

LAPORAN PRAKTIKUM

DESAIN PEMROGRAMAN WEB

JOBSHEET 7 - PHP FORM PROCESING

Dosen Pengampu :
Dimas Wahyu Wibowo, S.T., M.T.



Dibuat untuk memenuhi tugas mata kuliah
“Desain dan Pemrograman Web”

Disusun Oleh :

Ayleen Ruhul Qisthy

NIM : 2341720012

Kelas : 2H

No. Absen : 07

PROGRAM STUDI D-IV TEKNIK INFORMATIKA
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG

2024



Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-7: PHP - Form Proccessing

Mata Kuliah Desain dan Pemrograman Web

Pengampu: Tim Ajar Desain dan Pemrograman Web

Oktober 2024

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()**:

```
$nama = "John";  
if (isset($nama)) {  
    echo "Variabel 'nama' telah didefinisikan.";  
} else {  
    echo "Variabel 'nama' tidak ditemukan atau bernilai null.";  
}
```

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>
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> <div data-bbox="268 544 1481 636" style="border: 1px solid black; padding: 5px;"> <p>Anda belum dewasa atau variabel 'umur' tidak ditemukan.</p> </div> <p>Penjelasan : Kode tersebut memeriksa apakah variabel \$umur sudah didefinisikan. Pada kode tersebut variabel \$umur belum didefinisikan sehingga bernilai null dan akan menampilkan “Anda belum dewasa atau variabel 'umur' tidak ditemukan”</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>
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> <div data-bbox="268 1413 1481 1568" style="border: 1px solid black; padding: 5px;"> <p>Anda belum dewasa atau variabel 'umur' tidak ditemukan.</p> <p>Nama: Jane</p> </div> <p>Penjelasan : Kode yang telah ditambahkan tersebut terdapat variabel \$data yang telah didefinisikan dengan array berisi nama dan usia. Kemudian fungsi isset() memeriksa apakah variabel \$data telah didefinisikan. Pada kode tersebut variabel \$data telah didefinisikan, sehingga ” nama” akan ditampilkan.</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><code>\$myArray = array(); // Array kosong if (empty(\$myArray)) { echo "Array tidak terdefinisi atau kosong."; } else { echo "Array terdefinisi dan tidak kosong."; }</code></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> <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p>Array tidak terdefinisi atau kosong.</p> </div> <p>Penjelasan : Kode tersebut terdapat variabel <code>\$myArray</code> yang mendefinisikan array dengan array kosong. Terdapat fungsi <code>empty()</code> apakah variabel <code>\$myArray</code> kosong. Pada kode tersebut <code>\$myArray</code> didefinisikan dengan array kosong, sehingga kondisi <code>if</code> terpenuhi dan menampilkan "Array tidak terdefinisi atau kosong"</p>
4	<p>Tambahkan isi dari file <code>empty.php</code> tersebut dengan kode di bawah ini.</p> <pre><code>if (empty(\$nonExistentVar)) { echo "Variabel tidak terdefinisi atau kosong."; } else { echo "Variabel terdefinisi dan tidak kosong."; }</code></pre>

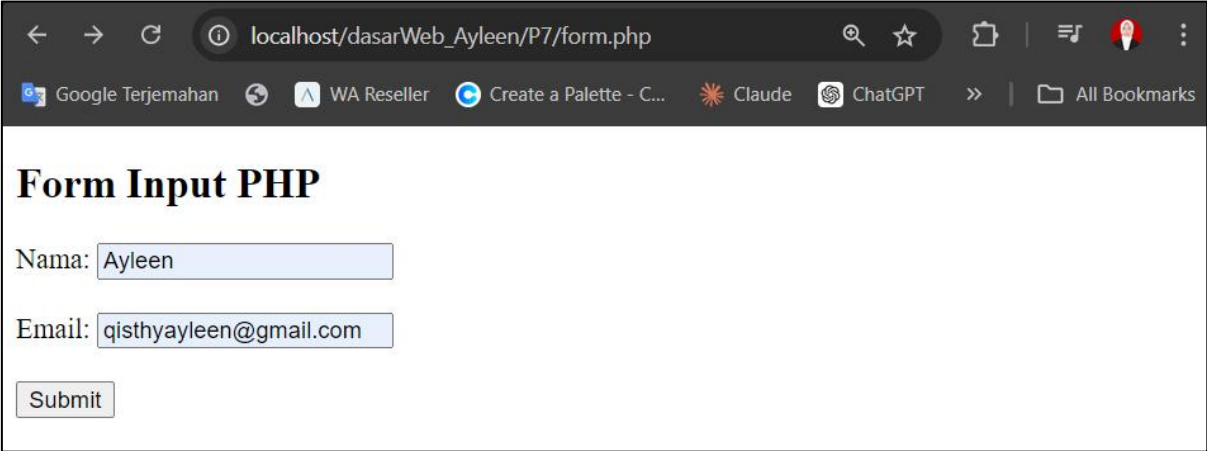
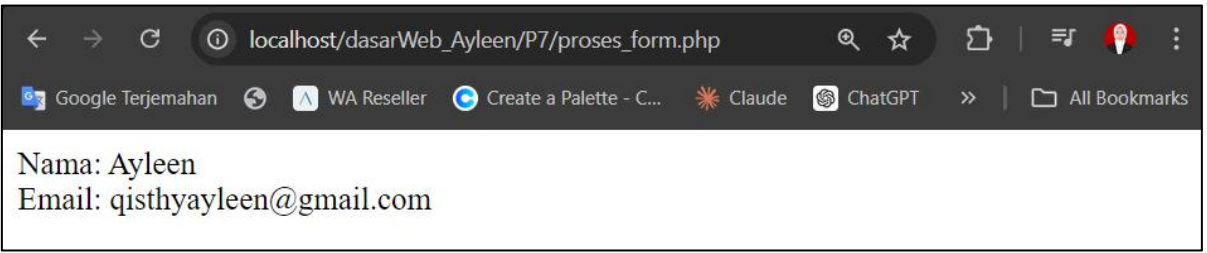
5	<p>Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/empty.php buat hasil tidak menjadi 1 baris, jadi tampilan dari echo harus terpisah. Apa yang anda pahami dari penggunaan empty pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 2.2)</p> <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p>Array tidak terdefinisi atau kosong.</p> <p>Variabel tidak terdefinisi atau kosong.</p> </div> <p>Penjelasan : Kode tersebut terdapat fungsi empty() untuk memeriksa apakah variabel \$nonExistentVar kosong atau tidak terdefinisikan. Pada kode tersebut variabel \$nonExistentVar tidak ada / kosong, sehingga menampilkan "Variabel tidak terdefinisi atau kosong".</p>
---	--

Praktikum Bagian 3 : Form Input PHP

Langkah	Keterangan
1	Buat satu file baru bernama form.php di dalam folder dasarWeb. Ketikkan kode padalangkah 2 di dalam form.php
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>

3	<p>Buat satu file baru bernama <code>proses_form.php</code> di dalam folder <code>dasarWeb</code>. Ketikkan kode pada langkah 2 di dalam <code>proses_form.php</code></p> <pre> <?php if (\$_SERVER["REQUEST_METHOD"] == "POST") { \$nama = \$_POST["nama"]; \$email = \$_POST["email"]; echo "Nama: " . \$nama . "
"; echo "Email: " . \$email; } ?> </pre>
4	<p>Simpan file tersebut, kemudian buka browser dan jalankan <code>localhost/dasarWeb/proses_form.php</code>. Apa yang anda pahami dari penggunaan form pada file tersebut. Catatdi bawah ini pemahaman anda. (soal no 3.1)</p> <p>,</p>   <p>Penjelasan : Kode tersebut terdapat file form html yang meminta input (nama, email) dari user yang kemudian data dikirimkan menggunakan metode POST ke file <code>proses_form.php</code>. Saat form tersebut dikirim dengan method form, maka data yang diinputkan akan diambil menggunakan post dan menampilkan nama dan email yang telah diinputkan.</p>
5	<p>Buat satu file baru bernama <code>form_self.php</code> di dalam folder <code>dasarWeb</code>. Ketikkan kode pada langkah 2 di dalam <code>form_self.php</code></p>


```

<!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; ?>">
        <span class="error"><?php echo $namaErr; ?></span><br><br>

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

```

6

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

Form Input PHP

Nama:

Form Input PHP

Data berhasil disimpan!

Nama:

Penjelasan : Kode html tersebut digunakan untuk membuat form. Membuat variabel \$namaErr dan \$nama kemudian cek apakah form sudah dikirim, jika form sudah dikirim maka terdapat pemilihan kondisi untuk memastikan nama tidak kosong. Jika nama kosong, maka akan menampilkan “Nama harus diisi!” tetapi jika nama tidak kosong maka menampilkan “Data berhasil disimpan!”. Fungsi htmlspecialchars() digunakan untuk menghindari konversi karakter spesial menjadi entitas HTML.

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 antarmuka 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:

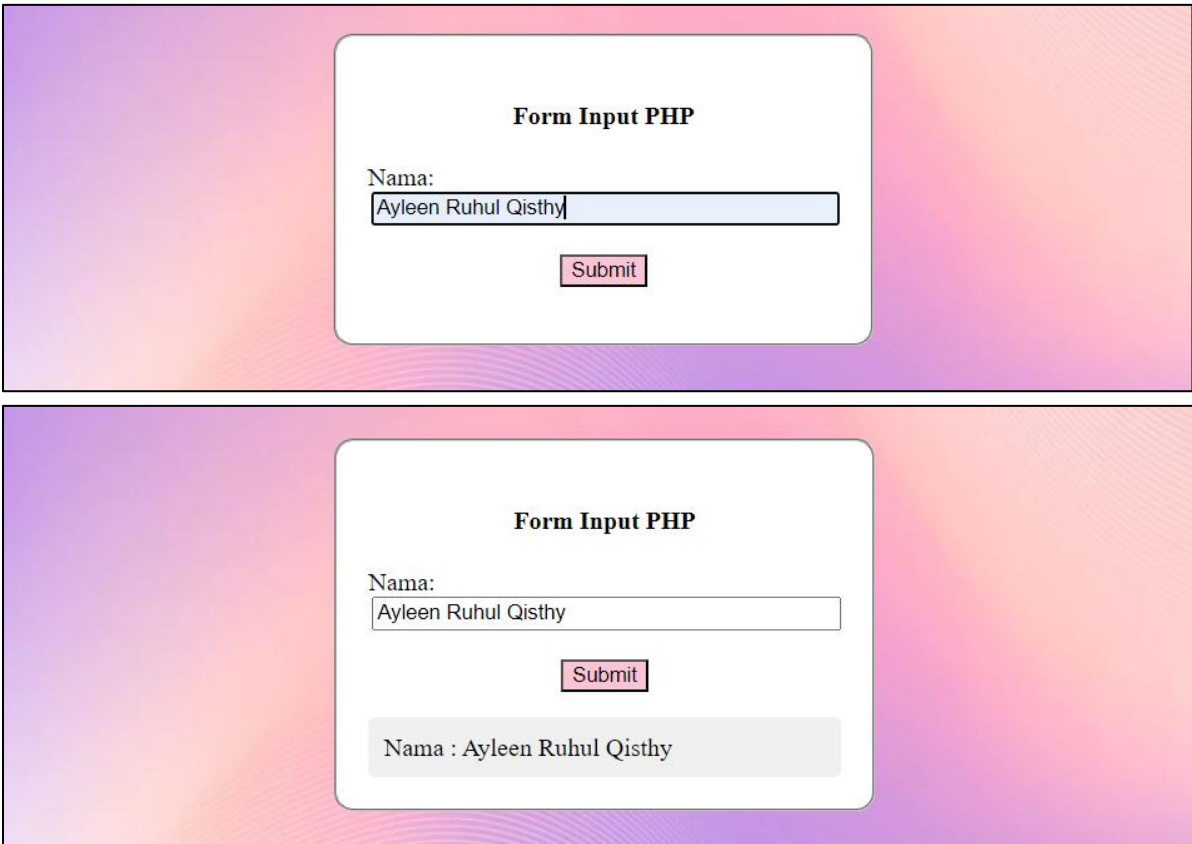

```
// Menggunakan filter_input untuk validasi input
$user_input = filter_input(INPUT_POST, 'user_input', FILTER_SANITIZE_STRING);

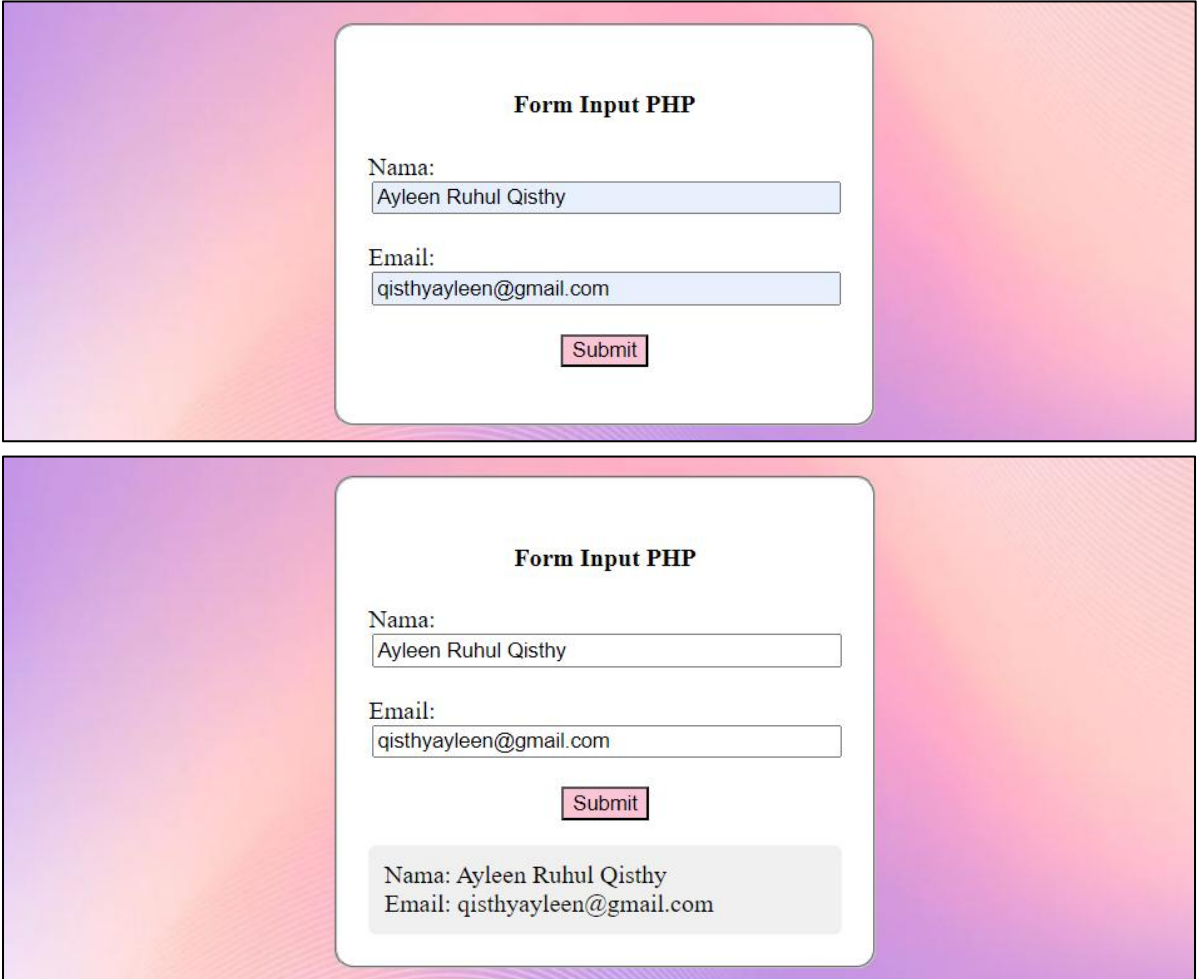
// Menggunakan htmlspecialchars untuk menghindari HTML injection saat menampilkan data
echo '<div>' . htmlspecialchars($user_input, ENT_QUOTES, 'UTF-8') . '</div>';
```

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 lainnya supaya langkah 2 bisa berjalan normal. Lalu simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/html_aman.php</code>
4	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.1)</p> 

	<p>Penjelasan : Kode tersebut mengambil input dari form POST dan diubah menjadi entitas HTML untuk mencegah serangan XSS menggunakan fungsi htmlspecialchars().</p>
5	<p>Ketikkan kode tambahan pada langkah 6 di dalam <code>html_aman.php</code></p>
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	<p>Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/html_aman.php</code></p>
8	<div> <div> <p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.2)</p> </div> <div>  </div> </div> <p>Penjelasan : Kode tersebut melakukan validasi email menggunakan <code>filter_var</code> dengan flag <code>FILTER_VALIDATE_EMAIL</code> untuk memeriksa apakah input email memiliki format yang valid.</p>

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>
3	Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/regex.php</code>
4	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.1)</p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;">Huruf kecil ditemukan!</div> <p>Penjelasan : Kode tersebut digunakan untuk mencari dan mencocokkan huruf kecil pada variabel <code>\$pattern</code> dengan teks pada variabel <code>\$text</code>. Kemudian text pada variabel <code>\$text</code> tersebut dicari dengan huruf yang cocok pada variabel <code>\$pattern</code> yaitu huruf kecil. Sehingga menampilkan “Huruf kecil ditemukan!”</p>
5	Ketikkan kode tambahan pada langkah 6 di dalam <code>regex.php</code>
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	Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/regex.php</code>
8	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.2)</p> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;">Huruf kecil ditemukan!</div> <div style="border: 1px solid black; padding: 5px; margin: 5px 0;">Cocokkan: 123</div> <p>Penjelasan : Kode tambahan tersebut digunakan untuk mencocokkan sebuah huruf pada variabel <code>\$pattern</code> dengan teks angka pada variabel <code>text</code>. Kemudian angka pada variabel <code>\$text</code> dicocokkan dengan angka pada variabel <code>\$pattern</code> yaitu <code>[0-9]</code>. Sehingga angka yang cocok yaitu 1, 2, dan 3.</p>

9	Ketikkan kode tambahan pada langkah 10 di dalam <code>regex.php</code>
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	Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/regex.php</code>
12	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.3)</p> <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p>Huruf kecil ditemukan!</p> <p>Cocokkan: 123</p> <p>I like banana pie.</p> </div> <p>Penjelasan : Kode tersebut terdapat variabel <code>pattern</code> dengan kata yang dicari yaitu <code>/apple/</code>. Variabel <code>\$replacement</code> terdapat kata pengganti yaitu <code>'banana'</code>. Kemudian terdapat variabel <code>\$text</code> yaitu <code>'I like apple pie.'</code>. Kemudian mencari kata baru dengan variabel <code>\$new_text</code> menggunakan fungsi <code>preg_replace</code> yang digunakan untuk mengganti regex dengan kata pengganti, yaitu menggantikan kata <code>apple</code> dengan <code>banana</code> dari variabel <code>\$text</code> tersebut. Sehingga menampilkan <code>'I like banana pie.'</code></p>
13	Ketikkan kode tambahan pada langkah 14 di dalam <code>regex.php</code>
14	<pre> \$pattern = '/go*d/'; // Cocokkan "god", "good", "gooooo", dll. \$text = 'god is good.'; if (preg_match(\$pattern, \$text, \$matches)) { echo "Cocokkan: " . \$matches[0]; } else { echo "Tidak ada yang cocok!"; } </pre>
15	Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/regex.php</code>
16	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.4)</p> <div style="border: 1px solid black; padding: 10px; margin: 10px 0;"> <p>Huruf kecil ditemukan!</p> <p>Cocokkan: 123</p> <p>I like banana pie.</p> <p>Cocokkan: god</p> </div>

	<p>Penjelasan : Kode tersebut digunakan untuk mencari dan mencocokkan huruf /go*d/ pada variabel \$pattern dengan teks pada variabel \$text. Fungsi preg_match digunakan untuk mencocokkan kata dari variabel \$pattern yaitu kata dengan awal huruf g dan o (o > 0) dan diakhiri dengan huruf d. Teks “god is good” terdapat kecocokan yaitu kata god, sehingga menampilkan “Cocokan: god”</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> <pre> 43 44 \$pattern = '/appl?e/'; 45 \$text = 'I like apple pie. I like ape pie.'; 46 \$new_text = preg_replace(\$pattern, 'banana', \$text); 47 echo \$new_text; 48 </pre> <p>I like banana pie. I like ape pie.</p> <p>Penjelasan : Kode tersebut terdapat variabel \$pattern dengan nilai /appl?e/ berarti karakter l opsional muncul 0/1 kali. Sehingga text dalam variabel \$text yang cocok dengan pola tersebut adalah apple. Fungsi preg_replace() untuk mengganti text yang cocok dengan kata baru yaitu banana. Sehingga menampilkan "I like banana pie. I like ape pie."</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> <pre> 50 51 \$pattern = '/o{1,2}/'; // Mencocokkan karakter 'o' yang muncul 1 hingga 2 kali 52 \$text = 'good food ghost box cool'; 53 54 if (preg_match_all(\$pattern, \$text, \$matches)) { 55 echo "Cocokkan ditemukan: " . implode(', ', \$matches[0]); 56 } else { 57 echo "Tidak ada yang cocok!"; 58 } 59 </pre> <p>Cocokkan ditemukan: oo, oo, o, o, oo</p> <p>Penjelasan : Kode tersebut terdapat variable \$pattern /o{1,2}/ untuk mencocokkan karakter huruf o yang muncul 1/2 kali. Fungsi preg_match_all() digunakan untuk mencari pola karakter tersebut dalam text pada variable \$text, yaitu 'good food ghost box cool'. Hasil pencocokan tersebut kemudian disimpan dalam array \$matches. Sehingga jika ada kecocokan, maka akan dicetak sehingga menampilkan “oo, oo, o, o, oo”</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

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

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

 <input type="submit" value="Submit"> </form> </body> </html> </pre> <pre> 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 . "
"; if (!empty(\$selectedWarna)) { echo "Warna favorit Anda: " . implode(", ", \$selectedWarna) . "
"; } else { echo "Anda tidak memilih warna favorit.
"; } echo "Jenis kelamin Anda: " . \$selectedJenisKelamin; } </pre>
3	Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/form_lanjut.php
4	Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 6.1)

Form Contoh

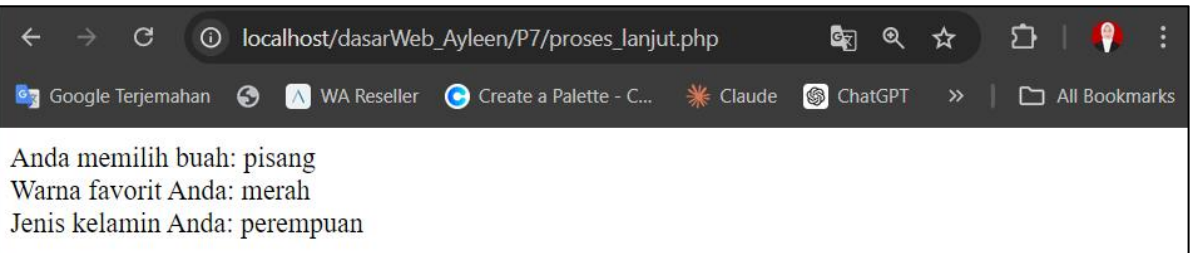
Pilih Buah:

Pilih Warna Favorit:

- ☒ Merah
☐ Biru
☐ Hijau

Pilih Jenis Kelamin:

- ☐ Laki-laki
☒ Perempuan



Penjelasan : Kode tersebut menampilkan form dan pengguna dapat memilih satu dari menu dropdown, memilih beberapa warna dari checkbox, dan memilih jenis kelamin dari radio button. Saat form dikirimkan dengan metode POST, input tersebut akan diproses. Sehingga halaman web browser menampilkan seperti pada gambar tersebut.

5

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>
  </form>
</body>
```

```

        <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>

```

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:

Pilih Warna Favorit:

- ☒ Merah
- ☒ Biru
- ☒ Hijau

Pilih Jenis Kelamin:

- ☐ Laki-laki
- ☒ Perempuan

Anda memilih buah: apel

Warna favorit Anda: merah, biru, hijau

Jenis kelamin Anda: perempuan

Penjelasan : Kode tersebut membuat form dengan jquery yaitu pengguna dapat memilih buah, warna favorit, dan jenis kelamin. Saat pengguna menekan tombol "Submit", jQuery mengumpulkan data dari form dan mengirimkannya secara asinkron (tanpa me-refresh halaman) ke file PHP (proses_lanjut.php) menggunakan metode POST. Respons dari server kemudian ditampilkan di div dengan id "hasil". Sehingga pengguna dapat melihat langsung hasil tampilan tanpa harus membuka halaman baru.

Praktikum 7 : Validasi Form

Langkah-langkah Praktikum :

Langkah	Keterangan
1	Buat satu file baru bernama <code>form_validasi.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>form_validasi.php</code>
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> <pre> <?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 . "
"; }) } 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"; } } ?> </pre>
3	Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/form_validasi.php</code>
4	Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.1)

Form Input dengan Validasi

Nama:

Email:

Data berhasil dikirim: Nama = Ayleen Ruhul Qisthy, Email = qisthyayleen@gmail.com

Jika submit tanpa mengirimkan input

Form Input dengan Validasi

Nama:

Email:

Nama harus diisi.
Email harus diisi.

Penjelasan : Kode tersebut membuat sebuah form input dengan validasi sederhana. Form HTML meminta input "nama" dan "email" dari pengguna, kemudian dikirimkan ke file PHP untuk diproses. Pada PHP, terdapat proses validasi untuk memeriksa apakah nama dan email telah diisi, dan memastikan format email valid menggunakan fungsi `filter_var`. Jika terdapat kesalahan, pesan error akan ditampilkan; jika tidak, data berhasil diproses dan ditampilkan seperti pada gambar diatas.

5

Kembangkan file bernama `form_validasi.php` di dalam folder `dasarWeb`. Ketikkan kode pada langkah 6 di dalam `form_validasi.php`

6

```
<!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">
    <span id="nama-error" style="color: red;"></span><br>

    <label for="email">Email:</label>
    <input type="text" id="email" name="email">
    <span id="email-error" style="color: red;"></span><br>

    <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("");
        }
      });
    });
  </script>
</body>
</html>
```


	<pre> 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>
8	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.2)</p> <div> <h3>Form Input dengan Validasi</h3> <p>Nama: <input type="text" value="Ayleen Ruhul Qisthy"/></p> <p>Email: <input type="text"/> Email harus diisi.</p> <p><input type="button" value="Submit"/></p> </div> <p>Penjelasan : Kode tersebut terdapat validasi form dengan menggunakan jQuery dan PHP. Jika input kosong atau email tidak valid, jQuery mencegah pengiriman form dan menampilkan pesan kesalahan seperti pada gambar diatas. Validasi tersebut memastikan agar data yang dikirim benar sebelum diproses lebih lanjut.</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> <pre> 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 if (valid) { \$.ajax({ url: "proses_validasi.php", type: "POST", data: { nama: nama, email: email }, success: function(response) { \$("#result").html(response); }, error: function() { \$("#result").html("Terjadi kesalahan saat mengirim data."); } }); } </pre> <div> <h3>Form Input dengan Validasi dan AJAX</h3> <p>Nama: <input type="text" value="Ayleen Ruhul Qisthy"/></p> <p>Email: <input type="text" value="qisthyayleen@gmail.com"/></p> <p><input type="button" value="Submit"/></p> <p>Data berhasil dikirim: Nama = Ayleen Ruhul Qisthy, Email = qisthyayleen@gmail.com</p> </div> <p>Penjelasan : Kode tersebut menambahkan fitur untuk mengirim form data dengan AJAX tanpa me-refresh halaman. Jika input valid, maka data akan dikirim ke `proses_validasi.php` dan memeriksa nama dan email, kemudian pesan kesalahan atau konfirmasi akan ditampilkan.</p>

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)

jQuery

```

51
52     if (password === "") {
53         $("#password-error").text("Password harus diisi.");
54         valid = false;
55     } else if (password.length < 8) {
56         $("#password-error").text("Password harus minimal 8 karakter.");
57         valid = false;
58     } else {
59         $("#password-error").text("");
60     }
61

```

PHP

```

22
23     // Validasi password
24     if (empty($password)) {
25         $errors[] = 'Password harus diisi.';
26     } elseif (strlen($password) < 8) {
27         $errors[] = 'Password harus minimal 8 karakter.';
28     }
29

```

Form Input dengan Validasi dan AJAX

Nama:

Email:

Password: Password harus minimal 8 karakter.

Form Input dengan Validasi dan AJAX

Nama:

Email:

Password:

Data berhasil dikirim: Nama = Ayleen Ruhul Qisthy, Email = qisthyayleen@gmail.com, Password valid

Penjelasan : Kode tersebut menambahkan validasi untuk password. Password harus diisi dengan minimal 8 karakter. Jika validasi berhasil, data dikirim ke server menggunakan AJAX, dan PHP memeriksa ulang semua input (nama, email, password). Kemudian hasilnya akan ditampilkan di halaman web. Jika terdapat kesalahan, maka pesan error akan ditampilkan di sebelah from password.