

MODUL 01 - Mengenal PHP & Proses Form dengan Metode POST dan GET

Perkenalan dengan PHP

PHP adalah bahasa scripting (bahasa pemrograman script) server-side yang dirancang untuk mengembangkan web. PHP pertama kali dikembangkan oleh Rasmus Lerdorf pada tahun 1994 dan saat ini dikelola oleh The PHP Group. (Wikipedia)

PHP adalah bahasa pemrograman script server-side yang digunakan untuk pengembangan web. PHP disebut bahasa pemrograman server-side karena kode PHP diproses pada komputer server. Hal ini berbeda dengan bahasa pemrograman client-side seperti JavaScript yang diproses pada web browser (client).

Pada awalnya PHP merupakan singkatan dari Personal Home Page. Sesuai dengan namanya, PHP digunakan untuk membuat website pribadi. Dalam beberapa tahun perkembangannya, PHP menjelma menjadi bahasa pemrograman web yang powerful dan tidak hanya digunakan untuk membuat web sederhana, tetapi juga website populer yang digunakan oleh jutaan orang seperti Wikipedia, wordpress, facebook, dll. Saat ini PHP adalah singkatan dari PHP: Hypertext Preprocessor, sebuah kepanjangan rekursif.

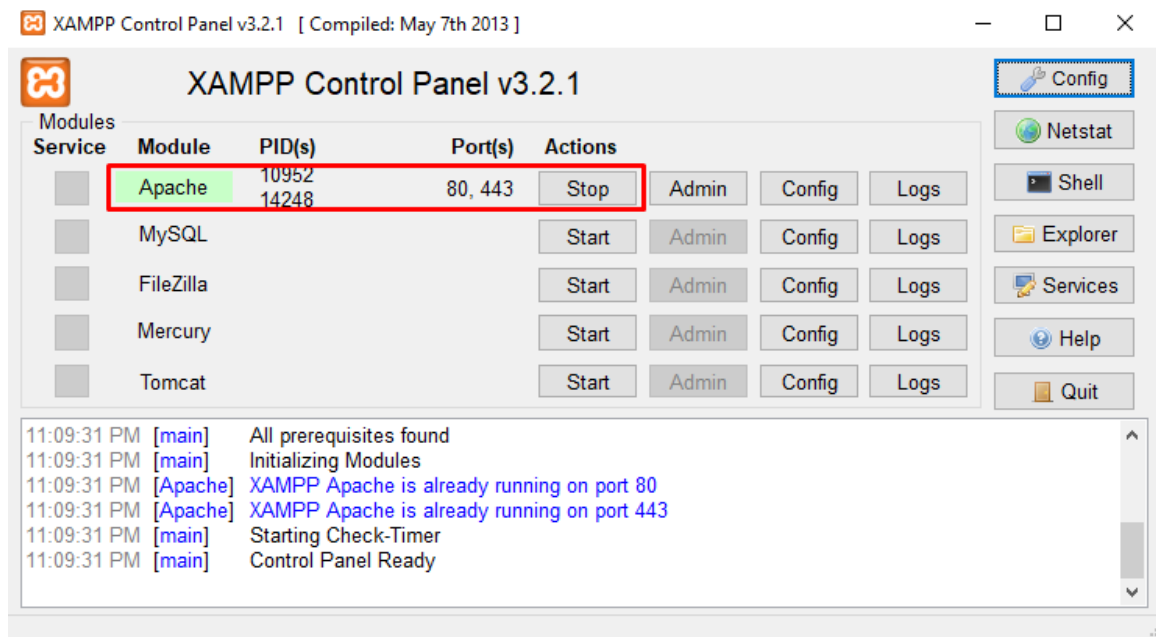
Fungsi PHP dalam Pemrograman Web

Untuk membuat halaman web, sebenarnya PHP bukanlah bahasa pemrograman yang wajib digunakan. Kita bisa saja membuat website hanya menggunakan HTML saja. Web yang dihasilkan dengan HTML (dan CSS) dikenal dengan website statis, dimana konten dan halaman web bersifat tetap.

Sebagai perbandingan, website dinamis yang bisa dibuat menggunakan PHP adalah situs web yang bisa menyesuaikan tampilan konten tergantung situasi. Website dinamis juga bisa menyimpan data ke dalam database, membuat halaman yang berubah-ubah sesuai input dari user, memproses form, dll.

Latihan 1: Menampilkan Hello World

1. Buka aplikasi XAMPP dan aktifkan modul Apache.



2. Buat sebuah file “latihan1.php” dan simpan pada direktori tempat XAMPP terinstall. Sebagai contoh, jika XAMPP terinstall pada drive D.

D:\xampp\htdocs\web_lanjut\latihan1.php

```
<!doctype html>
<html>
<head>
  <title>Latihan PHP</title>
</head>
<body>

<?php

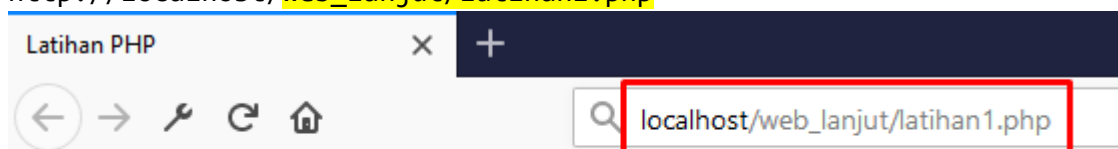
echo "<h1>Hello World</h1>";

?>

</body>
</html>
```

3. Untuk melihat hasilnya di web browser, ketikkan URL berikut:

http://localhost/web_lanjut/latihan1.php



Hello World

Variabel dalam PHP

1. Penulisan variabel harus diawali dengan tanda \$

Variabel dalam PHP harus diawali dengan tanda dollar(\$). Setelah tanda \$, sebuah variabel PHP harus diikuti dengan karakter pertama berupa huruf atau underscore (_), kemudian karakter kedua dan seterusnya bisa menggunakan huruf, angka atau underscore (_). Dengan aturan tersebut, variabel di dalam PHP tidak bisa diawali dengan angka.

Berikut adalah contoh penulisan variabel yang benar dalam PHP:

```
<?php
$i;
$nama;
$Umur;
$_lokasi_memori;
$ANGKA_MAKSIMUM;
?>
```

Berikut adalah contoh penulisan variabel yang salah dalam PHP:

```
<?php
$4ever; // variabel tidak boleh diawali angka
$_salah satu; // variabel tidak boleh mengandung spasi
$nama*^; // variabel tidak boleh mengandung karakter khusus * dan ^
?>
```

2. Variabel dalam PHP bersifat case sensitive

PHP membedakan variabel yang ditulis dengan huruf besar dan kecil (bersifat case sensitive), sehingga \$belajar tidak sama dengan \$Belajar dan \$BELAJAR, ketiganya akan dianggap sebagai variabel yang berbeda.

Untuk menghindari kesalahan program dikarenakan salah merujuk variabel, disarankan menggunakan huruf kecil untuk seluruh nama variabel.

3. Variabel dalam PHP tidak memerlukan deklarasi terlebih dahulu

Jika Anda pernah mempelajari bahasa pemrograman seperti C, C++ dan Java, di dalam bahasa pemrograman tersebut sebuah variabel harus dideklarasikan terlebih dahulu sebelum digunakan.

Namun di dalam PHP, variabel tidak perlu dideklarasikan terlebih dahulu. Anda bebas membuat variabel baru di tengah-tengah kode program, dan langsung menggunakannya tanpa dideklarasikan terlebih dahulu.

4. Variabel dalam PHP tidak bertipe

Dalam kelompok bahasa pemrograman, PHP termasuk Loosely Type Language, yaitu jenis bahasa pemrograman yang variabelnya tidak terikat pada sebuah tipe tertentu. Hal ini berbeda jika dibandingkan dengan bahasa pemrograman seperti C atau Java, dimana jika Anda membuat sebuah variabel bertipe integer, maka variabel itu hanya bisa menampung nilai angka.

Di dalam PHP, setiap variabel bebas diisi dengan nilai apa saja, seperti contoh berikut:

```
<?php
$a = 17; // nilai variabel a berisi angka (integer)
$a = "aku"; // nilai variabel a diubah menjadi kata (string)
$a = 17.42; // nilai variabel a diubah menjadi decimal (float)
?>
```

5. Variabel Sistem PHP (Predefined Variables)

Predefined Variables atau terjemahan bebasnya Variabel Sistem PHP, adalah beberapa variabel yang telah didefinisikan secara sistem oleh PHP, dan kita sebaiknya tidak membuat variabel dengan nama yang sama.

Beberapa contoh Predefined Variables:

- \$GLOBALS
- \$_SERVER
- \$_GET
- \$_POST
- \$_FILES
- \$_REQUEST
- \$_SESSION
- \$_ENV
- \$_COOKIE
- \$php_errormsg
- \$HTTP_RAW_POST_DATA
- \$http_response_header
- \$argc
- \$argv

Sumber: <http://www.php.net/reserved.variables>

Latihan 2: Menampilkan Hasil Input Form HTML dengan PHP (Metode POST)

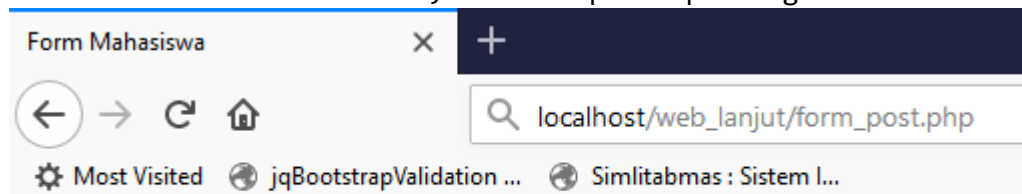
1. Buatlah file `form_post.php` dan simpan pada direktori yang sama dengan Latihan sebelumnya. Ketikkan kode berikut:

```
<!doctype html>
<html>
<head>
  <title>Form Mahasiswa</title>
</head>
<body>

<h2>Form Mahasiswa</h2>
<form action="proses_post.php" method="post">
  NIM:
  <input type="text" name="nim_mhs" />
  <br />
  Nama:
  <input type="text" name="nama_mhs" />
  <br />
  Jurusan:
  <select name="jur_mhs">
    <option value="IF">Informatika</option>
    <option value="SI">Sistem Informasi</option>
    <option value="IK">Ilmu Komunikasi</option>
    <option value="MN">Manajemen</option>
    <option value="AK">Akuntansi</option>
    <option value="ARK">Arsitektur</option>
    <option value="DKV">Desain Komunikasi Visual</option>
  </select>
  <br />
  <input type="submit" value="Kirim" />
</form>

</body>
</html>
```

Jika dibuka di web browser, akan tampil seperti gambar berikut:



Form Mahasiswa

NIM:

Nama:

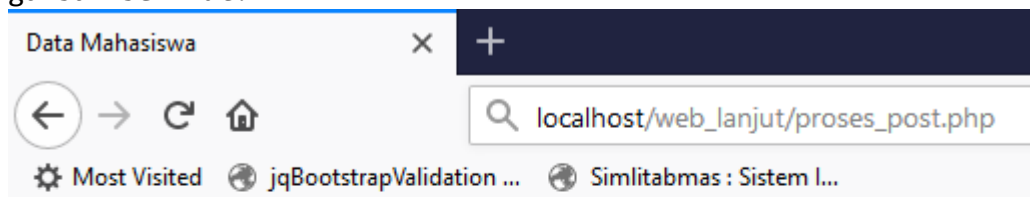
Jurusan:

2. Buat file proses_post.php untuk menampilkan data yang diinput di form sebelumnya.

```
<!doctype html>
<html>
<head>
    <title>Data Mahasiswa</title>
</head>
<body>
    <h2>Data Mahasiswa</h2>
    <?php
        $nim = $_POST['nim_mhs'];
        $nama = $_POST['nama_mhs'];
        $jurusan = $_POST['jur_mhs'];

        echo "NIM: $nim";
        echo "<br />";
        echo "Nama: $nama";
        echo "<br />";
        echo "Jurusan: $jurusan";
    ?>
</body>
</html>
```

Jika form diisi dan tombol “Kirim” di-klik, maka akan tampil hasil seperti gambar berikut:



Data Mahasiswa

NIM: 2017100100

Nama: Magnum Ice

Jurusan: IF

Latihan 3: Menampilkan Hasil Input Form HTML dengan PHP (Metode GET)

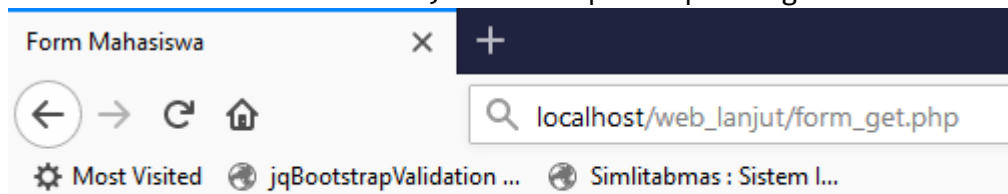
1. Buatlah file `form_get.php` dan simpan pada direktori yang sama dengan Latihan sebelumnya. Ketikkan kode berikut:

```
<!doctype html>
<html>
<head>
  <title>Form Mahasiswa</title>
</head>
<body>

<h2>Form Mahasiswa</h2>
<form action="proses_get.php" method="get">
  NIM:
  <input type="text" name="nim_mhs" />
  <br />
  Nama:
  <input type="text" name="nama_mhs" />
  <br />
  Jurusan:
  <select name="jur_mhs">
    <option value="IF">Informatika</option>
    <option value="SI">Sistem Informasi</option>
    <option value="IK">Ilmu Komunikasi</option>
    <option value="MN">Manajemen</option>
    <option value="AK">Akuntansi</option>
    <option value="ARK">Arsitektur</option>
    <option value="DKV">Desain Komunikasi Visual</option>
  </select>
  <br />
  <input type="submit" value="Kirim" />
</form>

</body>
</html>
```

Jika dibuka di web browser, akan tampil seperti gambar berikut:



Form Mahasiswa

NIM:

Nama:

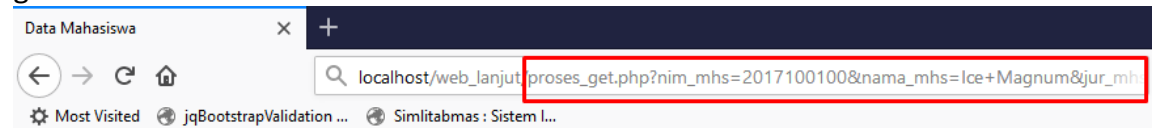
Jurusan:

2. Buat file proses_get.php untuk menampilkan data yang diinput di form sebelumnya.

```
<!doctype html>
<html>
<head>
    <title>Data Mahasiswa</title>
</head>
<body>
    <h2>Data Mahasiswa</h2>
    <?php
        $nim = $_GET['nim_mhs'];
        $nama = $_GET['nama_mhs'];
        $jurusan = $_GET['jur_mhs'];

        echo "NIM: $nim";
        echo "<br />";
        echo "Nama: $nama";
        echo "<br />";
        echo "Jurusan: $jurusan";
    ?>
</body>
</html>
```

Jika form diisi dan tombol “Kirim” di-klik, maka akan tampil hasil seperti gambar berikut:



Data Mahasiswa

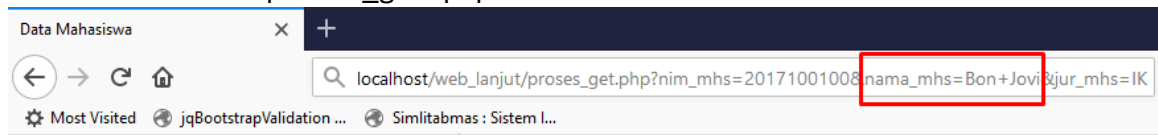
NIM: 2017100100
Nama: Ice Magnum
Jurusan: IK

Perhatikan URL yang dihasilkan, bandingkan dengan latihan sebelumnya. URL yang dihasilkan adalah sebagai berikut:

localhost/web_lanjut/proses_get.php?nim_mhs=2017100100&nama_mhs=Ice+Magnum&jur_mhs=IK

Perhatikan bahwa data dari form dikirim ke URL, dan file proses_get.php mengambil data dari URL.

3. Coba ganti data nama_mhs pada URL menjadi "nama_mhs=Bon+Jovi", kemudian refresh halaman proses_get.php.



Data Mahasiswa

NIM: 2017100100

Nama: Bon Jovi

Jurusan: IK

Data Nama berubah walaupun tidak melalui input form.

Latihan 4: Menerima Data Dari Link (URL)

1. Buat file "link_get.php" dan ketikkan kode berikut:

```
<!doctype html>
<html>
<head>
    <title>Link Get</title>
</head>
<body>

<a href="read_get.php?pesan=Selamat+Belajar+Pemrograman+Web">Click
Me!</a>

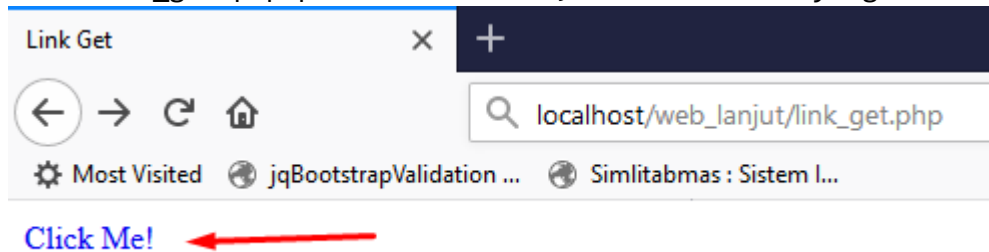
</body>
</html>
```

2. Buat file "read_get.php" dan ketikkan kode berikut:

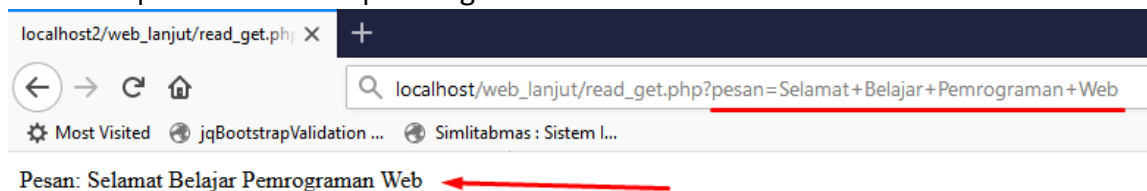
```
<?php

if(isset($_GET['pesan'])){
    echo "Pesan: $_GET[pesan]";
}else{
    echo "Tidak Ada Pesan";
}
```

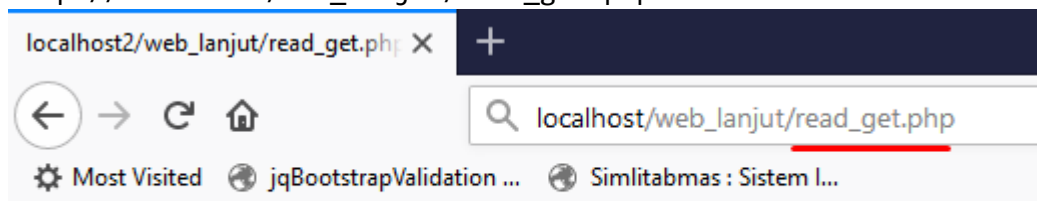
3. Buka link_get.php pada web browser, dan klik link yang ada.



4. Akan tampil halaman seperti gambar berikut:



5. Coba buka tab baru, dan ketikkan URL:
http://localhost/web_lanjut/read_get.php



Tidak Ada Pesan

Latihan 5: Latihan Mandiri

Buatlah form seperti tampilan dibawah ini, kemudian buat kode PHP yang akan dijalankan ketika tombol “Kirim” di-klik untuk menampilkan data-data yang diinput pada form.

The image shows a web browser window with the title "Latihan Mandiri". The address bar displays "http://localhost/web_lanjut/latihan2.php". The main content area contains a form titled "Data Mahasiswa". The form includes the following fields and controls:

- NIM**: A single-line text input field.
- Nama**: A single-line text input field.
- Jurusan**: A dropdown menu with "Informatika" selected.
- Jenis Kelamin**: Three radio buttons labeled "Laki-laki", "Perempuan", and "Tidak tahu".
- Minat**: A list of checkboxes for "Teknologi", "Bisnis", "Sastra", "Seni", "Olah Raga", and "Pendidikan".
- Alamat**: A multi-line text input field.
- Kirim**: A button to submit the form.