

JOBSHEET 7
PHP - FORM PROCESSING
DESAIN DAN PEMROGRAMAN WEB



Disusun Oleh:
Muhammad Naufal Assyauqi Ardyawan
2241760046
Presensi 16 / Kelas 2E

PROGRAM STUDI D-IV SISTEM INFORMASI BISNIS
JURUSAN TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
TAHUN 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

Langka h	Keterangan
1	Buatlah satu file baru di dalam direktori <code>dasarWeb</code> , 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> <p>Output:</p> <p>Anda belum dewasa atau variabel 'umur' tidak ditemukan.</p> <p>fungsi isset() digunakan untuk mengecek apakah sebuah variable memiliki nilai atau tidak. Jika sebuah variable memiliki nilai, maka fungsi akan mengembalikan nilai TRUE, dan akan mengembalikan nilai FALSE jika tidak memiliki nilai.</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> <p>Kode:</p> <pre> <?php \$umur; if (isset(\$umur) && \$umur >= 18) { echo "Anda sudah dewasa."; } else { echo "Anda belum dewasa atau variabel 'umur' tidak ditemukan."; } </pre>

```

echo "<br>";

$data = array("nama" => "Jane", "usia" => 25);
if (isset($data["nama"])) {
    echo " Nama: " . $data["nama"];
}
else {
    echo "Variabel 'nama' tidak ditemukan dalam array.";
}
?>

```

Output:

Anda belum dewasa atau variabel 'umur' tidak ditemukan.
 Nama: Jane

Pemilihan pertama berjalan seperti sebelumnya, pemilihan kedua berjalan karena terdapat data nama pada array \$data, sehingga program meng echo.

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>
3	<p>Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/empty.php. Apa yang anda pahami dari penggunaan empty pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 2.1)</p> <p>Output:</p> <p>Array tidak terdefinisi atau kosong.</p> <p>fungsi empty() digunakan untuk mengecek apakah sebuah array kosong atau tidak. Apabila array tersebut kosong, maka fungsi akan mengembalikan nilai TRUE. Apabila array memiliki data, maka fungsi akan mengembalikan nilai FALSE.</p>
4	<p>Tambahkan isi dari file empty.php 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>
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> <p>Output:</p> <p>Array tidak terdefinisi atau kosong. variabel tidak terdefinisikan atau kosong.</p> <p>empty() juga dapat digunakan untuk mengecek apakah sebuah variable telah didefinisikan atau tidak ada. Jika variable tidak ada, fungsi akan mengembalikan nilai TRUE.</p>

Praktikum Bagian 3 : Form Input PHP

Langka h	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>
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. Catat di bawah ini pemahaman anda. (soal no 3.1)</p> <p>Output:</p>

	<h2>Form Input PHP</h2> <p>Nama: <input type="text" value="Naufal Assyauqi"/></p> <p>Email: <input type="text" value="syauqi.ardyawan@gmail.com"/></p> <p><input type="submit" value="Submit"/></p> <p>Setelah Submit: Nama: Naufal Assyauqi Email: syauqi.ardyawan@gmail.com</p> <p>File form.php akan menampilkan sebuah form, setelah data diisi dan diSubmit, data akan dikirim ke file proses_form.php dengan method POST untuk di echo.</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>
6	<p>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)</p> <p>Output:</p>

Form Input PHP

Data berhasil disimpan!

Nama:

Program akan menampilkan form yang dapat diisi. Apabila ada data yang diisi kemudian di klik submit, maka akan menampilkan pesan “Data berhasil disimpan”. Jika tombol submit di klik tanpa mengisi nama, akan tampil pesan “Nama harus diisi”.

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 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>Output:</p> <p>Input yang aman: <code><script>alert("Contoh");</script></code></p> <p>Masukkan Input</p> <p>Input: <input type="text"/></p> <p><input type="button" value="Kirim"/></p> <p>Input yang ditulis adalah sebuah contoh script yang akan menampilkan pesan alert. Setelah teks di kirim, pesan alert tidak muncul. Namun, sintaks ditampilkan setelah melalui fungsi <code>htmlspecialchars()</code></p>

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>
8	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.2)</p> <p>Output:</p> <p>Input yang aman: <code><script>alert("Contoh");</script></code> Email yang dimasukkan valid: <code>test@yahoo.com</code></p> <h3>Masukkan Input</h3> <p>Input: <input type="text"/></p> <p>Email: <input type="text"/></p> <p><input type="button" value="Kirim"/></p> <p>Melakukan penambahan field form Email. Email akan dicek validasinya sebelum ditampilkan. Apabila valid, maka teks akan ditampilkan. Apabila tidak valid, akan muncul pesan “Email yang dimasukkan tidak 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> <p>Output: Huruf kecil ditemukan!</p> <p>fungsi <code>preg_match()</code> dapat digunakan untuk menemukan satu karakter yang sama dari sebuah variable String. Karakter yang dicari ditentukan dengan variable <code>\$pattern</code>. Apabila sebuah karakter ditemukan, maka fungsi akan bernilai TRUE.</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> <p>Output:</p>

	<p>Huruf kecil ditemukan! Cocokkan: 123</p> <p>nilai yang dicari kemudian ditemukan dapat dimasukkan ke dalam sebuah variable yang baru. Variable tersebut dapat digunakan untuk sintaks echo.</p>
9	Ketikkan kode tambahan pada langkah 10 di dalam <code>regex.php</code>
10	<pre><code>\$pattern = '/apple/'; \$replacement = 'banana'; \$text = 'I like apple pie.'; \$new_text = preg_replace(\$pattern, \$replacement, \$text); echo \$new_text; // Output: "I like banana pie."</code></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> <p>Output:</p> <p>Huruf kecil ditemukan! Cocokkan: 123 I like banana pie.</p> <p>fungsi <code>preg_replace</code> dapat digunakan untuk menggantikan suatu data dalam sebuah variabel teks dengan isi data <code>\$replacement</code>. Kata yang akan diubah adalah yang sama dengan isi data <code>\$pattern</code>.</p>
13	Ketikkan kode tambahan pada langkah 14 di dalam <code>regex.php</code>
14	<pre><code>\$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!"; }</code></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> <p>Output:</p>

	<p>Huruf kecil ditemukan! Cocokkan: 123 I like banana pie. Cocokkan: god</p> <p>Penggunaan kuantifier “*” akan mencari kata dalam \$text yang memiliki jumlah huruf o sebanyak 0 atau lebih.</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>Kode</p> <pre>/* Soal 5.5 */ \$pattern = '/go?d/'; \$text = 'god is good'; if (preg_match(\$pattern, \$text, \$matches)) { echo "Cocokkan: " . \$matches[0] . "
"; } else { echo "Tidak ada yang cocok!
"; }</pre> <p>Output</p> <p>Huruf kecil ditemukan! Cocokkan: 123 I like banana pie. Cocokkan: god Cocokkan: god</p> <p>Kuantifier “?” akan mencari kata yang memiliki jumlah huruf o sebanyak 0 atau 1 kali. Kata yang sama akan disimpan dalam array \$matches</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>Kode:</p> <pre>\$pattern = '/go{1,3}d/'; \$text = 'god is good'; if (preg_match(\$pattern, \$text, \$matches)) { echo "Cocokkan: " . \$matches[0] . "
"; } else { echo "Tidak ada yang cocok!
"; }</pre> <p>Output:</p>

	<p>Huruf kecil ditemukan!</p> <p>Cocokkan: 123</p> <p>I like banana pie.</p> <p>Cocokkan: god</p> <p>Cocokkan: god</p> <p>Cocokkan: god</p> <p>Kuantifier "{n,m}" akan menemukan kata-kata yang mengandung huruf o berjumlah minimal 1 dan maksimal 3. Kata yang ditemukan akan masuk ke array \$matches.</p>
--	---

Praktikum 6 : Form Lanjut

Langkah-langkah Praktikum :

Langkah	Keterangan
1	Buat satu file baru bernama <code>form_lanjut.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>form_lanjut.php</code>
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> 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	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 6.1)</p> <p>Output</p> <p>Form Contoh</p> <p>Pilih Buah: <input type="text" value="Apel"/></p> <p>Pilih Warna Favorit:</p> <p><input type="checkbox"/> Merah</p> <p><input checked="" type="checkbox"/> Biru</p> <p><input type="checkbox"/> Hijau</p> <p>Pilih Jenis Kelamin:</p> <p><input checked="" type="radio"/> Laki-laki</p> <p><input type="radio"/> Perempuan</p> <p><input type="button" value="Submit"/></p> <p>Anda memilih buah: apel Warna favorit Anda: biru Jenis kelamin Anda: laki-laki</p> <p>Akan muncul form yang perlu diisi. Setelah tombol submit di klik, pengguna akan <i>redirect</i> ke file proses_lanjut.php yang berada pada URL lain.</p>
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>

    <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)

Output :

	<h2>Form Contoh</h2> <p>Pilih Buah: <input type="text" value="Pisang"/></p> <p>Pilih Warna Favorit:</p> <p><input type="checkbox"/> Merah</p> <p><input checked="" type="checkbox"/> Biru</p> <p><input type="checkbox"/> Hijau</p> <p>Pilih Jenis Kelamin:</p> <p><input checked="" type="radio"/> Laki-laki</p> <p><input type="radio"/> Perempuan</p> <p><input type="button" value="Submit"/></p> <p>Anda memilih buah: pisang Warna favorit Anda: biru Jenis kelamin Anda: laki-laki</p> <p>Dengan menggunakan AJAX, form yang ditampilkan sama. Namun, hasil form ditampilkan tanpa <i>redirect</i> ke URL lain.</p>
--	--

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	<p>Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/form_validasi.php</p>
4	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.1)</p> <p>Output</p> <h2>Form Input dengan Validasi</h2> <div> <p>Nama: <input type="text" value="Syauqi"/></p> <p>Email: <input type="text" value="syauqi.ardyawan@gmail.com"/></p> <p><input type="button" value="Submit"/></p> </div> <p>Setelah klik submit:</p> <p>Data berhasil dikirim: Nama Syauqi, Email syauqi.ardyawan@gmail.com</p> <p>data yang telah diisi pada Form Input akan dikirim kepada file proses_validasi.php. Kemudian, data di validasi. Setelah selesai, program akan meng-echo data yang telah diinput.</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>
8	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.2)</p> <p>Output:</p> <h2 style="text-align: center;">Form Input dengan Validasi</h2> <div style="display: flex; align-items: flex-start;"> <div style="margin-right: 10px;"> Nama: <input type="text"/> Email: <input type="text"/> <input type="button" value="Submit"/> </div> <div> Nama harus diisi. Email harus diisi. </div> </div> <p>Penambahan kode akan menampilkan pesan error di sebelah kanan form yang menyatakan bahwa field nama dan email harus diisi. Tombol submit tidak akan melakukan <i>redirect</i> apabila kolom nama dan email belum 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>Kode Program:</p> <pre> <html> <head> </pre>

```

<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("");
    }
    if (valid) {
      $.ajax({
        type: "POST",
        url: "proses_validasi.php",
        data: {nama: nama, email: email},
        success: function(response) {
          $("#response").html(response);
        }
      });
    }
    else {
      !event.preventDefault();
    }
  });
});
</script>
</body>
</html>

```

Output

Form Input dengan Validasi

Nama:	<input type="text" value="Syauqi"/>
Email:	<input type="text" value="syauqi.ardyawan@gmail.com"/>
<input type="button" value="Submit"/>	

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)

Kode Program:

-file form_validasi.php

```
<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>

    <label for="password">Password:</label>
    <input type="password" id="password" name="password">
    <span id="password-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 password = $("#password").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;
        }
      });
    });
  </script>
</body>
</html>
```

10

```

    }
    else {
        $("#email-error").text("");
    }

    if (password.length < 8) {
        $("#password-error").text("Password minimal 8 karakter.");
        valid = false;
    }
    else {
        $("#password-error").text("");
    }
    if (valid) {
        $.ajax({
            type: "POST",
            url: "proses_validasi.php",
            data: {nama: nama, email: email, password: password},
            success: function(response) {
                $("#response").html(response);
            }
        });
    }
    else {
        !event.preventDefault();
    }
});
});
</script>
</body>
</html>

```

-file proses_validasi.php:

```

<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $nama = $_POST["nama"];
    $email = $_POST["email"];
    $password = $_POST["password"];
    $errors = array();
    if (empty($nama)) {
        $errors[] = "Nama harus diisi.";
    }
    if (empty($email)) {
        $errors[] = "Email harus diisi.";
    }
    else {
        if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
            $errors[] = "Format email tidak valid.";
        }
    }
    if (strlen($password) < 8) {
        $errors[] = "Password minimal 8 karakter.";
    }
    if (!empty($errors)) {
        foreach ($errors as $error) {
            echo $error . "<br>";
        }
    }
    else {
        echo "Data berhasil dikirim: Nama $nama, Email $email, Password $password";
    }
}

```

```
}  
?>
```

Output:
Sebelum diisi

Form Input dengan Validasi

Nama: Nama harus diisi.
Email: Email harus diisi.
Password: Password minimal 8 karakter.

Setelah diisi dan password hanya 6 digit.

Form Input dengan Validasi

Nama:
Email:
Password: Password minimal 8 karakter.

setelah diisi dan password 8 digit

Data berhasil dikirim: Nama Syauqi, Email syauqi.ardyawana@gmail.com, Password 12345678