



**Nama : Jessica Amelia**  
**Kelas : SIB 2F**  
**Absen : 15**  
**NIM : 2341760185**  
**Topik**

- Konsep Form Processing 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><b>Jawab :</b> Penggunaan isset memastikan bahwa variable ada sebelum dicek atau diproses lebih lanjut dalam pernyataan kondisi. Dalam kode tersebut fungsi isset(\$umur) digunakan untuk memeriksa apakah variabel \$umur telah diinisialisasi atau didefinisikan, kode juga memeriksa apakah nilai \$umur lebih besar atau sama dengan 18 dengan (&amp;&amp; \$umur &gt;= 18), jika tidak memenuhi kedua kondisi tersebut maka akan menampilkan seperti dibawah ini.</p> <pre> Anda belum dewasa atau variabel 'umur' tidak ditemukan. </pre>
5	<p>Tambahkan isi dari file isset.php tersebut dengan kode di bawah ini.</p> <pre> \$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><b>Jawab :</b> Pada kode PHP tersebut, fungsi isset(\$data["nama"]) digunakan untuk memeriksa apakah elemen array dengan key "nama" ada di dalam array \$data. Jika elemen "nama" ada di dalam array \$data, maka echo "Nama: " . \$data["nama"]; akan menampilkan nilai dari key "nama", yaitu "Jane". Jika elemen "nama" tidak ada di dalam array, pesan "Variabel 'nama' tidak ditemukan dalam array." akan ditampilkan.</p> <pre> Anda belum dewasa atau variabel 'umur' tidak ditemukan. Nama: Jane </pre>

### 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> <p><b>Jawab :</b> Pada kode diatas, fungsi <code>empty(\$myArray)</code> digunakan untuk memeriksa apakah array <code>\$myArray</code> kosong atau belum terdefinisi. Jika <code>\$myArray</code> kosong atau tidak didefinisikan, maka pernyataan dalam blok <code>if</code> akan dijalankan, dalam hal ini menampilkan pesan "Array tidak terdefinisi atau kosong.". Jika array memiliki elemen, blok <code>else</code> akan dijalankan dan menampilkan "Array terdefinisi dan tidak kosong".</p> <pre><code>Array tidak terdefinisi atau kosong.</code></pre>
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	Simpan file tersebut, kemudian buka browser dan jalankan <code>localhost/dasarWeb/empty.php</code> buat hasil tidak menjadi 1 baris, jadi tampilan dari <code>echo</code> harus terpisah. Apa

	<p>yang anda pahami dari penggunaan empty pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 2.2)</p> <p><b>Jawab :</b> Pada kode diatas, fungsi empty(\$nonExistentVar) digunakan untuk memeriksa apakah variabel \$nonExistentVar kosong atau tidak terdefinisi. empty(\$nonExistentVar) mengembalikan true jika variabel belum didefinisikan atau memiliki nilai kosong. Pesan "Array tidak terdefinisi atau kosong." akan muncul jika \$nonExistentVar belum didefinisikan atau kosong. Pesan "Array terdefinisi dan tidak kosong." akan muncul hanya jika variabel didefinisikan dan memiliki nilai yang tidak kosong. Dan dari kode diatas menampilkan pesan "Array tidak terdefinisi atau kosong" karena \$nonExistentVar belum didefinisikan.</p> <p>Array tidak terdefinisi atau kosong. Array 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 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	Buat satu file baru bernama proses_form.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam proses_form.php

	<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><b>Jawab :</b> Form pada file digunakan untuk megumpulkan input dari pengguna, seperti nama dan email, yang kemudian akan dikirim ke server untuk di proses. Dalam form ini, metode pengiriman data adalah POST (ditentukan oleh atribut method="post"), yang berarti data yang diinput oleh pengguna akan dikirim ke server secara tersembunyi (tidak terlihat di URL). Atribut action="proses_form.php" berarti form akan mengirim data ke halaman tersebut.</p> <div> <div> <h3>Form Input PHP</h3> <p>Nama: <input type="text"/></p> <p>Email: <input type="text"/></p> <p><input type="submit" value="Submit"/></p> </div> <div> <h3>Form Input PHP</h3> <p>Nama: <input type="text" value="Jessica Amelia"/></p> <p>Email: <input type="text" value="ameliajessica997@gmail.co"/></p> <p><input type="submit" value="Submit"/></p> </div> <div> <p>Nama: Jessica Amelia Email: ameliajessica997@gmail.com</p> </div> </div>
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>

```

<!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>

```

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

**Jawab :** Form ini digunakan untuk mengumpulkan input dari pengguna, yaitu nama. Form ini menggunakan metode POST untuk mengirimkan data ke server, dan atribut action diarahkan ke halaman itu sendiri (<?php echo htmlspecialchars(\$\_SERVER["PHP\_SELF"]); ?>) untuk memproses data yang dikirimkan. Variabel \$namaErr digunakan untuk menyimpan pesan error jika input nama kosong, dan variabel \$nama digunakan untuk menyimpan nilai input nama jika valid. Metode \$\_SERVER["REQUEST\_METHOD"] untuk mengetahui apakah formulir telah disubmit atau belum. Jika formulir telah disubmit, maka PHP akan menjalankan kode di dalam blok if. Dalam blok if, PHP melakukan validasi input nama menggunakan fungsi empty(). Jika input nama kosong, maka PHP akan mengisi variabel \$namaErr dengan pesan error. Jika input nama tidak kosong, maka PHP akan mengisi variabel \$nama dengan nilai input nama dan menampilkan pesan "Data berhasil disimpan!".

<p><b>Form Input PHP</b></p> <p>Nama: <input type="text"/></p> <p><input type="submit" value="Submit"/></p>	<p><b>Form Input PHP</b></p> <p>Data berhasil disimpan!</p> <p>Nama: <input type="text" value="Jessica Amelia"/></p> <p><input type="submit" value="Submit"/></p>
<p><b>Form Input PHP</b></p> <p>Nama: <input type="text"/> Nama harus diisi!</p> <p><input type="submit" value="Submit"/></p>	

## HTML Injection

HTML injection (atau disebut juga "HTML injection attack" atau "client-side injection") adalah serangan keamanan yang terjadi ketika seorang penyerang memasukkan kode HTML atau JavaScript yang berbahaya ke dalam input yang diterima oleh sebuah aplikasi web. Kode berbahaya tersebut

akan dieksekusi oleh browser pengguna yang melihat halaman web yang terpengaruh, yang dapat mengakibatkan akses ilegal ke data, perubahan tampilan halaman, atau serangan lainnya.

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

### Contoh HTML Injection:

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

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

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

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

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

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

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

### Cara Mengangani HTML Injection

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

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

5. **Escape Output:** Ketika Anda menampilkan data di halaman web, pastikan Anda menghindari injeksi JavaScript dengan menggunakan `htmlspecialchars()` atau metode serupa.

Berikut contoh sederhana dalam PHP untuk mengatasi HTML injection:

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

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

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

#### Praktikum 4 : HTML Injection

Langkah-langkah Praktikum :

Langkah	Keterangan
1	Buat satu file baru bernama <code>html_aman.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>html_aman.php</code>
2	<pre>\$input = \$_POST['input']; \$input = htmlspecialchars(\$input, ENT_QUOTES, 'UTF-8');</pre>
3	Tambahkan script 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><b>Jawab :</b> Penambahan kode <code>htmlspecialchars()</code> pada PHP di atas berfungsi untuk mengamankan data yang diterima dari formulir (form) HTML melalui metode POST. Fungsi ini mengubah karakter-karakter khusus dalam string menjadi entitas HTML, sehingga dapat mencegah serangan XSS (Cross-Site Scripting). Misalnya, jika pengguna mencoba memasukkan kode JavaScript atau HTML ke dalam formulir, kode tersebut tidak akan dieksekusi dan akan ditampilkan sebagai teks biasa.</p> <div> <div> <h3>Input HTML Aman</h3> <p>Nama: <input type="text"/> <input type="submit" value="Submit"/></p> </div> <div> <h3>Input HTML Aman</h3> <p>Nama: <input type="text"/> <input type="submit" value="Submit"/></p> <p>Nama harus diisi!</p> </div> </div> <p>Nama berhasil disimpan: Jessica Amelia</p>
5	Ketikkan kode tambahan pada langkah 6 di dalam <code>html_aman.php</code>



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	<p>Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/html_aman.php</p>
8	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.2)</p> <p><b>Jawab :</b> Dengan penambahan kode digunakan untuk memvalidasi input email dari pengguna. Variabel \$email menyimpan data dari input form yang dikirim menggunakan metode POST dengan key 'email'. Fungsi filter_var() digunakan untuk memvalidasi apakah email yang dimasukkan sesuai format email yang valid. Fungsi ini menggunakan filter FILTER_VALIDATE_EMAIL. Jika email valid, maka program akan melanjutkan dengan pemrosesan email yang aman. Jika email tidak valid, program akan menangani input yang tidak valid. Kode ini bertujuan untuk memastikan bahwa email yang dimasukkan oleh pengguna memiliki format yang benar sebelum diproses lebih lanjut.</p> <hr/> <p><b>Input HTML Aman</b></p> <p>Nama: <input type="text"/></p> <p>Email: <input type="text"/></p> <p><input type="button" value="Submit"/></p> <p><b>Input HTML Aman</b></p> <p>Nama: <input type="text" value="Jessica Amelia"/></p> <p>Email: <input type="text" value="ameliajessica997@gmail.cc"/></p> <p><input type="button" value="Submit"/></p> <p>Nama dan email berhasil diproses: Nama = Jessica Amelia, Email = ameliajessica997@gmail.com</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!";
}

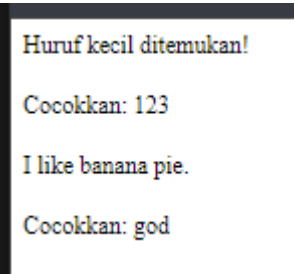
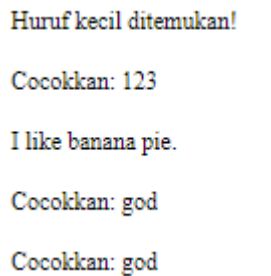
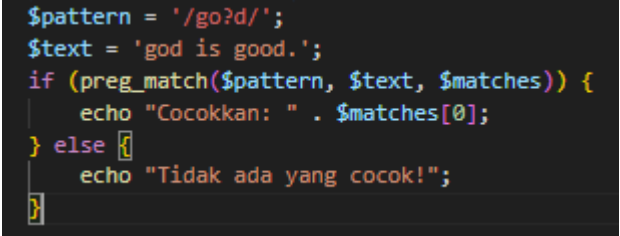
```

## Praktikum 5 : Penggunaan Regex Pada PHP

Langkah-langkah Praktikum :

Langkah	Keterangan
1	Buat satu file baru bernama 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)</p> <p><b>Jawab :</b> Kode di atas adalah contoh penggunaan fungsi preg_match() di PHP untuk mencocokkan pola dengan menggunakan regular expression (regex). (\$pattern): /[a-z]/ digunakan untuk mencari huruf kecil dalam teks. Rentang [a-z] merepresentasikan semua huruf kecil dari "a" hingga "z". String yang dicek adalah "This is a Sample Text.". Fungsi preg_match(\$pattern, \$text) akan memeriksa apakah string tersebut mengandung huruf kecil. Jika ditemukan huruf kecil, maka akan menampilkan output "Huruf kecil ditemukan!". Jika tidak ada huruf kecil, akan menampilkan "Tidak ada huruf kecil!". Dan hasilnya dari trks yang dicek huruf kecil ditemukan.</p> <pre> Huruf kecil ditemukan! </pre>
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)</p> <p><b>Jawab :</b> Kode di atas menggunakan preg_match untuk mencari digit angka dalam sebuah string. Pola yang digunakan adalah '/[0-9]+/' untuk mencocokkan satu atau lebih digit angka dalam string. String yang dicek adalah 'There are 123 apples.'. Fungsi preg_match mencari angka dalam teks tersebut. Jika ditemukan, hasilnya disimpan di array \$matches, dan elemen pertama (\$matches[0]) yang cocok akan ditampilkan, dalam hal ini angka 123. Jika tidak ada angka yang ditemukan, akan menampilkan "Tidak ada yang cocok!". Dan dari teks tersebut ditemukan angka 123.</p> <pre> Huruf kecil ditemukan! Cocokkan: 123 </pre>
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)</p> <p><b>Jawab :</b> Kode di atas menggunakan fungsi preg_replace untuk mengganti kata dalam sebuah string. Pola yang digunakan adalah '/apple/', yang mencari kata "apple" dalam string. Kata "apple" akan diganti dengan kata "banana" yang didefinisikan dalam variabel \$replacement. Teks awalnya adalah 'I like apple pie.'. Fungsi preg_replace mengganti kata "apple" menjadi "banana", sehingga teks hasilnya menjadi 'I like banana pie.'.</p> <pre> Huruf kecil ditemukan! Cocokkan: 123 I like banana pie. </pre>
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)</p> <p><b>Jawab :</b> Kode di atas menggunakan preg_match dengan pola regex '/go*d/' untuk mencocokkan teks yang mengandung pola khusus. Pola '/go*d/' berarti: g diikuti oleh nol atau lebih huruf o (diwakili oleh *), dan diakhiri dengan huruf d. Contoh kata yang cocok: "gd", "god", "good", "goood", dll. String yang diperiksa adalah 'god is good.'. Fungsi preg_match akan menemukan kecocokan pertama, yaitu kata "god" yang sesuai dengan pola, dan mencetak "Cocokkan: god". Jika tidak ada kecocokan ditemukan, maka akan mencetak "Tidak ada yang cocok!".</p> 
17	<p>Soal 5.5: Buatlah script langkah 14 dengan variable pattern yang di ubah menggunakan `?` (0 atau 1 kali). Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.5)</p> <p><b>Jawab:</b> Pola '/go?d/' berarti g diikuti oleh nol atau satu huruf o (diwakili oleh ?), lalu diakhiri dengan huruf d. Contoh kata yang cocok: "gd" dan "god", karena huruf o bisa muncul sekali atau tidak sama sekali. String yang diperiksa adalah 'god is good.'. Fungsi preg_match akan menemukan kecocokan pertama, yaitu kata "god", karena sesuai dengan pola, dan mencetak "Cocokkan: god". Jika tidak ada kecocokan ditemukan, maka akan mencetak "Tidak ada yang cocok!".</p>  
18	<p>Soal 5.6: Buatlah script langkah 14 dengan variable pattern yang di ubah menggunakan `{n,m}`. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.6)</p>

**Jawab:** Pola `/go{2,4}d/` berarti g diikuti oleh 2 hingga 4 huruf o (diwakili oleh `{2,4}`), dan diakhiri dengan huruf d. Contoh kata yang cocok: "good", "goood", "gooodd" (kata dengan 2, 3, atau 4 huruf o). String yang diperiksa adalah 'god is good. gooodd.'. Ini berisi beberapa variasi kata yang dapat dicocokkan. Fungsi `preg_match` akan menemukan kecocokan pertama, yaitu "good", karena memiliki 2 huruf o yang sesuai dengan pola. Hasilnya mencetak "Cocokkan: good". Jika tidak ada kecocokan, akan mencetak "Tidak ada yang cocok!".

Huruf kecil ditemukan!

Cocokkan: 123

I like banana pie.

Cocokkan: god

Cocokkan: good

```
$pattern = '/go{2,4}d/'; // Mencocokkan 'g'
$text = 'god is good. gooodd.'; // Menambahk
if (preg_match($pattern, $text, $matches)){
    echo "Cocokkan: " . $matches[0];
} else {
    echo "Tidak ada yang cocok!";
}
```

## 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)</p> <p><b>Jawab :</b> Kode ini membuat formulir di halaman web yang memungkinkan pengguna memilih buah, warna favorit, dan jenis kelamin. Setelah pengguna mengisi formulir dan mengklik tombol "Submit", data yang dipilih akan dikirim ke server menggunakan metode POST. Di bagian PHP, kode memeriksa apakah data telah dikirim. Kemudian, ia mengambil pilihan yang dibuat oleh pengguna, seperti buah yang dipilih, warna favorit (bisa lebih dari satu), dan jenis kelamin. Hasilnya ditampilkan kembali di layar, menunjukkan apa yang telah dipilih oleh</p>

	<p>pengguna. Ini adalah cara yang baik untuk membuat formulir yang interaktif dan mudah digunakan.</p> <div> <div> <h3>Form Contoh</h3> <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> </div> <div> <h3>Form Contoh</h3> <p>Pilih Buah: <input type="text" value="Apel"/></p> <p>Pilih Warna Favorit:</p> <p><input checked="" 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 checked="" type="radio"/> Perempuan</p> <p><input type="button" value="Submit"/></p> </div> <div> <hr/> <p>Anda memilih buah: apel</p> <p>Warna favorit Anda: merah</p> <p>Jenis kelamin Anda: perempuan</p> </div> </div>
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 :** Kode HTML ini membuat form interaktif dengan dropdown, checkbox, dan radio button. Form ini menggunakan jQuery untuk mengirimkan data secara asinkron melalui AJAX

ke proses\_lanjut.php, memungkinkan hasilnya ditampilkan di div #hasil tanpa perlu memuat ulang halaman.

### Form Contoh

Pilih Buah:

Pilih Warna Favorit:

☐ Merah

☒ Biru

☐ Hijau

Pilih Jenis Kelamin:

☐ Laki-laki

☒ Perempuan

Anda memilih buah: apel

Warna favorit Anda: biru

Jenis kelamin Anda: perempuan

## Praktikum 7 : Validasi Form

Langkah-langkah Praktikum :

Langkah	Keterangan
1	Buat satu file baru bernama form_validasi.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam form_validasi.php
2	<pre> &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)</p> <p><b>Jawab :</b> Kode diatas digunakan untuk memvalidasi nama dan email yang diinputkan, jika nama atau email tidak diisi dan di klik submit maka akan ke halaman lain dengan pesan nama harus diisi dan email harus diisi, jika sudah diisi dengan benar maka akan divalidasi dengan menampilkan data berhasil disimpan dengan nama dan email yang telah diinputkan.</p> <div> <h3>Form Input dengan Validasi</h3> <p>Nama: <input type="text"/></p> <p>Email: <input type="text"/></p> <p><input type="submit" value="Submit"/></p> </div> <div> <h3>Form Input dengan Validasi</h3> <p>Nama: <input type="text" value="Jessica Amelia"/></p> <p>Email: <input type="text" value="ameliajessica997@gmail.com"/></p> <p><input type="submit" value="Submit"/></p> </div> <div> <p>Data berhasil dikirim: Nama = Jessica Amelia. Email = ameliajessica997@gmail.com</p> </div> <div> <p>Nama harus diisi.</p> <p>Email harus diisi.</p> </div> <div> <h3>Form Input dengan Validasi</h3> <p>Nama: <input type="text" value="Jessie"/></p> <p>Email: <input type="text" value="123jes"/></p> <p><input type="submit" value="Submit"/></p> </div>

	<p>Format email tidak valid.</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)</p> <p><b>Jawab :</b> Kode diatas digunakan untuk memvalidasi nama dan email yang diinputkan, jika nama atau email tidak diisi maka akan muncul pesan diamping kolom inputan berwarna merah, jika sudah diisi dengan benar maka akan divalidasi dengan menampilkan data berhasil disimpan dengan nama dan email yang telah diinputkan.</p> <p><b>Form Input dengan Validasi</b></p> <p>Nama: <input type="text" value="Jessica Amelia"/></p> <p>Email: <input type="text" value="ameliajessica997@gmail.cc"/></p> <p><input type="button" value="Submit"/></p> <p>Data berhasil dikirim: Nama = Jessica Amelia. Email = ameliajessica997@gmail.com</p>

## Form Input dengan Validasi

Nama:

Email:  Email harus diisi.

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 :** Dengan menggunakan ajax halaman tidak akan reload saat form dikirimkan, pengguna tetap berada di halaman yang sama dan hanya melihat pesan keberhasilan/kegagalan tanpa refresh. Jika pengiriman berhasil (success callback), akan muncul alert yang menunjukkan respons dari server dan form akan direset. Jika ada kesalahan (error callback), pesan kesalahan akan muncul dalam bentuk alert.

## Form Input dengan Validasi

Nama:

Email:

9

## Form Input

Nama:

Email:

localhost says

Form berhasil dikirim: Data berhasil dikirim: Nama = Jessica Amelia.  
Email = ameliajessica997@gmail.com

OK

## Form Input

Nama:

Email:

localhost says

Form berhasil dikirim: Format email tidak valid.<br>

OK

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Form Input dengan Validasi</title>
5      <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6  </head>
7  <body>
8      <h1>Form Input dengan Validasi</h1>
9      <form id="myForm">
10         <label for="nama">Nama:</label>
11         <input type="text" id="nama" name="nama">
12         <span id="nama-error" style="color: red;"></span><br>
13
14         <label for="email">Email:</label>
15         <input type="text" id="email" name="email">
16         <span id="email-error" style="color: red;"></span><br>
17
18         <input type="submit" value="Submit">
19     </form>
20
21     <script>
22     $(document).ready(function() {
23         $("#myForm").submit(function(event) {
24             event.preventDefault(); // Mencegah pengiriman form secara tradisional
25
26             var nama = $("#nama").val();
27             var email = $("#email").val();
28             var valid = true;
29
30             // Reset pesan error
31             $("#nama-error").text("");
32             $("#email-error").text("");
33
34             // Validasi form
35             if (nama === "") {
36                 $("#nama-error").text("Nama harus diisi.");
37                 valid = false;
38             }
39
40             if (email === "") {
41                 $("#email-error").text("Email harus diisi.");
42                 valid = false;
43             }
44
45             if (valid) {
46                 // Kirim data menggunakan AJAX
47                 $.ajax({
48                     url: 'proses_validasi.php',
49                     type: 'POST',
50                     data: { nama: nama, email: email },
51                     success: function(response) {
52                         alert("Form berhasil dikirim: " + response);
53                         // Misalnya kita bisa reset form setelah sukses
54                         $("#myForm")[0].reset();
55                     },
56                     error: function(xhr, status, error) {
57                         alert("Terjadi kesalahan: " + error);
58                     }
59                 });
60             }
61         });
62     });
63     </script>
64 </body>
65 </html>
66

```

10

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

**Jawab :** Saat memasukkan password kurang dari 8 karakter maka akan muncul pesan disamping kotak inputan password "Password harus minimal 8 karakter", Ketika password sudah memenuhi syarat itu maka data berhasil disimpan.

## Form Input dengan Validasi

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

localhost says

Form berhasil dikirim: Data berhasil diproses.

OK

Nama:   
Email:   
Password:

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

    $errors = [];

    // 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 (strlen($password) < 8) {
        $errors[] = "Password harus minimal 8 karakter.";
    }

    // Jika tidak ada error
    if (empty($errors)) {
        // Proses data (misalnya, simpan ke database)
        echo "Data berhasil diproses.";
    } else {
        echo implode(", ", $errors);
    }
}
?>
```

```

1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Form Input dengan Validasi</title>
5      <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6  </head>
7  <body>
8      <h1>Form Input dengan Validasi</h1>
9      <form id="myForm">
10         <label for="nama">Nama:</label>
11         <input type="text" id="nama" name="nama">
12         <span id="nama-error" style="color: red;"></span><br>
13
14         <label for="email">Email:</label>
15         <input type="text" id="email" name="email">
16         <span id="email-error" style="color: red;"></span><br>
17
18         <label for="password">Password:</label>
19         <input type="password" id="password" name="password">
20         <span id="password-error" style="color: red;"></span><br>
21
22         <input type="submit" value="Submit">
23     </form>
24
25     <script>
26     $(document).ready(function() {
27         $("#myForm").submit(function(event) {
28             event.preventDefault(); // Mencegah pengiriman form secara tradisional
29
30             var nama = $("#nama").val();
31             var email = $("#email").val();
32             var password = $("#password").val();
33             var valid = true;
34
35             // Reset pesan error
36             $("#nama-error").text("");
37             $("#email-error").text("");
38             $("#password-error").text("");
39
40             // Validasi form
41             if (nama === "") {
42                 $("#nama-error").text("Nama harus diisi.");
43                 valid = false;
44             }
45
46             if (email === "") {
47                 $("#email-error").text("Email harus diisi.");
48                 valid = false;
49             }
50
51             // Validasi password
52             if (password.length < 8) {
53                 $("#password-error").text("Password harus minimal 8 karakter.");
54                 valid = false;
55             }
56
57             if (valid) {
58                 // Kirim data menggunakan AJAX
59                 $.ajax({
60                     url: 'proses_validasi.php',
61                     type: 'POST',
62                     data: { nama: nama, email: email, password: password },
63                     success: function(response) {
64                         alert("Form berhasil dikirim: " + response);
65                         // Misalnya kita bisa reset form setelah sukses
66                         $("#myForm")[0].reset();
67                     },
68                     error: function(xhr, status, error) {
69                         alert("Terjadi kesalahan: " + error);
70                     }
71                 });
72             }
73         });
74     });
75     </script>
76 </body>
77 </html>
78

```