

**LAPORAN PRAKTIKUM  
DESPRO WEB  
PERTEMUAN 7 PHP JOBSHEET 7**



**RAFFI ILHAM MAULANA  
2241720226  
1F / 24**



## 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

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

## 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

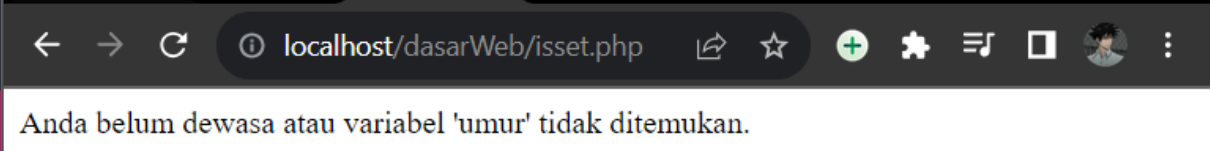
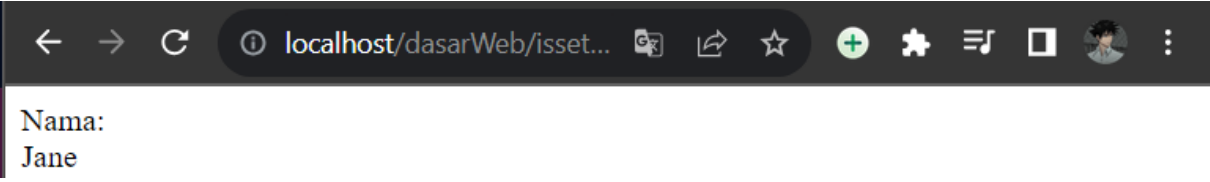
### Function isset

**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></p> <p>isset(\$umur): Ini menguji apakah variabel \$umur sudah didefinisikan atau belum. Jika variabel \$umur sudah didefinisikan, maka ekspresi ini akan mengembalikan true. Jika tidak, maka ekspresi ini akan mengembalikan false.</p> 
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></p> <p>untuk memeriksa keberadaan kunci "nama" dalam array \$data dan mencetak nilai yang sesuai jika kunci tersebut ada. Jika kunci "nama" tidak ditemukan, maka akan mencetak pesan kesalahan. Dalam contoh ini, karena kunci "nama" ada dalam array, maka hasilnya akan mencetak "Nama: Jane".</p> 

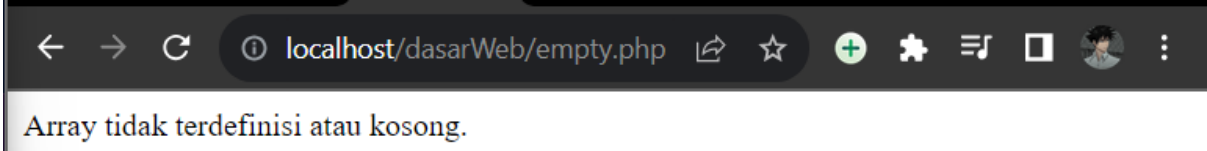
### Function empty

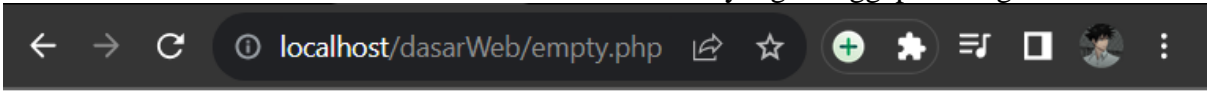
Fungsi **empty()** pada PHP digunakan untuk memeriksa apakah suatu variabel kosong atau tidak terdefinisi. Fungsi ini mengembalikan **true** jika variabel kosong atau tidak terdefinisi, dan **false** jika variabel memiliki nilai atau telah didefinisikan. Berikut adalah penjelasan dan contoh penggunaan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

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

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

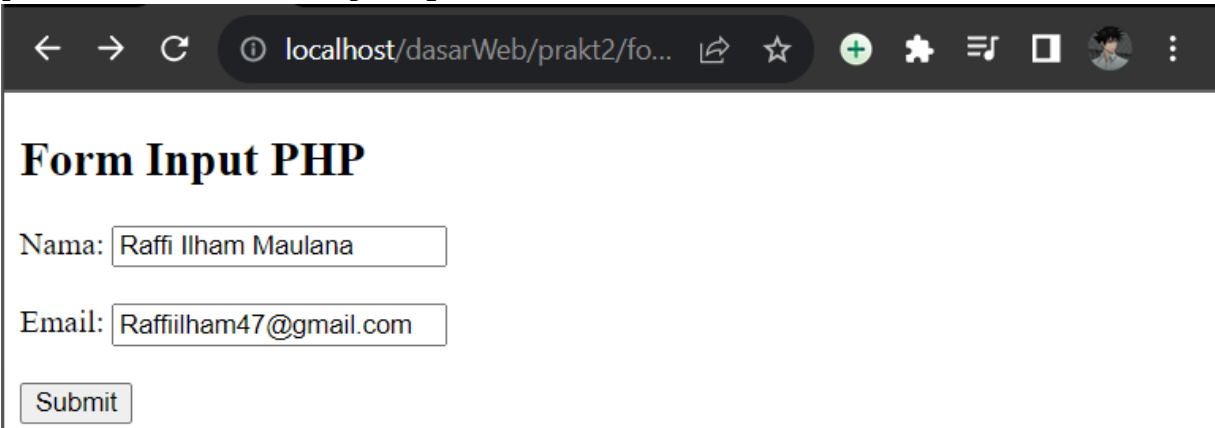
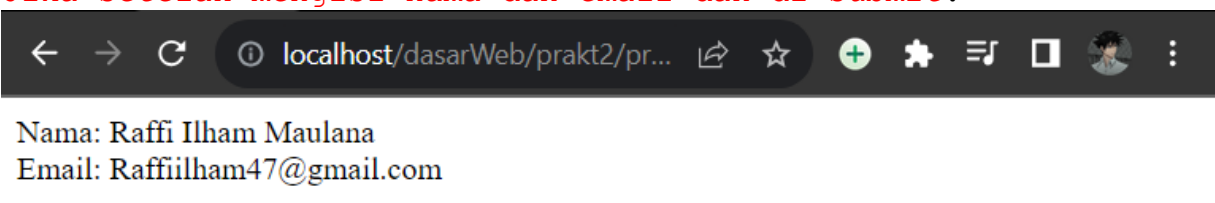
## Praktikum 2. Function empty

Langkah	Keterangan
1	Buat satu file baru bernama <code>empty.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>empty.php</code>
2	<pre><code>\$myArray = array(); // Array kosong if (empty(\$myArray)) {     echo "Array tidak terdefinisi atau kosong."; } else {     echo "Array terdefinisi dan tidak kosong."; }</code></pre>
3	<p>Simpan file tersebut, kemudian buka browser dan jalankan <code>localhost/dasarWeb/empty.php</code>. Apa yang anda pahami dari penggunaan <code>empty</code> pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 2.1)</p> <p><b>Jawab:</b></p> <p>sebenarnya konsepnya sama saja dengan "isset" namun "empty" ini adalah kebalikannya, jadi kode ini adalah untuk menginformasikan apakah array <code>\$myArray</code> tidak didefinisikan atau kosong. Jika kosong atau tidak didefinisikan, maka pesan "Array tidak terdefinisi atau kosong." akan dicetak. Jika tidak kosong dan sudah didefinisikan, maka pesan "Array terdefinisi dan tidak kosong." akan dicetak. Dalam contoh ini, karena array <code>\$myArray</code> awalnya didefinisikan tetapi kosong, pesan pertama akan dicetak.</p> 
4	<p>Tambahkan isi dari file <code>empty.php</code> tersebut dengan kode di bawah ini.</p> <pre><code>if (empty(\$nonExistentVar)) {     echo "Variabel tidak terdefinisi atau kosong."; } else {     echo "Variabel terdefinisi dan tidak kosong."; }</code></pre>

5	<p>Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/empty.php buat hasil tidak menjadi 1 baris, jadi tampilan dari echo harus terpisah. Apa yang anda pahami dari penggunaan empty pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 2.2)</p> <p><b>Jawab:</b></p> <p>Jadi, akibat dari penggunaan empty(\$nonExistentVar) adalah bahwa pesan "Variabel tidak terdefinisi atau kosong." akan dicetak ke layar untuk mengindikasikan bahwa variabel \$nonExistentVar tidak didefinisikan atau memiliki nilai yang dianggap kosong.</p>  <p>Array tidak terdefinisi atau kosong. Variabel tidak terdefinisi atau kosong.</p>
---	---

### Praktikum Bagian 3 : Form Input PHP

Langkah	Keterangan
1	Buat satu file baru bernama form.php di dalam folder dasarWeb. Ketikkan kode 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></p> <p>Form ini akan mengirimkan data ke URL proses_form.php ketika pengguna menekan tombol "Submit". Ini berarti bahwa semua data yang dimasukkan oleh pengguna akan dikirimkan ke skrip PHP proses_form.php untuk diproses lebih lanjut, yang akan dicetak pada halaman selanjutnya</p>  <p>Jika setelah mengisi nama dan email dan di submit:</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>

```

<!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 HTML ini memiliki atribut method="post" yang mengindikasikan bahwa data akan dikirimkan menggunakan metode POST saat formulir dikirimkan. Atribut action menunjukkan bahwa data akan diproses oleh halaman yang sama (htmlspecialchars(\$\_SERVER["PHP\_SELF"])).

Jadi,

Kode ini membantu mengumpulkan input dari pengguna, melakukan validasi sederhana, dan memberikan pesan umpan balik sesuai dengan hasil validasi.

**Tampilan awal:**

**Jika nama diisi**

Data berhasil disimpan!

Nama:

**Jika nama tidak diisi:**

Nama: <input type="text"/>	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);

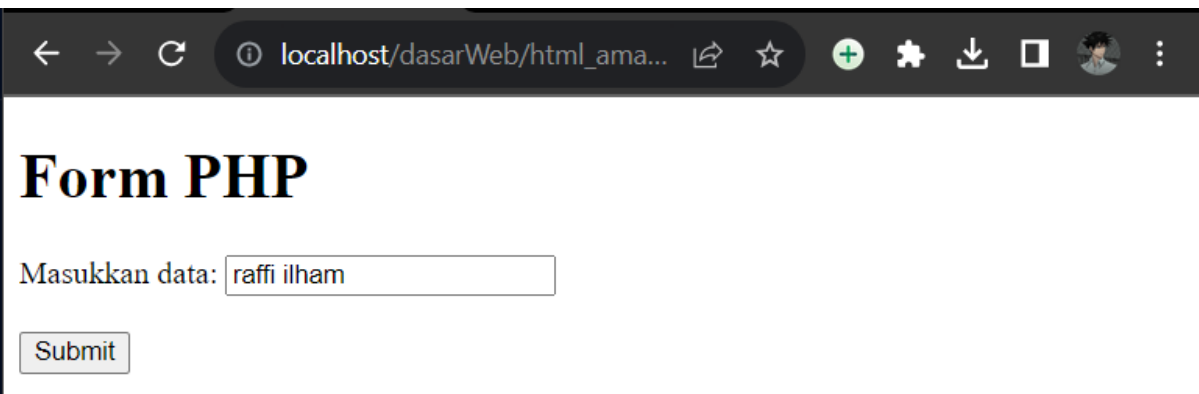
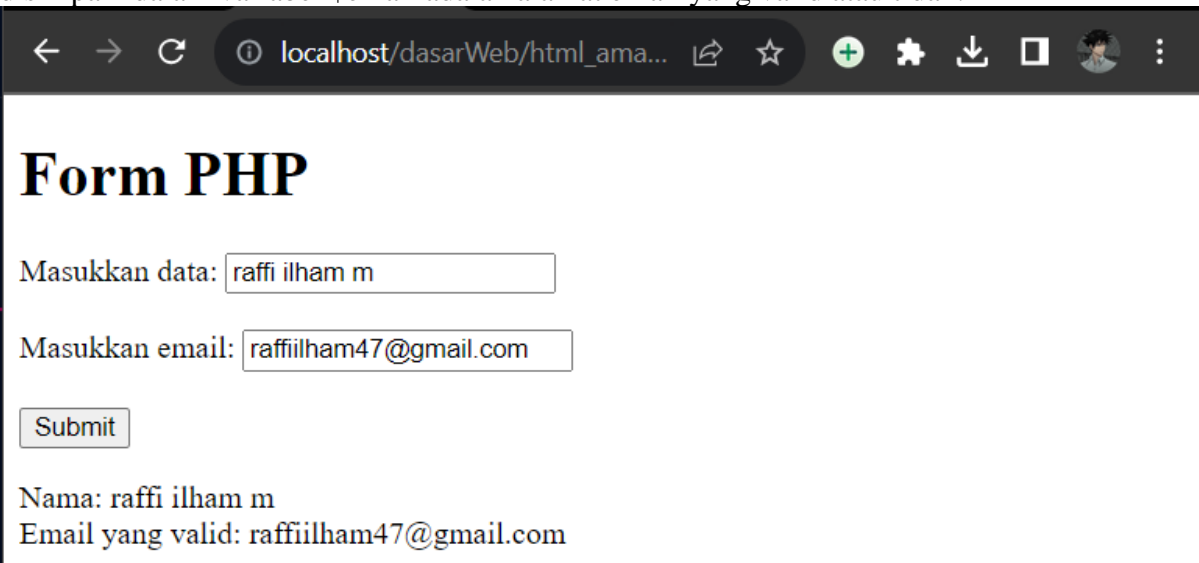
// Menggunakan htmlspecialchars untuk menghindari HTML injection saat menampilkan data
echo '<div>' . htmlspecialchars($user_input, ENT_QUOTES, 'UTF-8') . '</div>';
```

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	Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.1) Jawab: Kode program tersebut akan memproses input dengan menggunakan <code>htmlspecialchars()</code> untuk mengamankan input dari potensi XSS. Hasilnya akan ditampilkan di halaman web sebagai teks yang aman dari serangan XSS.

	
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	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>Jawab:</p> <p>Kode program tersebut digunakan untuk validasi alamat email dalam bahasa PHP. Dalam kode tersebut, fungsi <code>filter_var()</code> digunakan untuk memeriksa apakah suatu string yang disimpan dalam variabel <code>\$email</code> adalah alamat email yang valid atau tidak.</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

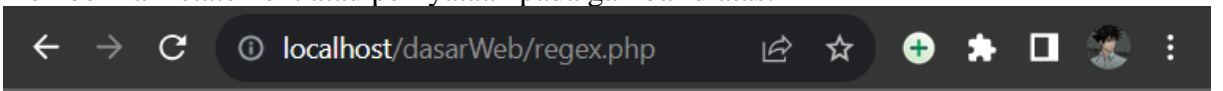
```
$pattern = '/hello/';
$text = 'Hello, world!';
if (preg_match($pattern, $text)) {
    echo "Cocokkan ditemukan!";
} else {
    echo "Tidak ada cocokan!";
}
```

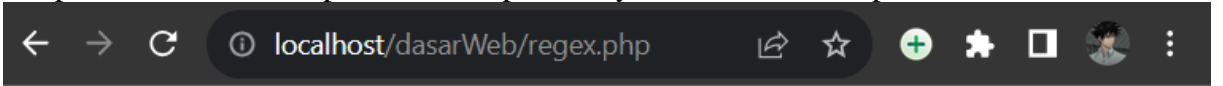
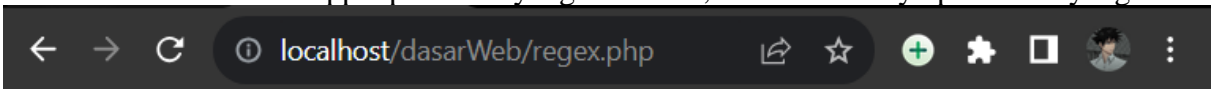
2. Mencocokkan karakter set

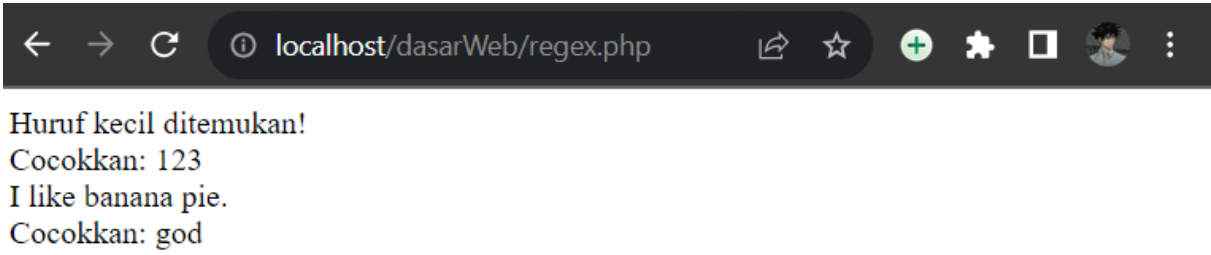
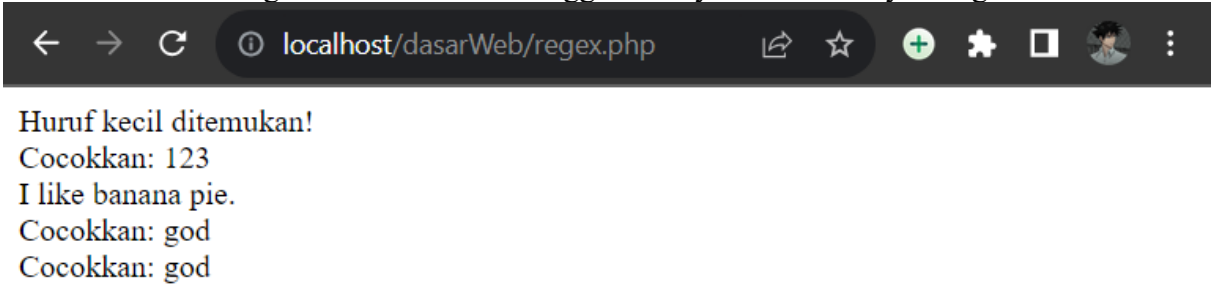
```
$pattern = '/[0-9]/'; // Cocokkan satu digit.
$text = 'There are 3 apples.';
if (preg_match($pattern, $text)) {
    echo "Digit ditemukan!";
} else {
    echo "Tidak ada digit!";
}
```

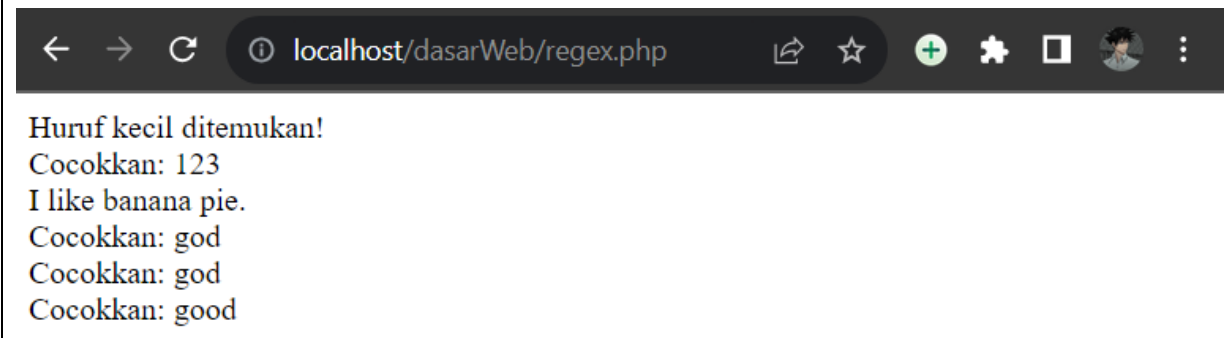
## Praktikum 5 : Penggunaan Regex Pada PHP

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><code>\$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!"; }</code></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></p> <p>Dari hasil diatas kita membandingkan sebuah sample teks dengan pembanding nya yaitu 'a-z' huruf kecil dari a sampai z. Lalu, kita bandingkan dengan menggunakan if else apakah pada sample teks nya tersebut memiliki huruf kecil atau tidak. Dan hasil yang didapatkan adalah benar atau bernilai true bahwa pada sample teks nya memiliki huruf kecil dan memberikan statement atau pernyataan pada gambar diatas.</p>  <p>Huruf kecil ditemukan!</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)</p> <p><b>Jawab:</b></p> <p>Dari hasil diatas kita juga membandingkan digit angka yang kita miliki pada variabel pattern dan sebuah sampel kalimat pada variabel text. Ketika pada perbandingan antara sampel kalimat tersebut pada variabel pattern nya maka akan ditampilkan.</p>  <p>Huruf kecil ditemukan! Cocokkan: 123</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)</p> <p><b>Jawab:</b></p> <p>Dari hasil yang terlihat disini kita membandingkan sebuah kata dengan kalimat yang ada dan setelah diketahui memiliki kesamaan atau sama nilainya maka akan ditimpa kata tersebut dengan kata yang baru. Pada variabel pattern kita menaruh isinya adalah apple. Disini kita akan mencari nilai apple pada text yang ada. Lalu, kita akan menyiapkan kata yang akan</p>  <p>Huruf kecil ditemukan! Cocokkan: 123 I like banana pie.</p>
13	Ketikkan kode tambahan pada langkah 14 di dalam regex.php

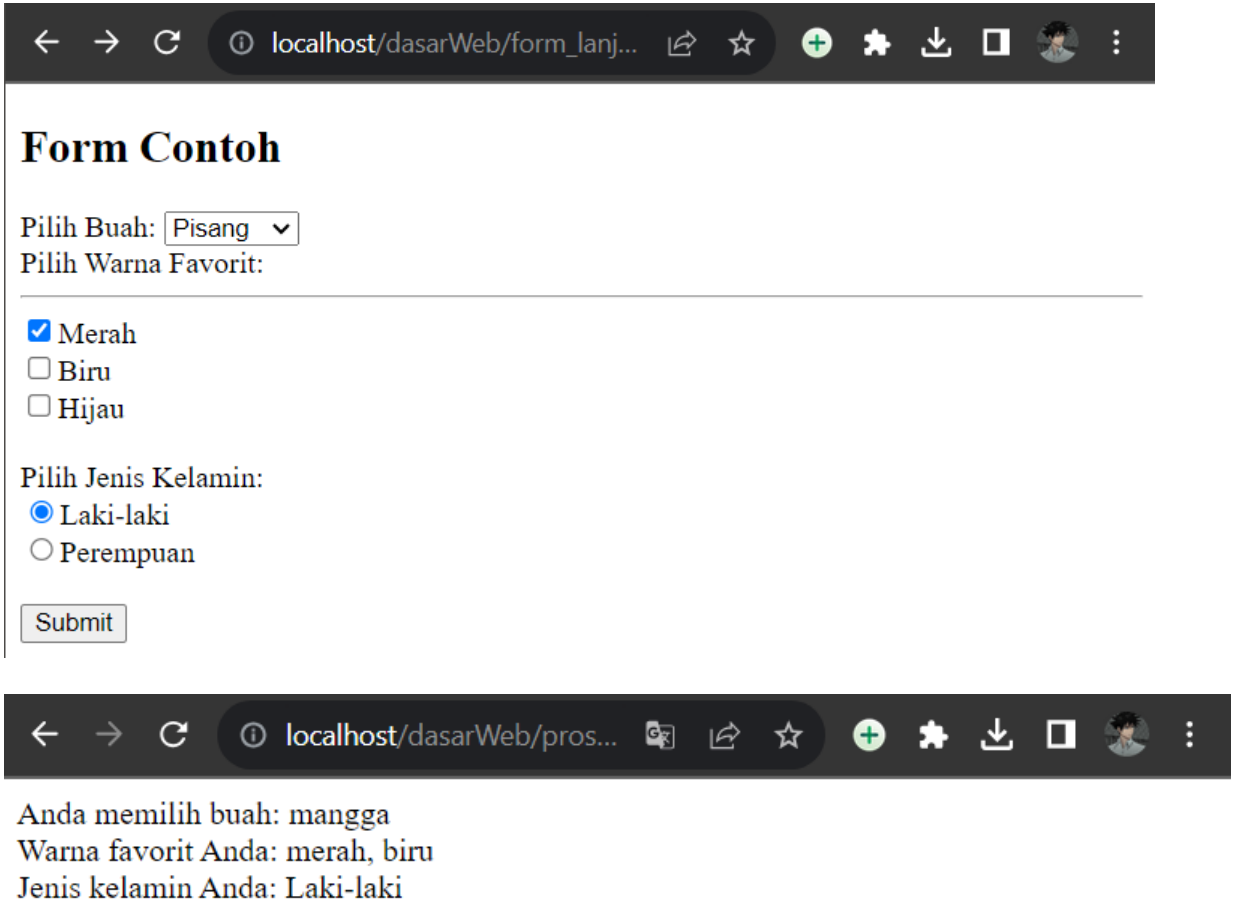
14	<pre> \$pattern = '/go*d/'; // Cocokkan "god", "good", "goood", dll. \$text = 'god is good.'; if (preg_match(\$pattern, \$text, \$matches)) {     echo "Cocokkan: " . \$matches[0]; } else {     echo "Tidak ada yang cocok!"; } </pre>
15	<p>Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php</p>
16	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.4)  Jawab:  Dari hasil yang didapatkan kita mendapatkan kata yang cocok adalah 'god'. Sedangkan pada sampel teks nya adalah "god is good". Lalu, pada pattern nya adalah "go*d". Disini kenapa hasilnya adalah god bukan good adalah pada variabel matches nya. Disini kita mentarget kata pertama yang cocok. Sedangkan kata pertama yang cocok adalah god. Maka itu akan menjadi hasilnya.</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:  Dari hasil diatas tetap sama seperti sebelumnya. Dikarenakan pada pola /go?d/ akan mencocokkan dengan teks yang memiliki karakter "g", diikuti oleh nol atau satu karakter "o" dan diakhiri dengan karakter "d". Sehingga hasilnya akan cocok yaitu "god".</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)  Jawab:  Dari hasil kode program diatas kita mendapatkan hasil "good". Hal tersebut bisa terjadi dikarenakan pada pattern kita menggunakan {2,3}. Jadi dimulai dengan karakter "g" dan diikuti oleh dua atau tiga karakter "o" dan diakhiri oleh "d". Dan hasil terakhirnya adalah "good" seperti pada gambar diatas.</p>

	
--	--

**Praktikum 6 : Form Lanjut**

Langkah-langkah Praktikum :

Langka h	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; </pre>

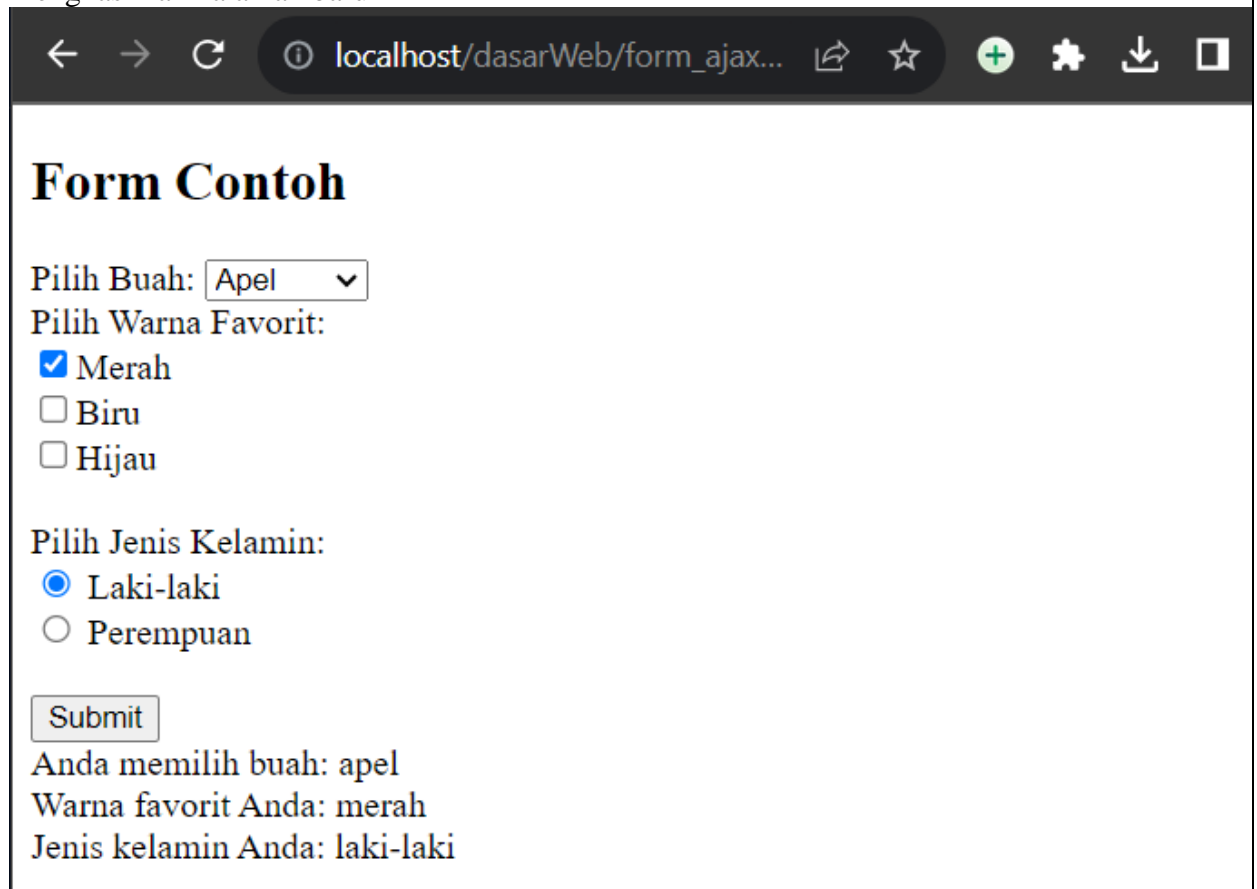
	<pre> if (\$_SERVER["REQUEST_METHOD"] == "POST") {     \$selectedBuah = \$_POST['buah'];      if (isset(\$_POST['warna'])) {         \$selectedWarna = \$_POST['warna'];     } else {         \$selectedWarna = [];     }      \$selectedJenisKelamin = \$_POST['jenis_kelamin'];      echo "Anda memilih buah: " . \$selectedBuah . "&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>Jawab:</p> <p>Akan tampil sebuah form ketika kita mengisi form tersebut dan mengirimnya maka akan menampilkan kembali sesuai dengan apa yang kita pilih di awal tadi, dan menghasilkannya pada halaman selanjutnya.</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> <pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;   &lt;title&gt;Contoh Form dengan PHP dan jQuery&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;h2&gt;Form Contoh&lt;/h2&gt;   &lt;form id="myForm"&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;div id="hasil"&gt;     &lt;!-- Hasil akan ditampilkan di sini --&gt;   &lt;/div&gt;    &lt;script&gt;     \$(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);           }         });       });     });   &lt;/script&gt; &lt;/body&gt; &lt;/html&gt; </pre>
6	<p>Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/form_ajax.php</code>. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 6.2)</p>

Jawab:

Akan tampil sebuah form ketika user mengisi form tersebut dan mengirimnya maka akan menampilkan kembali sesuai dengan apa yang kita pilih di awal tadi dibawahnya, tanpa menghasilkan halaman baru



**Form Contoh**

Pilih Buah:

Pilih Warna Favorit:

☒ Merah

☐ Biru

☐ Hijau

Pilih Jenis Kelamin:

☒ Laki-laki

☐ Perempuan

Anda memilih buah: apel

Warna favorit Anda: merah


Jenis kelamin Anda: laki-laki



## Praktikum 7 : Validasi Form

Langkah-langkah Praktikum :

Langka h	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>

	<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>
2	<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:  Kode program diatas akan meghasilkan keluaran nama dan email yang dua duanya terdapat kotak pengisian yang dimana user akan di request oleh kode program untuk mengisi kedua kotak tersebut, setelah ketua kotak tersebut diisi, lalu klik submit maka akan diarahkan ke halaman selanjutnya yang berisi cetakan validasi bahwa data berhasil dikirim dan kedua data yang sudah di isi tadi di form pada halaman sebelumnya</p>

	
5	<p>Kembangkan file bernama <code>form_validasi.php</code> di dalam folder <code>dasarWeb</code>. Ketikkan kode pada langkah 6 di dalam <code>form_validasi.php</code></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 <code>localhost/dasarWeb/form_validasi.php</code></p>
8	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.2) Jawab:</p>

	<p>Form itu meminta user untuk memasukkan nama dan alamat email setelah user mengisi form dan mengirimkannya, kode program akan memproses yang dikirim melalui method post, kode tsb juga akan melakukan validasi memastikan nama dan email tidak boleh kosong, jika kosong kode tsb akan muncul output "nama dan email tidak boleh ksong.</p>  <p>Setelah di submit:</p> <p>Data berhasil dikirim: Nama = Raffi Ilham Maulana, Email = Raffiilham47@gmail.com</p>
9	<p>Soal 7.3: Buatlah script langkah 6 dengan menggunakan ajax. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.3)</p> <p>Jawab:</p> <p>Dari penggunaan ajax tsb alurnya akan sama saja namun ada perbedaaan pada proses setelah data diisikan, yang mana setelah di kirimkan akan muncul output langsung tepat dibawah form, inilah yang jadi perbedaan pada method ajax, jika tidak diberi ajax, cetakannya akan muncul pada halaman selanjutnya</p> 
10	<p>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>jawab:</p> <p>di penambahan ini akan ada cetakan dan kotak permintaan sama seperti nama dan email tepat dibawah nama dan email tsb, password ini meminta format 8 isian,tidak sama seperti requestan lainnya, yang akan berisi sesuai yng kita ketik, disini hanya berbentuk tanda Bintang “*” jika kurang dari 8 akan muncul cetakan disebelah kanan password, “password harus berisi 8 karakter” selanjutnya setelah sesuai format dan data terkirim akan muncul output di hasil</p>

selanjutnya setelah email, yang berisi password dan hasil password yang kita isi dapat terlihat disini



← → ↻ ⓘ localhost/dasarWeb/form\_validasi.p... 🔗 ☆ + ⚙ □ 👤 ⋮

## Form Input dengan Validasi

Nama:

Email:

Password:

Data berhasil dikirim: Nama = Raffi Ilham Maulana, Email = Raffiilham47@gmail.com,  
Password = abcdefgh