

BAB 7

PEMBUATAN FORM

7.1. Form dalam PHP

Dengan memanfaatkan form yang terdapat dalam HTML, kita dapat membuat program untuk menangani masukan yang bervariasi, pada dasarnya ada 5 jenis masukan yang biasa digunakan untuk memasukan data, yaitu: text, radio, checkbox, select, dan textarea

7.2. Form Tipe Text

Masukan bertipe text atau textbox merupakan masukan dalam html yang dapat digunakan untuk memasukan data dalam satu baris.

Berikut adalah contoh untuk memasukan input yang bertipe text

Sintak dasar:

```
<input type=text name=fields size=length>
```

Namafile: **input1.html**

```
<html>
<form action="proses1.php" method=post>
<center>
<h1>Program Penjualan
<hr>
<table>
<tr><td>Nama Barang<td>:<input name=namabarang size=20>
<tr><td>Harga<td>:<input name=harga size=10>
<tr><td>Jumlah<td>:<input name=jumlah size=2>
</table>
<hr>
<input type=submit value=Proses>
<input type=reset value=Reset>
```

Program Penjualan

Nama Barang :

Harga :

Jumlah :

Gambar 7.1 Input Program Data Barang

Sedangkan untuk menangani form input tadi, dapat dilakukan dengan menggunakan perintah POST untuk membaca isi variabel yang dikirim.

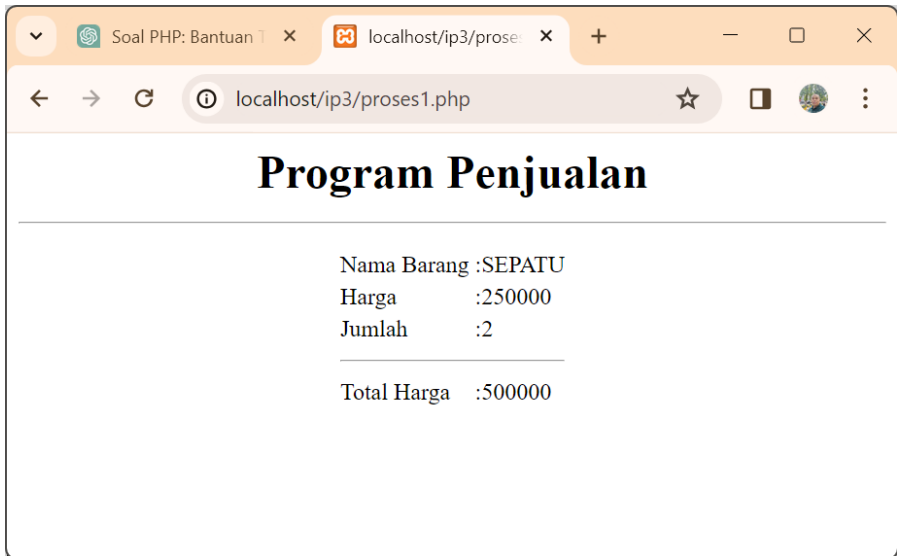
NamaFile: Proses1.php

```
<?php
echo "<center>";
echo "<h1>";
echo "Program Penjualan ";
echo "<hr>";
echo "<table>";
$namabarang=$_POST['namabarang'];
$harga=$_POST['harga'];
$jumlah=$_POST['jumlah'];

$total=$harga*$jumlah;

echo "<tr><td>Nama Barang<td>:$namabarang";
echo "<tr><td>Harga<td>:$harga";
echo "<tr><td>Jumlah<td>:$jumlah";
echo "<tr><td colspan=3><hr>";
echo "<tr><td>Total Harga<td>:$total";
?>
```

Selanjutnya ketika program dijalankan dan di Click tombol Proses, maka hasilnya akan ditampilkan seperti pada gambar 7.2.



Gambar 7.2. Menampilkan Hasil Proses

7.3. Form Tipe Radio

Input type radio berfungsi untuk membuat tombol **radio** atau tombol pilihan yang diisi dengan cara memilih salah satu tombol **radio** yang ada. **Radio** biasa digunakan untuk pilihan yang membatasi user untuk memilih satu dari pilihan yang ada.

Dalam penggunaan type **radio**, kita hanya memerlukan **tag input** dengan sebuah **atribut type radio**. Berikut format dasar **radio** dalam HTML:

Sintak:

```
<input type="radio" name=fields value=isivalue>Label
```

NamaFile:radio.html

```
<form action="proses_radio.php" method=post>
<center>
<font size=5>
Menggunakan Input Type Radio
<hr>
<table>
<tr><td>Tujuan
<td><input type=radio name=tujuan value=Bandung>Bandung
<tr><td><td><input type=radio name=tujuan value=Jakarta>Jakarta
<tr><td><td><input type=radio name=tujuan value=Bogor>Bogor
</table>
<hr>
<input type=submit value=Proses>
<input type=reset value=Reset>
```



Gambar 7.3. Input Tipe Radio

Untuk menangani input type Radio, dapat dilakukan dengan membaca value dari fields tersebut.

NamaFile:Proses_radio.php

```
<?php
echo "<center>";
echo "<font size=8>";
echo "Membaca Input Type Radio";
echo "<hr>";
$tujuan=$_POST['tujuan'];
echo "<tr><td>Tujuan<td>:$tujuan";
?>
```



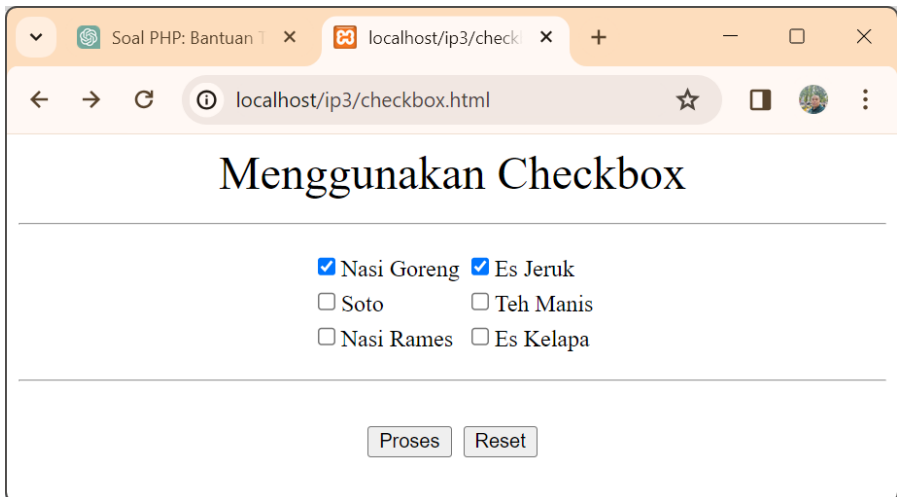
Gambar 7.4.Menampilkan data tipe Radio

7.4. Form Tipe Checkbox

Input type Checkbox berfungsi untuk membuat pilihan dengan cara men-checklist pilihan tersebut, pilihan bisa lebih dari satu pilihan.

Checkbox.html

```
<form action="proses_checkbox.php" method=post>
<center>
<font size=6>
Menggunakan Checkbox
<hr>
<table>
<tr><td><input type=checkbox name=nasigoreng>Nasi Goreng
<td><input type=checkbox name=esjeruk>Es Jeruk
<tr><td><input type=checkbox name=soto>Soto
<td><input type=checkbox name=tehmanis>Teh Manis
<tr><td><input type=checkbox name=nasirames>Nasi Rames
<td><input type=checkbox name=eskelapa>Es Kelapa
</table>
<hr>
<input type=submit value=Proses>
<input type=reset value=Reset>
```

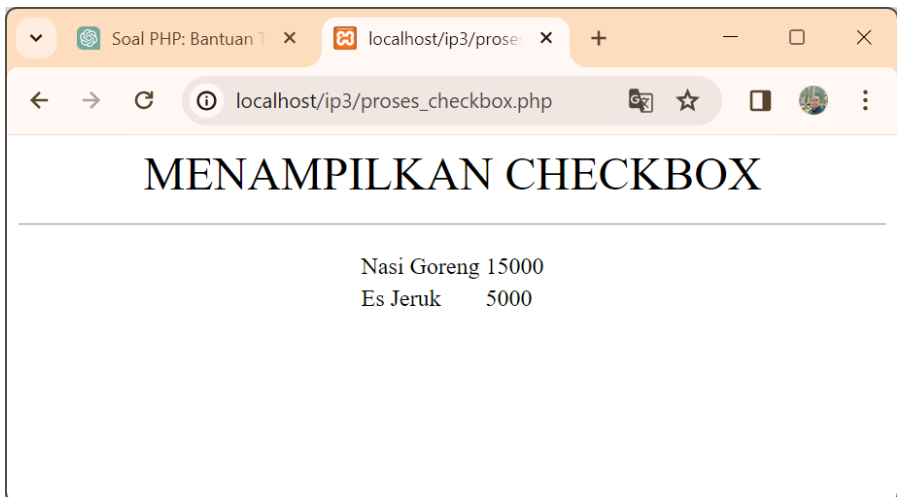


Gambar 7.5.Objek Tipe Checkbox

Untuk membaca field dengan tipe checkbox dapat dilakukan dengan membaca value dari fields tersebut, jika isi value adalah on artinya dipilih.

Proses_checkbox.php

```
<?php
error_reporting(0);
echo "<center>";
echo "<font size=6>";
echo "MENAMPILKAN CHECKBOX";
echo "<hr>";
echo "<table>";
$nasigoreng=$_POST['nasigoreng'];
if($nasigoreng)
{
$hnasgor=15000;
echo "<tr><td>Nasi Goreng <td>$hnasgor";
}
$esjeruk=$_POST['esjeruk'];
if($esjeruk)
{
$hesjeruk=5000;
echo "<tr><td>Es Jeruk<td>$hesjeruk";
}
?>
```



Gambar 7.6.Menangani Objek Tipe Checkbox

7.5. Form Tipe Select

Masukan menggunakan select digunakan untuk memasukan data dengan cara memilih salah satu pilihan dari beberapa pilihan yang disediakan

Sintak Dasar:

```
<select name=namavariabel>  
<option>Pilihan1  
<option>Pilihan2  
</select>
```

Namafile:select.html

```
<html>  
<form action="proses_select.php" method=post>  
<center>  
<font size=8>  
Program Gaji Pegawai  
<hr>  
<table>  
<tr><td>Nama Pegawai <td><input name=namapegawai size=20>  
<tr><td>Jabatan<td><select name=jabatan>  
<option>MANAGER  
<option>MARKETING  
<option>PRODUKSI  
<option>HRD  
<option>OB  
<tr><td>Tahun Masuk <td><input name=tahunmasuk size=4>  
</table>  
<hr>  
<input type=submit value=Proses>  
<input type=reset value=Reset>
```



Gambar 7.7.Input Select

Untuk membaca select, dapat dilakukan dengan cara yang sama dengan membaca tipe text atau textbox

Proses_select.php

```
<?php
echo "<center>";
echo "<font size=10>";
echo "<font color=blue>";
echo "Program Gaji Pegawai ";
echo "<hr>";
echo "<table>";
$namepegawai=$_POST['namepegawai'];
$jabatan=$_POST['jabatan'];
$tahunmasuk=$_POST['tahunmasuk'];
$lamakerja=2017 - $tahunmasuk;
if($jabatan=="MANAGER") $gajipokok=10000000;
if($jabatan=="MARKETING") $gajipokok=8000000;

$tunjangan=0.05*$gajipokok*$lamakerja;
$gajitotal=$gajipokok+$tunjangan;
echo "<tr><td>Nama Pegawai<td>:$namepegawai";
echo "<tr><td>Jabatan<td>:$jabatan";
```

```

echo "<tr><td>Tahun Masuk<td>:$tahunmasuk";
echo "<tr><td>Lama Kerja<td>:$lamakerja";
echo "<tr><td>Gaji Pokok<td>:$gajipokok";
echo "<tr><td>Tunjangan<td>:$tunjangan";
echo "<tr><td>Gaji Total<td>:$gajitotal";
?>

```



Gambar 7.8. Membaca Select

7.6. Form Tipe TextArea

Masukan menggunakan textarea digunakan untuk memasukan data lebih dari satu baris.

NamaFile:textarea.html

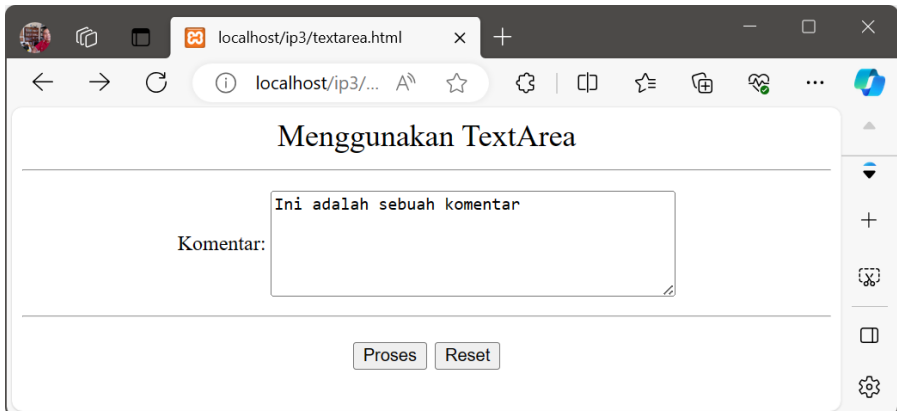
```

<center>
<form action="proses_textarea.php" method=post>
<font size=5>
Menggunakan TextArea
<hr>
<table>
<tr><td>Komentar:<td><textarea name=komen rows=5 cols=40></textarea>
</table>
</table>
<hr>
<input type=submit value=Proses>
<input type=reset value=Reset>

```

NamaFile:proses_textarea.php

```
<?php
error_reporting(0);
echo "<center>";
echo "<font size=6>";
echo "MENAMPILKAN TEXTAREA";
echo "<hr>";
echo "<table>";
$komen=$_POST['komen'];
echo "<tr><td>Komen<td>:$komen";
?>
```



Gambar 7.7. Membaca TextArea

7.7. Mengirim dan Menangani Data Array

Apabila anda ingin mengirimkan beberapa data sekaligus maka anda dapat melakukannya dengan menggunakan array.

Misalnya akan dikirim data penjualan seperti berikut:

FORM PENJUALAN BARANG

Nama Barang	Harga Satuan	Qty
Komputer	Rp. 4000000	
Laptop	Rp. 7500000	
SSD	Rp. 350000	

Hitung

Gambar 7.10. Form Array

Kemudian Proses Menangani Array nya akan dilakukan di program PHP

Nama File: **Form_penjualan.html**

```
<html>
<center>
<form method="POST" action="proses_hitung.php">
<center>
FORM PENJUALAN BARANG
<hr>
<table border="0" width=400>
<td colspan="3" align="center">
<b><font color=white>PENJUALAN BARANG</font></b>
<tr align=center bgcolor=#888888><th>Nama Barang<th>Harga Satuan<th>Qty
<tr><td>Komputer<td align=right>Rp. 4000000
<td align=center>
<input type=hidden name="nama[]" value="Komputer">
<input type=hidden name="harga[]" value="4000000">
<input type=text name="qty[]" size=5 maxlength=4>

<tr><td>Laptop<td align=right>Rp. 7500000
<td align=center>
<input type=hidden name="nama[]" value="Laptop">
<input type=hidden name="harga[]" value="7500000">
<input type=text name="qty[]" size=5 maxlength=4>
```

```

<tr><td>SSD<td align=right>Rp. 350000
<td align=center>
<input type=hidden name="nama[]" value="SSD">
<input type=hidden name="harga[]" value="350000">
<input type=text name="qty[]" size=5 maxlength=4>
</table>
<hr>
<input type=submit value="Hitung" name="submit">
<input type=reset value="Reset" name="reset">

```

proses_hitung.php

```

<Center>
<?php
error_reporting(0);
if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST['submit'])) {
    // Mengambil data dari form
    $namaBarang = $_POST['nama'];
    $hargaSatuan = $_POST['harga'];
    $qty = $_POST['qty'];
    // Menampilkan data yang diterima
    echo "<h2>DATA PENJUALAN BARANG</h2>";
    echo "<table border='1' width='400'>";
    echo "<tr bgcolor='#888888'><th>Nama Barang<th>Harga Satuan<th>Qty</tr>";
    for ($i = 0; $i < count($namaBarang); $i++) {
        if($qty[$i]== "") $qty[$i]=0;
        echo "<tr><td>$namaBarang[$i]<td align='right'>Rp. " .
        number_format($hargaSatuan[$i], 0, ',', '.') . "<td align='center'>$qty[$i]</tr>";
    }
    echo "</table>";
    // Hitung total
    $total = array_sum(array_map(function ($harga, $qty) {
        return $harga * $qty;
    }, $hargaSatuan, $qty));
    echo "<h3>Total Penjualan: Rp. " . number_format($total, 0, ',', '.') . "</h3>";
} else {
    echo "Akses tidak sah!";
}
?>

```

DATA PENJUALAN BARANG		
Nama Barang	Harga Satuan	Qty
Komputer	Rp. 4.000.000	1
Laptop	Rp. 7.500.000	2
SSD	Rp. 350.000	3

Total Penjualan: Rp. 20.050.000

Gambar 7.11. Menangani Variabel Array

7.8. PHP Form Validation

Pada bagian ini kita akan belajar bagaimana cara memproses formulir PHP dengan mempertimbangkan keamanan. Validasi data formulir yang tepat penting untuk melindungi formulir Anda dari peretas dan spammer!

7.8.1. Fungsi htmlspecialchars()

Fungsi htmlspecialchars() mengubah karakter khusus menjadi entitas HTML. Artinya akan mengganti karakter HTML seperti < dan > dengan < dan >. Hal ini mencegah penyerang mengeksploitasi kode dengan memasukkan kode HTML atau Javascript (*serangan Cross-site Scripting*) ke dalam formulir.

```
<html>
<?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>Contoh Validasi FORM PHP</h2>
<form method="post" action="<?php echo
htmlspecialchars($_SERVER["PHP_SELF"]);?>">
<table>
<tr><td>Nama lengkap<td>: <input type="text" name="name">
<tr><td>E-mail<td>: <input type="text" name="email">
<tr><td>Website<td>: <input type="text" name="website">
<tr><td>Pesanan<td>: <textarea name="comment" rows="5"
cols="40"></textarea>
<tr><td>Gender<td>:
    <input type="radio" name="gender" value="female">Female
    <input type="radio" name="gender" value="male">Male
    <input type="radio" name="gender" value="other">Other
</table>
<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>

```

Contoh Validasi FORM PHP

Nama lengkap :

E-mail :

Website :

Pesan :

Gender : ☒ Female ☐ Male ☐ Other

Your Input:

Gambar 7.12. Menangani Validasi Data

7.8.2. Bidang yang Wajib diisi

Dari tabel aturan validasi di halaman sebelumnya, kita melihat bahwa kolom "Nama", "Email", dan "Jenis Kelamin" wajib diisi.

Bidang ini tidak boleh kosong dan harus diisi dalam formulir 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_SELF"]);?>">
    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>
```

Langkah selanjutnya adalah melakukan validasi data masukan yaitu “Apakah kolom Nama hanya berisi huruf dan spasi?”, dan “Apakah kolom Email berisi sintaks alamat email yang valid?”, dan jika diisi, “Apakah bidang Situs Web berisi URL yang valid?”.

7.8.3 PHP - Validasi Nama

Kode di bawah ini menunjukkan cara sederhana untuk memeriksa apakah kolom nama hanya berisi huruf, tanda hubung, apostrof, dan spasi. Jika nilai bidang nama tidak valid, simpan pesan kesalahan:

```
$nama = test_input($_POST["nama"]);  
if (!preg_match("/^[a-zA-Z-' ]*$/", $nama)) {  
    $namaErr = "Only letters and white space allowed";  
}
```

7.8.4. PHP - Validasi Email

Cara termudah dan teraman untuk memeriksa apakah alamat email sudah terbentuk dengan baik adalah dengan menggunakan fungsi `filter_var()` PHP. Pada kode di bawah ini, jika alamat email tidak tersusun dengan baik, simpan pesan kesalahan:

```
$email = test_input($_POST["email"]);  
if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {  
    $emailErr = "Invalid email format";  
}
```

7.8.5. PHP - Validasi URL

Kode di bawah ini menunjukkan cara untuk memeriksa apakah sintaks alamat URL valid (ekspresi reguler ini juga mengizinkan tanda hubung di URL). Jika sintaks alamat URL tidak valid, simpan pesan kesalahan:

```
$website = test_input($_POST["website"]);
if (!preg_match("/\b(?:?:https?|ftp):\\\/|www\\.)([-a-z0-7+&@#\\/%?~_!:,.;]*[-a-z0-7+&@#\\/%?~_]|/i",$website)) {
    $websiteErr = "Invalid URL";
}
```

```
<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"]);
        // check if name only contains letters and whitespace
        if (!preg_match("/^[a-zA-Z-' ]*$/",$name)) {
            $nameErr = "Only letters and white space allowed";
        }
    }

    if (empty($_POST["email"])) {
        $emailErr = "Email is required";
    } else {
        $email = test_input($_POST["email"]);
        // check if e-mail address is well-formed
        if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
            $emailErr = "Invalid email format";
        }
    }

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

```

    // check if URL address syntax is valid
    if (!preg_match("/\b(?:?:https?|ftp):\\\/\\\/www\\.)[-a-z0-7+&@#\\/%?=~_!|:,.;]*[-a-z0-7+&@#\\/%?=~_]/i",$website)) {
        $websiteErr = "Invalid URL";
    }
}

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_SELF"]);?>">
    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>

```

PHP Form Validation Example

* required field

Name: * Name is required

E-mail: * Email is required

Website:

Comment:

Gender: ☐ Female ☐ Male ☐ Other * Gender is required

Gambar 7.13. Validasi Name dan E-mail

Untuk Kode Lengkapnya adalah sebagai berikut:

```

<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"]);
        // check if name only contains letters and whitespace
        if (!preg_match("/^[a-zA-Z- ]*$/", $name)) {
            $nameErr = "Only letters and white space allowed";
        }
    }

    if (empty($_POST["email"])) {
        $emailErr = "Email is required";
    } else {
        $email = test_input($_POST["email"]);
        // check if e-mail address is well-formed
        if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
            $emailErr = "Invalid email format";
        }
    }

    if (empty($_POST["website"])) {
        $website = "";
    } else {
        $website = test_input($_POST["website"]);
        // check if URL address syntax is valid (this regular expression
also allows dashes in the URL)
        if (!preg_match("/\b(?:(:https?|ftp):\/\/|www\.)[-a-z0-7+&@#\/%?~_!:,.;]*[-a-z0-7+&@#\/%?~_]/i", $website)) {
            $websiteErr = "Invalid URL";
        }
    }

    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_SELF"]);?>">
  Name: <input type="text" name="name" value="<?php echo $name;?>">
  <span class="error">* <?php echo $nameErr;?></span>
  <br><br>
  E-mail: <input type="text" name="email" value="<?php echo
$email;?>">
  <span class="error">* <?php echo $emailErr;?></span>
  <br><br>
  Website: <input type="text" name="website" value="<?php echo
$website;?>">
  <span class="error"><?php echo $websiteErr;?></span>
  <br><br>
  Comment: <textarea name="comment" rows="5" cols="40"><?php echo
$comment;?></textarea>
  <br><br>
  Gender:
  <input type="radio" name="gender" <?php if (isset($gender) &&
$gender=="female") echo "checked";?> value="female">Female
  <input type="radio" name="gender" <?php if (isset($gender) &&
$gender=="male") echo "checked";?> value="male">Male
  <input type="radio" name="gender" <?php if (isset($gender) &&
$gender=="other") echo "checked";?> 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>

```