

2018

[FORM HTML DAN PHP]

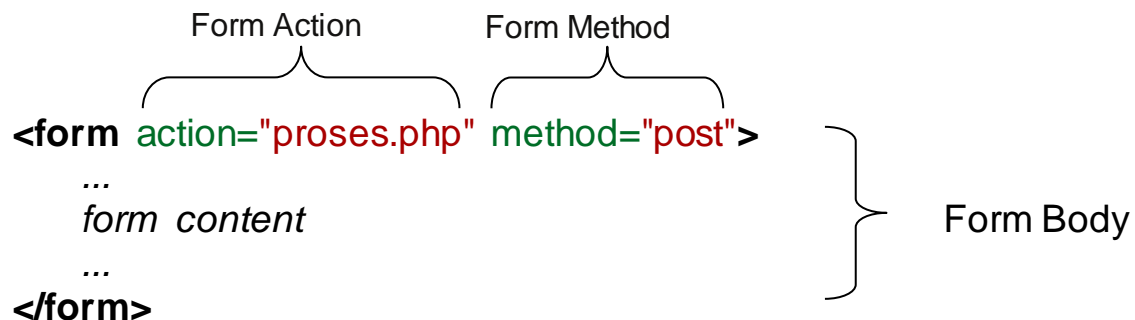
PEMROGRAMAN WEB LANJUT - S1 INFORMATIKA

AINUL YAQIN, M.KOM

1. Konsep penggunaan Form
2. Form Handling
 - a. Method GET
 - b. Method POST
3. Form Validation
4. Form Required

Form adalah mekanisme input / pengiriman data melalui sebuah halaman web. Sebuah web form memungkinkan user untuk memasukkan data yang akan dikirim ke server untuk diolah lebih lanjut. Sedangkan *method* adalah mekanisme atau cara yang digunakan sebuah halaman untuk mengirimkan informasi tersebut.

1. Struktur Umum



Struktur umum dari form diatas dapat dijelaskan sebagai berikut:

- Form element adalah elemen HTML yang digunakan untuk membuat sebuah form. Pembuatannya didahului dengan syntax **<form>** dan ditutup dengan **</form>**.
- Form body adalah tempat dimana elemen-elemen input data akan diletakkan.
- Form action menentukan kemana dikirimnya data yang telah dimasukkan user. Dalam contoh struktur diatas, data yang dimasukkan user akan dikirimkan ke halaman **proses.php** untuk diolah lebih lanjut. Jika value pada atribut **action** tidak ditentukan / dikosongkan, maka informasi itu akan dikirimkan ke halaman itu sendiri
- Form method akan menentukan metode yang akan digunakan dalam pengiriman informasi. Lebih lanjut mengenai form method akan dijelaskan secara detail pada sub bab "Method".

2. HTML Form Element

Sebuah form tidak dapat berdiri sendiri, ia memerlukan elemen HTML lain yang berfungsi sebagai properti masukan atau penampung data dan informasi yang akan dikirimkan oleh user. Diantara elemen-elemen tersebut adalah:

a. Text Box

Digunakan untuk memasukkan input data berupa text. Syntax umum dari elemen input text box adalah `<input type="text" />`. Berikut ini atribut yang mungkin muncul pada elemen input text box :

- size = ukuran dari text box dalam bentuk karakter
- maxsize = maksimal karakter yang dapat ditampung dalam sebuah text box
- name = nama dari elemen input
- value = menampilkan isi / data default dari text box

b. Password

Digunakan untuk menangani input data berupa password. Syntax umum dari elemen input password adalah `<input type="password" />`. Berikut ini atribut yang mungkin muncul pada elemen input password :

- size = ukuran dari password dalam bentuk karakter
- maxsize = maksimal karakter yang dapat ditampung dalam sebuah elemen input password
- name = nama dari elemen input
- value = menampilkan isi default dari password

c. Hidden Input Element

Digunakan untuk mengirimkan data yang tidak terlihat pada *browser* user. Syntax umum dari elemen input hidden adalah `<input type="hidden" />`. Berikut ini atribut yang mungkin muncul pada elemen input hidden :

- name = nama dari elemen input
- value = isi / data dari elemen input

d. Check Box

Check box digunakan untuk mengirimkan data yang berisifat *multiple choice*. Yaitu pengiriman data dengan banyak pilihan sekaligus. Syntax umum dari elemen input check box adalah `<input type="checkbox" />`. Berikut ini atribut yang mungkin muncul pada elemen input check box :

- name = nama dari elemen input
- value = isi / data dari elemen input
- checked = menandai secara otomatis elemen input check box

e. Radio Button

Digunakan untuk mengirimkan data yang berisifat *single choice*. Yaitu pengiriman data dengan hanya satu pilihan saja. Seperti halnya radio tape pada kehidupan nyata yang memiliki banyak pilihan tombol, tetapi kita hanya dapat memilih satu tombol saja pada satu waktu.

Syntax umum dari elemen input radio button adalah `<input type="radio" />`. Berikut ini atribut yang mungkin muncul pada elemen input radio button :

- name = nama dari elemen input
- value = isi / data dari elemen input
- checked = menandai secara otomatis elemen input radio button

f. Push Button

Biasanya digunakan / dikombinasikan bersamaan dengan Javascript atau VBScript untuk menghasilkan sebuah aksi.

Syntax umum dari elemen input radio button adalah `<input type="button" />`. Berikut ini atribut yang mungkin muncul pada elemen input radio button :

- name = nama dari elemen input
- value = isi / data dari elemen input push button

g. Submit Button

Setiap web form membutuhkan sebuah tombol submit untuk memulai pengiriman data yang telah dimasukkan oleh user, hal inilah yang kita sebut sebagai submit button.

Syntax umum dari elemen input submit button adalah `<input type="submit" />`. Berikut ini atribut yang mungkin muncul pada elemen input submit button :

- name = nama dari elemen input
- value = isi / data dari elemen input submit button

h. Image Submit Button

Digunakan untuk menggantikan tombol standar submit button dengan sebuah gambar.

Syntax umum dari elemen input ini adalah `<input type="image" src="image_location.jpg" />`. Berikut ini atribut yang mungkin muncul pada elemen input image submit button :

- name = nama dari elemen input image submit button

i. Reset Button

Digunakan untuk mengembalikan data pada sebuah form kepada keadaan semula (reset). Syntax umum dari elemen input ini adalah `<input type="reset" />`.

Berikut ini atribut yang mungkin muncul pada elemen input reset button :

- name = nama dari elemen input reset button

j. Text Area

Digunakan untuk memasukkan text dengan lebih leluasa seperti halnya mengetikkan data pada text editor. Tidak seperti text box, text area memungkinkan kita untuk mengetikkan data lebih dari 1 baris informasi.

Syntax umum dari elemen ini adalah `<textarea>...</textarea>`. Berikut ini atribut yang mungkin muncul pada elemen input text area :

- name = nama dari elemen text area
- rows = banyaknya baris yang ditampilkan dalam text area
- cols = lebar element input text area

k. Select

Sama halnya dengan radio button, elemen select juga menyediakan banyak pilihan data dan hanya satu item saja yang bisa dipilih. Yang membedakannya keduanya adalah cara penyajiannya, yaitu dengan *drop down list*.

Syntax umum dari elemen input ini adalah `<select>...</select>`. Berikut ini atribut yang mungkin muncul pada elemen select :

- name = nama dari elemen input
- size = membatasi jumlah pilihan yang dapat dilihat

Berbeda dengan elemen input yang lain, select membutuhkan elemen HTML lain yang digunakan untuk mendefinisikan pilihan-pilihan yang terdapat pada select. Elemen yang dimaksud adalah option.

Syntax umum dari elemen option ini adalah `<option>...</option>`. Berikut ini atribut yang mungkin muncul pada elemen option

- selected = menandai elemen option yang menjadi pilihan default element

04.latihan_01.html

```
<html>
<head>
<title>Contoh element Select & Option</title>
</head>
<body>
<form action="" method="post">
Warna favorit kamu :
<select name="warna">
<option value="red">Merah</option>
<option value="yellow">Kuning</option>
<option value="green" selected="selected">Hijau</option>
<option value="black">Hitam</option>
</select>
</form>
</body>
</html>
```



3. Method

Method adalah atribut sebuah form yang digunakan untuk menentukan bagaimana informasi / data yang telah dimasukkan user akan dikirimkan. Terdapat dua cara yang dapat digunakan:

- a. Method GET
- b. Method POST

Berikut ini akan dijelaskan cara kerja dua metode diatas beserta pengolahan hasil pengiriman data / informasi oleh bahasa pemrograman PHP.

3.1 Metode GET

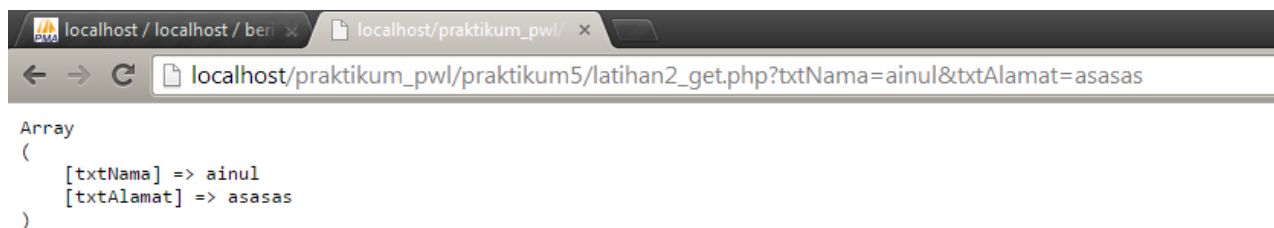
Metode GET memungkinkan kita untuk mengirimkan data / informasi yang melalui URI (*browser address bar*). Untuk lebih jelasnya lihat contoh berikut:

04.latihan_02.html

```
<html>
  <head>
    <title>Contoh Form</title>
  </head>
  <body>
    <form action="latihan_02_get.php" method="get">
      Nama :
      &nbsp;<input type="text" name="txtNama" />
      <br />
      Alamat :
      <textarea name="txtAlamat"></textarea>
      <br />
      <input type="submit" value="Submit" />
    </form>
  </body>
</html>
```

04.latihan_02_get.php

```
<?php
echo "<pre>";
print_r($_GET); echo
"</pre>";
?>
```

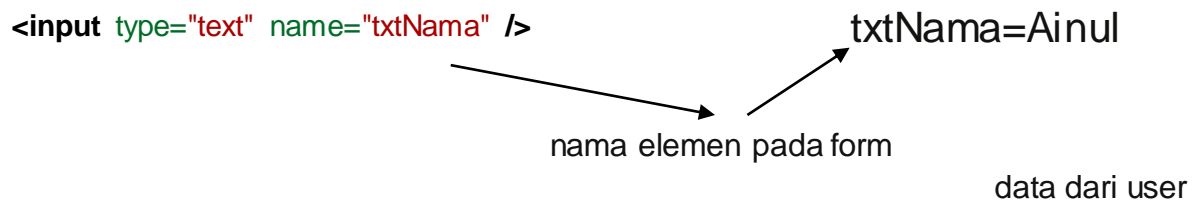


Kedua program diatas saling berkaitan satu sama lain. Skrip 04.latihan_02.html berfungsi untuk membuat sebuah form. Sedangkan 04.latihan_02_get.php berguna untuk memproses data yang telah dimasukkan oleh user. Berikut ini adalah cara kerja kedua program tersebut:

1. Skrip 04.latihan_02.html menyediakan sebuah form dengan tiga elemen. Yaitu element *input* dengan tipe *text*, elemen *textarea* dan sebuah elemen *input* dengan tipe *submit*.
2. User mengisikan informasi yang dibutuhkan pada form yang telah dibuat.
3. User menekan tombol "Submit" untuk mengirimkan data. Data dikirimkan melalui URI dan diproses oleh skrip tujuan (skrip tujuan didefinisikan pada form melalui atribut `action="04.latihan_02_get.php"`).

Perhatikan **keterangan a** pada gambar diatas, dimana **nama** elemen yang terletak pada form dan **data** yang diinputkan user akan dikirimkan melalui URI. Untuk pembahasan selanjutnya, elemen input yang dikirimkan melewati URI ini akan disebut sebagai variabel **GET**.

Berikut ini adalah contoh transformasi dari elemen input pada sebuah form menjadi variabel GET :



Jika terdapat lebih dari satu variabel GET, antara satu variabel GET satu dengan yang lainnya dipisahkan oleh tanda "&". Sedangkan tanda "?" (tanda tanya) memberitahukan kepada kita bahwa string setelah tanda ? merupakan awal dari variabel GET.

Contoh :

http://localhost/web/04.latihan_02_get.php?txtNama=Ainul&txtAlamat=Yogyakarta

4. Skrip `04.latihan_02_get.php` ditugaskan untuk memproses input data yang telah dikirimkan melalui form.
5. Dalam PHP, variabel GET yang dikirimkan tersebut akan diperlakukan sebagai sebuah array dengan index asosiatif. Setiap variabel GET akan ditangani / disimpan oleh satu array yang bernama `$_GET`. Oleh karena dianggap sebagai sebuah array, fungsi `print_r()` dapat digunakan untuk melihat deskripsi dari elemen array beserta index asosiatifnya.

`txtNama=Ainul` \longrightarrow `$_GET['txtNama'] = "Ainul";`

Dari keterangan diatas dapat kita ambil kesimpulan bahwa variabel GET `txtNama` dengan value `Ainul` dapat diperlakukan sebagai array `$_GET` dengan index asosiatif `'txtNama'` dan value `"Ainul"`. Begitu pula dengan cara navigasi (pengaksesan nilai, lihat lagi modul 3 pada bagian array) dari array tersebut. Untuk lebih jelasnya ubah skrip `04.latihan_02_get.php` menjadi seperti dibawah ini (pastikan atribut *form action* pada skrip anda menuju kepada `04.latihan_03.php`):

`04.latihan_03.php`

`<?php`

```
echo "Nama : ".$_GET['txtNama'];
echo "<br />";
echo "Alamat : ".$_GET['txtAlamat'];
```

`?>`



6. Dalam prakteknya, tidak semua variabel GET didapatkan dari pengiriman data dengan melewati sebuah form. Seringkali variabel GET juga didefinisikan langsung melalui tautan / link. Contoh : <http://google.com/search?q=apa+yang+dicari> atau <http://sesuatu.com/detail.php?id=15>.

3.2 Metode POST

Karena beberapa alasan, tidak semua data dapat dikirimkan melalui metode GET. Misalnya data login, akan sangat riskan jika username dan password yang kita inputkan juga dapat dilihat oleh orang lain.

Dalam kehidupan nyata, metode POST dapat diilustrasikan pada proses pengiriman surat melalui jasa POS Indonesia. Dimana isi surat dalam amplop (data / informasi) tidak dapat kita lihat. Yang terlihat hanya alamat penerima surat. Berikut ini adalah contoh program menggunakan metode POST yang merupakan modifikasi skrip 04.latihan_02.html.

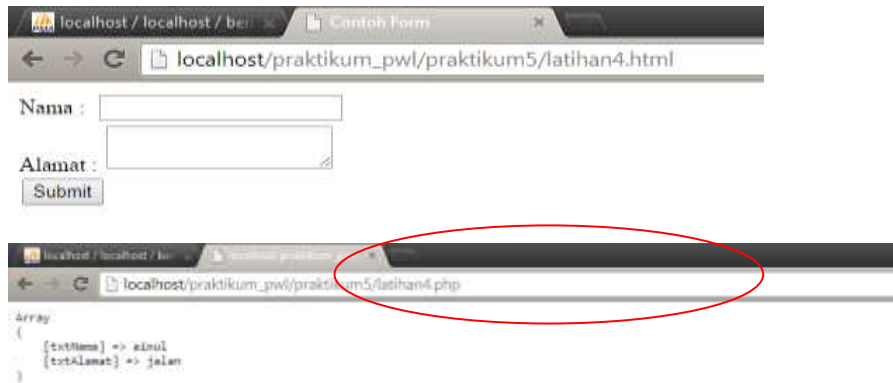
04.latihan_04.html

```
<html>
  <head>
    <title>Contoh Form</title>
  </head>
  <body>
    <form action="04.latihan_04.php" method="post">
      Nama :
      &nbsp;<input type="text" name="txtNama" />
      <br />
      Alamat :
      <textarea name="txtAlamat"></textarea>
      <br />
      <input type="submit" value="Submit" />
    </form>
  </body>
</html>
```

04.latihan_04.php

```
<?php
echo "<pre>";
print_r($_POST);
echo "</pre>";

?>
```



Kedua program diatas saling berkaitan satu sama lain. Skrip 04.latihan_04.html berfungsi untuk membuat sebuah form. Sedangkan 04.latihan_04.php berguna untuk memproses data yang telah dimasukkan oleh user. Berikut ini adalah cara kerja kedua program


tersebut:

1. Skrip `04.latihan_04.html` menyediakan sebuah form dengan tiga elemen. Yaitu element `input` dengan tipe `text`, elemen `textarea` dan sebuah elemen `input` dengan tipe `submit`.
2. User mengisi informasi yang dibutuhkan pada form yang telah dibuat.
3. User menekan tombol "Submit" untuk mengirimkan data. Data dikirimkan melalui metode POST dan diproses oleh skrip tujuan (skrip tujuan didefinisikan pada form melalui atribut `action="04.latihan_04.php"`).

Tidak seperti metode GET, metode POST tidak mengirimkan data / informasi melalui URI (variabel GET). Seperti yang telah diilustrasikan diatas, kita hanya dapat melihat skrip tujuan dimana data / informasi akan diproses. Sedangkan data / informasinya dikirimkan dibalik layar melalui metode POST. Pada gambar diatas, hal ini diperlihatkan pada **keterangan a**.

4. Skrip `04.latihan_04.php` ditugaskan untuk memproses input data yang telah dikirimkan melalui form.
5. Dalam PHP, variabel POST yang dikirimkan tersebut akan diperlakukan sebagai sebuah array dengan index asosiatif. Setiap variabel POST akan ditangani / disimpan oleh satu array yang bernama `$_POST`. Oleh karena dianggap sebagai sebuah array, fungsi `print_r()` dapat digunakan untuk melihat deskripsi dari elemen array beserta index asosiatifnya.

Berikut ini adalah contoh transformasi dari elemen input pada sebuah form menjadi variabel POST :

`<input type="text" name="txtNama" />`  `$_POST['txtNama']`

Dari keterangan diatas dapat kita lihat bahwa elemen `input text` pada skrip

`04.latihan_04.html` akan diolah oleh skrip `04.latihan_04.php` sebagai array `$_POST`.

Nama elemen `input text`, yaitu `txtNama` akan diperlakukan sebagai index asosiatif `'txtNama'`.

Secara lengkap elemen `input text` tersebut akan ditangani sebagai `$_POST['txtNama']`. Sedangkan data yang telah diisikan pada elemen `input text`

oleh user akan diperlakukan sebagai value dari elemen array tersebut. Lebih lengkapnya perhatikan keterangan berikut:

```
<input type="text" name="txtNama" />  
$_POST['txtNama'] = "Ainul";
```

Begitu pula dengan cara navigasi (pengaksesan nilai, lihat lagi modul 3 pada bagian array) dari array tersebut. Untuk lebih jelasnya ubah skrip 04.latihan_04.php menjadi seperti dibawah ini (pastikan atribut *form action* pada skrip anda menuju kepada 04.latihan_05.php):

04.latihan_05.php

```
<?php  
echo "Nama : ".$_POST['txtNama'];  
echo "<br />";  
echo "Alamat : ".$_POST['txtAlamat'];  
?>
```

4. Contoh - Contoh

4.1 Metode GET

04.latihan_06.html

```
<html>
  <head><title>Input Data - Form Metode GET</title></head>
  <body>
    <h3>Formulir Biodata</h3>
    <form action="04.latihan_06.php" method="get">
      <table width="600px">
        <tr>
          <td width="30%">Nama</td>
          <td width="2%">:</td>
          <td><input type="text" name="txtNama" size="45" /></td>
        </tr>
        <tr>
          <td>Alamat</td>
          <td>:</td>
          <td><textarea name="txtAlamat" cols="45"></textarea></td>
        </tr>
        <tr>
          <td>Jenis Kelamin</td>
          <td>:</td>
          <td>
            <input type="radio" name="rdJenisKelamin" value="Laki-Laki" />
            Laki-Laki
            <input type="radio" name="rdJenisKelamin" value="Perempuan" />
            Perempuan
          </td>
        </tr>
        <tr>
          <td>Agama</td>
          <td>:</td>
          <td>
            <select name="cbAgama">
              <option value="0">- pilih -</option>
              <option value="is">Islam</option>
              <option value="kk">Kristen Katolik</option>
              <option value="kp">Kristen Protestan</option>
              <option value="hd">Hindu</option>
              <option value="bd">Budha</option>
            </select>
          </td>
        </tr>
        <tr>
          <td></td><td></td>
          <td>
            <input type="submit" value="Kirim Informasi" />
            <input type="reset" value="Reset" />
          </td>
        </tr>
      </table>
    </form>
  </body>
</html>
```

04.latihan_06.php

```
<html>
  <head>
    <title>Pengolahan Data - Form Metode GET</title>
  </head>
  <body>
    <h3>Informasi Biodata</h3>
    <table width="600px">
      <tr>
        <td width="30%">Nama</td>
        <td width="2%">:</td>
        <td><?php echo $_GET['txtNama']; ?></td>
      </tr>
      <tr>
        <td>Alamat</td>
        <td>:</td>
        <td><?php echo $_GET['txtAlamat']; ?></td>
      </tr>
      <tr>
        <td>Jenis Kelamin</td>
        <td>:</td>
        <td><?php echo $_GET['rdJenisKelamin']; ?></td>
      </tr>
      <tr>
        <td>Agama</td>
        <td>:</td>
        <td>
          <?php
            if($_GET['cbAgama']=="is"){
              echo "Islam";
            } else if($_GET['cbAgama']=="kk"){
              echo "Kristen Katolik";
            } else if($_GET['cbAgama']=="kp"){
              echo "Kristen Protestan";
            } else if($_GET['cbAgama']=="hd"){
              echo "Hindu";
            } else if($_GET['cbAgama']=="bd"){
              echo "Budha";
            } else {
              echo "Tidak ditentukan";
            }
          ?>
        </td>
      </tr>
    </table>
  </body>
</html>
```

4.2 Metode POST

04.latihan_07.html

```
<html>
  <head>
    <title>Input Data - Form Metode POST</title>
  </head>
  <body>
    <h3>Formulir Biodata</h3>
    <form action="04.latihan_07.php" method="post">
      <table width="600px">
        <tr>
          <td width="30%">Nama</td>
          <td width="2%">:</td>
          <td><input type="text" name="txtNama" size="45" /></td>
        </tr>
        <tr>
          <td>Alamat</td>
          <td>:</td>
          <td><textarea name="txtAlamat" cols="45"></textarea></td>
        </tr>
        <tr>
          <td>Jenis Kelamin</td>
          <td>:</td>
          <td>
            <input type="radio" name="rdJenisKelamin" value="Laki-Laki" />
            Laki-Laki
            <input type="radio" name="rdJenisKelamin" value="Perempuan" />
            Perempuan
          </td>
        </tr>
        <tr>
          <td>Agama</td>
          <td>:</td>
          <td>
            <select name="cbAgama">
              <option value="0">- pilih -</option>
              <option value="is">Islam</option>
              <option value="kk">Kristen Katolik</option>
              <option value="kp">Kristen Protestan</option>
              <option value="hd">Hindu</option>
              <option value="bd">Budha</option>
            </select>
          </td>
        </tr>
        <tr>
          <td></td>
          <td></td>
          <td>
            <input type="submit" value="Kirim Informasi" />
            <input type="reset" value="Reset" />
          </td>
        </tr>
      </table>
    </form>
  </body>
</html>
```


04.latihan_07.php

```
<html>
  <head>
    <title>Pengolahan Data - Form Metode POST</title>
  </head>
  <body>
    <h3>Informasi Biodata</h3>
    <table width="600px">
      <tr>
        <td width="30%">Nama</td>
        <td width="2%">:</td>
        <td><?php echo $_POST ['txtNama']; ?></td>
      </tr>
      <tr>
        <td>Alamat</td>
        <td>:</td>
        <td><?php echo $_POST ['txtAlamat']; ?></td>
      </tr>
      <tr>
        <td>Jenis Kelamin</td>
        <td>:</td>
        <td><?php echo $_POST ['rdJenisKelamin']; ?></td>
      </tr>
      <tr>
        <td>Agama</td>
        <td>:</td>
        <td>
          <?php
            if($_POST ['cbAgama']=="is"){
              echo "Islam";
            } else if($_POST ['cbAgama']=="kk"){
              echo "Kristen Katolik";
            } else if($_POST ['cbAgama']=="kp"){
              echo "Kristen Protestan";
            } else if($_POST ['cbAgama']=="hd"){
              echo "Hindu";
            } else if($_POST ['cbAgama']=="bd"){
              echo "Budha";
            } else {
              echo "Tidak ditentukan";
            }
          ?>
        </td>
      </tr>
    </table>
  </body>
</html>
```

5. Form Validation

```

<!DOCTYPE HTML>
<html>
<head>
</head>
<body>

<?php
// define variables and set to empty values
$name = $email = $gender = $comment = $website = "";

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $name = test_input($_POST["name"]);
    $email = test_input($_POST["email"]);
    $website = test_input($_POST["website"]);
    $comment = test_input($_POST["comment"]);
    $gender = test_input($_POST["gender"]);
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
?>

<h2>PHP Form Validation Example</h2>
<form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_S
ELF"]);?>">
    Name: <input type="text" name="name">
    <br><br>
    E-mail: <input type="text" name="email">
    <br><br>
    Website: <input type="text" name="website">
    <br><br>
    Comment: <textarea name="comment" rows="5" cols="40"></textarea>
    <br><br>
    Gender:
    <input type="radio" name="gender" value="female">Female
    <input type="radio" name="gender" value="male">Male
    <input type="radio" name="gender" value="other">Other
    <br><br>
    <input type="submit" name="submit" value="Submit">
</form>

```

```

<?php
echo "<h2>Your Input:</h2>";
echo $name;
echo "<br>";
echo $email;
echo "<br>";
echo $website;
echo "<br>";
echo $comment;
echo "<br>";
echo $gender;
?>

</body>
</html>

```

6. Form required

```

<!DOCTYPE HTML>
<html>
<head>
<style>
.error {color: #FF0000;}
</style>
</head>
<body>

<?php
// define variables and set to empty values
$nameErr = $emailErr = $genderErr = $websiteErr = "";
$name = $email = $gender = $comment = $website = "";

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    if (empty($_POST["name"])) {
        $nameErr = "Name is required";
    } else {
        $name = test_input($_POST["name"]);
    }

    if (empty($_POST["email"])) {
        $emailErr = "Email is required";
    } else {
        $email = test_input($_POST["email"]);
    }

    if (empty($_POST["website"])) {
        $website = "";
    } else {

```

```

    $website = test_input($_POST["website"]);
}

if (empty($_POST["comment"])) {
    $comment = "";
} else {
    $comment = test_input($_POST["comment"]);
}

if (empty($_POST["gender"])) {
    $genderErr = "Gender is required";
} else {
    $gender = test_input($_POST["gender"]);
}
}

function test_input($data) {
    $data = trim($data);
    $data = stripslashes($data);
    $data = htmlspecialchars($data);
    return $data;
}
?>

<h2>PHP Form Validation Example</h2>
<p><span class="error">* required field</span></p>
<form method="post" action="<?php echo htmlspecialchars($_SERVER["PHP_S
ELF"]);?>">
    Name: <input type="text" name="name">
    <span class="error">* <?php echo $nameErr;?></span>
    <br><br>
    E-mail: <input type="text" name="email">
    <span class="error">* <?php echo $emailErr;?></span>
    <br><br>
    Website: <input type="text" name="website">
    <span class="error"><?php echo $websiteErr;?></span>
    <br><br>
    Comment: <textarea name="comment" rows="5" cols="40"></textarea>
    <br><br>
    Gender:
    <input type="radio" name="gender" value="female">Female
    <input type="radio" name="gender" value="male">Male
    <input type="radio" name="gender" value="other">Other
    <span class="error">* <?php echo $genderErr;?></span>
    <br><br>
    <input type="submit" name="submit" value="Submit">
</form>

<?php

```

```
echo "<h2>Your Input:</h2>";  
echo $name;  
echo "<br>";  
echo $email;  
echo "<br>";  
echo $website;  
echo "<br>";  
echo $comment;  
echo "<br>";  
echo $gender;  
?>  
  
</body>  
</html>
```

