

JOBSHEET 7

“PHP – Form Processing (jQuery dan PHP)”

Mata Kuliah Pemrograman Website

Dosen Pengampu: Moch. Zawaruddin Abdullah, S.ST., M.Kom.



Nama : Amanda Jasmyne Berliana Putri

NIM : 2241760081

Kelas/No Absen : SIB-2B/07

PROGRAM STUDI SISTEM INFORMASI BISNIS

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2024



Topik

- Konsep Form Proccessing dengan PHP dan JQuery

Tujuan

Mahasiswa diharapkan dapat:

1. Mahasiswa mampu membuat form menggunakan PHP
2. Mahasiswa mampu membuat form menggunakan jQuery

Perhatian

Jobsheet ini harus dikerjakan step-by-step sesuai langkah-langkah praktikum yang sudah diberikan. Soal dapat dijawab langsung di dalam kolom yang disediakan dengan menggunakan PDF Editor.

Pendahuluan

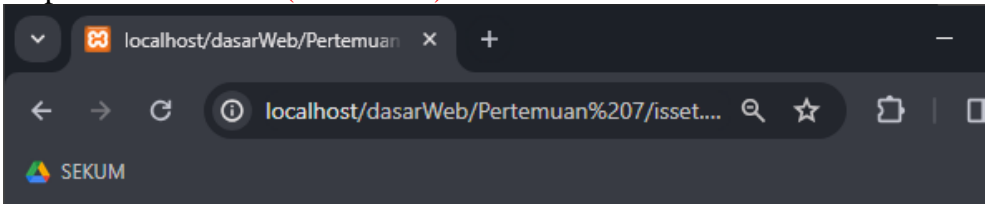
Function isset

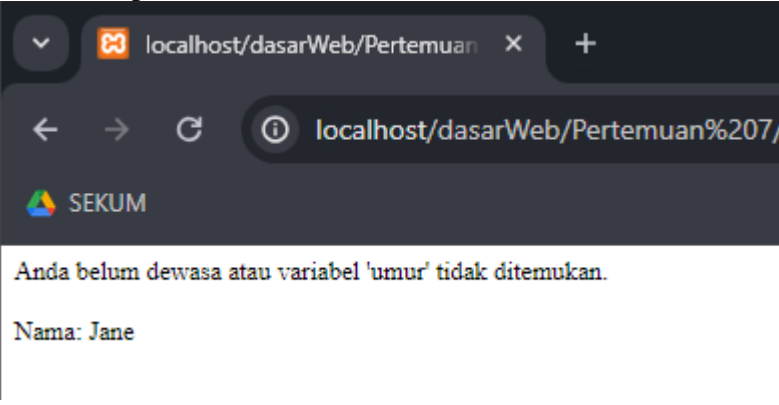
`isset()` adalah sebuah fungsi dalam PHP yang digunakan untuk memeriksa apakah suatu variabel telah didefinisikan (ada) atau tidak. Fungsi ini mengembalikan **true** jika variabel telah didefinisikan dan memiliki nilai, dan **false** jika variabel tidak ada atau bernilai **null**. Berikut adalah contoh script dan penjelasan minimal mengenai `isset()`:

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

Kode di atas memeriksa apakah variabel **\$nama** telah didefinisikan. Jika ya, maka pesan "Variabel 'nama' telah didefinisikan." akan ditampilkan, jika tidak, maka pesan "Variabel 'nama' tidak ditemukan atau bernilai null." akan ditampilkan.

Praktikum 1. Function Isset

Langkah	Keterangan
1	Buatlah satu file baru di dalam direktori dasarWeb, beri nama <code>isset.php</code> .
2	Ketikkan ke dalam file <code>isset.php</code> tersebut kode di bawah ini.
3	<pre>\$umur; if (isset(\$umur) && \$umur >= 18) { echo "Anda sudah dewasa."; } else { echo "Anda belum dewasa atau variabel 'umur' tidak ditemukan."; }</pre>
4	<p>Simpan file tersebut, kemudian buka browser dan jalankan <code>localhost/dasarWeb/isset.php</code>. Apa yang anda pahami dari penggunaan <code>isset</code> pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 1.1)</p>  <p>Anda belum dewasa atau variabel 'umur' tidak ditemukan.</p> <p>Penjelasan: Karena pada variable umur tidak memiliki value, pada pengecekan “if (isset(\$umur))” tidak dapat terpenuhi karena variabelnya null. Otomatis variable “\$umur >= 18” tidak akan terpenuhi karena variable umurnya 0. Sehingga yang terjadi adalah muncul statement seperti di atas.</p>
5	<p>Tambahkan isi dari file <code>isset.php</code> tersebut dengan kode di bawah ini.</p> <pre>\$data = array("nama" => "Jane", "usia" => 25); if (isset(\$data["nama"])) { echo "Nama: " . \$data["nama"]; } else { echo "Variabel 'nama' tidak ditemukan dalam array."; }</pre>

6	<p>Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/isset.php buat hasil tidak menjadi 1 baris, jadi tampilan dari echo harus terpisah. Apa yang anda pahami dari penggunaan isset pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 1.2)</p>  <p>Penjelasan: Setelah dimasukkan value, maka pengecekan akan bernilai true dan akan menjalankan perintah yaitu menampilkan nama yang ada pada array.</p>
---	---

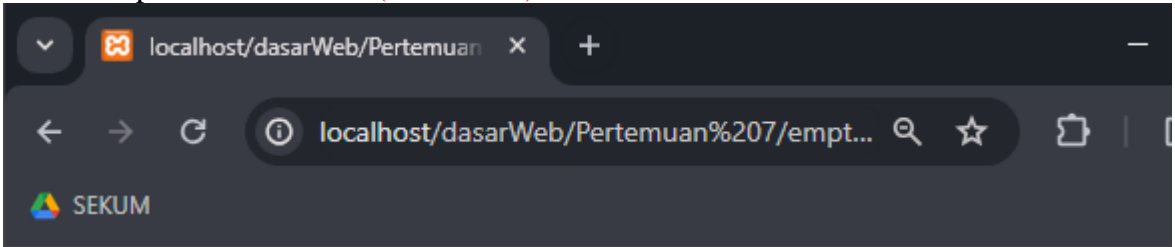
Function empty

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.";
}
```

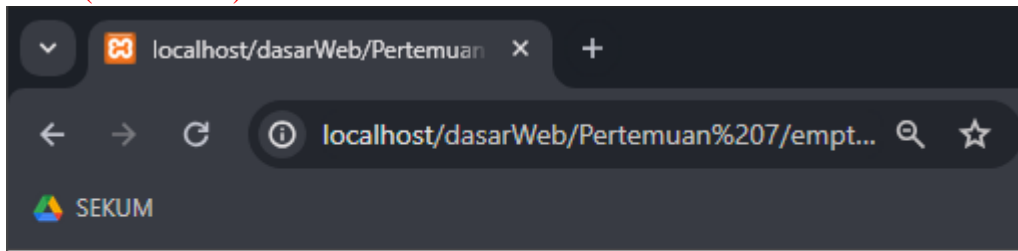
Fungsi **empty()** dapat digunakan untuk memeriksa apakah suatu string kosong atau tidak.

Praktikum 2. Function empty

Langkah	Keterangan
1	Buat satu file baru bernama <code>empty.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>empty.php</code>
2	<pre>\$myArray = array(); // Array kosong if (empty(\$myArray)) { echo "Array tidak terdefinisi atau kosong."; } else { echo "Array terdefinisi dan tidak kosong."; }</pre>
3	<p>Simpan file tersebut, kemudian buka browser dan jalankan <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>Array tidak terdefinisi atau kosong.</p> <p>Penjelasan: Karena kita tidak mendeklarasikan array dan tidak menaruh value, maka ketika dilakukan pengecekan akan muncul statement seperti di atas.</p>
4	<p>Tambahkan isi dari file <code>empty.php</code> 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

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)



Penjelasan: `Empty` digunakan untuk melakukan pengecekan apakah `variable/array` ada dan memiliki `value`.

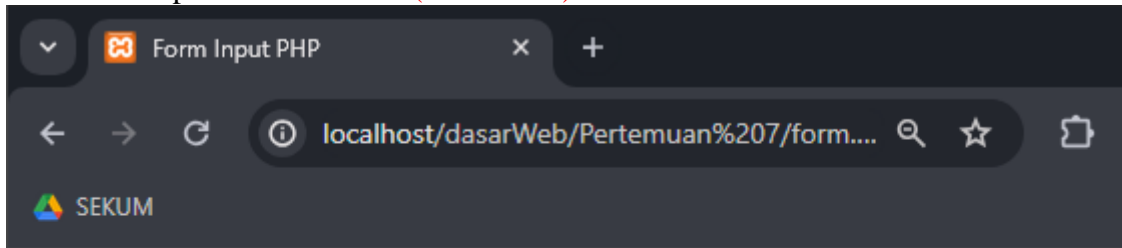
Praktikum Bagian 3 : Form Input PHP

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><!DOCTYPE html> <html> <head> <title>Form Input PHP</title> </head> <body> <h2>Form Input PHP</h2> <form method="post" action="proses_form.php"> <label for="nama">Nama:</label> <input type="text" name="nama" id="nama" required>

 <label for="email">Email:</label> <input type="email" name="email" id="email" required>

 <input type="submit" name="submit" value="Submit"> </form> </body> </html></pre>
3	<pre><?php if (\$_SERVER["REQUEST_METHOD"] == "POST") { \$nama = \$_POST["nama"]; \$email = \$_POST["email"]; echo "Nama: " . \$nama . "
"; echo "Email: " . \$email; } ?></pre>

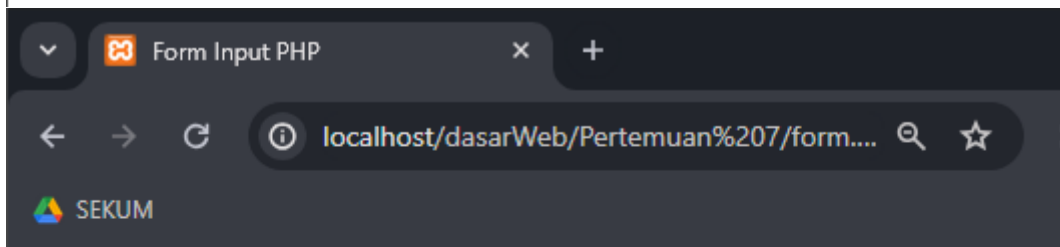
Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/proses_form.php. Apa yang anda pahami dari penggunaan form pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 3.1)



Form Input PHP

Nama:

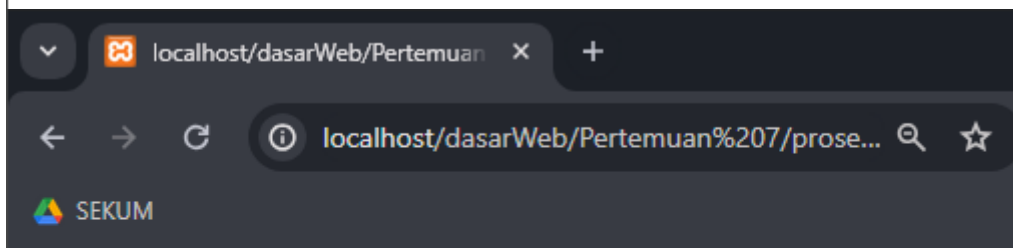
Email:



Form Input PHP

Nama:

Email:

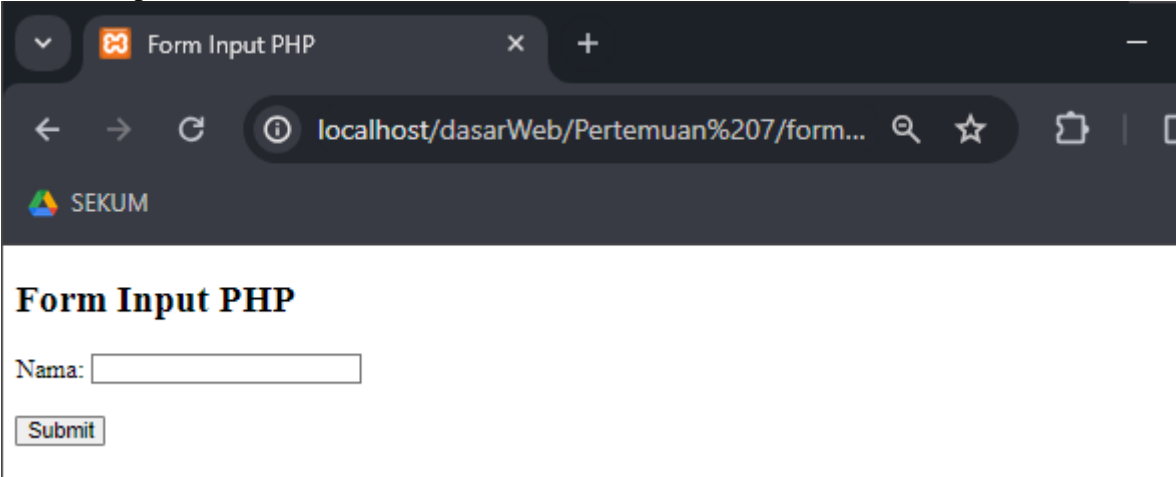


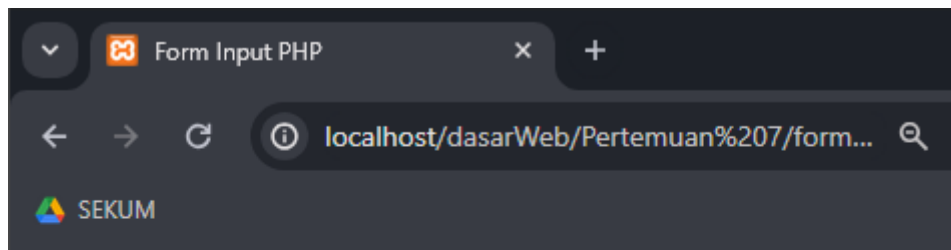
Nama: manda

Email: amandajbp04@gmail.com

Penjelasan: Ketika kita mengisi form.php dan klik submit, maka data yang kita masukkan akan ditampilkan pada proses_form.php.

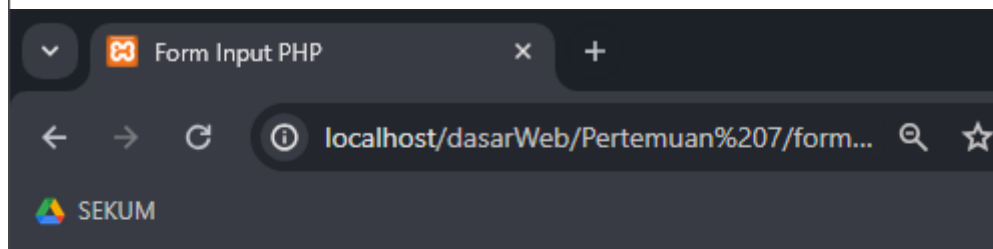
5	<p>Buat satu file baru bernama <code>form_self.php</code> di dalam folder <code>dasarWeb</code>. Ketikkan kode pada langkah 2 di dalam <code>form_self.php</code></p> <pre> <!DOCTYPE html> <html> <head> <title>Form Input PHP</title> </head> <body> <h2>Form Input PHP</h2> <?php // Inisialisasi variabel \$namaErr = ""; \$nama = ""; // Cek apakah form sudah disubmit if (\$_SERVER["REQUEST_METHOD"] == "POST") { // Validasi nama (contoh: pastikan nama tidak kosong) if (empty(\$_POST["nama"])) { \$namaErr = "Nama harus diisi!"; } else { \$nama = \$_POST["nama"]; echo "Data berhasil disimpan!"; } } ?> <form method="post" action="<?php echo htmlspecialchars(\$_SERVER["PHP_SELF"]); ?>"> <label for="nama">Nama:</label> <input type="text" name="nama" id="nama" value="<?php echo \$nama; ?>"> <?php echo \$namaErr; ?>

 <input type="submit" name="submit" value="Submit"> </form> </body> </html> </pre>
6	<p>Simpan file tersebut, kemudian buka browser dan jalankan <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> 



Form Input PHP

Nama:



Form Input PHP

Data berhasil disimpan

Nama:

Penjelasan: Ketika kita mengisi nama dan klik submit, maka akan ditampilkan statement “Data berhasil disimpan”, ketika nama tidak diisi dan klik submit maka akan muncul statement “Nama harus diisi.” Hal tersebut dapat terjadi karena itu termasuk penerapan if dan span-nya.

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:

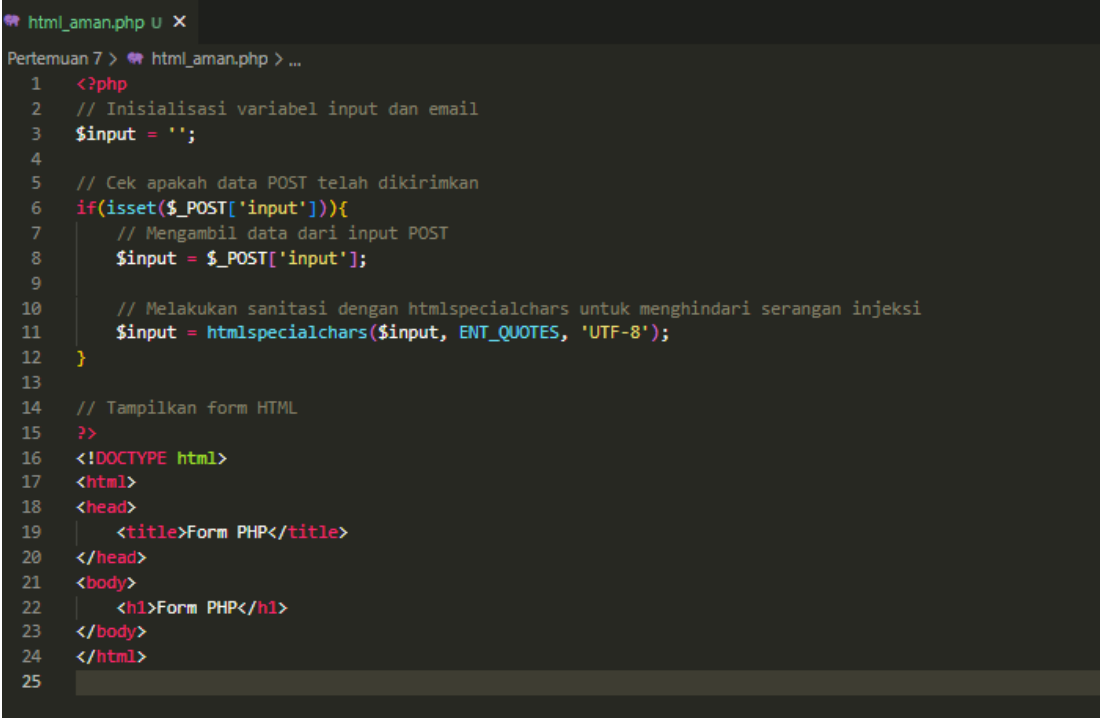
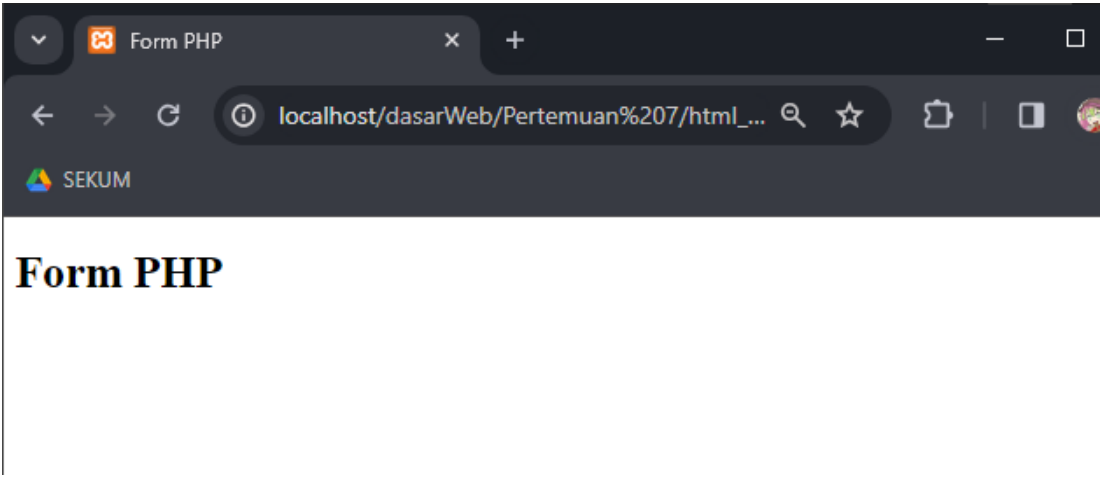
```
// Menggunakan filter_input untuk validasi input
$user_input = filter_input(INPUT_POST, 'user_input', FILTER_SANITIZE_STRING);

// 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 :

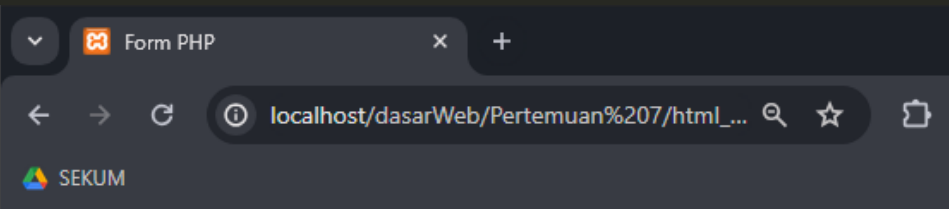
Langkah	Keterangan
1	Buat satu file baru bernama <code>html_aman.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>html_aman.php</code>
2	<pre>\$input = \$_POST['input']; \$input = htmlspecialchars(\$input, ENT_QUOTES, 'UTF-8');</pre>
3	<p>Tambahkan script lainnya supaya langkah 2 bisa berjalan normal. Lalu simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/html_aman.php</code></p>  
4	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.1)</p> <p>Penjelasan: Yang terjadi adalah muncul tulisan seperti gambar di atas.</p>

5	Ketikkan kode tambahan pada langkah 6 di dalam <code>html_aman.php</code>
6	<pre> // Periksa apakah input adalah email yang valid \$email = \$_POST['email']; if (filter_var(\$email, FILTER_VALIDATE_EMAIL)) { // Lanjutkan dengan pengolahan email yang aman } else { // Tangani input yang tidak valid } </pre>
7	<p>Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/html_aman.php</code></p> <pre> html_aman.php U X Pertemuan 7 > html_aman.php > html > body 1 <?php 2 // Inisialisasi variabel input dan email 3 \$input = ''; 4 \$email = ''; 5 \$result = ''; // Variabel untuk menyimpan hasil pengolahan 6 7 // Cek apakah data POST telah dikirimkan 8 if(isset(\$_POST['input'])){ 9 // Mengambil data dari input POST 10 \$input = \$_POST['input']; 11 12 // Melakukan sanitasi dengan htmlspecialchars untuk menghindari serangan injeksi 13 \$input = htmlspecialchars(\$input, ENT_QUOTES, 'UTF-8'); 14 } 15 16 // Cek apakah email POST telah dikirimkan 17 if(isset(\$_POST['email'])){ 18 \$email = \$_POST['email']; 19 20 // Melakukan validasi email dengan filter_var 21 if(filter_var(\$email, FILTER_VALIDATE_EMAIL)){ 22 // Lanjutkan dengan pengolahan email yang aman 23 24 // Misalnya, kita hanya akan menampilkan email yang valid 25 \$result = "Nama: " . \$input . "
Email yang valid: " . \$email; 26 }else{ 27 // Tangani input email yang tidak valid 28 \$result = "Nama: " . \$input . "
Email tidak valid."; 29 } 30 } 31 </pre>

```

32 // Tampilkan form HTML
33 >>
34 <!DOCTYPE html>
35 <html>
36 <head>
37   <title>Form PHP</title>
38 </head>
39 <body>
40   <h1>Form PHP</h1>
41   <form method="post" action="">
42     <label for="input">Masukkan data:</label>
43     <input type="text" name="input" id="input" value="<?php echo $input; ?>"><br><br>
44
45     <label for="email">Masukkan email:</label>
46     <input type="text" name="email" id="email" value="<?php echo $email; ?>"><br><br>
47
48     <input type="submit" value="Submit">
49   </form>
50
51   <?php
52   // Tampilkan hasil di bawah formulir jika ada hasil
53   if(!empty($result)){
54     echo "<p>" . $result . "</p>";
55   }
56   >>
57 </body>
58 </html>

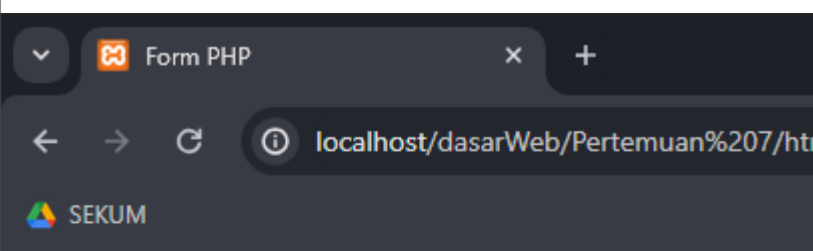
```



Form PHP

Masukkan data:

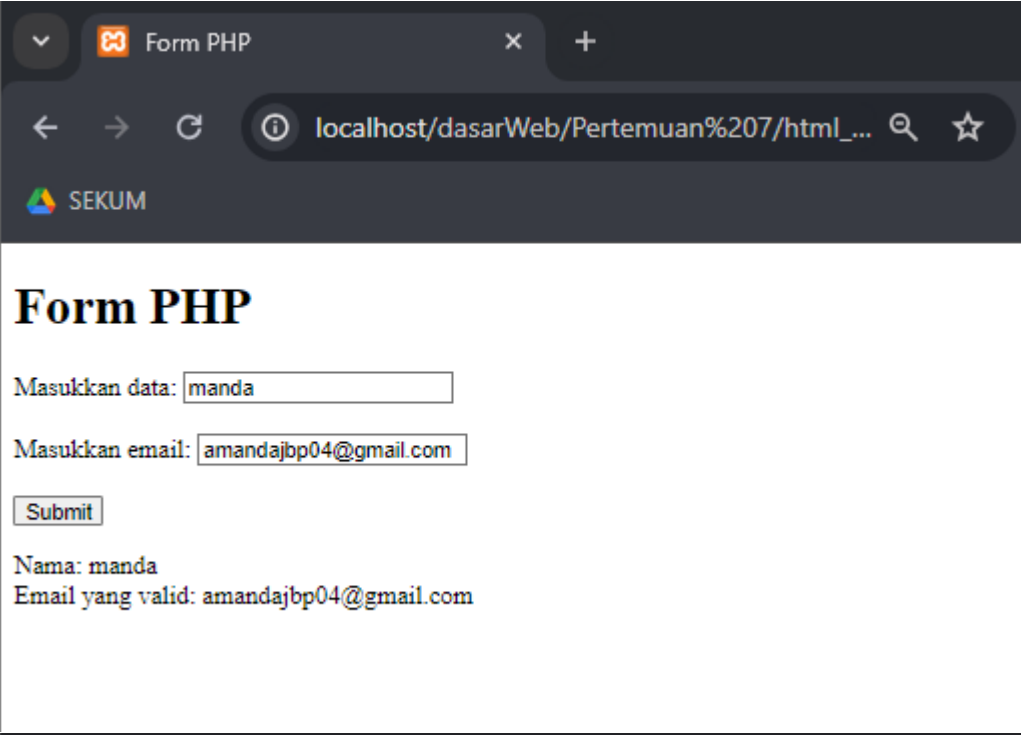
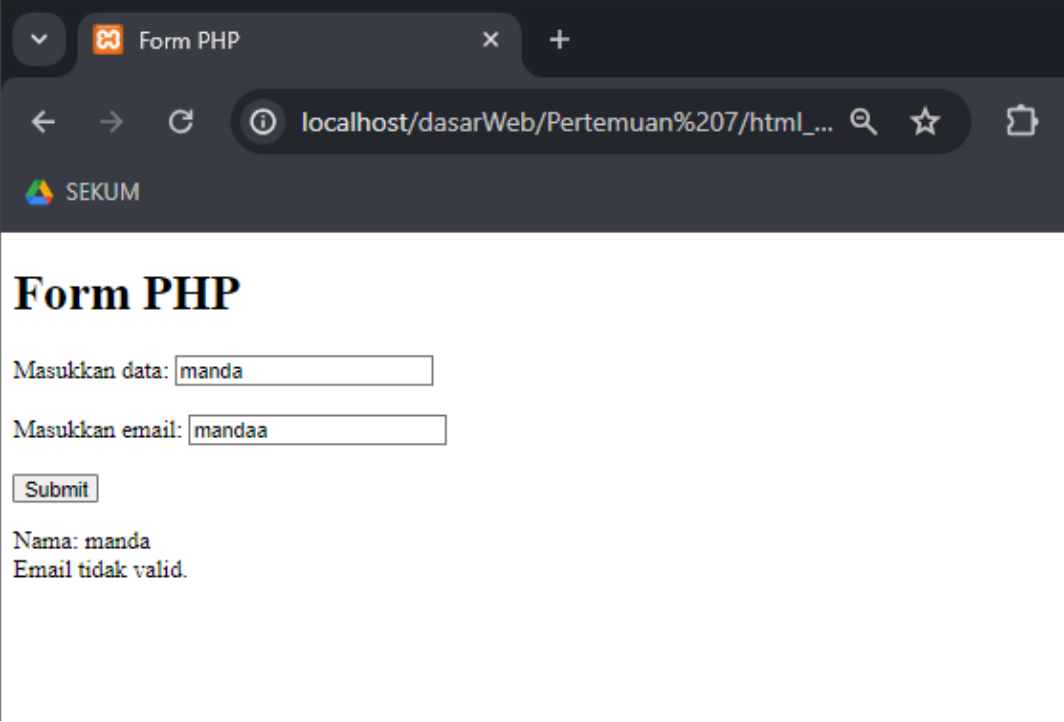
Masukkan email:



Form PHP

Masukkan data:

Masukkan email:

	 
8	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.2)</p> <p>Penjelasan: Digunakan “htmlspecialchars” dan “filter_var” dengan “FILTER_VALIDATE_EMAIL” untuk menghindari potensi injeksi dan melakukan validasi pada alamat email. Jika email yang dimasukkan valid (dengan @gmail.com), maka program akan menampilkan nama dan email yang valid.</p>

Regular Expression (Regex)

Regex adalah alat yang kuat untuk mencari, mencocokkan, atau memanipulasi teks berdasarkan pola tertentu. Anda dapat menggunakannya untuk berbagai tujuan seperti validasi input, pencarian teks, penggantian teks, dan banyak lagi.

Basic Regex Patterns:

- **Karakter Tunggal:** Karakter apa pun akan cocok dengan dirinya sendiri dalam Regex, kecuali karakter khusus yang harus dihindari dengan karakter pelarian \.
 - o Contoh: Pola `/a/` akan cocok dengan huruf "a" dalam teks.
- **Karakter Spesifik:** Anda dapat mencocokkan karakter tertentu dengan menyebutkannya dalam pola.
 - o Contoh: Pola `/hello/` akan cocok dengan teks "hello" dalam teks.
- **Karakter Khusus:** Beberapa karakter khusus dalam Regex harus dihindari dengan karakter pelarian (\) jika Anda ingin mencocokkannya secara harfiah. Contoh karakter khusus: `.`, `*`, `+`, `?`, `|`, `[`, `]`, `(`, `)`, `{`, `}`, `^`, `$`, `\`.
- **Karakter Set:** Anda dapat mencocokkan karakter dari satu set karakter dengan menggunakan `[]`. Misalnya, `/[aeiou]/` akan mencocokkan salah satu huruf vokal.
- **Rentang Karakter:** Anda dapat menggunakan `-` dalam karakter set untuk menentukan rentang karakter. Misalnya, `/[a-z]/` akan mencocokkan huruf kecil apa pun.
- **Kuantifier:** Anda dapat mengatur berapa kali karakter atau kelompok karakter sebelumnya harus muncul. Contoh kuantifier:
 - o `*`: 0 atau lebih kali
 - o `+`: 1 atau lebih kali
 - o `?`: 0 atau 1 kali
 - o `{n}`: Persis n kali
 - o `{n,}`: Setidaknya n kali
 - o `{n,m}`: Setidaknya n kali, maksimal m kali

Contoh Penggunaan Regex:

1. Mencocokkan pola

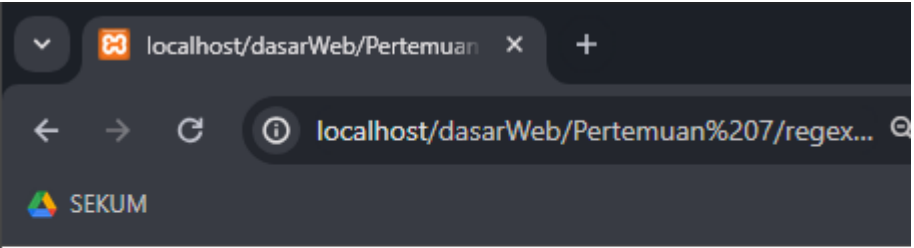
```
$pattern = '/hello/';
$text = 'Hello, world!';
if (preg_match($pattern, $text)) {
    echo "Cocokkan ditemukan!";
} else {
    echo "Tidak ada cocokan!";
}
```

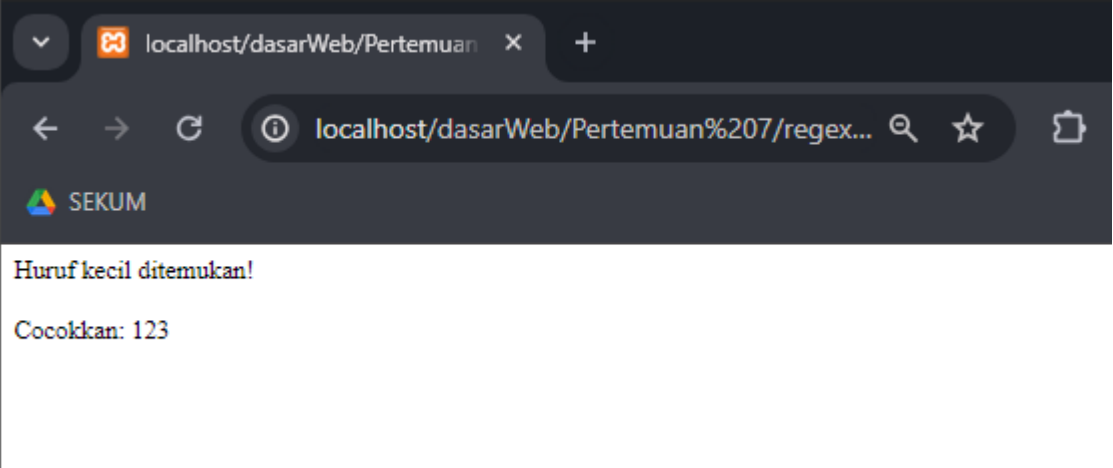
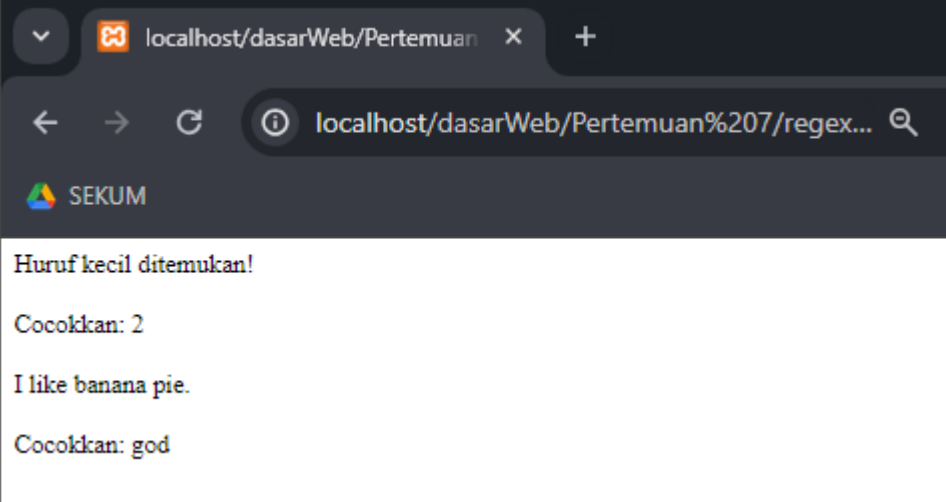
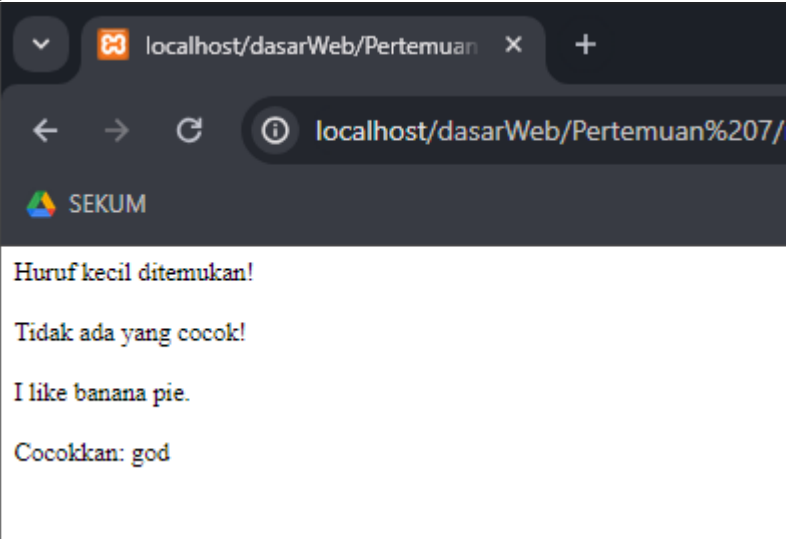
2. Mencocokkan karakter set

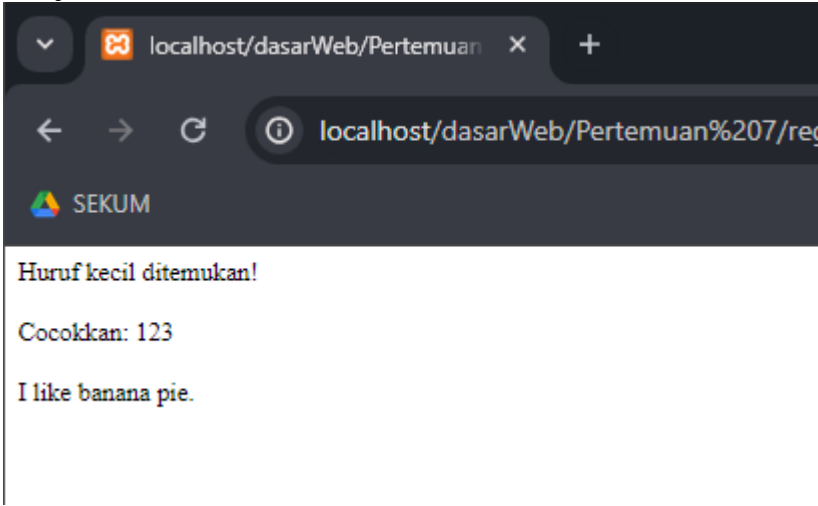
```
$pattern = '/[0-9]/'; // Cocokkan satu digit.
$text = 'There are 3 apples.';
if (preg_match($pattern, $text)) {
    echo "Digit ditemukan!";
} else {
    echo "Tidak ada digit!";
}
```


Praktikum 5 : Penggunaan Regex Pada PHP

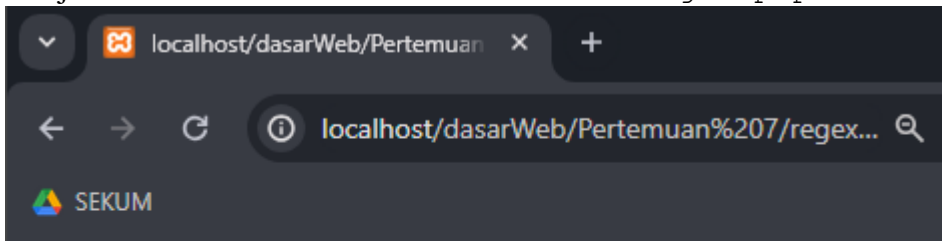
Langkah-langkah Praktikum :

Langkah	Keterangan
1	Buat satu file baru bernama 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	<p>Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php</p> 
4	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.1)</p> <p>Penjelasan: Dilakukan perbandingan sebuah sample text dengan pembanding 'a-z' yang dibandingkan dengan if-else. Hasilnya adalah true bahwa pada sample text memiliki huruf kecil.</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>

	<p>Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php</p>  <p>Huruf kecil ditemukan! Cocokkan: 123</p>  <p>Huruf kecil ditemukan! Cocokkan: 2 I like banana pie. Cocokkan: god</p>  <p>Huruf kecil ditemukan! Tidak ada yang cocok! I like banana pie. Cocokkan: god</p>
8	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.2)</p> <p>Penjelasan: Dilakukan perbandingan menggunakan digit angka. Ketika perbandingan antara sample kalimat dan pattern benar, maka statement true akan ditampilkan.</p>

9	Ketikkan kode tambahan pada langkah 10 di dalam <code>regex.php</code>
10	<pre> \$pattern = '/apple/'; \$replacement = 'banana'; \$text = 'I like apple pie.'; \$new_text = preg_replace(\$pattern, \$replacement, \$text); echo \$new_text; // Output: "I like banana pie." </pre>
11	<p>Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/regex.php</code></p> 
12	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.3)</p> <p>Penjelasan: Pada case ini, ketika saat dibandingkan dan ditemukan kesamaan, maka akan ditimpa kata tersebut dengan kata yang baru. Pada case ini adalah apple yang diganti dengan banana menggunakan <code>preg_replace</code>.</p>
13	Ketikkan kode tambahan pada langkah 14 di dalam <code>regex.php</code>
14	<pre> \$pattern = '/go*d/'; // Cocokkan "god", "good", "gooooo", dll. \$text = 'god is good.'; if (preg_match(\$pattern, \$text, \$matches)) { echo "Cocokkan: " . \$matches[0]; } else { echo "Tidak ada yang cocok!"; } </pre>

Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php

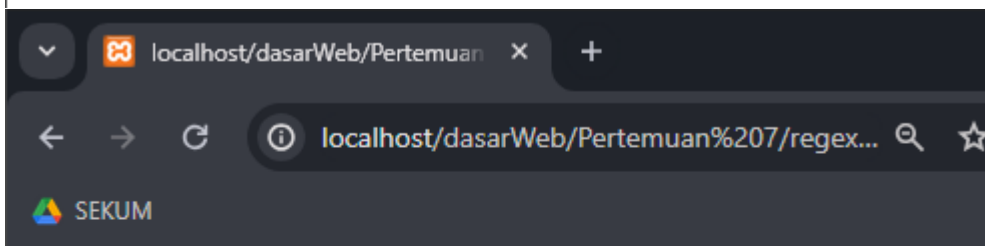


Huruf kecil ditemukan!

Cocokkan: 2

I like banana pie.

Cocokkan: god

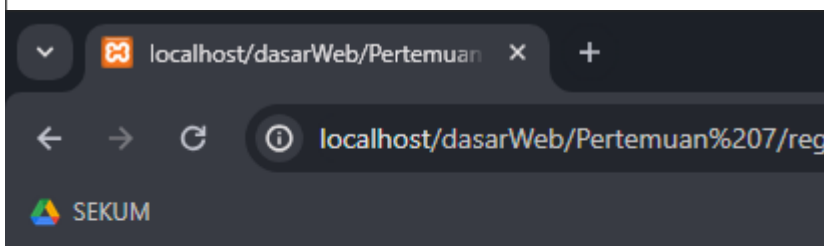


Huruf kecil ditemukan!

Cocokkan: 123

I like banana pie.

Cocokkan: good



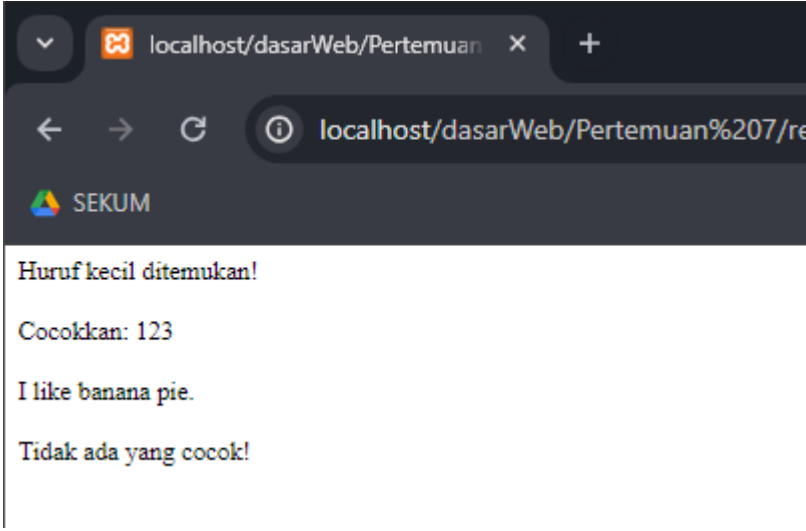
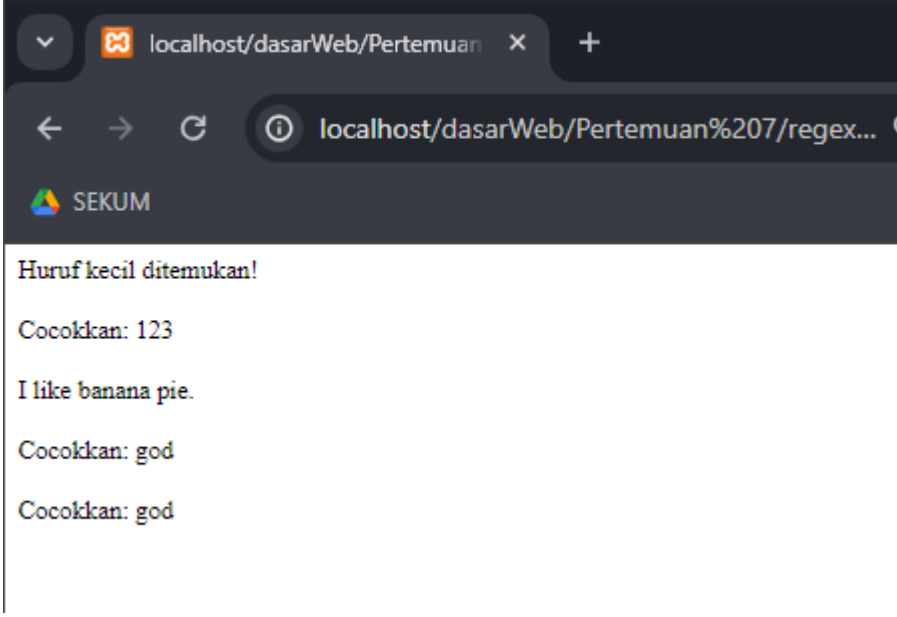
Huruf kecil ditemukan!

Cocokkan: 123

I like banana pie.

Cocokkan: goood

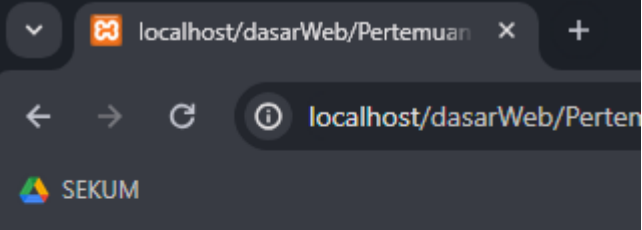
15

	
16	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.4)</p> <p>Penjelasan: Jika ada 2 kata yang sesuai dengan statement, maka akan ditampilkan yang awal saja karena kita mentarget kata pertama 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> " data-bbox="168 436 766 561"/> 

```

44 $pattern = '/go?od/'; // Cocokkan "god", "good", "goood", dll.
45 $text = 'goood is good.';
46
47 if (preg_match($pattern, $text, $matches)) {
48     echo "Cocokkan: " . $matches[0];
49 } else {
50     echo "Tidak ada yang cocok!";
51 }
52

```



Huruf kecil ditemukan!

Cocokkan: 123

I like banana pie.

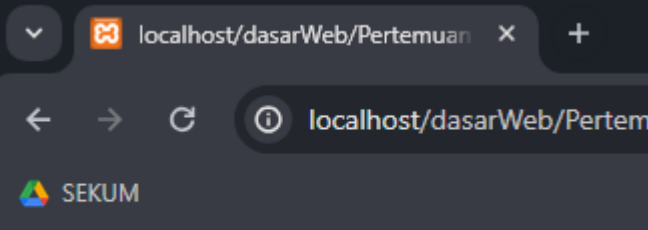
Cocokkan: god

Cocokkan: good

```

44 $pattern = '/go?od/'; // Cocokkan "god", "good", "goood", dll.
45 $text = 'goood is goood.';
46
47 if (preg_match($pattern, $text, $matches)) {
48     echo "Cocokkan: " . $matches[0];
49 } else {
50     echo "Tidak ada yang cocok!";
51 }
52

```



Huruf kecil ditemukan!

Cocokkan: 123

I like banana pie.

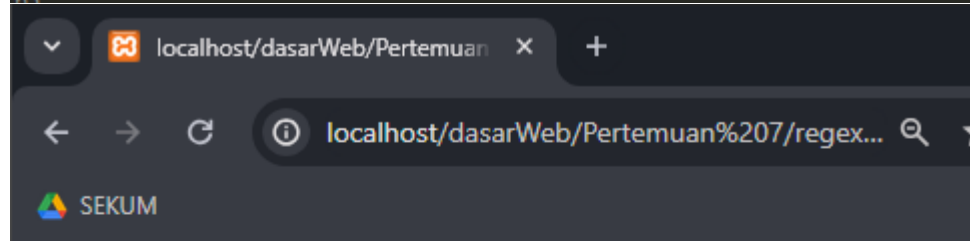
Cocokkan: god

Tidak ada yang cocok!

Penjelasan: Hasil yang muncul sama dengan hasil sebelumnya.

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)

```
55 $pattern = '/go{n,m}od/'; // Cocokkan "god", "good", "goood", dll.
56 $text = 'god is good.';
57
58 if (preg_match($pattern, $text, $matches)) {
59     echo "Cocokkan: " . $matches[0];
60 } else {
61     echo "Tidak ada yang cocok!";
62 }
63
```



Huruf kecil ditemukan!

Cocokkan: 123

I like banana pie.

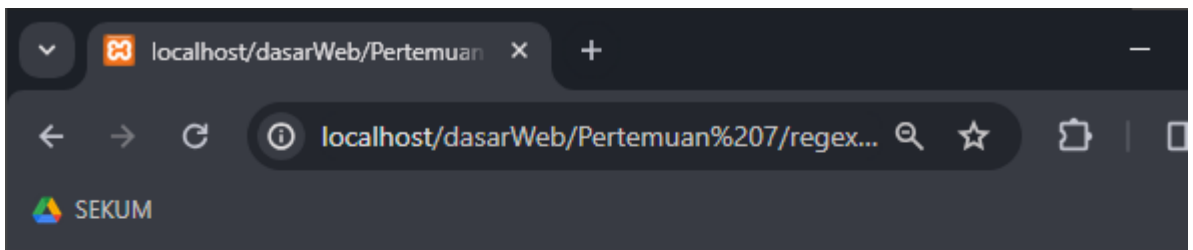
Cocokkan: god

Tidak ada yang cocok!

Tidak ada yang cocok!

```
55 $pattern = '/go{n,m}od/'; // Cocokkan "god", "good", "goood", dll.
56 $text = 'god is gomod.';
57
58 if (preg_match($pattern, $text, $matches)) {
59     echo "Cocokkan: " . $matches[0];
60 } else {
61     echo "Tidak ada yang cocok!";
62 }
63
```

```
55 $pattern = '/go{n,m}od/'; // Cocokkan "god", "good", "goood", dll.
56 $text = 'gonod is gomod.';
57
58 if (preg_match($pattern, $text, $matches)) {
59     echo "Cocokkan: " . $matches[0];
60 } else {
61     echo "Tidak ada yang cocok!";
62 }
63
```



Huruf kecil ditemukan!

Cocokkan: 123

I like banana pie.

Cocokkan: god

Tidak ada yang cocok!

Tidak ada yang cocok!

Penjelasan: Hasil yang harusnya muncul sama dengan praktikum 5 yang sebelumnya, disini saya salah menginputkan {n, m} yang seharusnya diganti dengan angka.

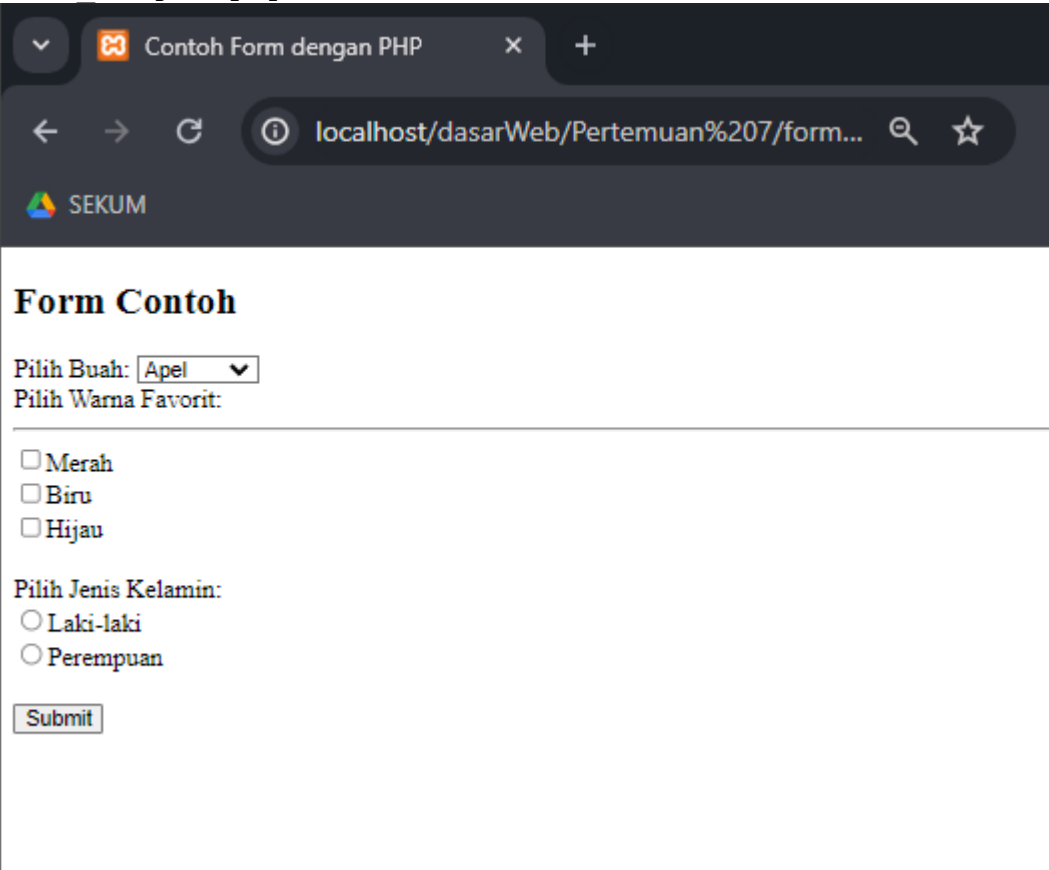
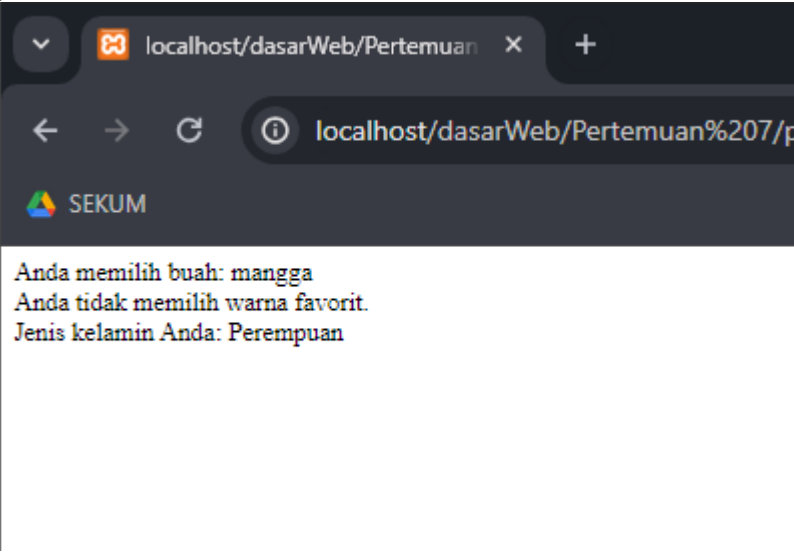
Praktikum 6 : Form Lanjut

Langkah-langkah Praktikum :

Langkah	Keterangan
1	Buat satu file baru bernama <code>form_lanjut.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>form_lanjut.php</code>
2	<pre> <!DOCTYPE html> <html> <head> <title>Contoh Form dengan PHP</title> </head> <body> <h2>Form Contoh</h2> <form method="POST" action="proses_lanjut.php"> <label for="buah">Pilih Buah:</label> <select name="buah" id="buah"> <option value="apel">Apel</option> <option value="pisang">Pisang</option> <option value="mangga">Mangga</option> <option value="jeruk">Jeruk</option> </select>
 <label>Pilih Warna Favorit:</label>
 <input type="checkbox" name="warna[]" value="merah"> Merah
 <input type="checkbox" name="warna[]" value="biru"> Biru
 <input type="checkbox" name="warna[]" value="hijau"> Hijau

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

 <input type="submit" value="Submit"> </form> </body> </html> </pre> <pre> if (\$_SERVER["REQUEST_METHOD"] == "POST") { \$selectedBuah = \$_POST['buah']; if (isset(\$_POST['warna'])) { \$selectedWarna = \$_POST['warna']; } else { \$selectedWarna = []; } \$selectedJenisKelamin = \$_POST['jenis_kelamin']; echo "Anda memilih buah: " . \$selectedBuah . "
"; if (!empty(\$selectedWarna)) { echo "Warna favorit Anda: " . implode(", ", \$selectedWarna) . "
"; } else { echo "Anda tidak memilih warna favorit.
"; } echo "Jenis kelamin Anda: " . \$selectedJenisKelamin; } </pre>

3	<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>Penjelasan: Yang akan muncul pada proses_lanjut adalah inputan yang sudah kita masukkan di form_lanjut sebelumnya. Jika tidak dipilih warna favorit, maka akan ditampilkan seperti statement di atas.</p>

Buat satu file baru bernama `form_ajax.php` di dalam folder `dasarWeb`. Ketikkan kode pada langkah 2 di dalam `form_ajax.php`

5

```
<!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>

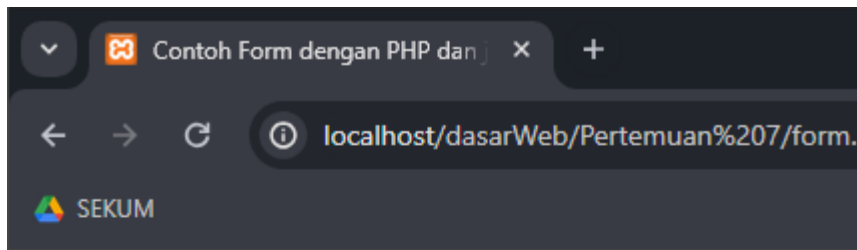
  <div id="hasil">
    <!-- Hasil akan ditampilkan di sini -->
  </div>

  <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>
```

Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/form_ajax.php. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 6.2)



Form Contoh

Pilih Buah:

Pilih Warna Favorit:

☒ Merah

☐ Biru

☐ Hijau

Pilih Jenis Kelamin:

☐ Laki-laki

☒ Perempuan

Anda memilih buah: mangga

Warna favorit Anda: merah

Jenis kelamin Anda: perempuan

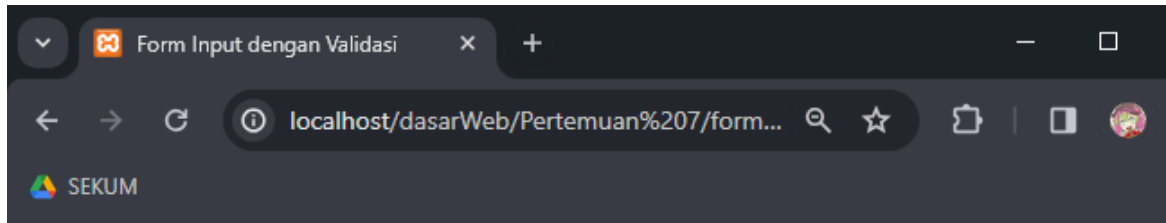
Penjelasan: Yang akan terjadi adalah javascript yang menggunakan jQuery akan mengumpulkan data hasil form dan mengirimkan ke PHP secara asinkron melalui AJAX. Lalu data akan dikirimkan ke proses_lanjut.php untuk diproses hasil inputannya.

Praktikum 7 : Validasi Form

Langkah-langkah Praktikum :

Langkah	Keterangan
1	Buat satu file baru bernama <code>form_validasi.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>form_validasi.php</code>
2	<pre><!DOCTYPE html> <html> <head> <title>Form Input dengan Validasi</title> </head> <body> <h1>Form Input dengan Validasi</h1> <form method="post" action="proses_validasi.php"> <label for="nama">Nama:</label> <input type="text" id="nama" name="nama">
 <label for="email">Email:</label> <input type="text" id="email" name="email">
 <input type="submit" value="Submit"> </form> </body> </html> <?php if (\$_SERVER["REQUEST_METHOD"] == "POST") { \$nama = \$_POST["nama"]; \$email = \$_POST["email"]; \$errors = array(); // Validasi Nama if (empty(\$nama)) { \$errors[] = "Nama harus diisi."; } // Validasi Email if (empty(\$email)) { \$errors[] = "Email harus diisi."; } elseif (filter_var(\$email, FILTER_VALIDATE_EMAIL)) { \$errors[] = "Format email tidak valid."; } // Jika ada kesalahan validasi if (empty(\$errors)) { (foreach (\$errors as \$error) { echo \$error . "
"; }) } else { // Lanjutkan dengan pemrosesan data jika semua validasi berhasil // Misalnya, menyimpan data ke database atau mengirim email echo "Data berhasil dikirim: Nama = \$nama, Email = \$email"; } } ?></pre>

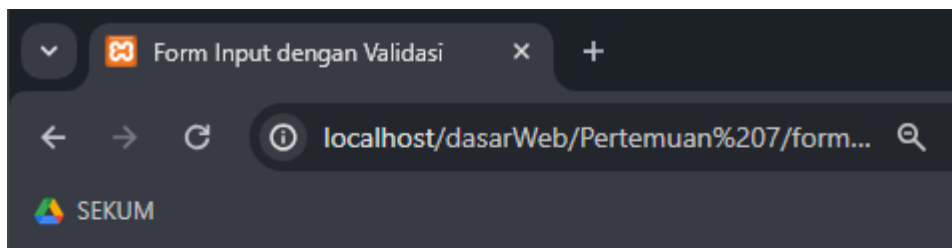
Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/form_validasi.php



Form Input dengan Validasi

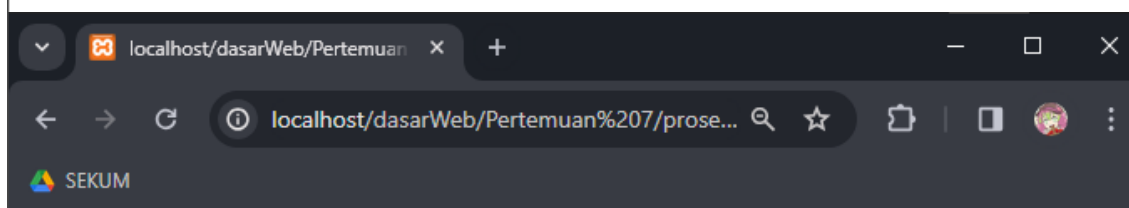
Nama:
Email:

3



Form Input dengan Validasi

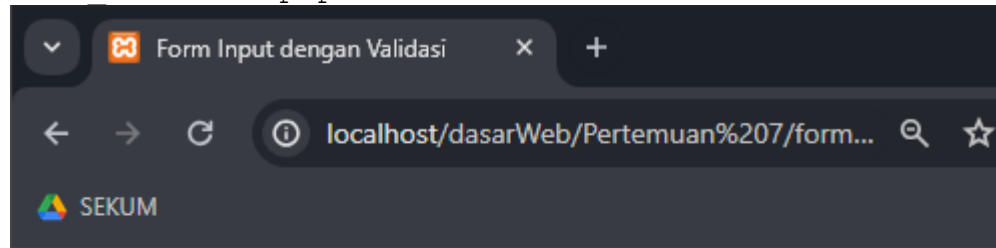
Nama:
Email:



Data berhasil dikirim: Nama = manda, Email = amandajbp04@gmail.com

4	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.1)</p> <p>Penjelasan: Ketika kita mengisi form pada form_validasi dan klik submit, maka akan dilakukan proses validasi dan hasilnya akan ditampilkan pada proses_validasi.</p>
5	<p>Kembangkan file bernama form_validasi.php di dalam folder dasarWeb. Ketikkan kode pada langkah 6 di dalam form_validasi.php</p>
6	<pre> <!DOCTYPE html> <html> <head> <title>Form Input dengan Validasi</title> <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script> </head> <body> <h1>Form Input dengan Validasi</h1> <form id="myForm" method="post" action="proses_validasi.php"> <label for="nama">Nama:</label> <input type="text" id="nama" name="nama">
 <label for="email">Email:</label> <input type="text" id="email" name="email">
 <input type="submit" value="Submit"> </form> <script> \$(document).ready(function() { \$("#myForm").submit(function(event) { var nama = \$("#nama").val(); var email = \$("#email").val(); var valid = true; if (nama === "") { \$("#nama-error").text("Nama harus diisi."); valid = false; } else { \$("#nama-error").text(""); } if (email === "") { \$("#email-error").text("Email harus diisi."); valid = false; } else { \$("#email-error").text(""); } if (!valid) { (event.preventDefault()); // Menghentikan pengiriman form jika validasi gagal } }); }); </script> </body> </html> </pre>

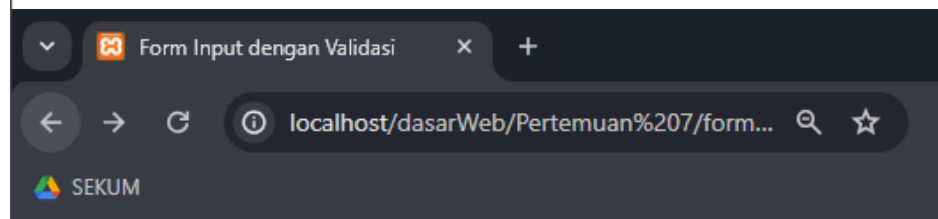
Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/form validasi.php



Form Input dengan Validasi

Nama:

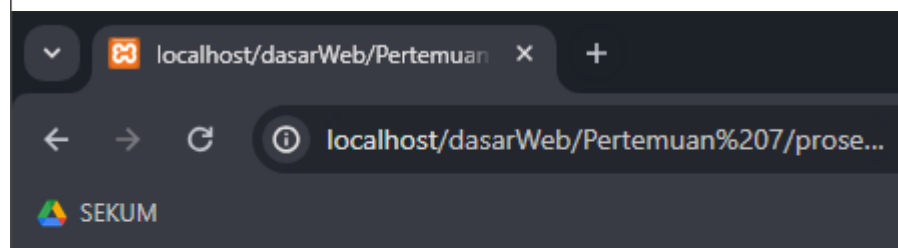
Email:



Form Input dengan Validasi

Nama: Nama harus diisi.

Email: Email harus diisi.



Nama harus diisi.
Email harus diisi

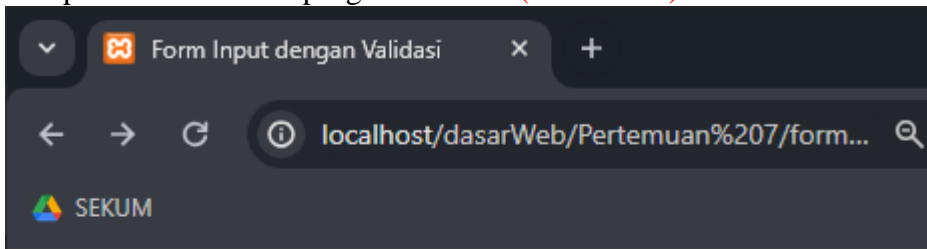
7

8

Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.2)

Penjelasan: Saat kita tidak mengisi form pada dorm_validasi, yang akan terjadi adalah seperti gambar di atas. Meskipun terdapat tulisan nama dan email harus diisi, tapi proses_validasi akan tetap terbuka dan menunjukkan statement bahwa nama dan email harus diisi.

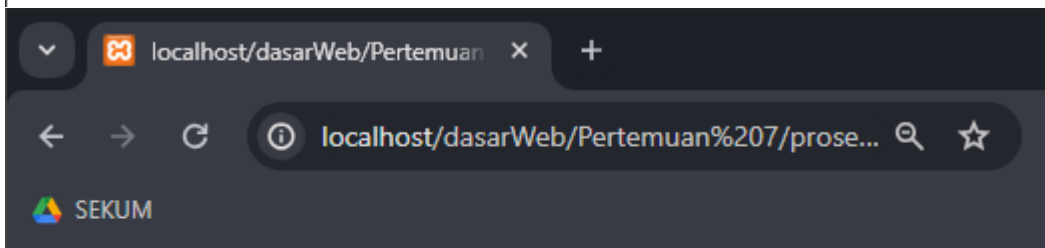
Soal 7.3: Buatlah script langkah 6 dengan menggunakan ajax. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.3)



Form Input dengan Validasi

Nama:
Email:

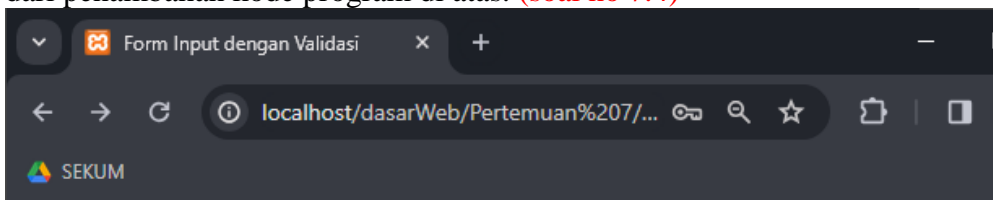
9



Data berhasil dikirim: Nama = manda, Email = amandajbp04@gmail.com

Penjelasan: Yang terjadi sama seperti sebelumnya.

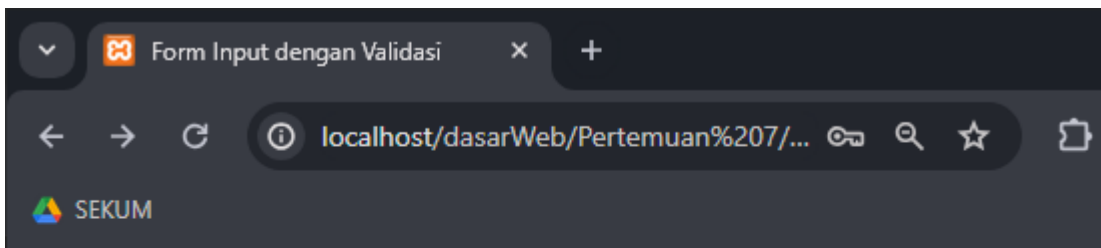
Soal 7.4: Tambahkan validasi untuk password dengan ketentuan minimal 8 karakter dengan menggunakan jQuery dan PHP tambahkan pada langkah 9. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.4)



Form Input dengan Validasi

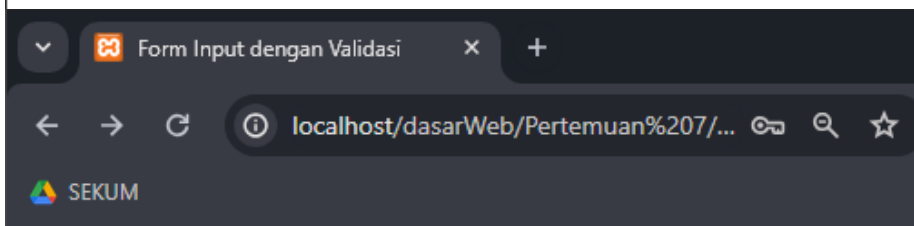
Nama:
Email:
Password:

10



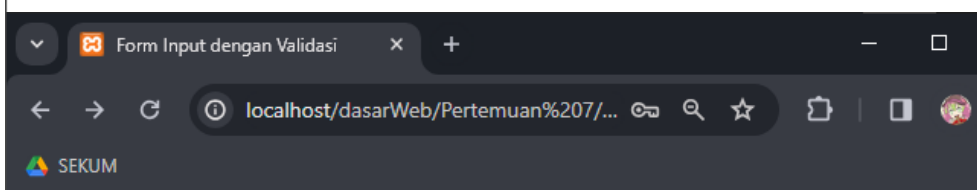
Form Input dengan Validasi

Nama:
Email:
Password: Password minimal 8 karakter.



Form Input dengan Validasi

Nama:
Email:
Password: Password minimal 8 karakter.



Form Input dengan Validasi

Nama:
Email:
Password:

Data berhasil dikirim: Nama = manda, Email = amandajbp04@gmail.com, Password = aaaaaaaa

Penjelasan: Jika kolom password tidak diisi/tidak diisi sesuai ketentuan yakni 8 karakter, maka yang terjadi adalah seperti gambar di atas. Ketika password sudah diisi sesuai ketentuan, maka akan ditampilkan di bagian bawah.