

MODUL 8

WEB PROGRAMMING : PHP 3

Tujuan :

1. Memahami tentang penggunaan Form
2. Memahami tentang perbedaan antara metode GET & POST
3. Memahami tentang pemrosesan file upload dengan nama yang sama
4. Memahami tentang pemrosesan file upload dengan nama yang berbeda

PENGOLAHAN DATA DARI FORM

Web menerima input dari user atau pengunjung menggunakan metode GET dan POST. GET akan mengirimkan data bersama dengan URL, sedangkan POST mengirimkannya secara terpisah.

User mengirimkan data input dengan mengisi teks atau pilihan pada attribut form html. Apa saja yang merupakan komponen form html diantaranya :

Form

```
<FORM ACTION=action base> form tags </FORM>  
<FORM METHOD=method> form tags </FORM>  
<FORM ENCTYPE=media type> form tags </FORM>  
<FORM ACTION=action base TARGET="target window name"> form tags </FORM>  
<FORM SCRIPT=URL> form tags </FORM>
```

Note : *target window name* diisikan berupa:

```
_blank  
_self  
_parent  
_top
```

Form Input Check Box

```
<INPUT TYPE=CHECKBOX NAME=name VALUE=value>  
<INPUT TYPE=CHECKBOX NAME=name VALUE=value CHECKED>
```

Form Input File

```
<INPUT TYPE=FILE NAME=name ACCEPT=mime type list>
```

Form Input Hidden

```
<INPUT TYPE=HIDDEN NAME=name VALUE=value>
```

Form Input Image

```
<INPUT TYPE=IMAGE NAME=name SRC="URL">  
<INPUT TYPE=IMAGE NAME=name SRC="URL" ALIGN="alignment">
```

Form Input Password

```
<INPUT TYPE=PASSWORD NAME=name>
<INPUT TYPE=PASSWORD NAME=name MAXLENGTH=length>
<INPUT TYPE=PASSWORD NAME=name SIZE=size>
<INPUT TYPE=PASSWORD NAME=name VALUE=value>
```

Form Input Radio Button

```
<INPUT TYPE=RADIO NAME=name VALUE=value>
<INPUT TYPE=RADIO NAME=name VALUE=value CHECKED>
```

Form Input Range

This is not supported in many browsers.

```
<INPUT TYPE=RANGE NAME=name MIN=min MAX=max>
<INPUT TYPE=RANGE NAME=name MIN=min> MAX=max VALUE=value>
```

Form Input Reset

```
<INPUT TYPE=RESET>
```

Form Input Submit

```
<INPUT TYPE=SUBMIT>
<INPUT TYPE=SUBMIT NAME=name>
<INPUT TYPE=SUBMIT VALUE=value>
```

Form Input Button

```
<INPUT TYPE=BUTTON>
<INPUT TYPE=BUTTON NAME=name>
<INPUT TYPE=BUTTON VALUE=value>
```

Form Input Text

```
<INPUT TYPE=TEXT NAME=name>
<INPUT TYPE=TEXT NAME=name MAXLENGTH=length>
<INPUT TYPE=TEXT NAME=name SIZE=size>
<INPUT TYPE=TEXT NAME=name VALUE=value>
```

Form Select

```
<SELECT NAME=name> option entries </SELECT>
<SELECT NAME=name MULTIPLE> option entries </SELECT>
<SELECT NAME=name SIZE=size> option entries </SELECT>
<SELECT NAME=name SRC=URL WIDTH=width HEIGHT=height UNITS=units> option entries </SELECT>
<OPTION> content
<OPTION SELECTED> content
<OPTION VALUE=value> content
<OPTION SHAPE=shape>
```

Form Text Area

```
<TEXTAREA NAME=name COLS=# columns ROWS=# rows> content </TEXTAREA>
```

UPLOAD FILE

Upload merupakan proses penyalinan data yang ada dari komputer client ke komputer server. Fasilitas upload memungkinkan aplikasi web misalnya pendaftaran keanggotaan yang menggunakan foto, atau lainnya. Namun biasanya upload digunakan untuk mempermudah administrasi.

LATIHAN – LATIHAN :

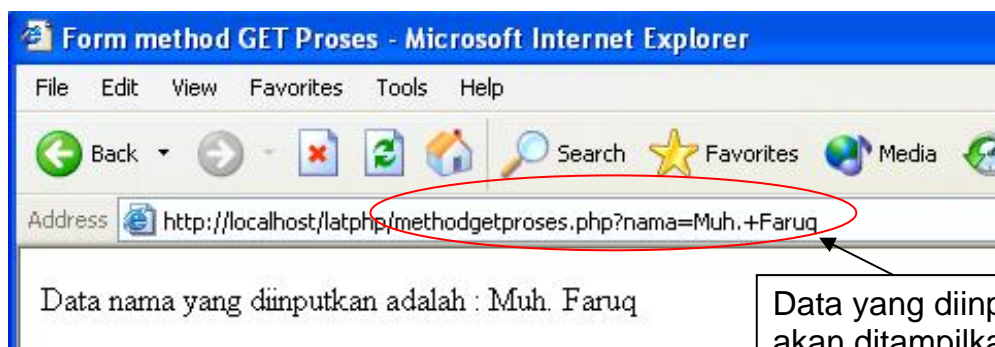
1. Proses Form menggunakan metode : GET

a. Untuk membuat inputan, dan beri nama file : metodeget.php

```
<html>
<head>
  <title> Form method GET </title>
</head>
<body>
<h1>Input</h1>
<form action="methodgetproses.php" method="get">
Masukkan nama : <input type="text" name="nama" size="25">
<input type="submit" value="Proses">
</form>
</body>
</html>
```

b. Buat file untuk memproses variabel yang diberikan oleh file metodeget.php, beri nama filenya : methodgetproses.php

```
<html>
<head>
  <title> Method GET proses </title>
</head>
<body>
Data nama yang diinputkan adalah : <?echo $_GET["nama"]?>
</body>
</html>
```



Keterangan :

Dengan metode GET maka data yang dikirimkan ke file php yang berikutnya akan ditampilkan, cara ini tidak cocok bila data yang dimasukkan berupa password.

2. Proses Form menggunakan metode : POST

a. Untuk membuat inputan, dan beri nama file : metodepost.php

```
<html>
<head>
  <title> Form method GET </title>
</head>
<body>
<h1>Input</h1>
<form action="methodpostproses.php" method="post">
Masukkan nama : <input type="text" name="nama" size="25">
<input type="submit" value="Proses">
</form>
</body>
</html>
```

b. Buat file untuk memproses variabel yang diberikan oleh file metodepost.php, beri nama filenya : methodpostproses.php

```
<html>
<head>
  <title> Method POST proses </title>
</head>
<body>
Data nama yang diinputkan adalah : <?echo $_POST["nama"]?>
</body>
</html>
```



Keterangan :

Dengan metode POST, maka data yang dikirimkan akan disembunyikan, hal ini sangat cocok bila dipakai untuk inputan berupa password. Untuk selain password, pemakaian GET & POST tergantung pada kebutuhan.

3. Proses input data dengan password menggunakan metode : POST

a. Untuk membuat inputan, dan beri nama file : metodepost.php

```
<html>
<head>
    <title>Contoh Form dengan POST</title>
</head>
<body>
<form action="proc_login.php" method="post">
<h1>Only for authorized user</h1>
<hr>
Username : <input type="text" name="username">
<br>
Password : <input type="password" name="password">
<br>
<input type="submit" value="login">
</form>
</body>
</html>
```

b. Buat file untuk memproses variabel, beri nama filenya : proc_login.php

```
<html>
<head>
    <title>Proses Input</title>
</head>
<body>
<?
$username=$_POST["username"];
$password=$_POST["password"];
?>
Username : <?echo $username?>
<br>
Password : <?echo $password?>
<br>
</body>
</html>
```

4. Perbandingan 2 bilangan menggunakan metode : POST

a. Untuk membuat inputan, dan beri nama file : bandingnilai.php

```
<html>
<head>
  <title>Contoh Form dengan POST</title>
</head>
<body>
<form action="proc_post.php" method="post">
Bil I :
<input type="text" name="bil1">
<br>
Bil II:
<input type="text" name="bil2">
<br>
<input type="submit" value="Bandingkan">
</form>
</body>
</html>
```

b. Buat file untuk memproses variabel, beri nama filenya : proc_post.php

```
<html>
<head>
  <title>Proses Input</title>
</head>
<body>
<?
$bill=$_POST["bil1"];
$bil2=$_POST["bil2"];
?>
<h1>Perbandingan Bilangan </h1>
<hr>
Bil I : <?echo $bill?>
<br>
Bil II: <?echo $bil2?>
<br>
<?
if ($bill<$bil2){
echo "$bill lebih kecil dari $bil2";
}
elseif($bill>$bil2){
echo "$bill lebih besar dari $bil2";
}
else{
echo "$bill sama dengan $bil2";
}
?>
</body>
</html>
```

5. Pemrosesan Buku Tamu dengan metode : POST

a. Untuk membuat inputan, dan beri nama file : bukutamu.php

```

<html>
<head>
  <title>Contoh Form dengan POST</title>
</head>
<body>
<h1>Buku Tamu</h1>
Komentar dan saran sangat kami butuhkan untuk
meningkatkan kualitas situs kami.
<hr>
<form action="proc_bukutamu.php" method="post">
<pre>
Nama anda      : <input type="text" name="nama" size="25"
                  maxlength="50">
Email address  : <input type="text" name="email" size="25"
                  maxlength="50">
Komentar       : <textarea name="komentar" cols="40"
                  rows="5"> </textarea>
<input type="submit" value="kirim">
<input type="reset" value="ulangi">
</pre>
</form>
</body>
</html>

```

b. Buat file untuk memproses variabel, beri nama filenya : proc_bukutamu.php

```

<html>
<head>
  <title>Buku Tamu</title>
</head>
<body>
<?
$nama=$_POST["nama"];
$email=$_POST["email"];
$komentar=$_POST["komentar"];
?>
<h1>Data Buku Tamu </h1>
<hr>
Nama anda      : <?echo $nama?>
<br>
Email address  : <?echo $email?>
<br>
Komentar       :
<textarea name="komentar" cols="40"
rows="5"><?echo $komentar?> </textarea>
<br>
</body>
</html>

```

6. Penyimpanan file yang di-Upload, dengan nama file yang tersimpan sama semuanya. Sebelum menjalankan program dibawah ini, maka register_globals yang ada di php.ini harus di-on-kan terlebih dahulu.

a. Untuk membuat inputan, dan beri nama file : fileupload.php

```
<html>
<head>
    <title>Form untuk input nama file</title>
</head>
<body>
<h1>Input nama file untuk Upload</h1>
<br>Klik Browse untuk memilih !
<form enctype="multipart/form-data" method="post"
action="do_upload.php">
<input type="hidden" name="MAX_FILE_SIZE" value="10000">
Nama File : <input type="file" name="file1" size="30">
<br>
<input type="submit" value="Upload">
</form>
</body>
</html>
```

b. Buat file untuk memproses variabel, beri nama filenya : do_upload.php, hasil upload bisa dilihat di direktori tempat menyimpan file php.

```
<html>
<head>
    <title></title>
</head>
<body>
<h1>Simpan file yang diupload</h1>
<br>
<?
if ($file1!="none") {
    copy("$file1","hasilupload.txt") or
    die ("No files");
}
else {
    die("Tidak ada file yang diupload");
}
?>
</body>
</html>
```

KETERANGAN :

Pada tag form_upload atribut enctype="multipart/form-data" harus ada. Atribut ini akan digunakan untuk memberitahu browser dan server bahwa yang dikirim bukan hanya data nama file, namun juga isi dari nama file.

Diperlukan juga sebuah variabel dengan nama MAX_FILE_SIZE dengan tipe hidden, yang harus disediakan untuk mengendalikan besar file maksimum yang diupload ke

server. Variabel ini harus diletakkan sebelum input dengan type 'file', nilai pada atribut value menunjukkan jumlah byte maksimum.

7. Cara lain untuk memproses file upload dengan nama file yang berbeda -> \$HTTP_POST_FILES

Ganti program 5.b. dengan program dibawah ini, dengan terlebih dahulu membuat folder dengan nama files di directory tempat menyimpan file php.

```

<html>
<head>
    <title></title>
</head>
<body>
<h1>Simpan file yang diupload</h1>
<?
$namafile = $HTTP_POST_FILES['file1']['name'];
?>
<p>Nama File : <?echo $namafile;?></p>
<br>
<?
if ($file1!="none") {
    copy("$file1","files/$namafile") or
    die ("No files");
}
else {
    die("Tidak ada file yang diupload");
}
?>
</body>
</html>

```

TUGAS :

1. Buatlah program untuk melakukan proses perhitungan dibawah ini, dengan GET.

