

## **Jobsheet-7: PHP – Form Proccessing**

**Tugas Mata Kuliah**  
Desain dan Pemrograman Web



oleh :  
Khoirul Hidayah  
2241760021  
SIB 2A (15)

**Program Studi Diploma IV Sistem Informasi Bisnis**  
**Jurusan Teknologi Informasi**  
**Politeknik Negeri Malang**



### 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

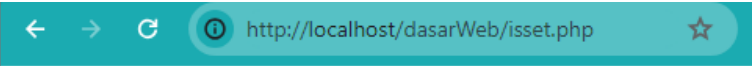
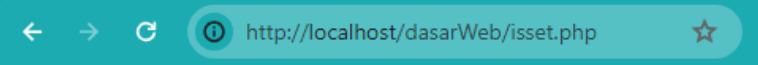
#### Function isset

**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) &amp;&amp; \$umur &gt;= 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>Jawab:</p>  <p>Penggunaan isset() pada file tersebut memastikan bahwa variabel \$umur sudah didefinisikan dan memiliki nilai sebelum dilakukan pengecekan kondisi.</p>
5	<p>Tambahkan isi dari file isset.php tersebut dengan kode di bawah ini.</p> <pre data-bbox="268 510 1332 817"> \$data = array("nama" =&gt; "Jane", "usia" =&gt; 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>Nama: Jane</p> <p>Kode tersebut memeriksa apakah key "nama" terdefinisi dalam array \$data, dan jika iya, maka akan mencetak nilai dari key tersebut; jika tidak, maka akan mencetak pesan bahwa 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()**:

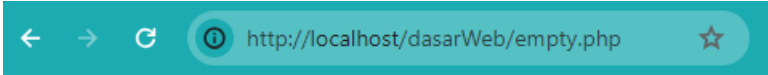
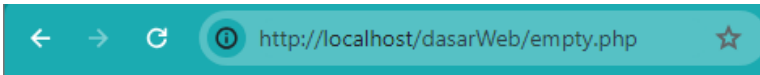
```

$nama = ""; // Variabel string kosong
if (empty($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

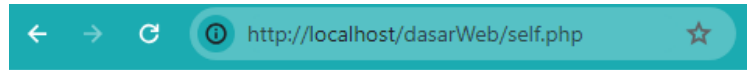
Langkah	Keterangan
1	Buat satu file baru bernama empty.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam empty.php
2	<pre> \$myArray = array(); // Array kosong if (empty(\$myArray)) {     echo "Array tidak terdefinisi atau kosong."; } else {     echo "Array terdefinisi dan tidak kosong."; } </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>Jawab:</p>  <p>Array tidak terdefinisi atau kosong.</p> <p>Kode tersebut menguji apakah array \$myArray kosong atau tidak terdefinisi. Jika array kosong atau tidak terdefinisi, maka pesan "Array tidak terdefinisi atau kosong." akan dicetak; jika tidak, maka pesan "Array terdefinisi dan tidak kosong." akan dicetak.</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>Jawab:</p>  <p>Array tidak terdefinisi atau kosong.</p> <p>Variabel tidak terdefinisi atau kosong.</p> <p>Kode tersebut menguji apakah variabel \$nonExistentVar tidak terdefinisi atau kosong. Jika variabel tidak terdefinisi atau kosong, maka pesan "Variabel tidak terdefinisi atau kosong." akan dicetak; jika tidak, maka pesan "Variabel terdefinisi dan tidak kosong." akan dicetak.</p>

## Praktikum Bagian 3 : Form Input PHP

Langkah	Keterangan
1	Buat satu file baru bernama form.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam form.php
2	<pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;   &lt;title&gt;Form Input PHP&lt;/title&gt; &lt;/head&gt; &lt;body&gt;   &lt;h2&gt;Form Input PHP&lt;/h2&gt;   &lt;form method="post" action="proses_form.php"&gt;     &lt;label for="nama"&gt;Nama:&lt;/label&gt;     &lt;input type="text" name="nama" id="nama" required&gt;&lt;br&gt;&lt;br&gt;      &lt;label for="email"&gt;Email:&lt;/label&gt;     &lt;input type="email" name="email" id="email" required&gt;&lt;br&gt;&lt;br&gt;      &lt;input type="submit" name="submit" value="Submit"&gt;   &lt;/form&gt; &lt;/body&gt; &lt;/html&gt; </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> &lt;?php if (\$_SERVER["REQUEST_METHOD"] == "POST") {   \$nama = \$_POST["nama"];   \$email = \$_POST["email"];    echo "Nama: " . \$nama . "&lt;br&gt;";   echo "Email: " . \$email; } ?&gt; </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><b>Form Input PHP</b></p> <p>Nama: <input type="text"/></p> <p>Email: <input type="text"/></p> <p><input type="submit" value="Submit"/></p> <p>Submit</p>  <p><b>Form Input PHP</b></p> <p>Nama: <input type="text" value="Khoirul Hidayah"/></p> <p>Email: <input type="text" value="rullhida23@gmail.com"/></p> <p><input type="submit" value="Submit"/></p> <p>Nama: Khoirul Hidayah Email: rullhida23@gmail.com</p> <ul style="list-style-type: none"> <li>• Dalam contoh ini, action dari formulir ditetapkan ke <code>&lt;?php echo \$_SERVER['PHP_SELF']; ?&gt;</code>, yang artinya formulir akan mengirimkan data ke halaman itu sendiri untuk diproses.</li> <li>• Kode pemrosesan formulir diletakkan setelah formulir HTML di dalam blok PHP, sehingga ketika formulir dikirimkan, data akan diproses dan hasilnya akan ditampilkan pada halaman yang sama.</li> </ul>
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> <pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;   &lt;title&gt;Form Input PHP&lt;/title&gt; &lt;/head&gt; &lt;body&gt;   &lt;h2&gt;Form Input PHP&lt;/h2&gt;   &lt;?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!";     }   }   ?&gt;    &lt;form method="post" action="&lt;?php echo htmlspecialchars(\$_SERVER["PHP_SELF"]); ?&gt;"&gt;     &lt;label for="nama"&gt;Nama:&lt;/label&gt;     &lt;input type="text" name="nama" id="nama" value="&lt;?php echo \$nama; ?&gt;"&gt;     &lt;span class="error"&gt;&lt;?php echo \$namaErr; ?&gt;&lt;/span&gt;&lt;br&gt;&lt;br&gt;     &lt;input type="submit" name="submit" value="Submit"&gt;   &lt;/form&gt; &lt;/body&gt; &lt;/html&gt; </pre>
6	<p>Simpan file tersebut, kemudian buka browser dan jalankan <code>localhost/dasarWeb/form_self.php</code>. Apa yang anda pahami dari penggunaan form pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 3.2)</p>

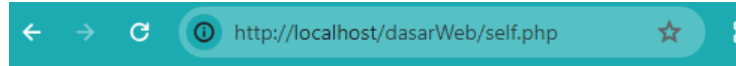
Jawab:



## Form Input PHP

Nama:

Submit



## Form Input PHP

Data berhasil disimpan!

Nama:

Dalam kode ini, saya telah menempatkan validasi nama di dalam blok PHP setelah pengecekan apakah formulir telah disubmit. Saya juga menggunakan fungsi htmlspecialchars() untuk mencegah serangan XSS (cross-site scripting) dengan mengonversi karakter khusus 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 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>';
```

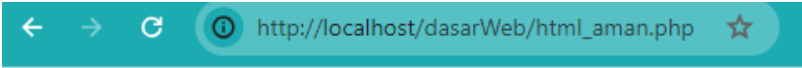
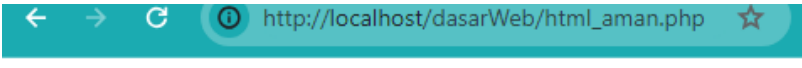
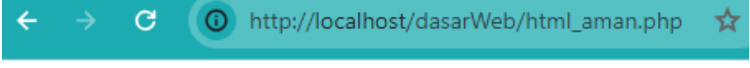
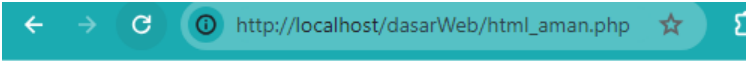
Dengan menggunakan langkah-langkah ini, Anda dapat memitigasi risiko HTML injection dalam aplikasi web Anda. Selalu penting untuk memvalidasi input, membersihkan data yang masuk, dan menghindari menampilkan data pengguna tanpa escaping yang benar.

### 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 localhost/dasarWeb/html_aman.php
4	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.1) Jawab:</p>  <p>Input: <input type="text"/> Submit</p> <p>Input yang diamankan:</p> <p>Submit</p>  <p>Input: Khoirul Hidayah Submit</p> <p>Input yang diamankan: Khoirul Hidayah</p> <p>Dengan perubahan tersebut, kita memastikan bahwa variabel \$input hanya diakses setelah memeriksa apakah data 'input' sudah ada dalam array \$_POST. Selain itu, kita juga menginisialisasi \$input dengan nilai kosong pada awalnya untuk menghindari peringatan deprecated dari htmlspecialchars().</p>
5	Ketikkan kode tambahan pada langkah 6 di dalam html_aman.php
6	<pre>// Memeriksa 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 localhost/dasarWeb/html_aman.php
8	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.2) Jawab:</p>  <p>Input: <input type="text"/></p> <p>Email: <input type="text"/></p> <p>Submit</p> <p>Input yang diamankan:</p> <p>Submit</p>  <p>Email yang dimasukkan adalah email yang valid.</p> <p>Input: Khoirul Hidayah</p> <p>Email: rullhida23@gmail.com</p> <p>Submit</p> <p>Input yang diamankan: Khoirul Hidayah</p>

Dalam kode di atas, saya telah menambahkan pengecekan untuk memastikan bahwa elemen 'email' telah didefinisikan sebelum mencoba mengaksesnya
--

## 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

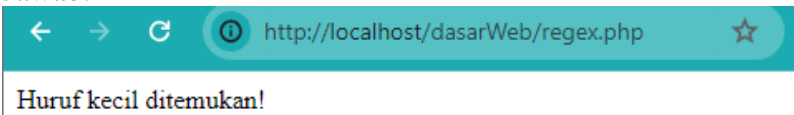
```

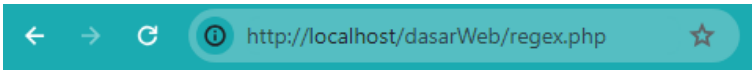
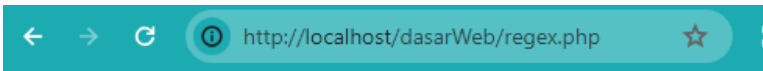
$pattern = '/[0-9]/'; // Cocokkan satu digit.
$text = 'There are 3 apples.';
if (preg_match($pattern, $text)) {
    echo "Digit ditemukan!";
} else {
    echo "Tidak ada digit!";
}

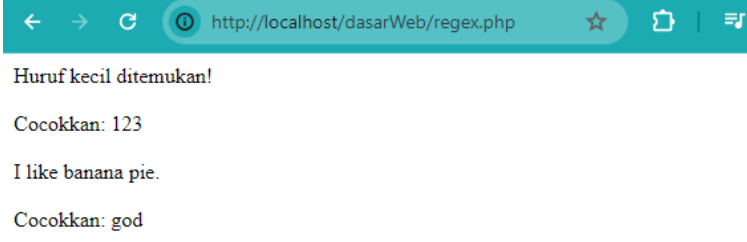
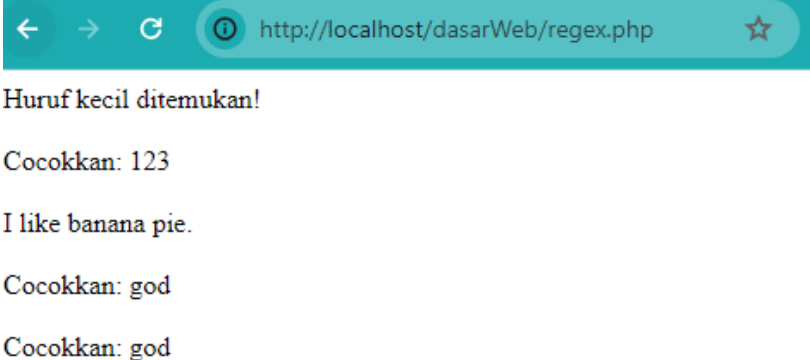
```

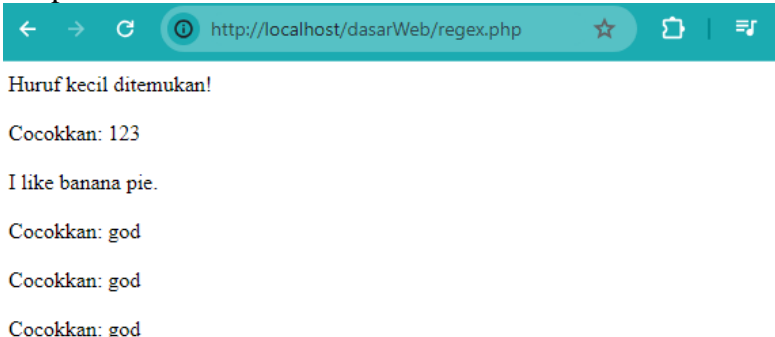
## Praktikum 5 : Penggunaan Regex Pada PHP

Langkah-langkah Praktikum :

Langkah	Keterangan
1	Buat satu file baru bernama regex.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam regex.php
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 localhost/dasarWeb/regex.php
4	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.1) Jawab:</p>  <p>Kode tersebut menguji apakah terdapat setidaknya satu huruf kecil dalam teks yang diberikan, dan mencetak pesan yang sesuai tergantung pada hasilnya.</p>
5	Ketikkan kode tambahan pada langkah 6 di dalam regex.php

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 localhost/dasarWeb/regex.php
8	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.2) Jawab:</p>  <p>Huruf kecil ditemukan!</p> <p>Cocokkan: 123</p> <p>Kode tersebut mencoba mencocokkan satu atau lebih digit dalam teks yang diberikan, dan jika ada, mencetak angka yang cocok. Jika tidak ada, maka mencetak pesan "Tidak ada yang cocok!".</p>
9	Ketikkan kode tambahan pada langkah 10 di dalam regex.php
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 localhost/dasarWeb/regex.php
12	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.3) Jawab:</p>  <p>Huruf kecil ditemukan!</p> <p>Cocokkan: 123</p> <p>I like banana pie.</p> <p>Kode tersebut mengganti setiap kemunculan pola '/apple/' dalam teks dengan kata 'banana' menggunakan fungsi preg_replace(). Hasilnya, kata 'apple' diganti dengan 'banana', dan teks baru ditampilkan</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>
16	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.4) Jawab:</p>  <p>Dengan perbaikan tersebut, output akan menampilkan kata yang cocok dengan pola yang ditentukan, dalam kasus ini "god" dan "good".</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) Jawab: Query:</p> <pre> 40 \$pattern = '/go?d/'; // Cocokkan "god", "good". 41 \$text = 'god is good.'; 42 if (preg_match(\$pattern, \$text, \$matches)) { 43     echo "Cocokkan: " . \$matches[0]; 44 } else { 45     echo "Tidak ada yang cocok!"; 46 } </pre> <p>Output:</p>  <p>Dalam kode ini, penggunaan `?` di dalam pola '/go?d/' memungkinkan huruf 'o' untuk muncul 0 atau 1 kali di dalam kata yang dicocokkan. Sehingga, kata "god" dan "good" akan cocok dengan pola yang ditentukan.</p>

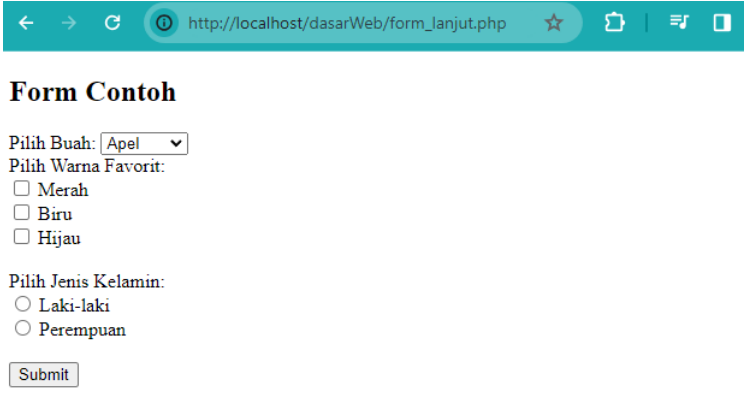
18	<p>Soal 5.6: Buatlah script langkah 14 dengan variable pattern yang di ubah menggunakan `{n,m}`. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.6)</p> <p>Jawab:</p> <p>Query:</p> <pre> 50 \$pattern = '/go{1,2}d/'; // Cocokkan "good", "gooood". 51 \$text = 'god is good.'; 52 if (preg_match(\$pattern, \$text, \$matches)) { 53     echo "Cocokkan: " . \$matches[0]; 54 } else { 55     echo "Tidak ada yang cocok!"; 56 } </pre> <p>Output:</p>  <p>Dalam kode ini, penggunaan <code>{1,2}</code> di dalam pola <code>/go{1,2}d/</code> memungkinkan huruf 'o' untuk muncul minimal 1 kali dan maksimal 2 kali di dalam kata yang dicocokkan. Sehingga, kata "good" dan "gooood" akan cocok dengan pola yang ditentukan.</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> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;   &lt;title&gt;Contoh Form dengan PHP&lt;/title&gt; &lt;/head&gt; &lt;body&gt;   &lt;h2&gt;Form Contoh&lt;/h2&gt;   &lt;form method="POST" action="proses_lanjut.php"&gt;     &lt;label for="buah"&gt;Pilih Buah:&lt;/label&gt;     &lt;select name="buah" id="buah"&gt;       &lt;option value="apel"&gt;Apel&lt;/option&gt;       &lt;option value="pisang"&gt;Pisang&lt;/option&gt;       &lt;option value="mangga"&gt;Mangga&lt;/option&gt;       &lt;option value="jeruk"&gt;Jeruk&lt;/option&gt;     &lt;/select&gt;      &lt;br&gt;      &lt;label&gt;Pilih Warna Favorit:&lt;/label&gt;&lt;br&gt;     &lt;input type="checkbox" name="warna[]" value="merah"&gt; Merah&lt;br&gt;     &lt;input type="checkbox" name="warna[]" value="biru"&gt; Biru&lt;br&gt;     &lt;input type="checkbox" name="warna[]" value="hijau"&gt; Hijau&lt;br&gt;      &lt;br&gt;      &lt;label&gt;Pilih Jenis Kelamin:&lt;/label&gt;&lt;br&gt;     &lt;input type="radio" name="jenis_kelamin" value="laki-laki"&gt; Laki-laki&lt;br&gt;     &lt;input type="radio" name="jenis_kelamin" value="perempuan"&gt; Perempuan&lt;br&gt;      &lt;br&gt;      &lt;input type="submit" value="Submit"&gt;   &lt;/form&gt; &lt;/body&gt; &lt;/html&gt; ' 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 . "&lt;br&gt;";    if (!empty(\$selectedWarna)) {     echo "Warna favorit Anda: " . implode(", ", \$selectedWarna) . "&lt;br&gt;";   } else {     echo "Anda tidak memilih warna favorit.&lt;br&gt;";   }    echo "Jenis kelamin Anda: " . \$selectedJenisKelamin; } </pre>
3	<p>Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/form_lanjut.php</p>
4	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 6.1) Jawab:</p>

	 <p><b>Form Contoh</b></p> <p>Pilih Buah: <input type="text" value="Apel"/></p> <p>Pilih Warna Favorit:</p> <p><input type="checkbox"/> Merah</p> <p><input type="checkbox"/> Biru</p> <p><input type="checkbox"/> Hijau</p> <p>Pilih Jenis Kelamin:</p> <p><input type="radio"/> Laki-laki</p> <p><input type="radio"/> Perempuan</p> <p><input type="button" value="Submit"/></p> <p>Script PHP di atas bertujuan untuk memproses data formulir yang dikirimkan melalui metode POST setelah formulir HTML yang berisi pilihan buah, warna favorit, dan jenis kelamin dipilih oleh pengguna.</p>
5	<p>Buat satu file baru bernama <code>form_ajax.php</code> di dalam folder <code>dasarWeb</code>. Ketikkan kode pada langkah 2 di dalam <code>form_ajax.php</code></p>



```

<!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)  
Jawab:

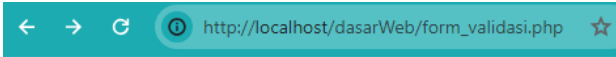
	 <p><b>Form Contoh</b></p> <p>Pilih Buah: <span>Apel</span></p> <p>Pilih Warna Favorit:</p> <p><input type="checkbox"/> Merah</p> <p><input type="checkbox"/> Biru</p> <p><input type="checkbox"/> Hijau</p> <p>Pilih Jenis Kelamin:</p> <p><input type="radio"/> Laki-laki</p> <p><input type="radio"/> Perempuan</p> <p><input type="button" value="Submit"/></p> <p>Form HTML di atas menggunakan jQuery untuk mengirim data formulir ke server secara asinkron menggunakan Ajax. Setelah data formulir dikirim, hasil dari server akan ditampilkan kembali ke halaman dalam div dengan id "hasil".</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> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;     &lt;title&gt;Form Input dengan Validasi&lt;/title&gt; &lt;/head&gt; &lt;body&gt;     &lt;h1&gt;Form Input dengan Validasi&lt;/h1&gt;     &lt;form method="post" action="proses_validasi.php"&gt;         &lt;label for="nama"&gt;Nama:&lt;/label&gt;         &lt;input type="text" id="nama" name="nama"&gt;         &lt;br&gt;           &lt;label for="email"&gt;Email:&lt;/label&gt;         &lt;input type="text" id="email" name="email"&gt;         &lt;br&gt;           &lt;input type="submit" value="Submit"&gt;     &lt;/form&gt; &lt;/body&gt; &lt;/html&gt; </pre>

	<pre> &lt;?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 . "&lt;br&gt;";         }     } 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";     } } ?&gt; </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) Jawab:</p>  <p>Script tersebut adalah formulir HTML yang meminta pengguna untuk memasukkan nama dan email. Kemudian, menggunakan PHP, dilakukan validasi pada kedua input tersebut. Jika ada kesalahan validasi, pesan kesalahan akan ditampilkan. Jika tidak, data yang dimasukkan akan diproses.</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> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;   &lt;title&gt;Form Input dengan Validasi&lt;/title&gt;   &lt;script src="https://code.jquery.com/jquery-3.6.0.min.js"&gt;&lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt;h1&gt;Form Input dengan Validasi&lt;/h1&gt;   &lt;form id="myForm" method="post" action="proses_validasi.php"&gt;     &lt;label for="nama"&gt;Nama:&lt;/label&gt;     &lt;input type="text" id="nama" name="nama"&gt;     &lt;span id="nama-error" style="color: red;"&gt;&lt;/span&gt;&lt;br&gt;      &lt;label for="email"&gt;Email:&lt;/label&gt;     &lt;input type="text" id="email" name="email"&gt;     &lt;span id="email-error" style="color: red;"&gt;&lt;/span&gt;&lt;br&gt;      &lt;input type="submit" value="Submit"&gt;   &lt;/form&gt;    &lt;script&gt;     \$(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         }       });     });   &lt;/script&gt; &lt;/body&gt; &lt;/html&gt; </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) Jawab:</p>  <p><b>Form Input dengan Validasi</b></p> <p>Nama: <input type="text"/>  Email: <input type="text"/>  <input type="button" value="Submit"/></p> <p><b>Form Input dengan Validasi</b></p> <p>Nama: <input type="text"/>  Email: <input type="text"/>  <input type="button" value="Submit"/></p> <p>Dari penambahan kode program di atas, dapat diamati bahwa:</p> <ul style="list-style-type: none"> <li>• Dilakukan validasi pada input nama dan email sebelum formulir dikirim.</li> <li>• Pesan kesalahan validasi ditampilkan di bawah input yang sesuai.</li> <li>• Formulir tidak akan dikirim jika ada kesalahan validasi yang terdeteksi.</li> <li>• Validasi dilakukan menggunakan JavaScript dengan bantuan jQuery untuk memastikan data yang dikirim sesuai dengan yang diharapkan.</li> </ul>

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)

Jawab:

Query:

```
//menggunakan ajax
<!DOCTYPE html>
<html>
<head>
  <title>Form Input dengan Validasi (Ajax)</title>
  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
</head>
<body>
  <h1>Form Input dengan Validasi (Ajax)</h1>
  <form id="myForm" method="post">
    <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="button" value="Submit" id="submit">
  </form>
  <div id="result"></div>
  <script>
    $(document).ready(function() {
      $("#submit").click(function() {
        var nama = $("#nama").val();
        var email = $("#email").val();
        $.ajax({
          type: "POST",
          url: "proses_validasi_ajax.php", // Ganti dengan URL file
          data: { nama: nama, email: email },
          success: function(response) {
            $("#result").html(response);
          }
        });
      });
    });
  </script>
</body>
</html>
```

PHP yang sesuai

Output:

	<div data-bbox="268 107 1099 154"> </div> <h2 data-bbox="272 188 798 235">Form Input dengan Validasi</h2> <div data-bbox="272 264 577 349"> <p>Nama: <input type="text"/></p> <p>Email: <input type="text"/></p> <p><input type="button" value="Submit"/></p> </div> <h2 data-bbox="272 383 798 430">Form Input dengan Validasi</h2> <div data-bbox="272 459 577 544"> <p>Nama: <input type="text"/></p> <p>Email: <input type="text"/></p> <p><input type="button" value="Submit"/></p> </div> <p data-bbox="272 544 450 566">//menggunakan ajax</p> <h2 data-bbox="272 600 925 647">Form Input dengan Validasi (Ajax)</h2> <div data-bbox="272 676 577 761"> <p>Nama: <input type="text"/></p> <p>Email: <input type="text"/></p> <p><input type="button" value="Submit"/></p> </div>
10	<p data-bbox="268 792 1485 902">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)</p> <p data-bbox="268 904 359 936">Jawab:</p> <p data-bbox="268 940 359 972">Query:</p> <div data-bbox="268 981 1372 2033"> <pre> /menambahkan password &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;   &lt;title&gt;Form Input dengan Validasi (Ajax)&lt;/title&gt;   &lt;script src="https://code.jquery.com/jquery-3.7.1.min.js"&gt;&lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt;h1&gt;Form Input dengan Validasi (Ajax)&lt;/h1&gt;   &lt;form id="myForm" method="post"&gt;     &lt;label for="nama"&gt;Nama:&lt;/label&gt;     &lt;input type="text" id="nama" name="nama"&gt;     &lt;span id="nama-error" style="color: red;"&gt;&lt;/span&gt;&lt;br&gt;     &lt;label for="email"&gt;Email:&lt;/label&gt;     &lt;input type="text" id="email" name="email"&gt;     &lt;span id="email-error" style="color: red;"&gt;&lt;/span&gt;&lt;br&gt;     &lt;label for="password"&gt;Password:&lt;/label&gt;     &lt;input type="password" id="password" name="password"&gt;     &lt;span id="password-error" style="color: red;"&gt;&lt;/span&gt;&lt;br&gt;     &lt;input type="button" value="Submit" id="submit"&gt;   &lt;/form&gt;   &lt;div id="result"&gt;&lt;/div&gt;   &lt;script&gt;     \$(document).ready(function() {       \$("#submit").click(function() {         var nama = \$("#nama").val();         var email = \$("#email").val();         var password = \$("#password").val(); </pre> </div>

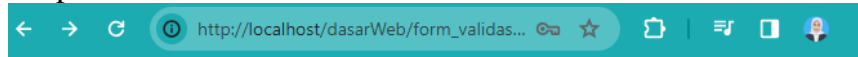
```

        $.ajax({
            type: "POST",
            url: "proses_validasi_ajax.php", // Ganti dengan URL file
            data: { nama: nama, email: email, password: password },
            success: function(response) {
                $("#result").html(response);
            }
        });
    });
</script>
</body>
</html>

<?php
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $nama = $_POST["nama"];
    $email = $_POST["email"];
    $password = $_POST["password"];
    $errors = array();
    // Validasi Nama
    if (empty($nama)) {
        $errors[] = "Nama harus diisi.";
    }
    // Validasi Email
    if (empty($email)) {
        $errors[] = "Email harus diisi.";
    } elseif (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        $errors[] = "Format email tidak valid.";
    }
    // Validasi Password
    if (empty($password)) {
        $errors[] = "Password harus diisi.";
    } elseif (strlen($password) < 8) {
        $errors[] = "Password minimal terdiri dari 8 karakter.";
    }
    // Jika ada kesalahan validasi
    if (!empty($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, Password = $password";
    }
}
?>

```

Output:



## Form Input dengan Validasi

Nama:

Email:



## Form Input dengan Validasi

Nama:

Email:

//menambahkan password

## Form Input dengan Validasi (Ajax)

Nama:

Email:

Password: