

PHP

Query String, Fungsi isset, Fungsi
Include, Fungsi Time & Date, File
Upload

Query String

- Variabel super global `$_GET` berisi nilai yang dikirimkan melalui query string
- Query String digunakan untuk mengirim data antar halaman
- `$_GET` dapat digunakan menggunakan form maupun tanpa form

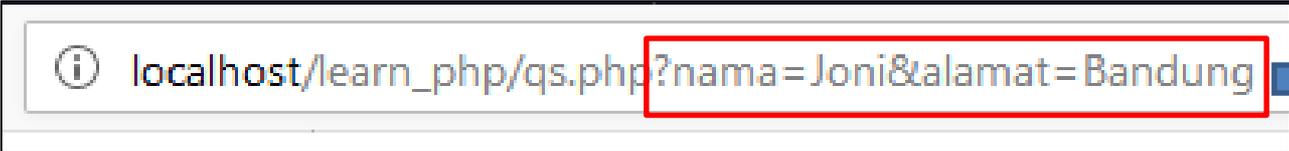
Query String

Contoh 1

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Belajar PHP</title>
5   </head>
6   <body>
7     <h1>Superglobals Variable $_GET</h1>
8     <?php
9       print_r($_GET);
10    ?>
11 </body>
12 </html>
```

Fungsi **print_r()** digunakan untuk melihat isi variable `$_GET`.

Pada URL, setelah tulisan `.php` tambahkan Query String

 `localhost/learn_php/qs.php?nama=Joni&alamat=Bandung` → Query String

Tanda “?” merupakan pemisah antara file.php dengan Query String. Tanda “&” merupakan pemisah antar Query String

Query String

Contoh 1

Dengan menggunakan fungsi **print_r()** maka isi dari variabel `$_GET` dapat terlihat. Isi variabel `$_GET` berbentuk Array

Superglobals Variable `$_GET`

Array ([nama] => Joni [alamat] => Bandung)



Index nama dan alamat merupakan hasil dari Query String

Query String

Contoh 1

Untuk menampilkan ke dalam web browser, ubahlah script php menjadi seperti ini

```
<?php
    $nama=$_GET["nama"];
    $alamat=$_GET["alamat"];

    echo "Nama Saya : $nama <br>";
    echo "Alamat Saya : $alamat";
?>
```

Maka outputnya akan menjadi seperti ini

Superglobals Variable \$_GET

Nama Saya : Joni

Alamat Saya : Bandung

Query String

Contoh 2. Mengirimkan data antar halaman

Query.php

```
1 <!DOCTYPE html>
2 <html>
3   <head>
4     <title>Belajar PHP</title>
5   </head>
6   <body>
7     <h1>Query String Antar Halaman </h1>
8     <?php
9       $nama="Joni";
10      $alamat="Bandung";
11
12      $query_string="?nama={$nama}&alamat={$alamat}";
13    ?>
14    <a href="proses.php<?php echo $query_string; ?>">Link Menuju proses.php</a>
15  </body>
16 </html>
```

Query String

Contoh 2. Mengirimkan data antar halaman

Proses.php

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Belajar PHP</title>
5      </head>
6      <body>
7          <h1>Halaman Proses</h1>
8          <?php
9              echo "Nama : " . $_GET["nama"] . "<br>";
10             echo "Alamat : " . $_GET["alamat"] . "<br>";
11         ?>
12     </body>
13 </html>
```

Query String

Contoh 2. Mengirimkan data antar halaman

Proses.php

```
1  <!DOCTYPE html>
2  <html>
3      <head>
4          <title>Belajar PHP</title>
5      </head>
6      <body>
7          <h1>Halaman Proses</h1>
8          <?php
9              echo "Nama : " . $_GET["nama"] . "<br>";
10             echo "Alamat : " . $_GET["alamat"] . "<br>";
11         ?>
12     </body>
13 </html>
```

Query String

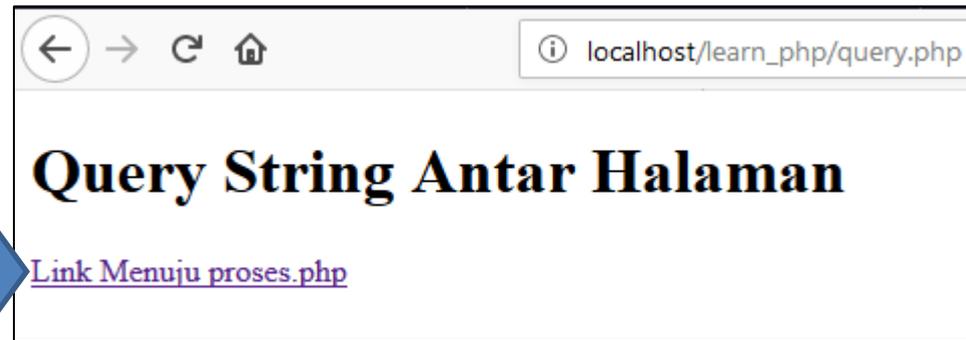
Contoh 2. Mengirimkan data antar halaman

Ketika menyorot link, maka di pojok kiri bawah browser akan muncul link preview seperti di bawah ini

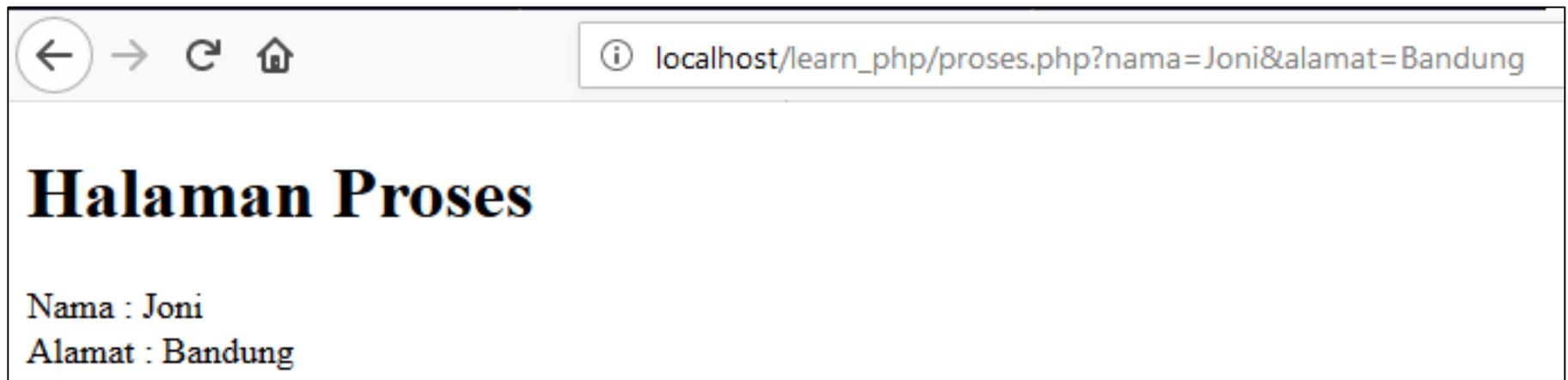
`localhost/learn_php/proses.php?nama=Joni&alamat=Bandung`



Output dari file query.php



Output dari file proses.php



Fungsi Isset

- Fungsi `isset()` digunakan untuk memeriksa apakah suatu `variable` sudah didefinisikan (`set`) atau tidak

Fungsi Isset

Contoh Penerapan Fungsi isset() pada Form Login

Login.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Isset</title>
5 </head>
6 <body>
7   <h1>Form Login</h1>
8   <form action="ProsesLogin.php" method="post">
9     <p>Username : <input type="text" name="user" ></p>
10    <p>Password: <input type="password" name="pass"></p>
11    <input type="submit" value="Login" name="submit">
12  </form>
13 </body>
14 </html>
```

Form Login

Username :

Password:

Login

Isikan username dan password, misal
username = admin, password = Unikom123

Fungsi Isset

Contoh Penerapan Fungsi isset() pada Form Login

ProsesLogin.php

```
1  <!DOCTYPE html>
2  <html>
3    <head>
4      <title>Isset</title>
5    </head>
6    <body>
7      <h1>Halaman Proses</h1>
8      <?php
9        print_r($_POST);
10     ?>
11    </body>
12  </html>
```

Halaman Proses

```
Array ( [user] => admin [pass] => Unikom123 [submit] => Login )
```

Untuk lebih memahami cara kerja fungsi isset(), lihat isi variable `$_POST` menggunakan fungsi `print_r()`. Terlihat bahwa isi variable `$_POST` berbentuk array dimana terdapat index user dengan value admin, pass dengan value Unikom123, dan submit dengan value Login

*[user], [pass], [submit] adalah nama dari elemen html pada file login.html

Fungsi Isset

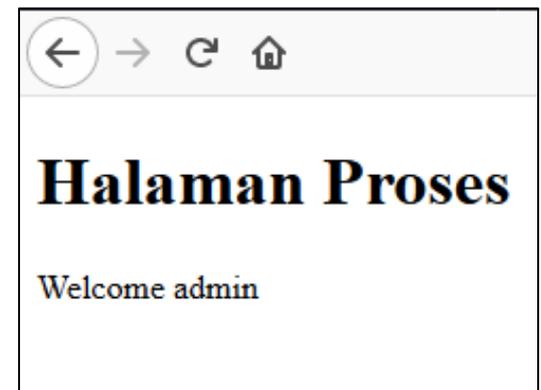
Contoh Penerapan Fungsi isset() pada Form Login

Ubah script php pada file ProsesLogin.php menjadi seperti di bawah ini

```
<?php
if (isset($_POST["submit"])) {
    echo "Welcome " . $_POST["user"];
}
else {
    echo "Dilarang Mengakses Halaman Ini, Silahkan Login Terlebih Dahulu!!";
}
?>
```

Pada script di atas telah ditambahkan fungsi isset() dimana akan memeriksa apakah variable submit sudah terdefiniskan atau tidak

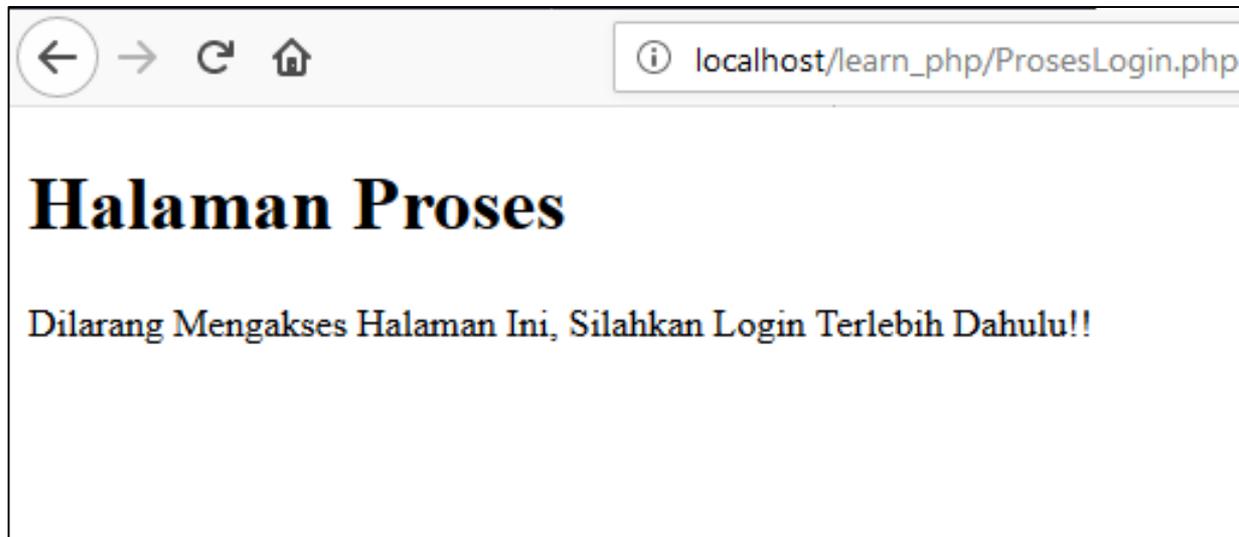
Output file ProsesLogin.php



Fungsi Isset

Contoh Penerapan Fungsi `isset()` pada Form Login

Jika mengakses langsung file `ProsesLogin.php` tanpa melalui Form Login, maka akan muncul tampilan seperti ini



Hal ini terjadi karena variable `submit` belum didefinisikan / terlewatkan

Fungsi Include

- Fungsi `include()` digunakan untuk memanggil kode program file lain ke dalam halaman PHP
- Dapat digunakan untuk memanggil file PHP maupun HTML
- Penulisan fungsi include :
`include("NamaFile.php")` atau
`include("direktori/NamaFile.php")`

Fungsi Include

- Fungsi include() untuk memanggil file HTML

header.html

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>Belajar PHP</title>
6 </head>
7 <body>
8
9 <header>
10  <h1>Ini adalah bagian header web</h1>
11 </header>
```

footer.html

```
1 <footer>
2   <p>Copyright @Sistem Informasi 2016</p>
3 </footer>
4 </body>
5 </html>
```

index.php

```
1 <?php
2   include("header.html");
3 ?>
4
5   <p>Ini adalah konten utama web</p>
6
7 <?php
8   include("footer.html");
9 ?>
```

File index.php memanggil file header.html dan footer.html dengan fungsi include()

Fungsi Include

Fungsi `include()` dapat digunakan untuk memanggil file HTML ketika kita membangun halaman web dengan cara memecah struktur halaman web (header, footer, dan lain-lain) menjadi beberapa file HTML yang terpisah

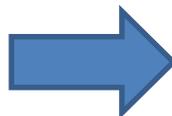
Output dari file `index.php`

Ini adalah bagian header web

Ini adalah konten utama web

Copyright @Sistem Informasi 2016

File `index.php` memanggil file `header.html` dan `footer.html` menjadi satu halaman web utuh



page source dari file `index.php`

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <meta charset="UTF-8">
5   <title>Belajar PHP</title>
6 </head>
7 <body>
8
9 <header>
10  <h1>Ini adalah bagian header web</h1>
11 </header>
12
13  <p>Ini adalah konten utama web</p>
14
15 <footer>
16   <p>Copyright @Sistem Informasi 2016</p>
17 </footer>
18 </body>
19 </html>
```

Fungsi Include

- Fungsi `include()` untuk memanggil file PHP (contoh fungsi `include` untuk memanggil function)

function.php

```
1  <?php
2      function welcome($nama){
3          return "Selamat Datang $nama di Pemrograman PHP";
4      }
5  ?>
```

index.php

```
1  <?php
2      include("function.php");
3      echo welcome("Joni");
4  ?>
```

Pada file `index.php`, file `function.php` yang berisi function dipanggil dengan menggunakan fungsi `include()`. Kemudian memanggil function `welcome()` dengan parameter "Joni" dan ditampilkan ke web browser

Fungsi Include

- Penulisan fungsi include -> include("NamaFile.php")
atau include("direktori/NamaFile.php")

function.php

```
1  <?php
2      function welcome($nama){
3          return "Selamat Datang $nama di Pemrograman PHP";
4      }
5  ?>
```

index.php

```
1  <?php
2      include("function.php");
3      echo welcome("Joni");
4  ?>
```

Pada file index.php, file function.php yang berisi function dipanggil dengan menggunakan fungsi include(). Kemudian memanggil function welcome() dengan parameter "Joni" dan ditampilkan ke web browser

Fungsi Time & Date

- Digunakan untuk mengolah data yang berhubungan dengan waktu
- Terdapat 2 fungsi yang dapat digunakan, yaitu `time()` dan `date()`

Fungsi Time

- Fungsi `time()` digunakan untuk menunjukkan waktu saat ini dalam satuan detik, dimulai dari tanggal 1 Januari 1970, 00:00:00 GMT

```
1  <?php
2  ..... echo time();
3  ?>
```

Fungsi Date

- Fungsi `date()` digunakan untuk menunjukkan waktu saat ini dalam bentuk format waktu umum

```
1  <?php
2  ..... echo date("d m Y");
3  ?>
```

Fungsi Date

- Terdapat beberapa format tanggal dan waktu pada fungsi date

Format angka/nama hari:

- **d**: Angka hari dengan format 2 digit dan didahului angka nol: 01 hingga 31.
- **j**: Angka hari dengan format 2 digit dan tanpa angka nol: 1 hingga 31.
- **w**: Angka hari untuk satu minggu, 0 hingga 6.
- **z**: Angka hari untuk satu tahun, 0 hingga 365.
- **D**: Nama hari dengan singkatan 3 karakter bahasa inggris: Sun, Mon hingga Sat.
- **I**: Nama hari dalam bahasa inggris: Sunday, Monday hingga Saturday.
- **S**: Awalan 2 digit karakter dalam bahasa inggris untuk nama hari: st, nd, rd dan th. Biasanya digunakan dengan format “j”.

Fungsi Date

Format angka/nama bulan:

- **m**: Angka bulan dengan format 2 digit dan didahului angka nol: 01 hingga 12.
- **n**: Angka bulan dengan format 2 digit dan tanpa angka nol: 1 hingga 12.
- **M**: Nama bulan dengan singkatan 3 karakter bahasa inggris: Jan, Feb hingga Dec.
- **F**: Nama bulan dalam bahasa inggris: January, February hingga December.
- **t**: Total jumlah hari dalam 1 bulan : 28 hingga 31.

Format angka/nama tahun:

- **L**: 1 jika tahun kabisat, 0 jika bukan tahun kabisat.
- **Y**: Angka tahun dengan 4 digit, seperti 1998, 2002 dan 2016
- **y**: Angka tahun dengan 2 digit, seperti 98, 02 dan 16

Fungsi Date

Format angka untuk waktu:

- **a**: am atau pm, dengan huruf kecil.
- **A**: AM atau PM, dengan huruf besar.
- **g**: Angka jam dalam format 12 jam, tanpa awalan nol: 1 hingga 12.
- **G**: Angka jam dalam format 24 jam, tanpa awalan nol: 0 hingga 23.
- **h**: Angka jam dalam format 12 jam, dengan awalan nol: 01 hingga 12.
- **H**: Angka jam dalam format 24 jam, dengan awalan nol: 01 hingga 23.
- **i**: Angka menit dengan awalan nol: 00 hingga 59.
- **s**: Angka detik dengan awal nol: 00 hingga 59.

```
1  <?php
2  ..... date_default_timezone_set("Asia/Jakarta");
3  ..... echo date("l, d F Y, H:i:s");
4  ?>
```

File Upload

- Mengupload suatu file melalui form
- Menggunakan superglobal variable **\$_FILES**
- Pada tag `<form>` tambahkan atribut **enctype="multipart/form-data"**

File Upload

FormUpload.php

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <meta charset="UTF-8">
5  |   <title>Belajar Upload File</title>
6  </head>
7  <body>
8  <h1>Upload File</h1>
9  |   <form action="" method="post" enctype="multipart/form-data">
10 |     <p>Upload File: <input type="file" name="file_upload"></p>
11 |     <input type="submit" name="submit" value="Proses Upload">
12 |   </form>
13 </body>
14 </html>
```

File Upload

Output

Upload File

Upload File: No file selected.

Tambahkan script php sebelum tag <html>

```
1  <?php
2      if (isset($_POST["submit"])) {
3          echo "<pre>";
4          print_r($_FILES);
5          echo "</pre>";
6      }
7  ?>
```

```
Array
(
    [file_upload] => Array
        (
            [name] => Koala.jpg
            [type] => image/jpeg
            [tmp_name] => C:\xampp\tmp\phpBB6F.tmp
            [error] => 0
            [size] => 244350
        )
)
```

*file berhasil di upload & disimpan temporary di C:\xampp\tmp

Memindahkan Lokasi File Upload

1. Buatlah folder “upload” pada direktori tempat menyimpan file php
2. Edit script php pada file FormUpload.php
3. Fungsi **move_uploaded_file** berfungsi untuk memindahkan file upload dari penyimpanan sementara ke folder “upload”

```
1 ▼ <?php
2 ▼   if (isset($_POST["submit"])) {
3     $nama_folder="upload";
4     $tmp = $_FILES["file_upload"]["tmp_name"];
5     $nama_file = $_FILES["file_upload"]["name"];
6     move_uploaded_file($tmp, "$nama_folder/$nama_file");
7   }
8   ?>
```

Thank you

