

PHP dan FORM

\$_GET

HTML Form: gunakan method="get"

```
<form method="get" action="">
    <input type="text" name="keyword"/>
    <input type="submit" name="submit"
    value="Search"/>
</form>
```

PHP Code:

```
if (isset($_GET['keyword'])) {
    $keyword = $_GET['keyword'];
    $sql = 'SELECT *
        FROM article
        WHERE title
            LIKE "%' . $keyword . '%"';
}
```

Pada GET, data dikirim lewat url, gunakan GET untuk data yang tidak sensitif, seperti untuk pencarian dan pagination.

\$_POST

HTML Form: gunakan method="post"

```
<form method="post" action="">
    <input type="password" name="pass"/>
    <input type="submit" name="submit"
    value="Login"/>
</form>
```

PHP Code:

```
if (isset($_POST['submit'])) {
    $sql = 'SELECT user, password
        FROM user
        WHERE password = "'.$_POST['pass'].''";
    $result = mysqli_query($sql);
}
```

Pada POST, data dikirim lewat HTTP HEADER, sehingga tidak terlihat. Gunakan POST untuk data sensitif, seperti password, atau ketika terjadi perubahan data pada database seperti form edit atau delete data.

\$_FILES

HTML Form: gunakan enctype="multipart/form-data"

```
<form method="post" action=""
enctype="multipart/form-data">
    <input type="file" name="user_file"/>
    <input type="submit" name="submit"
    value="Upload"/>
</form>
```

PHP Code:

```
if (isset($_POST['submit'])) {
    echo '<pre>'; print_r($_FILES); echo'</pre>';
}
```

Nilai Input sama dengan data yang disubmit

Pada contoh dibawah, jika form disubmit, maka input keyword akan berisi string yang dicari (Method GET):

```
<form method="get" action="">
    <input type="text" name="keyword"
        value="<?=@$_GET['keyword'] ?: ""?>"/>
    <input type="submit" name="submit"
        value="Search"/>
</form>
```

Penambahan @ untuk menghilangkan warning ketika variabel \$_GET['keyword'] belum terdefinisi, misal ketika form pertama kali dibuka. Contoh untuk method POST

```
<form method="post" action="">
    <input type="text" name="nama"
        placeholder="Nama Anda"
        value="<?=@$_POST['nama'] ?: ""?>"/>
    <input type="submit" name="submit"
        value="Simpan"/>
</form>
```

ELEMEN SELECT

Usahakan gunakan loop ketika membuat elemen select

```
<form method="post" action="">
    <?php
        $bulan = array('Januari',
                      'Februari',
                      'Maret'
                    );
        echo '<select name="bulan">';
        foreach ($bulan as $key => $nama_bulan) {
            $selected = @$_POST['bulan'] == $key
                ? ' SELECTED="SELECTED"' : '';
            echo '<option value="'.$key.'"'.
                $selected .'>' .
                $nama_bulan .
            '</option>';
        }
        echo '</select>';
    ?>
    <input type="submit" name="submit"
        value="Simpan"/>
</form>
```

Penggunaan loop akan memudahkan ketika ada penambahan maupun pengurangan elemen, disamping itu juga memudahkan pemberian atribut selected="selected". Opsi akan membuat opsi otomatis terpilih sesuai dengan yang dipilih user.

Example: Menampilkan data database

```
1. <?php
2. // SUBMIT
3. $hasil_query = false;
4. $error = $warning = array();
5. if (isset($_POST['submit']))
6. {
7.     $form_error = '';
8.     if (!trim($_POST['nama_barang'])) {
9.         $error = 'Nama barang harus diisi';
10.    }
11.
12.    if (!$error) {
13.        $conn = @mysqli_connect('localhost', 'root', '', 'db');
14.        if (!$conn) {
15.            $error[] = mysqli_connect_error($conn);
16.        } else {
17.            $where = trim($_POST['nama_barang'])
18.                  ? 'nama_barang LIKE "%'
19.                    .$_POST['nama_barang']. "%"
20.                  : '';
21.
22.            $sql = 'SELECT * FROM barang WHERE ' . $where;
23.            $result = mysqli_query ($conn, $sql);
24.
25.            // RESULT
26.            if (!$result) {
27.                $error[] = mysqli_error($conn)
28.                          . '<br><strong>SQL Query</strong>: ' . $sql;
29.            } else {
30.                $num_rows = mysqli_num_rows($result);
31.                if (!$num_rows) {
32.                    $warning[] = 'Data tidak ditemukan';
33.                } else {
34.                    $hasil_query = true;
35.                }
36.            }
37.        }
38.    }
39. }?>
<html>
<head> <title>PHP dan FORM</title> </head>
<body>
<?php
```

```
44. if ($error) {
45.     echo '<div>Error: ' . join($error, ', ') . '</div>';
46.
47.
48. if ($warning) {
49.     echo '<div>' . join($warning, ', ') . '</div>';
50. }
51. ?>
52. <form action="" method="post">
53.     <div>
54.         <label>Nama Barang</label>
55.         <input type="text" name="nama_barang"
56.                 value=<?=@$_POST['nama_barang'] ?: ''?> />
57.     </div>
58.     <div>
59.         <input type="submit" name="submit" value="Submit"/>
60.     </div>
61. </form>
62. <?php
63. if (isset($_POST['submit'])) {
64.
65.     if ($hasil_query)
66.     {
67.         echo 'Ditemukan ' . $num_rows . ' data';
68.         $thead = ' <tr>
69.             <th>No</th>
70.             <th>Nama Barang</th>
71.         </tr> ';
72.
73.         echo ' <table> <thead>' . $thead . '</thead>
74.             <tbody>';
75.
76.         $no = 1;
77.         while($row = mysqli_fetch_array($result)) {
78.             echo '<tr> <td>' . $no . '</td>
79.                   <td>' . $row['nama_barang'] . '</td>
80.             </tr> ';
81.             $no++;
82.         }
83.         echo '</tbody></table>';
84.     }
85. }
86. </div>
87. </body>
88. </html>
```