

MODUL 5-6 PHP: MENGOLAH DATA DENGAN PHP

1. KONEKSI DATABASE

Untuk bisa mengakses database dan tabel di MySQL, kita perlu melakukan koneksi terlebih dahulu agar bisa terhubung ke database tersebut. Sebaiknya skrip untuk melakukan koneksi dibuat tersendiri dalam sebuah file, karena setiap kali kita akan mengolah data di database, maka kita harus selalu melakukan koneksi terlebih dahulu. Disamping itu yang terlebih penting lagi, hal ini biasatna terjadi ketika kita melakukan proses upload di internet untuk mengonlinekan situs kita, maka biasanya Username dan Password akan ditentukan oleh web hosting (tempat kita meletakkan file-file situs kita). Sekarang coba bayangkan, apabila anda mempunyai sepuluh file, maka anda harus mengganti username dan password sepuluh kali, gimana kalau seratus file? Alangkah sangat merepotkan. Nah, dengan memisah file koneksi tersendiri dalam sebuah file, maka kita hanya perlu melakukan perubahan username dan passwordnya cukup sekali.

Oke, sekarang mari kita buat file koneksi sebagai berikut:

Skrip koneksi_bukutamu.php

```
<? mysql_connect("localhost","root","");  
mysql_select_db("bukutamu");  
?>
```

Keterangan skrip:

- `Mysql_connect("hostname","username","password");`
Untuk melakukan koneksi ke server MySQL. Biasanya pada komputer lokal kita belum ada username dan password databasenya, jadi dikosongkan saja menjadi:
`Mysql_connect("localhost","","");`
- `Mysql_select_db("nama_database");`
Untuk memilih sebuah database, karena tentunya dalam server database mysql terdapat banyak database, sehingga kita harus memilih salah satu database yang akan kita aktifkan/gunakan, dalam hal ini adalah database bukutamu.
- Untuk menyisipkan/memanggil skrip koneksi.php dalam sebuah skip lainnya, cukup tuliskan:
`Include "namafile";`

2. PROSES INPUT DATA

Proses Input Data adalah suatu proses untuk memasukkan data-data ke dalam database/table. Perintah SQL yang akan digunakan adalah:

```
INSERT INTO nama_tabel(field1,field2,...)
```

```
VALUES (data1,data2,.....)
```

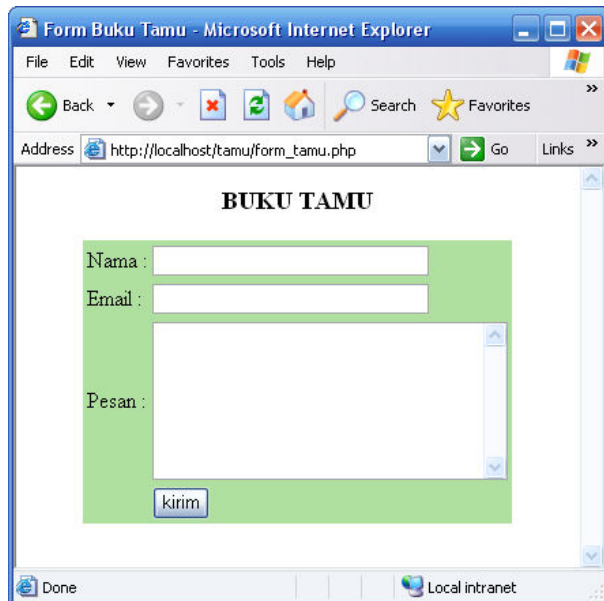
Untuk mempraktekkannya, cobalah tuliskan dua file berikut :

Skrip form_tamu.php

```
<html>
<head>
<title>Form Buku Tamu</title>
</head>
<body>
<h3 align=center>BUKU TAMU </H3>
<form method="post" action="input_tamu.php">
<table bgcolor="bluegray" align="center">
<tr>
<td>Nama : </td>
<td><Input type="text" name="nama" size=30></td>
</tr>
<tr>
<td>Email : </td>
<td><Input type="text" name="email" size=30></td>
</tr>
<tr>
<td>Pesan : </td>
<td><textarea name="pesan" rows=7
cols=30></textarea>
</tr>
<tr>
<td></td>
<td><input type=submit value=kirim></td>
</tr>
```

```
</table>
</form>
</body>
</html>
```

Akhir skrip



Skrip input_tamu.php

```
<?
include "koneksi_bukutamu.php";
$name=$_POST["nama"];
$email=$_POST["email"];
$message=$_POST["pesan"];
$masuk="INSERT INTO tamu(nama,email,pesan)VALUES
('$name','$email','$message)";
$result=mysql_query($masuk);
//apabila input data berhasil
if($result)
{
    echo "Proses Input data Sukses";
}
else
{
    echo "Proses Input data Gagal";
}
?>
```

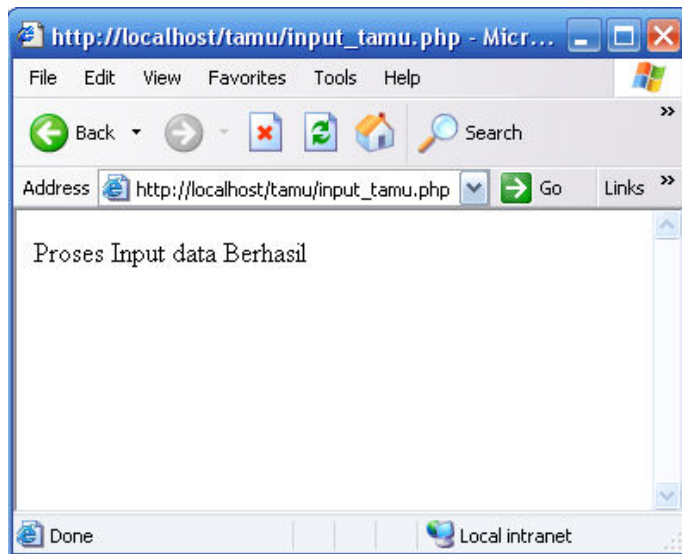
Akhir skrip

Keterangan skrip

Mysql_query("perintah query")

Untuk mengirimkan perintah SQL ke server database MySQL untuk dieksekusi. Sekarang coba jalankan skrip form_tamu.php, lalu isikan data pada bagian nama, email dan pesan.

Kemudian klik tombol kirim, maka data-data yang dimasukkan di form akan diproses oleh skrip input_tamu.php dan apabila tidak ada kesalahan maka proses input data berhasil.



3. MENAMPILKAN DATA DARI DATABASE

Untuk melihat hasil data-data yang telah dimasukkan ke database maka kita perlu menampilkannya ke browser. Perintah SQL yang akan digunakan adalah

```
SELECT * FROM nama_tabel;
```

Untuk mempraktekkannya, cobalah tuliskan skrip berikut:

Skrip tampil_tamu_user.php

```
<?
echo "<H3>DATA TAMU </H3>";
include "koneksi_bukutamu.php";
$stampil=mysql_query("SELECT * FROM tamu ORDER BY no_tamu
DESC");
while ($data=mysql_fetch_array($stampil))
{
    echo "nama: $data[nama]<br>";
    echo "Email: $data[email]<br>";
    echo "Pesan : $data[pesan]<br><hr>";
}
```

```
echo"<a href=tampil_tamu_tabel.php><h3><b><p align=center>Lihat  
Tamu</h3></b></p></a>" ;  
?>
```

Akhir skrip

Keterangan skrip

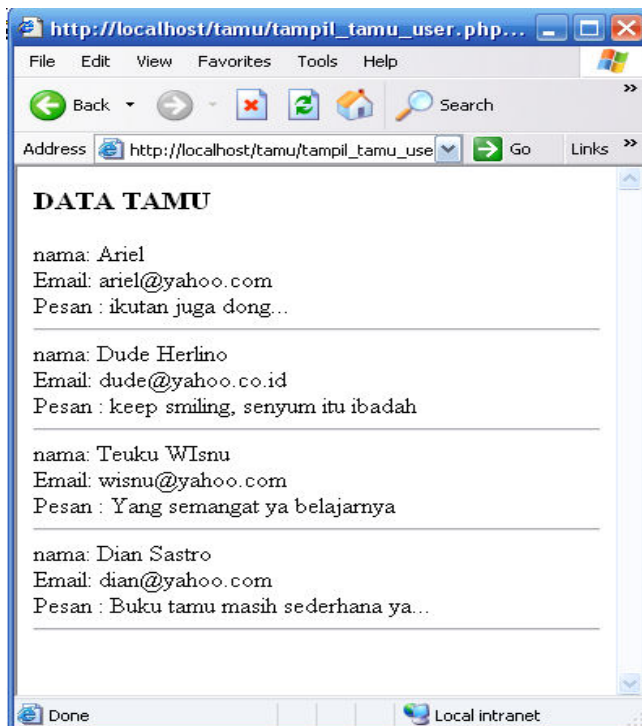
- SELECT * FORM tamu ORDER BY no_tamu DESC;

Tanda asterisk (*) menyatakan bahwa semua data di database akan ditampilkan dan ORDER BY berfungsi untuk mengurutkan data secara DESC (dari besar ke kecil, missal 100.....1, Z.....A). jadi perintah SQL tersebut artinya :”tampilkan semua data dari table tamu, kemudian urutkan berdasarkan no_tamu secara descending berarti diurutkan berdasarkan pada data terakhir yang diinputkan.

- Mysql_fetch_array (“hasil_perintah_SQL”);

Untuk menampilkan hasil dari perintah SQL dalam bentuk array.

Sekarang coba jalankan skrip tampil_tamu_user.php, maka hasilnya akan terlihat seperti berikut.



4. TEKNIK MERAPIKAN DATA

Biasanya kita lebih suka melihat tampilan data dalam bentuk table karena tampilanya lebih rapi dan enak dibaca. Sebenarnya pembuatan table menggunakan tag-tag HTML yang sudah pernah kita pelajari, namun tag-tag tersebut dikombinasikan di PHP. tag- tag HTML yang digunakan dalam membuat table :

```
<TABLE> .....</TABLE>
```

```
<TR>.....</TR>
```

```
<TD>.....</TD>
```

```
<TH>.....</TH>
```

Lalu bagaimana menerapkan tag-tag tersebut ke dalam PHP, mari kita praktekkan untuk menampilkan data tamu dengan sedikit modifikasi skrip tampil_tamu_user.php menjadi skrip berikut :

Skrip_tampil_tamu_tabel.php

```
<?
echo "<H3 ALIGN=CENTER>DATA TAMU</H3>";
include "koneksi_bukutamu.php";
$tampil=mysql_query("select*from          tamu      order      by      no_tamu
desc");
echo"<table border=1 align=center>
<th width=10%>Nama</th>
<th width=15%>E-mail</th>
<th width=30%>Pesan</th>
</tr>";

while ($data=mysql_fetch_array($tampil))
{
    echo "<tr>
    <td>$data[nama]</td>";
    echo "<td>$data[email]</td>";
    echo"<td>$data[pesan]</td></tr>";
}
echo "</table>";
```

akhir skrip

Sekarang coba jalankan skrip tampil_tamu_tabel.php maka hasilnya akan terlihat seperti gambar berikut:

DATA TAMU		
Nama	E-mail	Pesan
Ariel	ariel@yahoo.com	ikutan juga dong...
Dude Herlino	dude@yahoo.co.id	keep smiling, senyum itu ibadah
Teuku Wisnu	wisnu@yahoo.com	Yang semangat ya belajarnya
Dian Sastro	dian@yahoo.com	Buku tamu masih sederhana ya...

5. MENAMPILKAN DATA UNTUK ADMIN

Kalau pada trik sebelumnya, kita hanya menampilkan data untuk umum/user, sehingga dapat dilihat semua orang, sedangkan tampilan data untuk keperluan admin harus berbeda, karena ada tambahan tools untuk melakukan pengeditan dan penghapusan data. Disamping itu kita akan menampilkan total keseluruhan data tamu yang sudah terdaftar di database. Untuk lebih jelasnya, cobalah skrip berikut :

Skrip tampil_tamu_admin.php

```
<?
echo "<H3 ALIGN=CENTER>DATA TAMU</H3>";
include "koneksi_bukutamu.php";
$tampil=mysql_query("select*from tamu order by no_tamu desc");
$total=mysql_num_rows($tampil);
echo"<table border=1 align=center>
<th width=10%>Nama</th>
<th width=15%>E-mail</th>
<th width=30%>Pesan</th>
<th width=10%>Tools</th>
</tr>";

while ($data=mysql_fetch_array($tampil))
{
    echo "<tr>
    <td>$data[nama]</td>";
    echo "<td>$data[email]</td>";
    echo"<td>$data[pesan]</td>";
    //buat link untuk edit dan hapus data echo
    "<td> <a
href=edit_tamu.php?id=$data[no_tamu]>Edit</a>|
    <a
href=hapus_tamu.php?id=$data[no_tamu]>Hapus</a></td></tr> ";
```

```

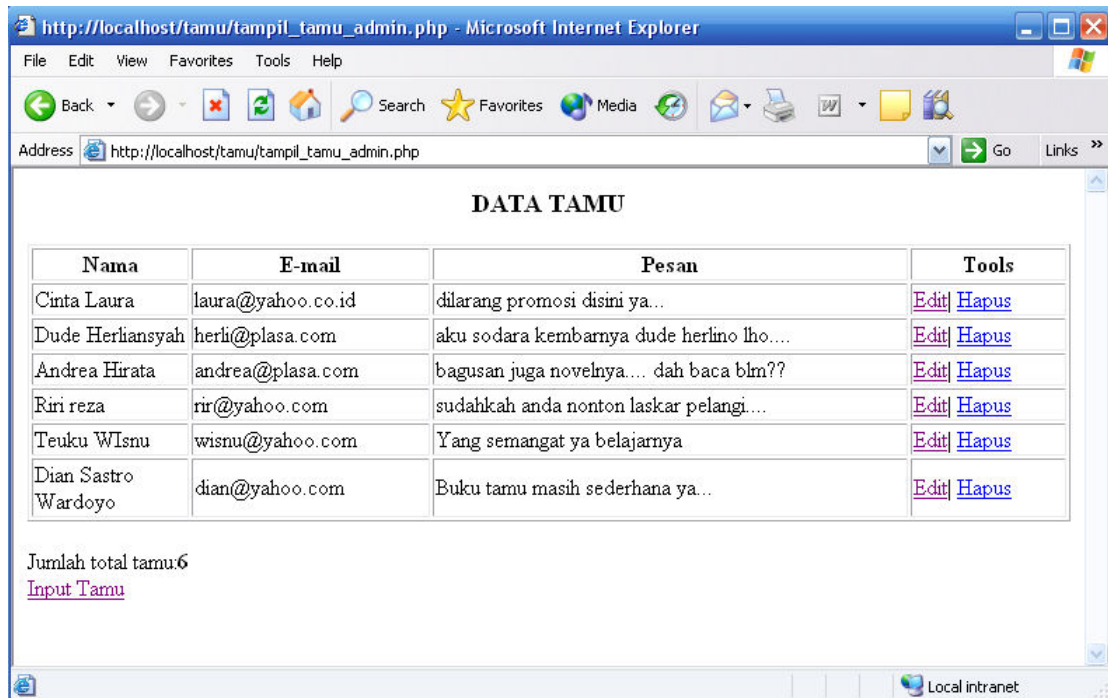
}
echo "</table>";
echo "<p>Jumlah total tamu:<b>$total</b>";
echo"<br><a href=form_tamu.php>Input Tamu</a>";
?>

```

Keterangan skrip

- `mysql_num_rows(perintah_sql);`
untuk menampilkan/menghitung jumlah total baris data yang ada di database. Hasilnya dalam bentuk angka.
- `Edit `
Untuk mem membuat link pada teks Edit dengan tujuan file edtt_tamu.php dan pada link tersebut diberi variable id yang berisi nilai yang diambilkan dari `no_tamu`. Hal tersebut juga berlaku untuk link Hapus.

Sekarang jalankanlah skrip `tampil_tamu_admin.php`, maka hasilnya akan terlihat seperti pada gambar berikut :



Cobalah arahkan kursor mouse anda ke link Edit, lalu lihat pada status bar browser, maka disana akan terlihat tujuan file yaitu `edit_tamu.php` dengan id sesuai dengan `no_tamu` dan `no_tamu` atau id tersebut tentunya berbeda dengan data-data lainnya, hal ini bisa anda coba dengan mengarahkan kursor mouse pada link Edit anta Hapus

didata-data lainnya dan perjakan perubahan id-nya pada status bar browser.

6. TEKNIK REDIRECT: TEKNIK PENGALIHAN HALAMAN

Redirect adalah suatu teknik untuk mengalihkan alamat atau halaman web secara otomatis. Adapun perintah untuk redirect adalah :

```
Header ("location:nama_file_yang_dituju");
```

Masih bingung? Oke, untuk lebih jelasnya, mari langsung kita praktekkan saja, misalnya pada skrip `input_tamu.php` dimana apabila input tersebut berhasil, maka akan tampil teks "Proses Input Data Berhasil". Nah, sekarang halaman `input_tamu.php` akan kita alihkan secara otomatis (redirect) ke halaman `tampil_tamu_admin`, sehingga kita bisa langsung melihat hasil data yang baru diinputkan. Untuk mewujudkan hal tersebut maka kita akan sedikit melakukan modifikasi pada skrip `input_tamu.php` menjadi seperti berikut:

Skrip `input_tamu.php`

```
<?
include "koneksi_bukutamu.php";
$nama=$_POST["nama"];
$email=$_POST["email"];
$pesan=$_POST["pesan"];
$masuk="INSERT INTO tamu(nama,email,pesan)VALUES
('$nama','$email','$pesan)";
$hasil=mysql_query($masuk);
//apabila input data berhasil
if($hasil)
{
    header("location:tampil_tamu_admin.php");
}
else
{
    echo "Proses Input data Gagal";
}
?>
```

Sekarang coba jalankan skrip form_tamu.php lalu isikan data-datanya pada form. Kemudian klik tombol kirim, maka data-data yang dimasukkan di form akan diproses oleh skrip oleh skrip input_tamu.php dan apabila proses input data berhasil, maka secara otomatis halaman akan dialihkan (proses pengalihan halaman tidak terlihat dilayar, karena prosesnya berjalan sangat cepat) ke halaman data tamu admin (tampil_tamu_admin.php) dan hasil input data tersebut langsung dapat dilihat.

7. PROSES PENGEDITAN DATA

Proses edit data adalah suatu proses untuk mengubah data-data yang ada dalam database/tabel. Ada dua tahapan yang perlu dilakukan dalam proses edit data ini, pertama terlebih dahulu id (kunci primer) dari data yang akan diedit, dalam hal ini adalah no_tamu, jadi id adalah sebuah variabel yang berisi nilai dari no_tamu. Adapun perintah SQL yang digunakan adalah :

```
SELECT * FROM nama_tabel WHERE $field_kunci="$id";
```

Setelah mendapat no_tamu yang akan diedit, tampilkan dalam bentuk form data-data yang akan diedit. Kemudian proseslah data-data tersebut untuk diubah (update) dengan perintah SQL berikut :

```
UPDATE nama_tabel set nama_field1='$data1' WHERE field_kunci='id';
```

Untuk mempraktekkannya, cobalah tuliskan dua file skrip berikut :

Skrip edit_tamu.php

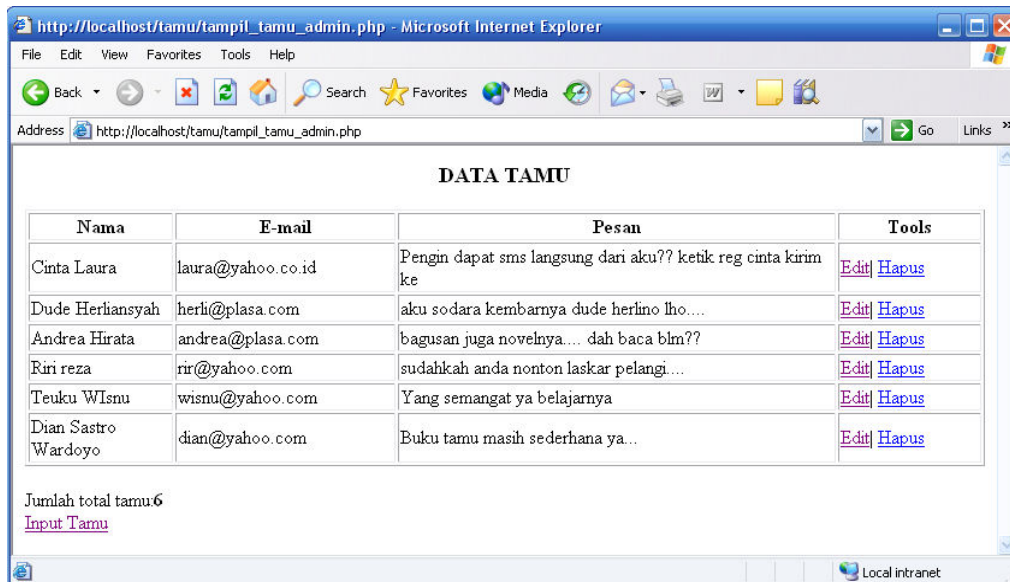
```
<?
include "koneksi_bukutamu.php";
$id=$_GET[id];
$edit="SELECT * FROM `tamu` WHERE `no_tamu` ='$id' ";
$hasil=mysql_query($edit);
$data=mysql_fetch_array($hasil);
//echo "Nama = $data[nama]";
echo "<H3>EDIT BUKU TAMU</h3>";
echo "<form method=get action=update_tamu.php>"; echo
"<input type=hidden name=id value='$id'>"; echo
>Nama:<Input type=text name=nama
value='$data[nama] '><br>";
echo "Email:<Input type=text name=email
value='$data[email] '><br>";
echo "Pesan<br><textarea name=pesan cols=20
rows=5>$data[pesan]</textarea><br>"; echo
"<input type=submit value=Edit>"; echo
"</form>";
```

?>

Skrip update_tamu.php

```
<?
include "koneksi_bukutamu.php";
//$id=$_GET[id];
$nama=$_GET[nama];
$email=$_GET[email];
$pesan=$_GET[pesan];
$proses="UPDATE tamu SET nama='$nama', email='$email',
pesan='$pesan' WHERE no_tamu=$_GET[id]";
$hasil=mysql_query($proses);
if ($hasil)
{
    header ("location:tampil_tamu_tabel.php");
}
else
{
    echo "Proses Update Data Gagal";
}
?>
```

Sekarang coba jalankan skrip `_tampil_tamu_admin.php`, maka akan terlihat tampilannya sebagai berikut :

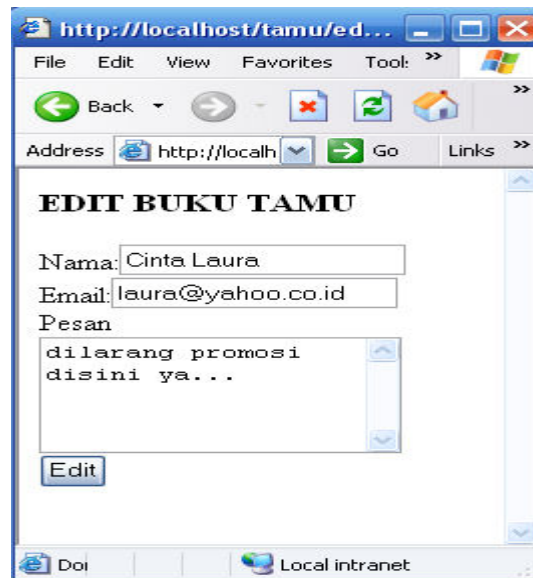


The screenshot shows a Microsoft Internet Explorer browser window displaying a web page titled "DATA TAMU". The page contains a table with four columns: Nama, E-mail, Pesan, and Tools. The table lists six guests with their names, email addresses, messages, and links to edit or delete their entries. Below the table, it shows the total number of guests as 6 and a link to "Input Tamu".

Nama	E-mail	Pesan	Tools
Cinta Laura	laura@yahoo.co.id	Pengin dapat sms langsung dari aku?? ketik reg cinta kirim ke	Edit Hapus
Dude Herliansyah	herli@plasa.com	aku sodara kembarnya dude herlino lho...	Edit Hapus
Andrea Hirata	andrea@plasa.com	bagusan juga novelnya... dah baca bim??	Edit Hapus
Riri reza	rir@yahoo.com	sudahkah anda nonton laskar pelangi...	Edit Hapus
Teuku Wisnu	wisnu@yahoo.com	Yang semangat ya belajarnya	Edit Hapus
Dian Sastro Wardoyo	dian@yahoo.com	Buku tamu masih sederhana ya...	Edit Hapus

Jumlah total tamu: 6
[Input Tamu](#)

Coba klik link edit untuk data paling atas, maka akan tampil halaman form edit (skrip edit_tamu.php) kemudian editlah/ubahlan datanya seperti pada gambar berikut :



EDIT BUKU TAMU

Nama: Cinta Laura

Email: laura@yahoo.co.id

Pesan
dilarang promosi disini ya...

Edit

Kemudian klik tombol Edit, maka data-data yang diubah di form akan diproses oleh skrip update_data.php dan apabila proses edit data berhasil, maka secara otomatis halaman akan dialihkan ke halaman data Tamu Admin dan hasil pengeditan data tersebut langsung dapat dilihat hasilnya seperti berikut.



DATA TAMU

Nama	E-mail	Pesan	Tools
Cinta Laura	laura@yahoo.co.id	dilarang promosi disini ya...	Edit Hapus
Dude Herliansyah	herli@plasa.com	aku sodara kembarnya dude herlino lho....	Edit Hapus
Andrea Hirata	andrea@plasa.com	bagusan juga novelnya.... dah baca blm??	Edit Hapus
Riri reza	riri@yahoo.com	sudahkah anda nonton laskar pelangi....	Edit Hapus
Teuku Wisnu	wisnu@yahoo.com	Yang semangat ya belajarnya	Edit Hapus
Dian Sastro Wardoyo	dian@yahoo.com	Buku tamu masih sederhana ya...	Edit Hapus

Jumlah total tamu:6
[Input Tamu](#)

8. PROSES PENGHAPUSAN DATA

Proses hapus data adalah proses untuk melakukan penghapusan data-data yang ada di database/tabel. Cara kerja proses penghapusan data sama dengan proses edit data yang telah diterangkan sebelumnya (ada dua tahap). Adapun perintah SQL yang akan digunakan adalah :

```
DELETE FROM nama_tabel WHERE feild_kunci="$id";
```

Untuk mempraktekkannya, cobalah tuliskan skrip berikut:

Skrip hapus_tamu.php

```
<?
include "koneksi_bukutamu.php";
$id=$_GET['id'];
$hapus="DELETE from tamu where no_tamu='$id'";
$hasil=mysql_query($hapus);
if ($hasil)
{
    header("location:tampil_tamu_admin.php");
}
else
{
    echo "Proses penghapusan data gagal";
}
?>
```

Sekarang coba jalankan skrip tampil_tamu_admin.php, aka akan terlihat tampilan seperti pada gambar diatas. Coba klik link Hapus pada salah satu data.

Setelah link Hapus diklik, maka skrip hapus_tamu.php akan memproses dta yang dihapus dan apabila proses penghapusan data berhasil, maka secara otomatis halaman akan dialihkan ke halaman data tamu admin dan hasil penghapusan data tersebut langsung dapat dilihat.

9. PROSES PENCARIAN DATA

Proses pengolahan data terakhir adalah proses pencarian data. Perintah SQL yang akan digunakan adalah :

```
SELECT * FROM tabel WHERE $field LIKE '%id%';
```

Untuk mempraktekannya, tuliskan dua buah file skrip berikut :

Skrip form_cari.php

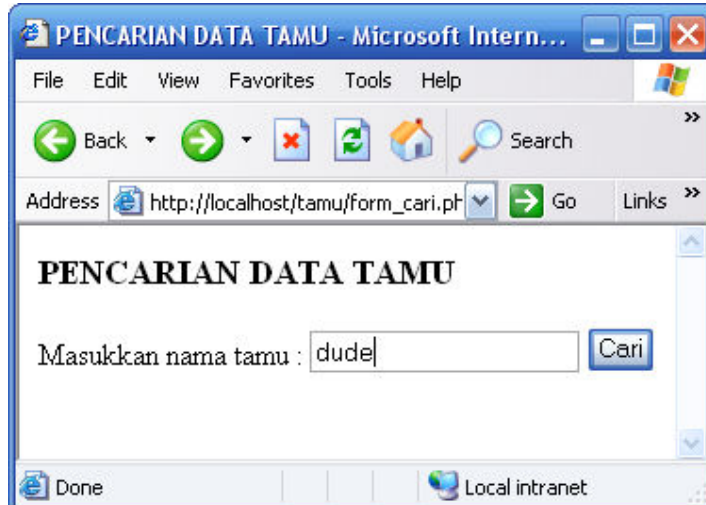
```
<html>
<head>
  <title>PENCARIAN DATA TAMU</title>
</head>
<body>
  <H3>PENCARIAN DATA TAMU</H3>
  <FORM METHOD=POST ACTION=cari_tamu.php>
  Masukkan nama tamu : <Input type=text name="nama">
  <input type=submit value=Cari>
</form>
</body>
</html>
```

Skrip cari_tamu.php

```
<?
echo "<h3 align=center>HASIL PENCARIAN</h3>";
include "koneksi_bukutamu.php";
$nama=$_POST[nama];
$cari=mysql_query("SELECT *FROM tamu WHERE nama like
'%"$nama%"");
$jumlah=mysql_num_rows($cari);
if($jumlah>0)
{
echo "Tamu yang bernama <B>$nama</b> ada : $jumlah
orang <br>";
echo "<table border=1> <tr> <th>Nama</th>
<th>Email</th> <th>Pesan</th> </tr>";
while ($data=mysql_fetch_array($cari))
{
echo "<tr><td>$data[nama]</td>";
echo "<td>$data[email]</td>";
echo "<td>$data[pesan]</td></tr>";
}
echo "</table>";
}
else
{
echo "Tamu dengan nama <b>$nama</b> tidak ada";
}
}
```

?>

Sekarang coba jalankan skrip form_cari.php. lalu isikan nama tamu yang akan dicari, misalnya dude. Lihat gambar dibawah ini.



Kemudian klik tombol Cari, maka skrip cari_tamu.php akan memproses data yang dicari dan apabila data yang dicari ada, maka datanya akan ditampilkan seperti gambar berikut:

