak tebal berikut:

```
<h4>Data Mahasiswa</td>
<table border=1>
<tr>
    <td>No</td>
    <td>NIM</td>
    <td>Nama</td>
    <td>Alamat</td>
    <td>Id Jurusan</td>
    <td colspan=2>Aksi</td>
</tr>

{section name=hasil loop=$data}
<tr>
    <td>{$data[hasil].no}</td>
    <td>{$data[hasil].nim}</td>
    <td>{$data[hasil].nama}</td>
    <td>{$data[hasil].alamat}</td>
    <td>{$data[hasil].id_jurusan}</td>
    <td><a href="edit.php?nim={$data[hasil].nim}">Edit</a></td>
    <td><a href="hapus.php?nim={$data[hasil].nim}">Hapus</a></td>
</tr>
{/section}
</table>
```

Refresh kembali pada halaman **tampil.php**, hasilnya dapat Anda lihat pada Gambar 3.6.



Gambar 3.6 Hasil skrip tampil.php

Buat file dengan nama **edit.php**, kemudian simpan dalam folder kerja smarty Anda. Adapun skripnya sebagai berikut:

```
<?php
error_reporting(0);
require('libs/Smarty.class.php');
$smarty = new Smarty;

include "koneksi.php";

if (!empty($_GET['nim'])){
    $sql = mysql_query("SELECT * FROM tmahasiswa WHERE nim =
'$_GET[nim]'");
    $data = mysql_fetch_array($sql);
    $dataresult = array(
        'nim' => $data[nim],
        'nama' => $data[nama],
        'alamat' => $data[alamat],
        'id_jurusan' => $data[id_jurusan]
    );
    $smarty->assign("data", $dataresult);

    if ($_POST['edit'] == 'Edit'){
        $sql = mysql_query("UPDATE tmahasiswa SET
nama = '$_POST[nama]',

        alamat = '$_POST[alamat]',
        id_jurusan = '$_POST[id_jurusan]'
        WHERE nim = '$_POST[nim]'");
        header("location: tampil.php");
    }
}
$smarty->display("edit.tpl");
?>
```

Setelah file logika kita buat, selanjutnya kita buat file layout bernama **edit.tpl** dan simpan dalam folder template. Adapun skripnya sebagai berikut:

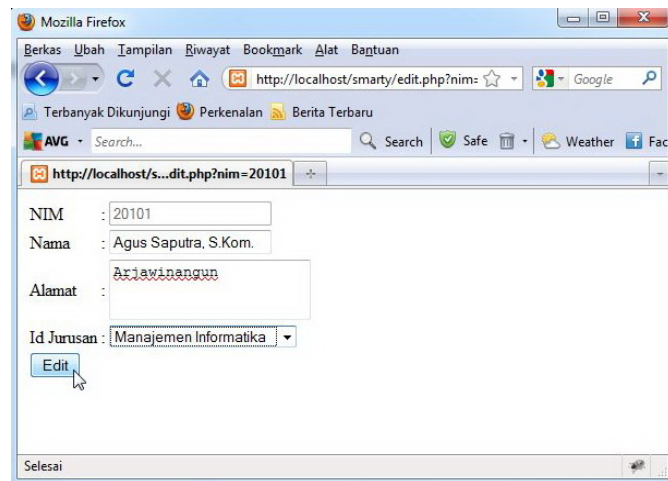
```
<form method="POST" action="">
<table>
<tr>
    <td>NIM</td>
    <td>:</td>
    <td><input type="text" name="nim" value="{ $data.nim}"
disabled>
    <input type="hidden" name="nim"
value="{ $data.nim}"></td>
</tr>
<tr>
    <td>Nama</td>
    <td>:</td>
    <td><input type="text" name="nama"
value="{ $data.nama}"></td>
</tr>
<tr>
```

```

        <td>Alamat</td>
        <td>:</td>
        <td><textarea name="alamat">{$data.alamat}</textarea></td>
    </tr>
    <tr>
        <td>Id Jurusan</td>
        <td>:</td>
        <td>
            <select name="id_jurusan">
                <option value="1" {if $data.id_jurusan ==
1}SELECTED{/if}>Teknik Informatika</option>
                <option value="2" {if $data.id_jurusan ==
2}SELECTED{/if}>Sistem Informasi</option>
                <option value="3" {if $data.id_jurusan ==
3}SELECTED{/if}>Manajemen Informatika</option>
                <option value="4" {if $data.id_jurusan ==
4}SELECTED{/if}>komputerisasi Akuntansi</option>
            </select>
        </td>
    </tr>
    <tr>
        <td><input type="submit" name="edit" value="Edit"></td>
    </tr>
</table>
</form>

```

Refresh pada halaman tampil data. Kemudian klik salah satu link edit dari daftar mahasiswa, maka akan ditampilkan form edit seperti yang terlihat pada Gambar 3.7.



**Gambar 3.7 Hasil skrip edit.php**

Silakan lakukan perubahan data dan akhiri dengan klik tombol edit.

### 3.4.4 Hapus Data

Untuk proses hapus data, kita memerlukan 1 file php saja (tanpa file layout). Karena apa? Karena hapus data itu kan tidak membutuhkan tampilan?

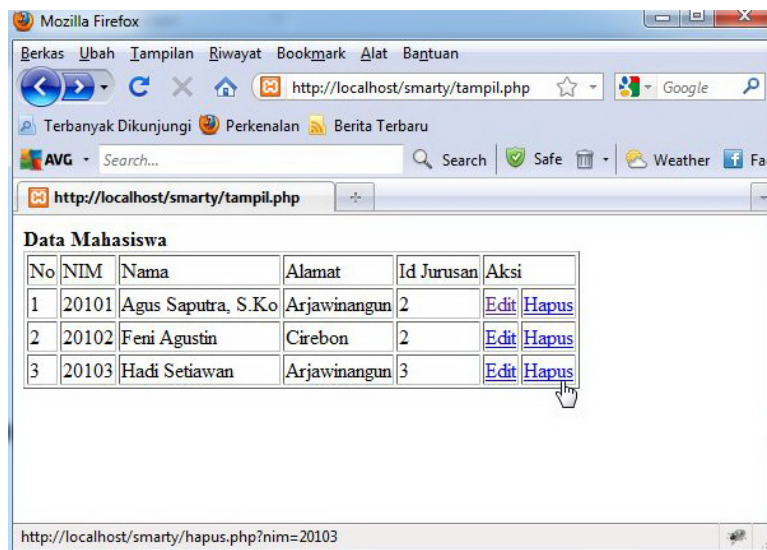
Jadi, langsung saja hapus, kemudian langsung diarahkan ke halaman semula. Buat file dengan nama **hapus.php**, simpan ke dalam folder smarty. Adapun skripnya sebagai berikut:

```
<?php
include "koneksi.php";

mysql_query("DELETE FROM tmahasiswa WHERE nim = '$_GET[nim]'");

header("location: tampil.php");
?>
```

Refresh kembali halaman tampil.php, dan klik salah satu link hapus, maka data yang Anda klik akan terhapus seketika itu juga. Lihat pada Gambar 3.8.



Gambar 3.8 Hasil skrip hapus.php

## 3.5 Upload File

Kita akan masuk ke dalam teknik upload file. Upload file bisa bermacam-macam, contohnya upload gambar, video, dokumen, dan lain-lain. Upload itu bisa diartikan proses perpindahan dari lokal ke server. Lokal means My Computer and Server means Hosting... halah.. belagu deh om, english-english-an segala.. stupid amat dah.. Yang pasti, kita langsung saja mulai prosesnya.

Oh iya, sebelumnya di sini penulis memberikan 2 contoh upload menggunakan `move_upload_file` dan juga copy, serta 1 tambahan thumbnail.

### 3.5.1 Menggunakan `Move_uploaded_file`

`Move_uploaded_file` merupakan salah satu fungsi atau perintah yang digunakan untuk upload data/file. Dengan menggunakan `move_uploaded_file`, data akan langsung ter-upload ke direktori yang dituju tanpa harus tersimpan ke dalam log temp terlebih dahulu. Artinya dengan menggunakan fungsi ini, akan menghemat space Anda. Untuk menggunakan fungsi ini dalam smarty.. yukk simak baik-baik.

Buat file dengan nama **upload\_move.php** dan simpan dalam folder kerja smarty Anda. Adapun skripnya sebagai berikut:

```
<?php
error_reporting(0);
require('libs/Smarty.class.php');
$smarty = new Smarty;

if ($_POST['upload'] == 'Upload'){
    $filename      = $_FILES['image']['tmp_name'];
    $name          = $_FILES['image']['name'];
    $direktori     = "images/$name";

    move_uploaded_file($filename, $direktori);
    $smarty->assign("success", 1);
    $smarty->assign("fn", $name);
}

$smarty->display("upload_move.tpl");
?>
```



### Penjelasan kode:

- `error_reporting(0);` artinya kita meng-hidden seluruh error agar tidak ditampilkan pada halaman web.
- `require('libs/Smarty.class.php');` skrip untuk memanggil library smarty.class.php.
- `$smarty = new Smarty;` membuat new objek data dari class Smarty.
- `if ($_POST['upload'] == 'Upload'){` jika file php mendapat parameter kiriman dari form (tpl) yang teridentifikasi sebagai Upload, maka lakukan proses.
- `$filename` dan `$name` membuat variabel tipe upload data dari form. `$name` adalah nama file.
- `$direktori` variabel direktori tujuan. Tentang ke mana file yang di-upload akan disimpan.
- `move_uploaded_file($filename, $direktori);` proses upload terjadi, perpindahan dari lokal menuju ke direktori tujuan.
- `$smarty->assign("success", 1);` membuat assign success dengan nilai 1. Ini akan digunakan pada file layout.
- `$smarty->assign("fn", $name);` membuat assign fn dengan nilai nama file.
- `$smarty->display("upload_move.tpl");` tempat skrip dan hasil ditampilkan, yaitu akan ditampilkan pada file `upload_move.tpl`.

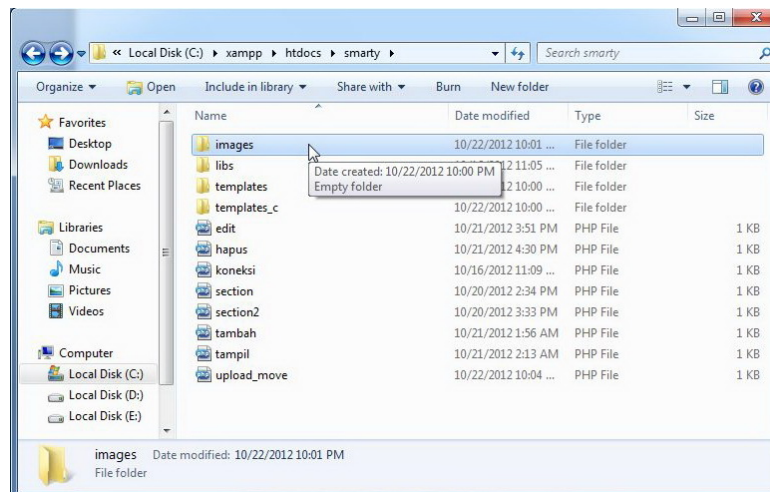
Selanjutnya buat file layout dengan nama `upload_move.tpl` dan simpan ke dalam folder template. Adapun skripnya sebagai berikut:

```
<form method="POST" action="" enctype="multipart/form-data">
<table>
<tr>
    <td>Upload File</td>
    <td>:</td>
    <td><input type="file" name="image"></td>
</tr>
<tr>
    <td><input type="submit" name="upload" value="Upload"></td>
</tr>
</table>
</form>
```

```
{if $success == '1'}
    
{/if}
```

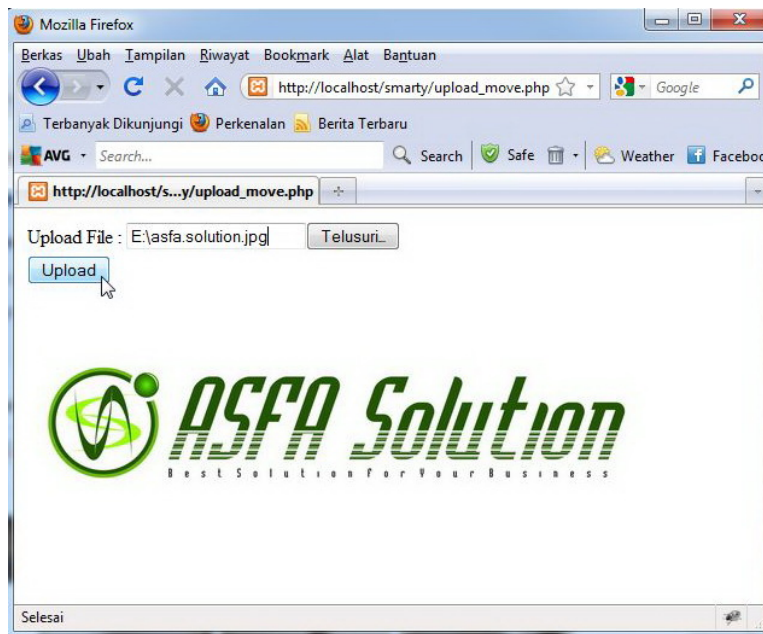
Yang menjadi titik utama mungkin adalah skrip yang tercetak tebal di atas. Itu artinya ketika variabel assign success bernilai sama dengan 1, maka akan langsung ditampilkan gambar hasil upload tersebut. Nama file didapat dari skrip assign fn yang terjadi pada file php.

Langkah terakhir adalah membuat folder baru dengan nama **images** di dalam folder kerja smarty Anda. Lihat pada Gambar 3.9.



**Gambar 3.9** Buat folder *images.php*

Jalankan skrip yang telah dibuat melalui web browser dengan mengetikkan url [http://localhost/smarty/upload\\_move.php](http://localhost/smarty/upload_move.php), maka Anda akan mendapatkan form seperti pada Gambar 3.10. Silakan lakukan upload file, kemudian akhiri dengan klik tombol Upload.



Gambar 3.10 Hasil skrip `upload_move.php`

### 3.5.2 Menggunakan Copy

Berbeda dengan `move_uploaded_file`, `copy` juga digunakan untuk melakukan proses upload file. Bedanya `copy` akan diletakkan pada log temp terlebih dahulu, lalu akan diteruskan ke direktori tujuan.

Buat file dengan nama `upload_copy.php` dan simpan ke dalam folder `smarty`. Adapun skripnya sebagai berikut:

```
<?php
error_reporting(0);
require('libs/Smarty.class.php');
$smarty = new Smarty;

if ($_POST['upload'] == 'Upload'){
    $f1 = $_FILES['image']['tmp_name'];
    $f1_name = $_FILES['image']['name'];
    $f1_type = $_FILES['image']['type'];
    $f1_size = $_FILES['image']['size'];

    if (!empty($f1_type)){
        switch ($f1_type) {
```

```

        case "image/jpeg" :
        copy($F1, "images/$F1_name");

        break;
        case "image/pjpeg" :
        copy($F1, "images/$F1_name");
    }

    $smarty->assign("success", 1);
    $smarty->assign("fn", $F1_name);
}

$smarty->display("upload_move.tpl");
?>

```

#### Penjelasan kode:

- `if(!empty($F1_type)){` jika type data tidak kosong, maka jalankan proses selanjutnya.
- `case "image/jpeg" : atau case "image/pjpeg" :` itu menandakan bahwa file yang di-upload berupa gambar dengan format .jpg.
- `copy($F1, "images/$F1_name");` lakukan copy dari lokal temp ke direktori tujuan.
- Untuk skrip lainnya, penjelasannya sama. Di sini kita menggunakan file layout **upload\_move.tpl**. Karena isinya sama saja, jadi untuk memperingkas pembahasan.

Silakan buka web browser Anda kembali dengan mengunjungi alamat [http://localhost/smarty/upload\\_copy.php](http://localhost/smarty/upload_copy.php). Maka hasilnya akan sama seperti pada Gambar 3.10.

### 3.5.3 Thumbnail

Thumbnail identik dengan gambar, *by the way*.. apa sih thumbnail itu? .. Begini sob.. dalam suatu website, pasti Anda pernah melihat suatu gambar dengan ukuran besar maupun kecil. Dan uniknya nih.. kedua gambar tersebut ditampilkan dalam kualitas sangat baik, dalam artian tidak pecah (padat resolusi). Itulah yang dimaksudkan dengan thumbnail, teknik me-resize suatu gambar.

Kita melakukan upload gambar dengan ukuran 1600 x 1200 piksel. Pada saat kita upload, gambar tersebut akan dilakukan resize

(pemisahan gambar asli dan gambar thumbnail) sehingga pada web konten, gambar thumbnail tersebutlah yang ditampilkan. Berarti gambar yang asli tidak berguna donk? .. ya enggak gitu .. Gambar asli tetap digunakan apabila kita ingin melihat gambar dalam ukuran yang sebenarnya (zooming). Hal ini dilakukan untuk memperingan daya akses web serta juga menghemat bandwith hosting. Nah.. pada subbab ini, kita akan mencoba untuk melakukan trik thumbnail ini. Berikut langkahnya.

### Langkah Pertama

Buat file yang isinya fungsi thumbnail. Simpan dengan nama **thumbnail.php** dan simpan dalam folder kerja smarty Anda. Adapun skripnya sebagai berikut:

```
<?php
function Upload($uploadName){
    // File gambar diupload
    $direktori = "images/";
    $direktoriThumb = "images/thumb/";
    $file = $direktori . $uploadName;

    // Simpan gambar dalam ukuran sebenarnya
    $realImagesName = $_FILES['image']['tmp_name'];
    move_uploaded_file($realImagesName, $file);

    // Identitas file gambar
    $realImages = imagecreatefromjpeg($file);
    $width = imageSX($realImages);
    $height = imageSY($realImages);

    // Simpan dalam ukuran file yang diinginkan (thumb)
    $thumbWidth = 150;
    $thumbHeight = ($thumbWidth / $width) * $height;

    // Fungsi untuk mengubah ukuran gambar
    $thumbImage = imagecreatetruecolor($thumbWidth,
$thumbHeight);
    imagecopyresampled($thumbImage, $realImages, 0, 0, 0, 0,
$thumbWidth, $thumbHeight, $width, $height);

    // Simpan gambar thumbnail
    imagejpeg($thumbImage, $direktoriThumb . "thumb_" .
$uploadName);

    // Hapus objek gambar dalam memori
    imagedestroy($realImages);
    imagedestroy($thumbImage);
}
?>
```

Pada skrip di atas, terdapat variabel `$direktori` dan `$direktoriThumb`. Variabel tersebut merupakan tempat folder untuk menyimpan gambar, seperti yang dilihat. Gambar asli akan disimpan dalam folder `images` sedangkan gambar thumbnail akan disimpan dalam folder `images/thumbs`. Oleh sebab itu, jangan lupa buat folder "thumb" dalam folder images... folder images-nya nggak usah dibuat lagi, kan sudah pernah kita buat sebelumnya.

Untuk file layout-nya, kita tetap sama menggunakan `upload_move.tpl`, hanya saja ada sedikit skrip yang perlu diubah. Perhatikan skrip yang tercetak tebal berikut:

```
{if $success == '1'}
     <br><br>
    Thumb:<br>
    
{/if}
```

Selanjutnya kita buat file `upload_thumbnail.php`, kemudian simpan ke dalam folder kerja smarty Anda. Dan jangan lupa isikan dengan skrip berikut:

```
<?php
error_reporting(0);
require('libs/Smarty.class.php');
$smarty = new Smarty;

// panggil file thumbnail
include "thumbnail.php";

if ($_POST['upload'] == 'Upload'){
    // deklarasi parameter POST ke dalam variabel
    $f1 = $_FILES['image']['tmp_name'];
    $f1Name = $_FILES['image']['name'];
    $f1Type = $_FILES['image']['type'];
    $f1Size = $_FILES['image']['size'];

    // Menggunakan fungsi Upload dengan mengirimkan variabel
    $f1Name
    Upload($f1Name);

    $smarty->assign("success", 1);
    $smarty->assign("fn", $f1Name);
}

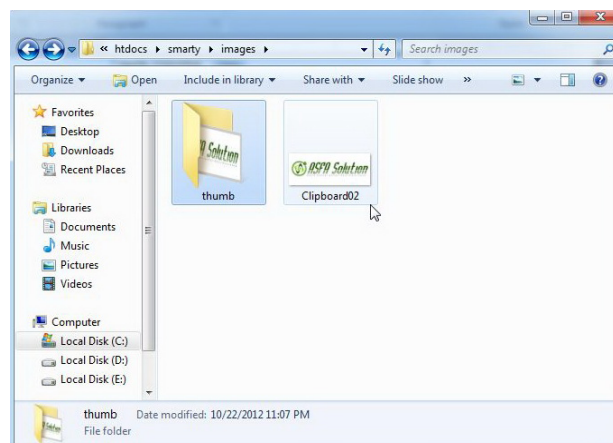
$smarty->display("upload_move.tpl");
?>
```

Buka web browser Anda dengan mengetikkan url [http://localhost/smarty/upload\\_thumbnail.php](http://localhost/smarty/upload_thumbnail.php), akan tampil hasil seperti Gambar 3.11. Browse image dan akhiri dengan klik Upload.



**Gambar 3.11 Hasil skrip upload\_thumbnail.php**

Jika Anda masih kurang percaya, bisa Anda lihat pada folder images-nya. Lihat Gambar 3.12.



**Gambar 3.12 Gambar telah ter-upload dalam folder images dan thumb**

### 3.6 Membuat Paging (Halaman)

Paginasi halaman tentu merupakan fitur utama yang harus ada dalam suatu web. Tapi dipikir-pikir, bukan harus ada sih, tapi tergantung dari kondisi web tersebut.. jiah... sih om plin-plan.. ☺

Untuk membuat paging menggunakan php biasa, sangat mudah, tutorialnya pun banyak sekali di internet. Cukup tanya ama mba google, dapet deh fungsi yang diinginkan.

Nah, jika kita ingin membuat paging menggunakan smarty? .. Bagaimana yah om? .. Sedikit sekali tutorial yang membahas mengenai smarty. Untuk itu, di sini penulis ingin berbagi ilmu dengan Anda.

Kita menggunakan tabel **tmahasiswa** yang telah dibuat sebelumnya.

Buat file dengan nama **paging.php** dan simpan dalam folder kerja smarty Anda. Adapun skripnya sebagai berikut:

```
<?php
error_reporting(0);
require('libs/Smarty.class.php');
$smarty = new Smarty;

include "koneksi.php";

$limit = 1;
$next = $_GET['next'];
if (empty ($next) ) {
    $posisi = 0;
    $next = 1;
}
else {
    $posisi = ($next - 1) * $limit;
}

$sql = mysql_query("SELECT * FROM tmahasiswa LIMIT $posisi,
$limit");
$no = 1+$posisi;
while ($data = mysql_fetch_array($sql)){
    $dataresult[] = array( 'no' => $no,
        'nim' => $data['nim'],
        'nama' => $data['nama'],
        'alamat' => $data['alamat'],
        'id_jurusan' => $data['id_jurusan']
    );
    $no++;
}

$sql_paging = mysql_query("SELECT * FROM tmahasiswa");
$jmldata = mysql_num_rows($sql_paging);
```



```

$jumlah_halaman = ceil($jmldata / $limit);
for($i = 1; $i <= $jumlah_halaman; $i++){
    if($i == $next) {
        $url = "yes";
    }
    else{
        $url = "no";
    }
    $num[] = array( 'i' => $i,
        'url' => $url);
    $smarty->assign("num", $num);
}
$smarty->assign("data", $dataresult);
$smarty->display("paging.tpl");
?>

```

#### Penjelasan kode:

- `error_reporting(0);` digunakan untuk menghilangkan semua error yang akan ditampilkan.
- `require('libs/Smarty.class.php');` digunakan untuk memanggil class library smarty.
- `$smarty = new Smarty;` membuat objek baru dari class smarty.
- `include "koneksi.php";` berfungsi untuk memanggil fungsi koneksi yang akan digunakan sebagai penghubung antara php dengan database mysql.
- `$limit = 1;` membuat variabel batas. Variabel ini akan digunakan sebagai limit atau pembatasan penampilan halaman. Artinya setiap halaman akan ditampilkan hanya 1 data.
- `$next = $_GET['next'];` digunakan untuk menangkap parameter variabel next untuk menentukan posisi tampilnya data.
- `if (empty ($next) ) {` jika parameter GET dari next adalah kosong, maka akan dibuat variabel `$posisi` dan `$next`. Ini juga akan menentukan posisi tampilnya halaman. Jika `$next` tidak kosong, maka akan dibuat variabel `$posisi`.
- `$sql = mysql_query("SELECT * FROM tmahasiswa LIMIT $posisi, $limit");` skrip yang digunakan untuk mengambil data mahasiswa dari tabel tmahasiswa. Hanya saja data ini akan dibatasi sesuai dengan variabel `$limit`.

- `$no = 1+$posisi;` digunakan untuk penomoran halaman.
- `while ($data = mysql_fetch_array($sql)){` digunakan untuk menampilkan data. Data ini akan disimpan ke dalam bentuk array (`$dataresult`).
- `$sql_paging = mysql_query("SELECT * FROM tmahasiswa");` digunakan untuk menyaring data terhadap tmahasiswa. Hanya saja pengambilan data mahasiswa ini tidak kita batasi.
- `$jmldata = mysql_num_rows($sql_paging);` digunakan untuk menghitung banyaknya data yang dihasilkan dari `$sql_paging`.
- `$jumlah_halaman = ceil($jmldata / $limit);` skrip untuk melakukan pembulatan ke atas terhadap angka yang dihasilkan.
- `for($i = 1; $i <= $jumlah_halaman; $i++){` melakukan skrip perulangan terhadap angka yang dihasilkan.
- `if($i == $next) {` ini sebenarnya gampang-gampangnya penulis aja. Jadi logikanya, jika nomor itu sama dengan parameter `next` yang ditangkap, maka kita buat variabel `$url` bernilai Yes. Jika tidak, maka kita buat `$url` bernilai No.
- `$num[]` karena ini bersifat perulangan angka yang kita tidak tahu entah berapa banyaknya perulangan yang dilakukan maka kita buat variabel `$num` dengan tambahan [], artinya data itu adalah lebih dari 1.
- `$smarty->assign("num", $num);` melakukan assign terhadap variabel `$num`. Ini akan dijadikan sebagai angka paging.
- `$smarty->assign("data", $dataresult);` melakukan assign terhadap variabel `$dataresult`. Ini akan dijadikan sebagai tampilnya data mahasiswa.
- `$smarty->display("paging.tpl");` kita atur agar semua skrip yang telah kita buat hanya akan ditampilkan pada file **paging.tpl**.

Selanjutnya buat file layout dengan nama **paging.tpl** dan simpan di dalam folder template. Adapun skripnya sebagai berikut:

```
<h4>Data Mahasiswa</td>
<table border=1>
<tr>
```

```

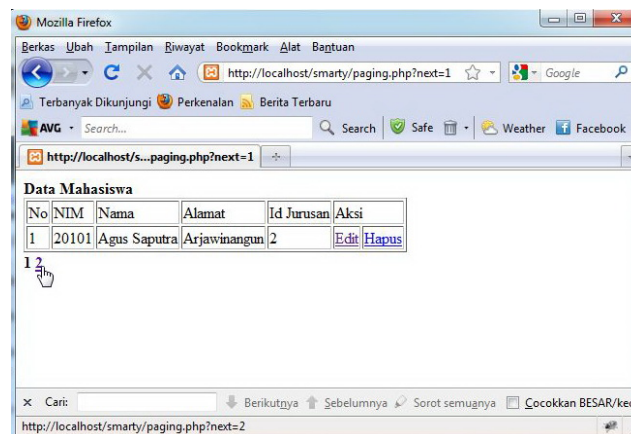
<td>No</td>
<td>NIM</td>
<td>Nama</td>
<td>Alamat</td>
<td>Id Jurusan</td>
<td colspan=2>Aksi</td>
</tr>

{section name=hasil loop=$data}
<tr>
<td>{$data[hasil].no}</td>
<td>{$data[hasil].nim}</td>
<td>{$data[hasil].nama}</td>
<td>{$data[hasil].alamat}</td>
<td>{$data[hasil].id_jurusan}</td>
<td><a href="edit.php?nim={$data[hasil].nim}">Edit</a></td>
<td><a
href="hapus.php?nim={$data[hasil].nim}">Hapus</a></td>
</tr>
{/section}
</table>

{section name=num loop=$num}
{if $num[num].url == "yes"}
{$num[num].i}
{else}
<a href="paging.php?next={$num[num].i}">{$num[num].i}</a>
{/if}
{/section}

```

Buka web browser Anda dengan mengetikkan url <http://localhost/smarty/paging.php>, maka akan tampil konten seperti Gambar 3.13. Lakukan klik data terhadap link paging yang tampil.



Gambar 3.13 Hasil skrip paging.php

### 3.7 Force Download

Upload file? Sudah...

Paging? Sudah juga..

Hmmm.. kali ini kita masuk saja ke dalam pembahasan force download. Soalnya kenapa penulis menyertakan fungsi ini dalam buku? .. Karena fitur ini hampir setiap kali ada dalam suatu website. Ditambah penulis rasa, fitur ini juga cukup penting. Apalagi 1 yang penulis tekankan, susah sekali mendapatkan tutorial mengenai trik-trik dalam bahasa smarty.

Force download merupakan suatu proses perpindahan dari sistem server ke sistem lokal kita. Dalam bahasa Indonesia, Download itu sama dengan Unduh data. Data yang di-unduh ini banyak sekali macamnya, bisa berupa images, video, dokumen, file, dan lain sebagainya. Namun karena pada latihan sebelumnya, kita melakukan download terhadap file gambar, maka di sini penulis juga akan memberikan sampel untuk proses download gambar juga.

Untuk mempersingkat pembahasan, buka file `move_upload.tpl`, kemudian tambahkan skrip yang tercetak tebal berikut:

```
{if $success == '1'}
    <a href="download.php?id={$fn}"><b>Download File</b></a>
<br><br>
     <br><br>
    Thumb:<br>
    
{/if}
```

Selanjutnya buat file php dengan nama `download.php`, simpan dalam folder kerja smarty Anda, dan tuliskan dengan skrip berikut:

```
<?php
$direktori = "images/";
$filename = $_GET['id'];

$file_extension = strtolower(substr(strrchr($filename,"."),1));

// cek ekstensi file (apakah jpeg, pdf, doc, dll)
switch($file_extension){
    case "pdf": $ctype="application/pdf"; break;
    case "exe": $ctype="application/octet-stream"; break;
    case "zip": $ctype="application/zip"; break;
    case "rar": $ctype="application/rar"; break;
    case "doc": $ctype="application/msword"; break;
    case "xls": $ctype="application/vnd.ms-excel"; break;
```

```

case "ppt": $ctype="application/vnd.ms-powerpoint"; break;
case "gif": $ctype="image/gif"; break;
case "png": $ctype="image/png"; break;
case "jpeg":
case "jpg": $ctype="image/jpg"; break;
default: $ctype="application/proses";
}

// Perintah untuk Force file
header("Pragma: private");
header("Expires: 0");
header("Cache-Control:      must-revalidate,      post-check=0,      pre-
check=0");
header("Cache-Control: private",false);
header("Content-Type: $ctype");
header("Content-Disposition:                                attachment;
filename=\"\".basename($filename).\"\"");
header("Content-Transfer-Encoding: binary");
header("Content-Length: \" .filesize($direktori.$filename));
readfile("$direktori$filename");
exit();
?>

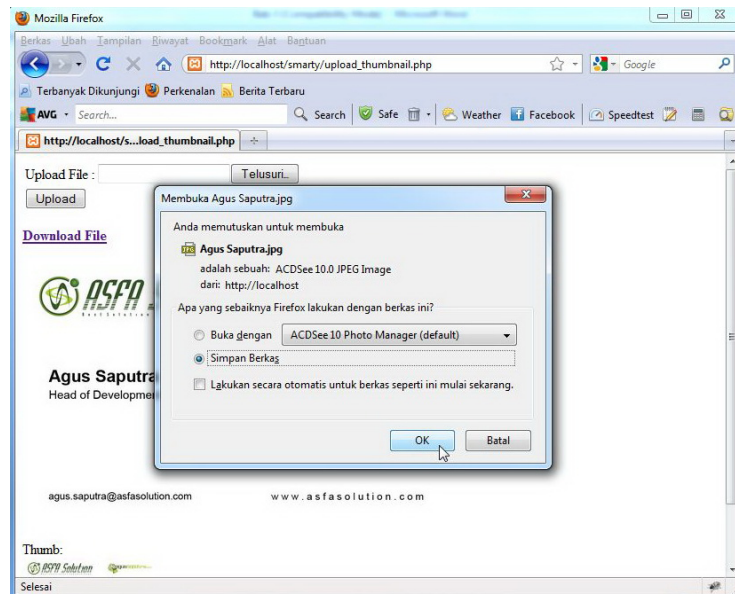
```

Buka web browser Anda, dan jalankan skrip `upload_thumbnail.php` ([http://localhost/smarty/upload\\_thumbnail.php](http://localhost/smarty/upload_thumbnail.php)). Lakukan upload, ketika berhasil, akan tampil link **Download**. Lihat Gambar 3.14.



Gambar 3.14 Hasil skrip `download.php`

Klik link download, maka akan ditampilkan jendela download seperti pada Gambar 3.15. Pilih Open jika ingin membuka, dan Save jika Anda ingin menyimpannya ke dalam komputer lokal Anda.



Gambar 3.15 Download file

### 3.8 Import File Excel

Adakalanya setiap inputan data tidak bisa dilakukan satu per satu. Sebabnya, bisa jadi karena banyaknya data, jadi tidak dimungkinkan untuk input data satu per satu. Seperti contohnya pengalaman penulis bekerja di salah satu perusahaan swasta. Penulis diharuskan memasukkan data yang jumlahnya lebih dari 20.000 data setiap harinya. Kalo penulis buat sistem manual, bisa-bisa tahun 2015 baru selesai tuh kerjaan. Mau nggak mau, penulis buat suatu fitur, di mana dalam 1 kali klik, seluruh data akan tersimpan semuanya dalam waktu singkat. Fitur tersebut dinamakan "Import". Nantinya data yang diimpor, akan disimpan dalam bentuk dokumen Excel, dan kemudian akan diimpor melalui sebuah sistem, dan akan disimpan dalam database.

## Langkah Pertama

Download file **excel\_reader** dari situs [http://www.4shared.com/file/1xmtEZeZ/excel\\_reader2.html](http://www.4shared.com/file/1xmtEZeZ/excel_reader2.html) (akan menghasilkan file zip). Kemudian lakukan ekstrak pada file zip tersebut dan letakkan pada folder kerja Anda di httdocs.

## Langkah Kedua

Di sini kita masih tetap menggunakan tabel mahasiswa sebagai studi kasus database-nya. So, kita lanjut saja membuat file php dengan nama **import.php** dan simpan dalam folder kerja smarty Anda. Adapun skripnya sebagai berikut:

```
<?php
error_reporting(0);
require('libs/Smarty.class.php');
$smarty = new Smarty;

// panggil file excel_reader
include "excel_reader2.php";

// koneksi ke mysql
include "koneksi.php";

if ($_POST['upload'] == 'Import'){
    // membaca file excel yang diupload
    $data = new Spreadsheet_Excel_Reader($_FILES['userfile']['tmp_name']);

    // membaca jumlah baris (excel)
    $baris = $data->rowcount($sheet_index=0);

    // nilai awal counter untuk jumlah data yang sukses dan
    yang gagal diimport
    $sukses = 0;
    $gagal = 0;

    // import data excel mulai baris ke-2 (karena baris pertama
    adalah nama kolom)
    for ($i=2; $i<=$baris; $i++) {
        // membaca data nim (kolom ke-1)
        $nim = $data->val($i, 1);
        // membaca data nama (kolom ke-2)
        $nama = $data->val($i, 2);
        // membaca data alamat (kolom ke-3)
        $alamat = $data->val($i, 3);
        // membaca data id jurusan (kolom ke-4)
        $idjurusan = $data->val($i, 4);

        // setelah data dibaca, simpan ke dalam tabel
        mahasiswa
        $hasil = mysql_query("INSERT INTO tmahasiswa VALUES
        ('$nim', '$nama', '$alamat', '$idjurusan')");
    }
}
```

```

    }
    $smarty->assign("success", "Data berhasil diimport");
}
$smarty->display("import.tpl");
?>

```

### Langkah Ketiga

Buat file layout dengan nama **import.tpl** (karena file php akan ditampilkan pada file import.tpl). Simpan ke dalam folder template. Adapun skripnya sebagai berikut:

```

{$success}

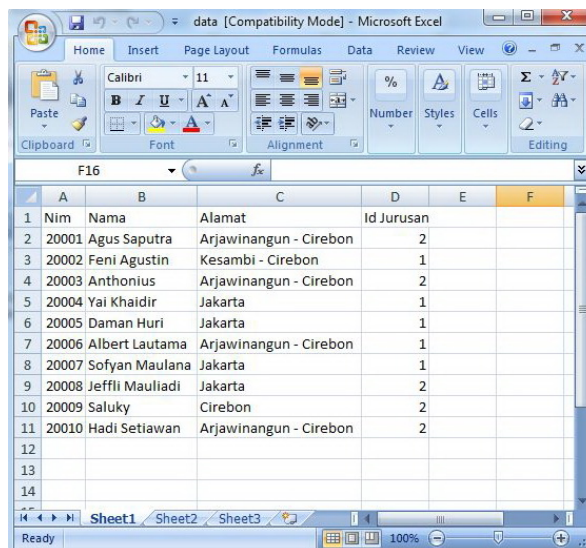
<h1>Import Data Mahasiswa</h1>

<form method="post" enctype="multipart/form-data" action="" >
Unggah File Excel: <input name="userfile" type="file">
<input name="upload" type="submit" value="Import">
</form>

```

### Langkah Keempat

Buat data dalam format Excel (.xls) untuk kita upload pada skrip yang telah kita buat. Formatnya dapat Anda lihat seperti gambar ini.

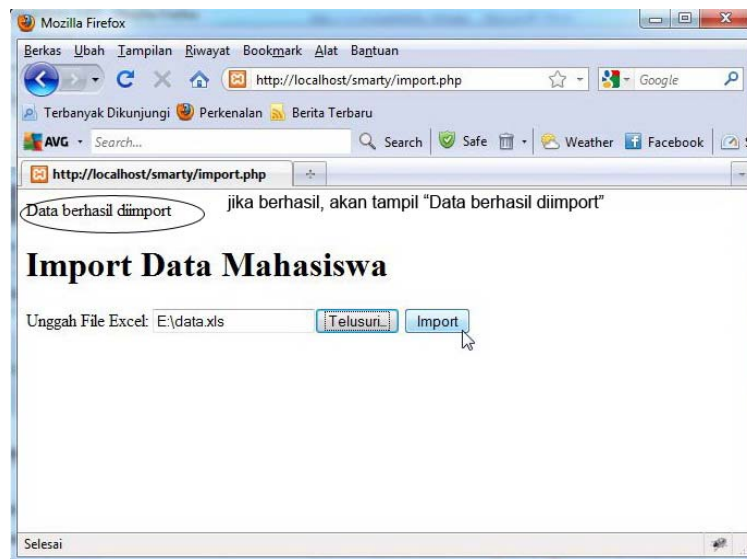


	A	B	C	D	E	F
1	Nim	Nama	Alamat	Id Jurusan		
2	20001	Agus Saputra	Arjawinangun - Cirebon	2		
3	20002	Feni Agustin	Kesambi - Cirebon	1		
4	20003	Anthonius	Arjawinangun - Cirebon	2		
5	20004	Yai Khaidir	Jakarta	1		
6	20005	Daman Huri	Jakarta	1		
7	20006	Albert Lautama	Arjawinangun - Cirebon	1		
8	20007	Sofyan Maulana	Jakarta	1		
9	20008	Jeffli Mauliadi	Jakarta	2		
10	20009	Saluky	Cirebon	2		
11	20010	Hadi Setiawan	Arjawinangun - Cirebon	2		
12						
13						
14						

Gambar 3.16 Format data Excel

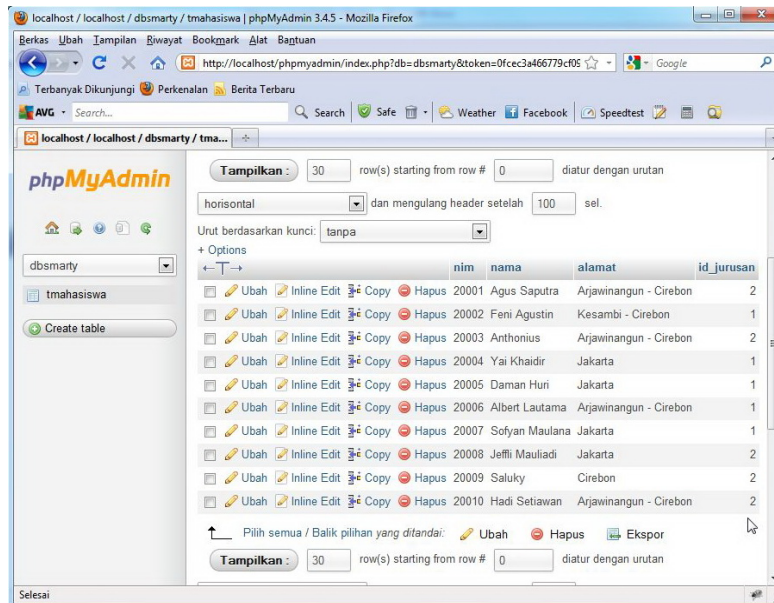


Untuk melihat hasilnya, bisa Anda buka web browser Anda dan ketikkan url <http://localhost/smarty/import.php>. Import file Excel dengan format yang sudah ditentukan seperti pada Gambar 3.16. Klik tombol Import dan lihat hasilnya seperti pada Gambar 3.17.



*Gambar 3.17 Hasil skrip import.php*

Untuk memastikannya, bisa Anda lihat pada database-nya secara langsung, data sudah tersimpan. Lihat Gambar 3.18.



Gambar 3.18 Data berhasil disimpan