

LAPORAN TUGAS JOBSHEET VII “PHP - Form Proccessing”

Mata Kuliah: Pemrograman Web
Dosen Pengampu: Dimas Wahyu Wibowo, S.T., M.T.



Disusun Oleh:

Nama : Syava Aprilia P
NIM : 2241760129
Absen :25

**PROGRAM STUDI SISTEM INFORMASI BISNIS
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2023**



Topik

- Konsep Form Proccessing dengan PHP dan JQuery

Tujuan

Mahasiswa diharapkan dapat:

1. Mahasiswa mampu membuat form menggunakan PHP
2. Mahasiswa mampu membuat form menggunakan jQuery

Perhatian

Jobsheet ini harus dikerjakan step-by-step sesuai langkah-langkah praktikum yang sudah diberikan. Soal dapat dijawab langsung di dalam kolom yang disediakan dengan menggunakan PDF Editor.

Pendahuluan

Function isset

isset() adalah sebuah fungsi dalam PHP yang digunakan untuk memeriksa apakah suatu variabel telah didefinisikan (ada) atau tidak. Fungsi ini mengembalikan **true** jika variabel telah didefinisikan dan memiliki nilai, dan **false** jika variabel tidak ada atau bernilai **null**. Berikut adalah contoh script dan penjelasan minimal mengenai **isset()**:

Kode di atas memeriksa apakah variabel **\$nama** telah didefinisikan. Jika ya, maka pesan "Variabel 'nama' telah didefinisikan." akan ditampilkan, jika tidak, maka pesan "Variabel 'nama' tidak ditemukan atau bernilai null." akan ditampilkan.

Praktikum 1. Function Isset

| Langkah | Keterangan |
|---------|--|
| 1 | Buatlah satu file baru di dalam direktori dasarWeb, beri nama <code>isset.php</code> . |
| 2 | Ketikkan ke dalam file <code>isset.php</code> tersebut kode di bawah ini. |
| 3 | <pre>\$umur; if (isset(\$umur) && \$umur >= 18) { echo "Anda sudah dewasa."; } else { echo "Anda belum dewasa atau variabel 'umur' tidak ditemukan."; }</pre> <p>Jawab:</p> |

| | |
|---|--|
| | <pre> Pertemuan 7 > ❄ isset.php 1 <?php 2 \$umur; 3 if (isset(\$umur) && \$umur >= 18) { 4 echo "Anda sudah dewasa."; 5 } else { 6 echo "Anda belum dewasa atau variabel 'umur' tidak ditemukan."; 7 } 8 ?> </pre> |
| 4 | <p>Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/isset.php. Apa yang anda pahami dari penggunaan isset pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 1.1)</p> <p>Jawab:</p> <p>Anda belum dewasa atau variabel 'umur' tidak ditemukan.</p> <p>Penggunaan isset() dalam file tersebut digunakan untuk memeriksa apakah variabel \$umur telah didefinisikan dan memiliki nilai atau tidak sebelum mencoba untuk membandingkannya dengan angka 18. Jika \$umur telah didefinisikan dan nilainya tidak null, maka kondisi isset(\$umur) akan bernilai true, dan kode di dalam blok if akan dieksekusi. Jika tidak, maka pesan akan menunjukkan bahwa variabel 'umur' tidak ditemukan atau belum diinisialisasi.</p> |
| 5 | <p>Tambahkan isi dari file isset.php tersebut dengan kode di bawah ini.</p> <pre> \$data = array("nama" => "Jane", "usia" => 25); if (isset(\$data["nama"])) { echo "Nama: " . \$data["nama"]; } else { echo "Variabel 'nama' tidak ditemukan dalam array."; } </pre> <pre> //nomor 5 \$data = array("nama" => "Jane", "usia" => 25); if (isset(\$data["nama"])) { echo "Nama: " . \$data["nama"]; } else { echo "Variabel 'nama' tidak ditemukan dalam array."; } ?> </pre> |
| 6 | <p>Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/isset.php buat hasil tidak menjadi 1 baris, jadi tampilan dari echo harus terpisah. Apa yang anda pahami dari penggunaan isset pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 1.2)</p> <p>Jawab:</p> <p>Anda belum dewasa atau variabel 'umur' tidak ditemukan>Nama: Jane</p> <p>Agar tidak menjadi 1 baris maka ditambahkan echo" "; setelah kode nomor 3.</p> <p>Anda belum dewasa atau variabel 'umur' tidak ditemukan. Nama: Jane</p> <p>isset() digunakan untuk memeriksa keberadaan elemen dengan kunci "nama" dalam array \$data. Jika elemen tersebut ada, maka akan menampilkan nama yang terkandung dalam array. Jika tidak, maka akan menampilkan pesan "Variabel 'nama' tidak ditemukan dalam array."</p> |

Function empty

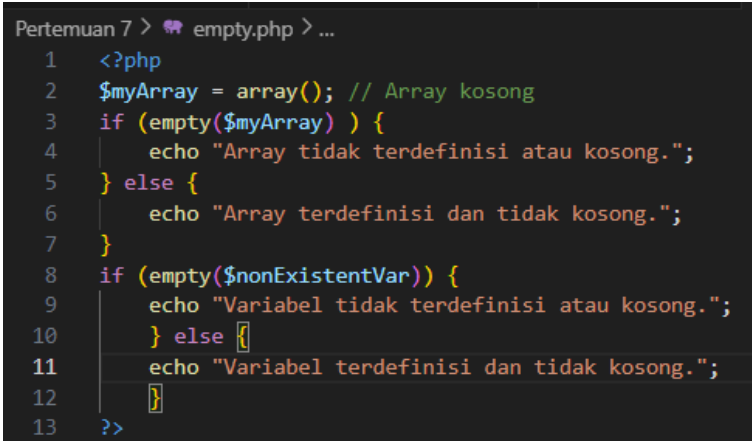
Fungsi **empty()** pada PHP digunakan untuk memeriksa apakah suatu variabel kosong atau tidak terdefinisi. Fungsi ini mengembalikan **true** jika variabel kosong atau tidak terdefinisi, dan **false** jika variabel memiliki nilai atau telah didefinisikan. Berikut adalah penjelasan dan contoh penggunaan **empty()**:

```
$nama = ""; // Variabel string kosong
if (empty($nama)) {
    echo "Nama tidak terdefinisi atau kosong.";
} else {
    echo "Nama terdefinisi dan tidak kosong.";
}
```

Fungsi **empty()** dapat digunakan untuk memeriksa apakah suatu string kosong atau tidak.

Praktikum 2. Function empty

| Langkah | Keterangan |
|---------|---|
| 1 | Buat satu file baru bernama <code>empty.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>empty.php</code> |
| 2 | <pre>\$myArray = array(); // Array kosong if (empty(\$myArray)) { echo "Array tidak terdefinisi atau kosong."; } else { echo "Array terdefinisi dan tidak kosong."; }</pre> <pre>Pertemuan 7 > empty.php > ... 1 <?php 2 \$myArray = array(); // Array kosong 3 if (empty(\$myArray)) { 4 echo "Array tidak terdefinisi atau kosong."; 5 } else { 6 echo "Array terdefinisi dan tidak kosong."; 7 } 8 ?></pre> |
| 3 | <p>Simpan file tersebut, kemudian buka browser dan jalankan <code>localhost/dasarWeb/empty.php</code>. Apa yang anda pahami dari penggunaan <code>empty</code> pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 2.1)</p> <p>Jawab:</p> <p>Array tidak terdefinisi atau kosong.</p> <p>Penggunaan <code>empty()</code> dalam file tersebut digunakan untuk memeriksa apakah sebuah array kosong atau tidak terdefinisi. Jika array <code>\$myArray</code> tidak memiliki elemen (yaitu, kosong) atau belum didefinisikan sama sekali, maka kondisi <code>empty(\$myArray)</code> akan bernilai <code>true</code>, dan pesan "Array tidak terdefinisi atau kosong." akan ditampilkan. Namun, jika array memiliki setidaknya satu elemen, maka kondisi <code>empty(\$myArray)</code> akan bernilai <code>false</code>, dan pesan "Array</p> |

| | |
|---|---|
| | terdefinisi dan tidak kosong." akan ditampilkan. Ini memungkinkan untuk menghindari kesalahan saat mencoba mengakses atau memanipulasi array yang belum diinisialisasi atau kosong. |
| 4 | <p>Tambahkan isi dari file <code>empty.php</code> tersebut dengan kode di bawah ini.</p> <pre> if (empty(\$nonExistentVar)) { echo "Variabel tidak terdefinisi atau kosong."; } else { echo "Variabel terdefinisi dan tidak kosong."; } </pre>  <pre> Pertemuan 7 > empty.php > ... 1 <?php 2 \$myArray = array(); // Array kosong 3 if (empty(\$myArray)) { 4 echo "Array tidak terdefinisi atau kosong."; 5 } else { 6 echo "Array terdefinisi dan tidak kosong."; 7 } 8 if (empty(\$nonExistentVar)) { 9 echo "Variabel tidak terdefinisi atau kosong."; 10 } else { 11 echo "Variabel terdefinisi dan tidak kosong."; 12 } 13 ?> </pre> |
| 5 | <p>Simpan file tersebut, kemudian buka browser dan jalankan <code>localhost/dasarWeb/empty.php</code> buat hasil tidak menjadi 1 baris, jadi tampilan dari echo harus terpisah. Apa yang anda pahami dari penggunaan <code>empty</code> pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 2.2)</p> <p>Jawab:</p> <p>Agar tidak menjadi satu baris maka tambahkan <code> </code> didalam echo setelah kalimat.</p> <p>Array tidak terdefinisi atau kosong. Variabel tidak terdefinisi atau kosong.</p> <p>Pada bagian kedua, <code>empty(\$nonExistentVar)</code> digunakan untuk memeriksa apakah variabel <code>\$nonExistentVar</code> didefinisikan dan memiliki nilai atau tidak. Karena variabel tersebut belum didefinisikan, maka kondisi <code>empty(\$nonExistentVar)</code> akan bernilai true, dan pesan "Variabel tidak terdefinisi atau kosong." akan ditampilkan.</p> |

Praktikum Bagian 3 : Form Input PHP

| Langkah | Keterangan |
|---------|---|
| 1 | Buat satu file baru bernama <code>form.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>form.php</code> |

| | |
|---|--|
| 2 | <pre> <!DOCTYPE html> <html> <head> <title>Form Input PHP</title> </head> <body> <h2>Form Input PHP</h2> <form method="post" action="proses_form.php"> <label for="nama">Nama:</label> <input type="text" name="nama" id="nama" required> <label for="email">Email:</label> <input type="email" name="email" id="email" required> <input type="submit" name="submit" value="Submit"> </form> </body> </html> </pre> <p>Pertemuan 7 > form.php > html > body > form</p> <pre> 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>Form Input PHP</title> 5 </head> 6 7 <body> 8 <h2>Form Input PHP</h2> 9 <form method="post" action="proses_form.php"> 10 <label for="nama">Nama:</label> 11 <input type="text" name="nama" id="nama" required> 12 13 <label for="email">Email:</label> 14 <input type="email" name="email" id="email" required> 15 16 <input type="submit" name="submit" value="Submit"> 17 </form> 18 </body> 19 </html> </pre> |
| 3 | <p>Buat satu file baru bernama proses_form.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam proses_form.php</p> <pre> <?php if (\$_SERVER["REQUEST_METHOD"] == "POST") { \$nama = \$_POST["nama"]; \$email = \$_POST["email"]; echo "Nama: " . \$nama . " "; echo "Email: " . \$email; } ?> </pre> |

| | |
|---|--|
| | <pre> Pertemuan 7 > proses_form.php > ... 1 <?php 2 if (\$_SERVER["REQUEST_METHOD"] == "POST") { 3 \$nama = \$_POST["nama"]; 4 \$email = \$_POST["email"]; 5 6 echo "Nama: " . \$nama . " "; 7 echo "Email: " . \$email; 8 } 9 ?> </pre> |
| 4 | <p>Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/proses_form.php. Apa yang anda pahami dari penggunaan form pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 3.1)</p> <p>Jawab:</p> <p>Buka form.php terlebih dahulu, lalu isi form dan submit</p> <p>Form Input PHP</p> <p>Nama : <input type="text" value="syava"/></p> <p>Email: <input type="text" value="syafaaps3@gmail.com"/></p> <p><input type="submit" value="Submit"/></p> <p>Lalu akan dialihkan ke proses_form.php dan berikut hasilnya</p> <p>Nama: syava Email: syafaaps3@gmail.com</p> <p>formulir tersebut memungkinkan pengguna untuk memasukkan nama dan email, dan ketika formulir tersebut dikirimkan, halaman proses_form.php akan mengambil nilai-nilai tersebut dan menampilkannya kembali kepada pengguna.</p> |
| 5 | <p>Buat satu file baru bernama form_self.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam form_self.php</p> <pre> <!DOCTYPE html> <html> <head> <title>Form Input PHP</title> </head> <body> <h2>Form Input PHP</h2> <?php // Inisialisasi variabel \$namaErr = ""; \$nama = ""; // Cek apakah form sudah disubmit if (\$_SERVER["REQUEST_METHOD"] == "POST") { // Validasi nama (contoh: pastikan nama tidak kosong) if (empty(\$_POST["nama"])) { \$namaErr = "Nama harus diisi!"; } else { \$nama = \$_POST["nama"]; echo "Data berhasil disimpan!"; } } ?> <form method="post" action="<?php echo htmlspecialchars(\$_SERVER["PHP_SELF"]); ?>"> <label for="nama">Nama:</label> <input type="text" name="nama" id="nama" value="<?php echo \$nama; ?>"> <?php echo \$namaErr; ?> <input type="submit" name="submit" value="Submit"> </form> </body> </html> </pre> |

```

Pertemuan 7 > form_self.php > html > body > form > span.error
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Form Input PHP</title>
5 </head>
6 <body>
7 <h2>Form Input PHP</h2>
8 <?php
9 // Inisialisasi variabel
10 $namaErr = "";
11 $nama = "";
12
13 // Cek apakah form sudah disubmit
14 if ($_SERVER["REQUEST_METHOD"] == "POST") {
15     // Validasi nama (contoh: pastikan nama tidak kosong)
16     if (empty($_POST["nama"])) {
17         $namaErr = "Nama harus diisi!";
18     } else {
19         $nama = $_POST["nama"];
20         echo "Data berhasil disimpan!";
21     }
22 }
23 ?>
24 <form method="post" action="" <?php echo htmlspecialchars($_SERVER["PHP_SELF"]); ?>>
25 <label for="nama">Nama:</label>
26 <input type="text" name="nama" id="nama" value="" <?php echo $nama; ?>>
27 <span class="error"> <?php echo $namaErr; ?></span><br><br>
28
29 <input type="submit" name="submit" value="Submit">
30 </form>
31 </body>
32 </html>

```

Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/form_self.php. Apa yang anda pahami dari penggunaan form pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 3.2)

Jawab:

Form Input PHP

Nama:

- Setelah submit

Form Input PHP

Data berhasil disimpan!

Nama:

Yang saya pahami tentang penggunaan formulir (form) pada file tersebut:

- Elemen Form HTML: Terdapat sebuah formulir HTML yang terdiri dari satu input teks untuk nama (<input type="text">) dan satu tombol submit (<input type="submit">).
- Penggunaan PHP untuk Validasi Form: PHP digunakan untuk memproses data formulir setelah dikirimkan. Pada saat formulir dikirimkan (\$_SERVER["REQUEST_METHOD"] == "POST"), PHP akan memeriksa apakah nama telah diisi. Jika tidak, akan ditampilkan pesan kesalahan ("Nama harus diisi!") di samping kolom input nama.
- Sanitasi Input: Fungsi htmlspecialchars() digunakan untuk mengonversi karakter khusus HTML menjadi entitas HTML. Hal ini dilakukan untuk mencegah serangan XSS (Cross-Site Scripting).
- Menampilkan Pesan Kesalahan: Jika terjadi kesalahan validasi, pesan kesalahan akan ditampilkan di samping kolom input nama.

| | |
|--|--|
| | <ul style="list-style-type: none"> - Menampilkan Data yang Sudah Diisi: Setelah formulir disubmit, data yang sudah dimasukkan (misalnya, nama yang sudah diisi) akan tetap muncul di dalam kolom input nama. Ini membantu pengguna untuk melihat kembali data yang telah dimasukkan dan memperbaiki jika diperlukan. <p>Dengan demikian, formulir tersebut memungkinkan pengguna untuk memasukkan nama, melakukan validasi, dan menampilkan pesan kesalahan jika perlu.</p> |
|--|--|

HTML Injection

HTML injection (atau disebut juga "HTML injection attack" atau "client-side injection") adalah serangan keamanan yang terjadi ketika seorang penyerang memasukkan kode HTML atau JavaScript yang berbahaya ke dalam input yang diterima oleh sebuah aplikasi web. Kode berbahaya tersebut akan dieksekusi oleh browser pengguna yang melihat halaman web yang terpengaruh, yang dapat mengakibatkan akses ilegal ke data, perubahan tampilan halaman, atau serangan lainnya.

Seorang penyerang dapat mencoba menyisipkan kode HTML atau JavaScript yang berbahaya ke dalam input yang diterima oleh aplikasi web. Jika aplikasi web tersebut tidak melakukan sanitasi atau escape input dengan benar, maka kode berbahaya tersebut akan dieksekusi oleh browser pengguna, yang dapat menyebabkan masalah keamanan.

Contoh HTML Injection:

Misalnya, kita memiliki formulir pencarian sederhana di situs web yang mencari kata kunci di antara ulasan pengguna:

```
<form action="search.php" method="post">
  <input type="text" name="keyword">
  <input type="submit" value="Search">
</form>
```

Sekarang, jika aplikasi web tidak memvalidasi input dengan benar, seorang penyerang dapat memasukkan input berbahaya seperti ini:

```
<script>alert('You have been hacked!');</script>
```

Jika aplikasi web tidak menghindari atau membersihkan input ini sebelum menampilkannya di halaman hasil pencarian, maka hasilnya akan menjadi seperti ini:

```
<div>Hasil pencarian untuk "<script>alert('You have been hacked!');</script>":</div>
```

Kode JavaScript berbahaya tersebut akan dieksekusi oleh browser pengguna dan akan memunculkan kotak peringatan "You have been hacked!". Ini adalah contoh HTML injection yang sederhana. Sebagai pengembang web, penting untuk selalu melakukan sanitasi dan validasi input dari pengguna, serta menghindari menampilkan input pengguna langsung di halaman web tanpa escaping atau sanitasi yang tepat untuk menghindari serangan semacam ini.

Cara Mengangani HTML Injection

Cara menangani HTML injection adalah dengan memvalidasi, menyaring, dan menghindari menampilkan input pengguna tanpa escaping yang tepat. Berikut adalah beberapa langkah untuk melindungi aplikasi Anda dari HTML injection:

1. **Validasi Input:** Validasi input dari pengguna untuk memastikan bahwa hanya data yang valid yang diterima. Anda bisa menggunakan fungsi seperti `filter_input()` atau `filter_var()` untuk melakukan validasi.
2. **Saring Input:** Saring input pengguna untuk menghapus atau menggantikan karakter yang berpotensi berbahaya, seperti `<`, `>`, `&`, dan lainnya. Anda bisa menggunakan fungsi seperti `htmlspecialchars()` atau `strip_tags()` untuk ini.
3. **Parameterized Statements (Query):** Jika Anda menghasilkan SQL query dengan input pengguna, gunakan parameterized statements atau prepared statements untuk mencegah SQL injection, yang bisa menjadi bentuk serangan yang mirip dengan HTML injection.
4. **Content Security Policy (CSP):** Terapkan Content Security Policy (CSP) pada header HTTP Anda untuk mengontrol sumber daya yang dapat digunakan dalam halaman web Anda.
5. **Escape Output:** Ketika Anda menampilkan data di halaman web, pastikan Anda menghindari injeksi JavaScript dengan menggunakan `htmlspecialchars()` atau metode serupa.

Berikut contoh sederhana dalam PHP untuk mengatasi HTML injection:

```
// Menggunakan filter_input untuk validasi input
$user_input = filter_input(INPUT_POST, 'user_input', FILTER_SANITIZE_STRING);

// Menggunakan htmlspecialchars untuk menghindari HTML injection saat menampilkan data
echo '<div>' . htmlspecialchars($user_input, ENT_QUOTES, 'UTF-8') . '</div>';
```

Dengan menggunakan langkah-langkah ini, Anda dapat memitigasi risiko HTML injection dalam aplikasi web Anda. Selalu penting untuk memvalidasi input, membersihkan data yang masuk, dan menghindari menampilkan data pengguna tanpa escaping yang benar.

Praktikum 4 : HTML Injection

Langkah-langkah Praktikum :

| Langkah | Keterangan |
|---------|--|
| 1 | Buat satu file baru bernama <code>html_aman.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>html_aman.php</code> |
| 2 | <pre>\$input = \$_POST['input']; \$input = htmlspecialchars(\$input, ENT_QUOTES, 'UTF-8');</pre> |
| 3 | Tambahkan script lainya supaya langkah 2 bisa berjalan normal. Lalu simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/html_aman.php</code> |

| | |
|---|---|
| | <pre> Pertemuan 7 > ❏ html_aman.php > ... 1 <?php 2 // Mendapatkan nilai dari input 3 \$input = \$_POST['input']?\$_POST['input'] : ''; 4 5 // Melakukan sanitasi input 6 \$input_sanitized = htmlspecialchars(\$input, ENT_QUOTES, 'UTF-8'); 7 ?> 8 9 <!DOCTYPE html> 10 <html> 11 <head> 12 <title>Sanitasi Input</title> 13 </head> 14 <body> 15 <h2>Sanitasi Input</h2> 16 <form method="post" action="<?php echo htmlspecialchars(\$_SERVER['PHP_SELF']); ?>" 17 <label for="input">Masukkan Input:</label> 18 <input type="text" name="input" id="input" value="<?php echo isset(\$input) ? \$input : ''; ?>" 19 <input type="submit" name="submit" value="Submit"> 20 </form> 21 22 <?php 23 // Menampilkan hasil sanitasi input 24 if (isset(\$input_sanitized)) { 25 echo "<p>Input yang disanitasi: \$input_sanitized</p>"; 26 } 27 ?> 28 </body> 29 </html> 30 </pre> |
| 4 | <p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.1)</p> <div> <div> <h3>Sanitasi Input</h3> <p>Masukkan Input: <input type="text"/> <input type="button" value="Submit"/></p> <p>Input yang disanitasi:</p> </div> <div> <h3>Sanitasi Input</h3> <p>Masukkan Input: <input type="text" value="syava"/> <input type="button" value="Submit"/></p> <p>Input yang disanitasi: syava</p> </div> </div> |
| 5 | Ketikkan kode tambahan pada langkah 6 di dalam <code>html_aman.php</code> |
| 6 | <pre> // Periksa apakah input adalah email yang valid \$email = \$_POST['email']; if (filter_var(\$email, FILTER_VALIDATE_EMAIL)) { // Lanjutkan dengan pengolahan email yang aman } else { // Tangani input yang tidak valid } </pre> |
| 7 | Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/html_aman.php</code> |

| | |
|---|---|
| | <pre> Pertemuan 7 > ❏ html_aman.php > ... 1 <?php 2 // Mendapatkan nilai dari input 3 \$input = isset(\$_POST['input']) ? \$_POST['input'] : ''; 4 5 // Melakukan sanitasi input 6 \$input_sanitized = htmlspecialchars(\$input, ENT_QUOTES, 'UTF-8'); 7 8 // Memeriksa apakah input adalah email yang valid 9 \$email = isset(\$_POST['email']) ? \$_POST['email'] : ''; 10 \$emailErr = ''; 11 if (\$_SERVER["REQUEST_METHOD"] == "POST") { 12 if (!empty(\$email)) { 13 if (filter_var(\$email, FILTER_VALIDATE_EMAIL)) { 14 // Lanjutkan dengan pengolahan email yang aman 15 } else { 16 // Tangani input yang tidak valid 17 \$emailErr = "Format email tidak valid!"; 18 } 19 } 20 } 21 ?> 22 23 <!DOCTYPE html> 24 <html> 25 <head> 26 <title>Sanitasi Input</title> 27 </head> 28 <body> 29 <h2>Sanitasi Input</h2> 30 <form method="post" action="<?php echo htmlspecialchars(\$_SERVER["PHP_SELF"]); ?>"> 31 <label for="input">Masukkan Input:</label> 32 <input type="text" name="input" id="input" value="<?php echo htmlspecialchars(\$input, ENT_QUOTES, 'UTF-8'); ?>"> 33 34 <label for="email">Email:</label> 35 <input type="email" name="email" id="email" value="<?php echo htmlspecialchars(\$email, ENT_QUOTES, 'UTF-8'); ?>"> 36 <?php echo \$emailErr; ?> 37 38 <input type="submit" name="submit" value="Submit"> 39 </form> 40 41 <?php 42 // Menampilkan hasil sanitasi input 43 if (isset(\$input_sanitized)) { 44 echo "<p>Input yang disanitasi: \$input_sanitized</p>"; 45 } 46 ?> 47 </body> 48 </html> </pre> |
| 8 | <p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.2)</p> <div data-bbox="279 1209 734 1512"> <h3>Sanitasi Input</h3> <p>Masukkan Input: <input type="text" value="syava"/></p> <p>Email: <input type="text" value="syafaaps3@gmail.com"/></p> <p><input type="button" value="Submit"/></p> <p>Input yang disanitasi: syava</p> </div> |

Regular Expression (Regex)

Regex adalah alat yang kuat untuk mencari, mencocokkan, atau memanipulasi teks berdasarkan pola tertentu. Anda dapat menggunakannya untuk berbagai tujuan seperti validasi input, pencarian teks, penggantian teks, dan banyak lagi.

Basic Regex Patterns:

- **Karakter Tunggal:** Karakter apa pun akan cocok dengan dirinya sendiri dalam Regex, kecuali karakter khusus yang harus dihindari dengan karakter pelarian \.
 - o Contoh: Pola **/a/** akan cocok dengan huruf "a" dalam teks.

- **Karakter Spesifik:** Anda dapat mencocokkan karakter tertentu dengan menyebutkannya dalam pola.
 - o Contoh: Pola `/hello/` akan cocok dengan teks "hello" dalam teks.
- **Karakter Khusus:** Beberapa karakter khusus dalam Regex harus dihindari dengan karakter pelarian (`\`) jika Anda ingin mencocokkannya secara harfiah. Contoh karakter khusus: `.`, `*`, `+`, `?`, `[`, `]`, `(`, `)`, `{`, `}`, `^`, `$`, `\`.
- **Karakter Set:** Anda dapat mencocokkan karakter dari satu set karakter dengan menggunakan `[]`. Misalnya, `/[aeiou]/` akan mencocokkan salah satu huruf vokal.
- **Rentang Karakter:** Anda dapat menggunakan `-` dalam karakter set untuk menentukan rentang karakter. Misalnya, `/[a-z]/` akan mencocokkan huruf kecil apa pun.
- **Kuantifier:** Anda dapat mengatur berapa kali karakter atau kelompok karakter sebelumnya harus muncul. Contoh kuantifier:
 - o `*`: 0 atau lebih kali
 - o `+`: 1 atau lebih kali
 - o `?`: 0 atau 1 kali
 - o `{n}`: Persis n kali
 - o `{n,}`: Setidaknya n kali
 - o `{n,m}`: Setidaknya n kali, maksimal m kali

Contoh Penggunaan Regex:

1. Mencocokkan pola

```
$pattern = '/hello/';
$text = 'Hello, world!';
if (preg_match($pattern, $text)) {
    echo "Cocokkan ditemukan!";
} else {
    echo "Tidak ada cocokan!";
}
```

2. Mencocokkan karakter set

```
$pattern = '/[0-9]/'; // Cocokkan satu digit.
$text = 'There are 3 apples.';
if (preg_match($pattern, $text)) {
    echo "Digit ditemukan!";
} else {
    echo "Tidak ada digit!";
}
```

Praktikum 5 : Penggunaan Regex Pada PHP

Langkah-langkah Praktikum :

| Langkah | Keterangan |
|---------|---|
| 1 | Buat satu file baru bernama <code>regex.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>regex.php</code> |

| | |
|---|--|
| 2 | <pre> \$pattern = '/[a-z]/'; // Cocokkan huruf kecil. \$text = 'This is a Sample Text.'; if (preg_match(\$pattern, \$text)) { echo "Huruf kecil ditemukan!"; } else { echo "Tidak ada huruf kecil!"; } </pre> <pre> Pertemuan 7 > regex.php > ... 1 <?php 2 \$pattern = '/[a-z]/'; // Cocokkan huruf kecil. 3 \$text = 'This is a Sample Text. '; 4 if (preg_match(\$pattern, \$text)) { 5 echo "Huruf kecil ditemukan!"; 6 } else { 7 echo "Tidak ada huruf kecil!"; 8 } 9 ?> </pre> |
| 3 | <p>Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php</p> <p>Huruf kecil ditemukan!</p> |
| 4 | <p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.1)</p> <p>Jawab:</p> <p>Dalam kode di atas, kita menggunakan regular expression (/[a-z]/) untuk mencocokkan apakah terdapat setidaknya satu huruf kecil dalam teks yang diberikan. Jika cocok, maka akan menampilkan pesan "Huruf kecil ditemukan!", dan jika tidak, akan menampilkan pesan "Tidak ada huruf kecil!".</p> |
| 5 | <p>Ketikkan kode tambahan pada langkah 6 di dalam regex.php</p> |
| 6 | <pre> \$pattern = '/[0-9]+/'; // Cocokkan satu atau lebih digit. \$text = 'There are 123 apples.'; if (preg_match(\$pattern, \$text, \$matches)) { echo "Cocokkan: " . \$matches[0]; } else { echo "Tidak ada yang cocok!"; } </pre> |
| 7 | <p>Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php</p> |

| | |
|----|--|
| | <pre> Pertemuan 7 > regex.php > ... 1 <?php 2 \$pattern = '/[a-z]/'; // Cocokkan huruf kecil. 3 \$text = 'This is a Sample Text. '; 4 if (preg_match(\$pattern, \$text)) { 5 echo "Huruf kecil ditemukan! "; 6 } else { 7 echo "Tidak ada huruf kecil! "; 8 } 9 10 \$pattern = '/[0-9]+/'; // Cocokkan satu atau lebih digit. 11 \$text = 'There are 123 apples. '; 12 if (preg_match(\$pattern, \$text, \$matches)) { 13 echo "Cocokkan: " . \$matches[0]; 14 } else { 15 echo "Tidak ada yang cocok! "; 16 } 17 ?> </pre> |
| 8 | <p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.2)</p> <p>Huruf kecil ditemukan! Cocokkan: 123</p> <p>Program tersebut menggunakan regular expression untuk mencocokkan pola tertentu dalam teks. Dua pola yang dicocokkan yaitu:</p> <ul style="list-style-type: none"> - Cocokkan Huruf Kecil: Program mencari setidaknya satu huruf kecil dalam teks. Jika ditemukan, program menampilkan pesan "Huruf kecil ditemukan!". Jika tidak ditemukan, program menampilkan pesan "Tidak ada huruf kecil!". - Cocokkan Satu atau Lebih Digit: Program mencari satu atau lebih digit dalam teks. Jika ditemukan, program menampilkan digit yang cocok. Jika tidak ditemukan, program menampilkan pesan "Tidak ada yang cocok!". <p>Keseluruhan program memberikan contoh penggunaan regular expression untuk mencocokkan dan menangani pola tertentu dalam teks.</p> |
| 9 | <p>Ketikkan kode tambahan pada langkah 10 di dalam regex.php</p> |
| 10 | <pre> \$pattern = '/apple/'; \$replacement = 'banana'; \$text = 'I like apple pie.'; \$new_text = preg_replace(\$pattern, \$replacement, \$text); echo \$new_text; // Output: "I like banana pie." </pre> |
| 11 | <p>Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php</p> <pre> Pertemuan 7 > regex.php > ... 1 <?php 2 \$pattern = '/[a-z]/'; // Cocokkan huruf kecil. 3 \$text = 'This is a Sample Text. '; 4 if (preg_match(\$pattern, \$text)) { 5 echo "Huruf kecil ditemukan! "; 6 } else { 7 echo "Tidak ada huruf kecil! "; 8 } 9 10 \$pattern = '/[0-9]+/'; // Cocokkan satu atau lebih digit. 11 \$text = 'There are 123 apples. '; 12 if (preg_match(\$pattern, \$text, \$matches)) { 13 echo "Cocokkan: " . \$matches[0] . " "; 14 } else { 15 echo "Tidak ada yang cocok! "; 16 } 17 18 \$pattern = '/apple/'; 19 \$replacement = 'banana'; 20 \$text = 'I like apple pie. '; 21 \$new_text = preg_replace(\$pattern, \$replacement, \$text); 22 echo \$new_text; // Output: "I like banana pie." 23 ?> </pre> |

| | |
|----|--|
| 12 | <p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.3)</p> <p>Huruf kecil ditemukan! Cocokkan: 123 I like banana pie.</p> <p>Pola dan Pengganti: Variabel \$pattern berisi pola regular expression yang ingin dicocokkan dalam teks, yaitu /apple/. Variabel \$replacement berisi string yang akan digunakan untuk menggantikan setiap kemunculan pola yang cocok, yaitu 'banana'.</p> <p>Penggunaan preg_replace(): Fungsi preg_replace() digunakan untuk mencari setiap kemunculan pola yang cocok dalam teks (\$text) dan menggantinya dengan string pengganti (\$replacement). Hasil dari penggantian disimpan dalam variabel \$new_text.</p> <p>Output: Dengan menggunakan echo, hasil dari penggantian (\$new_text) ditampilkan. Dalam contoh ini, teks "I like apple pie." akan diganti menjadi "I like banana pie.".</p> |
| 13 | Ketikkan kode tambahan pada langkah 14 di dalam regex.php |
| 14 | <pre> \$pattern = '/go*d/'; // Cocokkan "god", "good", "goood", dll. \$text = 'god is good.'; if (preg_match(\$pattern, \$text, \$matches)) { echo "Cocokkan: " . \$matches[0]; } else { echo "Tidak ada yang cocok!"; } </pre> |
| 15 | <p>Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php</p> <pre> 23 24 \$pattern = '/go*d/'; // Cocokkan "god", "good", "goood", dll. 25 \$text = 'god is good. '; 26 if (preg_match(\$pattern, \$text, \$matches)) { 27 echo "Cocokkan: " . \$matches[0] . " "; 28 } else { 29 echo "Tidak ada yang cocok! "; 30 } 31 ?> </pre> |
| 16 | <p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.4)</p> <p>Huruf kecil ditemukan! Cocokkan: 123 I like banana pie. Cocokkan: god</p> <p>Dari penambahan kode program di atas, yang diamati adalah:</p> <ul style="list-style-type: none"> - Pencocokan Pola dengan Karakter Khusus: Kode baru menambahkan pencocokan pola baru menggunakan karakter khusus "*" dalam regular expression. Variabel \$pattern diatur dengan pola /go*d/, yang akan mencocokkan kata-kata seperti "god", "good", "goood", dan seterusnya. - Penggunaan Karakter Khusus dalam Pola: Karakter "*" dalam pola /go*d/ digunakan untuk mencocokkan nol atau lebih kemunculan dari huruf "o" setelah huruf "g" dan sebelum huruf "d". |

| | |
|----|---|
| | <p>Hasil Pencocokan: Jika pola cocok dengan teks, maka akan ditampilkan kata yang cocok bersama dengan pesan "Cocokkan:". Jika tidak ada yang cocok, akan ditampilkan pesan "Tidak ada yang cocok!".</p> |
| 17 | <p>Soal 5.5: Buatlah script langkah 14 dengan variable pattern yang di ubah menggunakan `?` (0 atau 1 kali). Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.5)</p> <p>Jawab:</p> <pre>\$pattern = '/go?d/'; // Cocokkan "god", "good", "goood", dll. \$text = 'god is good. '; if (preg_match(\$pattern, \$text, \$matches)) { echo "Cocokkan: " . \$matches[0] . " "; } else { echo "Tidak ada yang cocok! "; } ?></pre> <p>Huruf kecil ditemukan! Cocokkan: 123 I like banana pie. Cocokkan: god</p> <p>Pencocokan Pola dengan Karakter "?": Kode baru mengubah pola menggunakan karakter "?" dalam regular expression. Variabel \$pattern diatur dengan pola /go?d/, yang akan mencocokkan kata-kata seperti "god" atau "good", di mana huruf "o" dapat muncul nol atau satu kali.</p> <p>Hasil Pencocokan: Jika pola cocok dengan teks, maka akan ditampilkan kata yang cocok bersama dengan pesan "Cocokkan:". Jika tidak ada yang cocok, akan ditampilkan pesan "Tidak ada yang cocok!".</p> |
| 18 | <p>Soal 5.6: Buatlah script langkah 14 dengan variable pattern yang di ubah menggunakan `{n,m}`. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.6)</p> <p>Jawab:</p> <pre>\$pattern = '/go{1,3}d/'; // Cocokkan "god", "good", "goood" \$text = 'god is good. '; if (preg_match(\$pattern, \$text, \$matches)) { echo "Cocokkan: " . \$matches[0] . " "; } else { echo "Tidak ada yang cocok! "; } ?></pre> <p>Huruf kecil ditemukan! Cocokkan: 123 I like banana pie. Cocokkan: god</p> <p>Pencocokan Pola dengan Kurung Kurawal {n,m}: Kode baru mengubah pola menggunakan kurung kurawal {n,m} dalam regular expression. Variabel \$pattern diatur dengan pola /go{1,3}d/, yang akan mencocokkan kata-kata seperti "god", "good", atau "goood", di mana huruf "o" muncul 1 hingga 3 kali.</p> <p>Hasil Pencocokan: Jika pola cocok dengan teks, maka akan ditampilkan kata yang cocok bersama dengan pesan "Cocokkan:". Jika tidak ada yang cocok, akan ditampilkan pesan "Tidak ada yang cocok!".</p> |

Praktikum 6 : Form Lanjut

Langkah-langkah Praktikum :

| Langkah | Keterangan |
|---------|---|
| 1 | Buat satu file baru bernama form_lanjut.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam form_lanjut.php |

```

<!DOCTYPE html>
<html>
<head>
  <title>Contoh Form dengan PHP</title>
</head>
<body>
  <h2>Form Contoh</h2>
  <form method="POST" action="proses_lanjut.php">
    <label for="buah">Pilih Buah:</label>
    <select name="buah" id="buah">
      <option value="apel">Apel</option>
      <option value="pisang">Pisang</option>
      <option value="mangga">Mangga</option>
      <option value="jeruk">Jeruk</option>
    </select>

    <br>

    <label>Pilih Warna Favorit:</label><br>
    <input type="checkbox" name="warna[]" value="merah"> Merah<br>
    <input type="checkbox" name="warna[]" value="biru"> Biru<br>
    <input type="checkbox" name="warna[]" value="hijau"> Hijau<br>

    <br>

    <label>Pilih Jenis Kelamin:</label><br>
    <input type="radio" name="jenis_kelamin" value="laki-laki"> Laki-laki<br>
    <input type="radio" name="jenis_kelamin" value="perempuan"> Perempuan<br>

    <br>

    <input type="submit" value="Submit">
  </form>
</body>
</html>

```

jawab:

2

```

Pertemuan 7 > form_lanjut.php > html > body > form
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>Contoh Form dengan PHP</title>
5  </head>
6  <body>
7    <h2>Form Contoh</h2>
8    <form method="POST" action="proses_lanjut.php">
9      <label for="buah">Pilih Buah:</label>
10     <select name="buah" id="buah">
11       <option value="apel">Apel</option>
12       <option value="pisang">Pisang</option>
13       <option value="mangga">Mangga</option>
14       <option value="jeruk">Jeruk</option>
15     </select>
16
17     <br>
18
19     <label>Pilih Warna Favorit:</label><br>
20     <input type="checkbox" name="warna[]" value="merah"> Merah<br>
21     <input type="checkbox" name="warna[]" value="biru"> Biru<br>
22     <input type="checkbox" name="warna[]" value="hijau"> Hijau<br>
23
24     <br>
25
26     <label>Pilih Jenis Kelamin:</label><br>
27     <input type="radio" name="jenis_kelamin" value="laki-laki"> Laki-laki<br>
28     <input type="radio" name="jenis_kelamin" value="perempuan"> Perempuan<br>
29
30     <br>
31
32     <input type="submit" value="Submit">
33   </form>
34 </body>
35 </html>
36

```

```

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $selectedBuah = $_POST['buah'];

    if (isset($_POST['warna'])) {
        $selectedWarna = $_POST['warna'];
    } else {
        $selectedWarna = [];
    }

    $selectedJenisKelamin = $_POST['jenis_kelamin'];

    echo "Anda memilih buah: " . $selectedBuah . "<br>";

    if (!empty($selectedWarna)) {
        echo "Warna favorit Anda: " . implode(", ", $selectedWarna) . "<br>";
    } else {
        echo "Anda tidak memilih warna favorit.<br>";
    }

    echo "Jenis kelamin Anda: " . $selectedJenisKelamin;
}

```

jawab:

```

Pertemuan 7 > proses_lanjut.php > ...
1  <?php
2  if ($_SERVER["REQUEST_METHOD"] == "POST") {
3      $selectedBuah = $_POST['buah'];
4
5      if (isset($_POST['warna'])) {
6          $selectedWarna = $_POST['warna'];
7      } else {
8          $selectedWarna = [];
9      }
10     $selectedJenisKelamin = $_POST['jenis_kelamin'];
11
12     echo "Anda memilih buah: " . $selectedBuah . "<br>";
13
14     if (isset($selectedWarna) && !empty($selectedWarna)) {
15         echo "Warna favorit Anda: " . implode(", ", $selectedWarna) . "<br>";
16     } else {
17         echo "Anda tidak memilih warna favorit.<br>";
18     }
19     echo "Jenis kelamin Anda: " . $selectedJenisKelamin;
20 }
21 ?>

```

Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/form_lanjut.php

Form Contoh

Pilih Buah:

Pilih Warna

☒ Merah
☐ Biru
☐ Hijau

Pilih Jenis Kelamin:
☐ Laki-laki
☒ Perempuan

Anda memilih buah: apel
 Warna favorit Anda: merah
 Jenis kelamin Anda: perempuan

Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 6.1)

Jawab:

Kode program tersebut mencetak hasil yang sudah diinput/dipilih sebelumnya.

Buat satu file baru bernama form_ajax.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam form_ajax.php

```

<!DOCTYPE html>
<html>
<head>
  <title>Contoh Form dengan PHP dan jQuery</title>
  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
</head>
<body>
  <h2>Form Contoh</h2>
  <form id="myForm">
    <label for="buah">Pilih Buah:</label>
    <select name="buah" id="buah">
      <option value="apel">Apel</option>
      <option value="pisang">Pisang</option>
      <option value="mangga">Mangga</option>
      <option value="jeruk">Jeruk</option>
    </select>

    <br>

    <label>Pilih Warna Favorit:</label><br>
    <input type="checkbox" name="warna[]" value="merah"> Merah<br>
    <input type="checkbox" name="warna[]" value="biru"> Biru<br>
    <input type="checkbox" name="warna[]" value="hijau"> Hijau<br>

    <br>

    <label>Pilih Jenis Kelamin:</label><br>
    <input type="radio" name="jenis_kelamin" value="laki-laki"> Laki-laki<br>
    <input type="radio" name="jenis_kelamin" value="perempuan"> Perempuan<br>

    <br>

    <input type="submit" value="Submit">
  </form>

  <div id="hasil">
    <!-- Hasil akan ditampilkan di sini -->
  </div>

  <script>
    $(document).ready(function () {
      $("#myForm").submit(function (e) {
        e.preventDefault(); // Mencegah pengiriman form secara default

        // Mengumpulkan data form
        var formData = $("#myForm").serialize();

        // Kirim data ke server PHP
        $.ajax({
          url: "proses_lanjut.php", // Ganti dengan nama file PHP yang sesuai
          type: "POST",
          data: formData,
          success: function (response) {
            // Tampilkan hasil dari server di div "hasil"
            $("#hasil").html(response);
          }
        });
      });
    });
  </script>
</body>
</html>

```

```

Pertemuan 7 > form_ajax.php > IDOCTYPE > html > body > form#myForm > select#buah > option
1 <!DOCTYPE html>
2 <html>
3 <head>
4 <title>Contoh Form dengan PHP dan jQuery</title>
5 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6 </head>
7 <body>
8 <h2>Form Contoh</h2>
9 <form id="myForm">
10 <label for="buah">Pilih Buah :</label>
11 <select name="buah" id="buah">
12 <option value="apel">Apel</option>
13 <option value="pisang">Pisang</option>
14 <option value="mangga">Mangga</option>
15 <option value="jeruk">Jeruk</option>
16 </select>
17
18 <br>
19
20 <label>Pilih Warna Favorit :</label><br>
21 <input type="checkbox" name="warna[]" value="merah"> Merah<br>
22 <input type="checkbox" name="warna[]" value="biru"> Biru<br>
23 <input type="checkbox" name="warna[]" value="hijau"> Hijau<br>
24
25 <br>
26
27 <label>Pilih Jenis Kelamin :</label><br>
28 <input type="radio" name="jenis_kelamin" value="laki-laki"> Laki-laki<br>
29 <input type="radio" name="jenis_kelamin" value="perempuan"> Perempuan<br>
30
31 <br>
32
33 <input type="submit" value="Submit">
34 </form>
35
36 <div id="hasil">
37 <!-- Hasil akan ditampilkan di sini -->
38 </div>
39
40 <script>
41 $(document).ready(function () {
42     $("#myForm").submit(function (e) {
43         e.preventDefault(); // Mencegah pengiriman form secara default
44
45         // Mengumpulkan data form
46         var formData = $("#myForm").serialize();
47
48         // Kirim data ke server PHP
49         $.ajax({
50             url: "proses_lanjut.php", // Ganti dengan nama file PHP yang sesuai
51             type: "POST",
52             data: formData,
53             success: function (response) {
54                 // Tampilkan hasil dari server di div "hasil"
55                 $("#hasil").html(response);
56             }
57         });
58     });
59 </script>
60
61 </body>
62 </html>

```

6

Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/form_ajax.php. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 6.2)

Form Contoh

Pilih Buah : Apel

Pilih Warna Favorit :

☒ Merah

☐ Biru

☐ Hijau

Pilih Jenis Kelamin :

☐ Laki-laki

☒ Perempuan

Praktikum 7 : Validasi Form

Langkah-langkah Praktikum :

| Langkah | Keterangan |
|---------|--|
| 1 | Buat satu file baru bernama form_validasi.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam form_validasi.php |
| 2 | <pre> <!DOCTYPE html> <html> <head> <title>Form Input dengan Validasi</title> </head> <body> <h1>Form Input dengan Validasi</h1> <form method="post" action="proses_validasi.php"> <label for="nama">Nama:</label> <input type="text" id="nama" name="nama"> <label for="email">Email:</label> <input type="text" id="email" name="email"> <input type="submit" value="Submit"> </form> </body> </html> </pre> <p>Jawab:</p> |

Pertemuan 7 > form_validasi.php > html > body > form > label

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Form Input dengan Validasi</title>
5 </head>
6 <body>
7   <h1>Form Input dengan Validasi</h1>
8   <form method="post" action="proses_validasi.php">
9     <label for="nama">Nama :</label>
10    <input type="text" id="nama" name="nama">
11    <br>
12
13    <label for="email">Email :</label>
14    <input type="text" id="email" name="email">
15    <br>
16
17    <input type="submit" value="Submit">
18  </form>
19 </body>
20 </html>
```

```
<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $nama = $_POST["nama"];
    $email = $_POST["email"];
    $errors = array();

    // Validasi Nama
    if (empty($nama)) {
        $errors[] = "Nama harus diisi.";
    }

    // Validasi Email
    if (empty($email)) {
        $errors[] = "Email harus diisi.";
    } elseif (filter_var($email, FILTER_VALIDATE_EMAIL)) {
        $errors[] = "Format email tidak valid.";
    }

    // Jika ada kesalahan validasi
    if (empty($errors)) {
        foreach ($errors as $error) {
            echo $error . "<br>";
        }
    } else {
        // Lanjutkan dengan pemrosesan data jika semua validasi berhasil
        // Misalnya, menyimpan data ke database atau mengirim email
        echo "Data berhasil dikirim: Nama = $nama, Email = $email";
    }
}
?>
```

Jawab:

Pertemuan 7 > proses_validasi.php > ...

```
1 <?php
2 if ($_SERVER["REQUEST_METHOD"] == "POST") {
3     $nama = $_POST["nama"];
4     $email = $_POST["email"];
5     $errors = array();
6     // Validasi Nama
7     if (empty($nama)) {
8         $errors[] = "Nama harus diisi.";
9     }
10    // Validasi Email
11    if (empty($email)) {
12        $errors[] = "Email harus diisi.";
13    } else if (filter_var($email, FILTER_VALIDATE_EMAIL)) {
14        $errors[] = "Format email tidak valid.";
15    }
16    // Jika ada kesalahan validasi
17    if (is_array($errors)){
18        foreach ($errors as $error): ?>
19            <?php echo $error; ?>
20            <br/>
21        <?php endforeach;
22    }
23    else {
24        // Lanjutkan dengan pemrosesan data jika semua validasi berhasil
25        // Misalnya, menyimpan data ke database atau mengirim email
26        echo "Data berhasil dikirim: Nama = $nama, Email = $email";
27    }
28 }
29 ?>
```

| | |
|---|---|
| 3 | <p>Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/form_validasi.php</p> <h3>Form Input dengan Validasi</h3> <div> <div> Nama : <input type="text" value="syava"/> Email : <input type="text" value="syafaaps3@gmail.com"/> <input type="button" value="Submit"/> </div> <div> Format email tidak valid. </div> </div> <p>Jika email tidak diisi</p> <h3>Form Input dengan Validasi</h3> <div> <div> Nama : <input type="text" value="syava"/> Email : <input type="text"/> <input type="button" value="Submit"/> </div> <div> Email harus diisi. </div> </div> |
| 4 | <p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.1)</p> <p>Program ini terdiri dari dua bagian:</p> <ol style="list-style-type: none"> 1. Form Input (index.html): <ul style="list-style-type: none"> Form HTML dengan dua field input: nama dan email. Ketika form disubmit, data di kirim ke proses_validasi.php menggunakan method POST. 2. Proses Validasi (proses_validasi.php): <ul style="list-style-type: none"> Memvalidasi input nama dan email: <ul style="list-style-type: none"> nama tidak boleh kosong. email tidak boleh kosong dan harus memiliki format yang valid. Jika terdapat error validasi, error akan ditampilkan di halaman. Jika semua validasi berhasil, data akan diproses (misalnya, disimpan ke database atau dikirim email). <p>Kesimpulan:</p> <p>Program ini mendemonstrasikan cara melakukan validasi email sederhana di PHP.</p> |
| 5 | <p>Kembangkan file bernama form_validasi.php di dalam folder dasarWeb. Ketikkan kode pada langkah 6 di dalam form_validasi.php</p> |
| 6 | <pre> <!DOCTYPE html> <html> <head> <title>Form Input dengan Validasi</title> <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script> </head> <body> <h1>Form Input dengan Validasi</h1> <form id="myForm" method="post" action="proses_validasi.php"> <label for="nama">Nama:</label> <input type="text" id="nama" name="nama"> <label for="email">Email:</label> <input type="text" id="email" name="email"> <input type="submit" value="Submit"> </form> <script> \$(document).ready(function() { \$("#myForm").submit(function(event) { var nama = \$("#nama").val(); var email = \$("#email").val(); var valid = true; if (nama == "") { \$("#nama-error").text("Nama harus diisi."); valid = false; } else { \$("#nama-error").text(""); } if (email == "") { \$("#email-error").text("Email harus diisi."); valid = false; } else { \$("#email-error").text(""); } if (valid) { (!event.preventDefault()); // Menghentikan pengiriman form jika validasi gagal } }); }); </script> </body> </html> </pre> |

| | |
|---|--|
| 7 | <p>Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/form validasi.php</p> <pre> Pertemuan 7 > form_validasi.php > html > body > script > ready() callback > submit() callback 1 <!DOCTYPE html> 2 <html> 3 <head> 4 <title>Form Input dengan Validasi</title> 5 <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script> 6 </head> 7 <body> 8 <h1>Form Input dengan Validasi</h1> 9 <form id="myForm" method="post" action="proses_validasi.php"> 10 <label for="nama">Nama :</label> 11 <input type="text" id="nama" name="nama"> 12 13 14 <label for="email">Email:</label> 15 <input type="text" id="email" name="email"> 16 17 18 <input type="submit" value="Submit"> 19 </form> 20 21 <script> 22 \$(document).ready(function() { 23 \$("#myForm").submit(function(event) { 24 var nama = \$("#nama").val(); 25 var email = \$("#email").val(); 26 var valid = true; 27 28 if (nama === "") { 29 \$("#nama-error").text("Nama harus diisi."); 30 valid = false; 31 } else { 32 \$("#nama-error").text(""); 33 } 34 35 if (email === "") { 36 \$("#email-error").text("Email harus diisi."); 37 valid = false; 38 } else { 39 \$("#email-error").text(""); 40 } 41 42 if (valid) { 43 event.preventDefault(); 44 // Menghentikan pengiriman form jika validasi gagal 45 } 46 }); 47 }); 48 </script> 49 </body> 50 </html> </pre> |
| 8 | <p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.2)</p> <h2>Form Input dengan Validasi</h2> <p>Nama : <input type="text"/> Nama harus diisi!</p> <p>Email : <input type="text"/> Email harus diisi!</p> |
| 9 | <p>Soal 7.3: Buatlah script langkah 6 dengan menggunakan ajax. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.3)</p> <p>Jawab:</p> |

```

Pertemuan 7 > form_validasi.php > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Form Input dengan Validasi</title>
5      <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6  </head>
7  <body>
8      <h1>Form Input dengan Validasi</h1>
9      <form id="myForm" method="post" action="proses_validasi.php">
10         <label for="nama">Nama :</label>
11         <input type="text" id="nama" name="nama">
12         <span id="nama-error" style="color: red;"></span><br>
13
14         <label for="email">Email:</label>
15         <input type="text" id="email" name="email">
16         <span id="email-error" style="color: red;"></span><br>
17
18         <input type="submit" value="Submit">
19     </form>
20
21     <script>
22         $(document).ready(function() {
23             $("#myForm").submit(function(event) {
24                 event.preventDefault(); // Menghentikan pengiriman form
25
26                 var nama = $("#nama").val();
27                 var email = $("#email").val();
28                 var valid = true;
29
30                 if (nama === "") {
31                     $("#nama-error").text("Nama harus diisi.");
32                     valid = false;
33                 } else {
34                     $("#nama-error").text("");
35                 }
36
37                 if (email === "") {
38                     $("#email-error").text("Email harus diisi.");
39                     valid = false;
40                 } else {
41                     $("#email-error").text("");
42                 }
43
44                 if (valid) {
45                     $.ajax({
46                         url: $(this).attr('action'),
47                         type: $(this).attr('method'),
48                         data: $(this).serialize(),
49                         success: function(response) {
50                             // Handle response from server
51                             console.log(response);
52                         },
53                         error: function(xhr, status, error) {
54                             // Handle error
55                             console.error(xhr.responseText);
56                         }
57                     });
58                 }
59             });
60         });
61     </script>
62 </body>
63 </html>
64

```

Form Input dengan Validasi

Nama :

Email:

Setelah perubahan ini, form akan dicegah dari pengiriman jika validasi gagal dan data form akan dikirimkan ke **proses_validasi.php** menggunakan AJAX, tanpa perbarui halaman.

10

Soal 7.4: Tambahkan validasi untuk password dengan ketentuan minimal 8 karakter dengan menggunakan jQuery dan PHP tambahkan pada langkah 9. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.4)


Jawab:

```
25 <script>
26     $(document).ready(function() {
27         $("#myForm").submit(function(event) {
28             event.preventDefault(); // Menghentikan pengiriman form
29
30             var nama = $("#nama").val();
31             var email = $("#email").val();
32             var password = $("#password").val();
33             var valid = true;
34
35             if (nama === "") {
36                 $("#nama-error").text("Nama harus diisi.");
37                 valid = false;
38             } else {
39                 $("#nama-error").text("");
40             }
41
42             if (email === "") {
43                 $("#email-error").text("Email harus diisi.");
44                 valid = false;
45             } else {
46                 $("#email-error").text("");
47             }
48
49             if (password === "") {
50                 $("#password-error").text("Password harus diisi.");
51                 valid = false;
52             } else if (password.length < 8) {
53                 $("#password-error").text("Password minimal harus 8 karakter.");
54                 valid = false;
55             } else {
56                 $("#password-error").text("");
57             }
58
59             if (valid) {
60                 $.ajax({
61                     url: $(this).attr('action'),
62                     type: $(this).attr('method'),
63                     data: $(this).serialize(),
64                     success: function(response) {
65                         // Handle response from server
66                         console.log(response);
67                     },
68                     error: function(xhr, status, error) {
69                         // Handle error
70                         console.error(xhr.responseText);
71                     }
72                 });
73             }
74         });
75     });
76 </script>
77 </body>
78 </html>
79
```

Form Input dengan Validasi

Nama :


Email:

Password: 

Form Input dengan Validasi

Nama :

Email:

Password: 

Form Input dengan Validasi

Nama :

Email:

Password: Password minimal harus 8 karakter.

form telah diperbarui dengan tambahan input password dan validasi baik di sisi client (JavaScript) maupun di sisi server (PHP). Jika password tidak memenuhi syarat minimal 8 karakter, pesan kesalahan akan ditampilkan. Jika semua input memenuhi syarat, data akan dikirimkan ke **proses_validasi.php**.