

**PRAKTIKUM DESAIN PEMROGRAMAN WEB**  
**JOBSHEET 7 Form Processing**



**Nama : Fatikah Salsabilla**

**Kelas : 2H**

**No. Absen : 14**

**NIM : 2341720003**

**PROGRAM STUDI TEKNIK INFORMATIKA**  
**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) &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> <hr/> <p>Anda belum dewasa atau variabel 'umur' tidak ditemukan</p> <p>Pada kode program tersebut isset digunakan untuk mengecek variabel \$umur sudah didefinisikan, kode program tersebut akan menampilkan nilai “null” karena \$umur tidak didefinisikan.</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> <hr/> <p>Anda belum dewasa atau variabel 'umur' tidak ditemukan Nama: Jane</p> <p>Pada kode program tersebut isset digunakan untuk mengecek variabel \$nama dalam array \$data sudah didefinisikan, kode program tersebut akan menampilkan nilai dari variabel \$nama “Nama: Jane” karena \$nama sudah didefinisikan.</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> <hr/> <p>Array tidak terdefinisi atau kosong.</p> <p>Pada kode program tersebut menghasilkan output seperti gambar diatas, karena variabel <code>\$myArray</code> bernilai <code>true</code> kosong atau tidak didefinisikan.</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	<p>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 yang anda pahami dari penggunaan <code>empty</code> pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 2.2)</p> <hr/> <p>Array tidak terdefinisi atau kosong.  Array tidak terdefinisi atau kosong.</p> <p>Pada kode program tersebut menghasilkan output seperti gambar diatas, karena variabel <code>\$nonExistentVar</code> bernilai <code>true</code> kosong atau belum didefinisikan.</p>

### Praktikum Bagian 3 : Form Input PHP

Langkah	Keterangan
1	Buat satu file baru bernama <code>form.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>form.php</code>

2	<pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;     &lt;title&gt;Form Input PHP&lt;/title&gt; &lt;/head&gt; &lt;body&gt;     &lt;h2&gt;Form Input PHP&lt;/h2&gt;     &lt;form method="post" action="proses_form.php"&gt;         &lt;label for="nama"&gt;Nama:&lt;/label&gt;         &lt;input type="text" name="nama" id="nama" required&gt;&lt;br&gt;&lt;br&gt;          &lt;label for="email"&gt;Email:&lt;/label&gt;         &lt;input type="email" name="email" id="email" required&gt;&lt;br&gt;&lt;br&gt;          &lt;input type="submit" name="submit" value="Submit"&gt;     &lt;/form&gt; &lt;/body&gt; &lt;/html&gt; </pre>
3	<p>Buat satu file baru bernama proses_form.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam proses_form.php</p> <pre> &lt;?php if (\$_SERVER["REQUEST_METHOD"] == "POST") {     \$nama = \$_POST["nama"];     \$email = \$_POST["email"];      echo "Nama: " . \$nama . "&lt;br&gt;";     echo "Email: " . \$email; } ?&gt; </pre>
4	<p>Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/proses_form.php. Apa yang anda pahami dari penggunaan form pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 3.1)</p> <hr/> <p><b>Form Input PHP</b></p> <p>Nama: <input type="text"/></p> <p>Email: <input type="email"/></p> <p><input type="submit" value="Submit"/></p>  <p>Nama: Fatimah Email: Fatimah@gmail.com</p>

	Pada 2 file tersebut form.php digunakan untuk menerima input dari pengguna, sedangkan proses_form.php digunakan untuk menampilkan data dari form.php
5	Buat satu file baru bernama form_self.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam form_self.php

```

<!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>

```

6

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)

Pada kode program tersebut terdapat validasi yang digunakan untuk memastikan pengguna meninputkan nama pada form tersebut.

## 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

## 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>Pada kode program tersebut terdapat validasi yang digunakan untuk memastikan pengguna meninputkan nama pada form tersebut dan menghapus atau menggantikan karakter yang berpotensi berbahaya.</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	Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/html_aman.php</code>



8

Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.2)

Pada kode program tersebut terdapat validasi yang digunakan untuk memastikan pengguna meninputkan nama pada form tersebut dan mengecek validasi email apakah valid atau tidak.

## Regular Expression (Regex)

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

### Basic Regex Patterns:

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

### Contoh Penggunaan Regex:

1. Mencocokkan pola

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

## 2. Mencocokkan karakter set

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

### Praktikum 5 : Penggunaan Regex Pada PHP

Langkah-langkah Praktikum :

Langkah	Keterangan
1	Buat satu file baru bernama <code>regex.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>regex.php</code>
2	<pre><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 <code>localhost/dasarWeb/regex.php</code>
4	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.1)</p> <p>Huruf kecil ditemukan</p> <p>Kode program tersebut berfungsi untuk mencocokkan variabel <code>\$text</code> apakah mengandung huruf kecil apapun, jika “ya” maka akan bernilai true dan jika “tidak” maka akan bernilai false.</p>
5	Ketikkan kode tambahan pada langkah 6 di dalam <code>regex.php</code>
6	<pre><code>\$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!"; }</code></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>Cocokkan: 1</p> <p>Kode program tersebut berfungsi untuk mencocokkan variabel \$text apakah mengandung angka dalam text tersebut,jika “ya” maka akan bernilai true dan menampilkan hasil dari angka yang pertama kali ditemukan dan jika “tidak” maka akan bernilai false.</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>I like banana pie.</p> <p>Pada kode program tersebut akan mencocokkan \$text dengan \$pattern apakah karakter spesifiknya sama, jika sama maka \$pattern akan di ganti dengan \$replacement sehingga menjadi “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)</p> <p>Cocokkan: god</p> <p>Pada kode program tersebut akan mencocokkan \$pattern digunakan untuk mencocokkan \$text dengan pola /go*d/, kemudian jika sudah ditemukan, maka akan disimpan dalam variabel \$matches dan akan menampilkan kata pertama yaitu “god”, jika tidak ditemukan maka akan menampilkan “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>Cocokkan: god</p> <div data-bbox="272 1144 608 1294"> <pre> \$pattern = '/god?/'; // Cocokkan "god", "g \$text = 'god.'; if (preg_match(\$pattern, \$text, \$matches)){     echo "Cocokkan: " . \$matches[0]; } else{     echo "Tidak ada yang cocok!"; } </pre> </div> <div data-bbox="639 1245 946 1294"> <p>Tidak ada yang cocok!</p> </div> <p>Pada kode program tersebut akan mencocokkan \$pattern digunakan untuk mencocokkan \$text dengan pola /go*d/ dengan karakter o sebanyak 0 atau 1 kali, kemudian jika sudah ditemukan, maka akan disimpan dalam variabel \$matches dan akan menampilkan kata pertama yaitu “god”, jika tidak ditemukan maka akan menampilkan “Tidak ada yang cocok!”. Saat \$text diganti dengan “good” maka akan bernilai “false” karena ada 2 karakter o.</p>
18	<p>Soal 5.6: Buatlah script langkah 14 dengan variable pattern yang di ubah menggunakan `{n,m}`. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.6)</p> <p>Cocokkan: god</p> <div data-bbox="272 1675 608 1720"> <pre>\$text = 'is good.';</pre> </div> <div data-bbox="639 1686 799 1720"> <p>Cocokkan: good</p> </div> <div data-bbox="272 1742 608 1776"> <pre>\$text = 'is goood.';</pre> </div> <div data-bbox="639 1742 839 1776"> <p>Tidak ada yang cocok!</p> </div> <p>Pada kode program tersebut akan mencocokkan \$pattern digunakan untuk mencocokkan \$text dengan pola /go*d/ dengan karakter o sebanyak n=1 sampai maksimal 3 kali, kemudian jika sudah ditemukan, maka akan disimpan dalam variabel \$matches dan akan menampilkan kata pertama yaitu “god”, jika tidak ditemukan maka akan menampilkan “Tidak ada yang cocok!”. Jika isi variabel \$text diubah menjadi “is good” maka hasilnya akan menampilkan “Cocokkan: good” karena ada 2 karakter o, jika isi variabel \$text diubah menjadi “is goood” maka akan bernilai false karena batas maksimal 3 karakter o.</p>

## Praktikum 6 : Form Lanjut

Langkah-langkah Praktikum :

Langkah	Keterangan
1	Buat satu file baru bernama <code>form_lanjut.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>form_lanjut.php</code>

2	<pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;     &lt;title&gt;Contoh Form dengan PHP&lt;/title&gt; &lt;/head&gt; &lt;body&gt;     &lt;h2&gt;Form Contoh&lt;/h2&gt;     &lt;form method="POST" action="proses_lanjut.php"&gt;         &lt;label for="buah"&gt;Pilih Buah:&lt;/label&gt;         &lt;select name="buah" id="buah"&gt;             &lt;option value="apel"&gt;Apel&lt;/option&gt;             &lt;option value="pisang"&gt;Pisang&lt;/option&gt;             &lt;option value="mangga"&gt;Mangga&lt;/option&gt;             &lt;option value="jeruk"&gt;Jeruk&lt;/option&gt;         &lt;/select&gt;          &lt;br&gt;          &lt;label&gt;Pilih Warna Favorit:&lt;/label&gt;&lt;br&gt;         &lt;input type="checkbox" name="warna[]" value="merah"&gt; Merah&lt;br&gt;         &lt;input type="checkbox" name="warna[]" value="biru"&gt; Biru&lt;br&gt;         &lt;input type="checkbox" name="warna[]" value="hijau"&gt; Hijau&lt;br&gt;          &lt;br&gt;          &lt;label&gt;Pilih Jenis Kelamin:&lt;/label&gt;&lt;br&gt;         &lt;input type="radio" name="jenis_kelamin" value="laki-laki"&gt; Laki-laki&lt;br&gt;         &lt;input type="radio" name="jenis_kelamin" value="perempuan"&gt; Perempuan&lt;br&gt;          &lt;br&gt;          &lt;input type="submit" value="Submit"&gt;     &lt;/form&gt; &lt;/body&gt; &lt;/html&gt;  if (\$_SERVER["REQUEST_METHOD"] == "POST") {     \$selectedBuah = \$_POST['buah'];      if (isset(\$_POST['warna'])) {         \$selectedWarna = \$_POST['warna'];     } else {         \$selectedWarna = [];     }      \$selectedJenisKelamin = \$_POST['jenis_kelamin'];      echo "Anda memilih buah: " . \$selectedBuah . "&lt;br&gt;";      if (!empty(\$selectedWarna)) {         echo "Warna favorit Anda: " . implode(", ", \$selectedWarna) . "&lt;br&gt;";     } else {         echo "Anda tidak memilih warna favorit.&lt;br&gt;";     }      echo "Jenis kelamin Anda: " . \$selectedJenisKelamin; } </pre>
3	Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/form_lanjut.php
4	Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 6.1)



	<p><b>Form Contoh</b></p> <p>Pilih Buah: <input type="text" value="Mangga"/></p> <p>Pilih Warna Favorit:</p> <p><input type="checkbox"/> Merah</p> <p><input checked="" type="checkbox"/> Biru</p> <p><input type="checkbox"/> Hijau</p> <p>Pilih Jenis Kelamin:</p> <p><input type="radio"/> Laki -Laki</p> <p><input checked="" type="radio"/> Perempuan</p> <p><input type="button" value="Submit"/></p> <div> <p>Anda memilih buah: mangga</p> <p>Warna favorit Anda: biru</p> <p>Jenis kelamin Anda: perempuan</p> </div> <p>Pada kode program tersebut file <code>form_lanjut.php</code> digunakan untuk mengirim data yang sudah di submit kemudian pada file <code>proses_lanjut.php</code> digunakan untuk menampilkan hasil dari inputan dimana <code>\$selectedBuah</code> menampilkan buah yang dipilih kemudian <code>\$selectedWarna</code> menampilkan warna favorit yang dipilih, lalu <code>\$selectedJenisKelamin</code> akan menampilkan jenis kelamin yang pilih.</p>
5	<p>Buat satu file baru bernama <code>form_ajax.php</code> di dalam folder <code>dasarWeb</code>. Ketikkan kode pada langkah 2 di dalam <code>form_ajax.php</code></p>

```

<!DOCTYPE html>
<html>
<head>
  <title>Contoh Form dengan PHP dan jQuery</title>
  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
</head>
<body>
  <h2>Form Contoh</h2>
  <form id="myForm">
    <label for="buah">Pilih Buah:</label>
    <select name="buah" id="buah">
      <option value="apel">Apel</option>
      <option value="pisang">Pisang</option>
      <option value="mangga">Mangga</option>
      <option value="jeruk">Jeruk</option>
    </select>

    <br>

    <label>Pilih Warna Favorit:</label><br>
    <input type="checkbox" name="warna[]" value="merah"> Merah<br>
    <input type="checkbox" name="warna[]" value="biru"> Biru<br>
    <input type="checkbox" name="warna[]" value="hijau"> Hijau<br>

    <br>

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

    <br>

    <input type="submit" value="Submit">
  </form>

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

  <script>
    $(document).ready(function () {
      $("#myForm").submit(function (e) {
        e.preventDefault(); // Mencegah pengiriman form secara default

        // Mengumpulkan data form
        var formData = $("#myForm").serialize();

        // Kirim data ke server PHP
        $.ajax({
          url: "proses_lanjut.php", // Ganti dengan nama file PHP yang sesuai
          type: "POST",
          data: formData,
          success: function (response) {
            // Tampilkan hasil dari server di div "hasil"
            $("#hasil").html(response);
          }
        });
      });
    });
  </script>
</body>
</html>

```

6

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

### Form Contoh

Pilih Buah:

Pilih Warna Favorit:

☒ Merah

☐ Biru

☐ Hijau

Pilih Jenis Kelamin:

☒ Laki-laki

☐ Perempuan

### Form Contoh

Pilih Buah:

Pilih Warna Favorit:

☒ Merah

☐ Biru

☐ Hijau

Pilih Jenis Kelamin:

☒ Laki-laki

☐ Perempuan

Anda memilih buah: jeruk

Warna favorit Anda: merah

Jenis kelamin Anda: laki-laki

Pada kode program tersebut data dikirim dengan metode ajax dengan menggunakan metode ajax kemudian setelah data disubmit hasilnya akan berada di halaman yang sama .

## Praktikum 7 : Validasi Form

Langkah-langkah Praktikum :

Langkah	Keterangan
1	Buat satu file baru bernama <code>form_validasi.php</code> di dalam folder <code>dasarWeb</code> . Ketikkan kode pada langkah 2 di dalam <code>form_validasi.php</code>
2	<pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;     &lt;title&gt;Form Input dengan Validasi&lt;/title&gt; &lt;/head&gt; &lt;body&gt;     &lt;h1&gt;Form Input dengan Validasi&lt;/h1&gt;     &lt;form method="post" action="proses_validasi.php"&gt;         &lt;label for="nama"&gt;Nama:&lt;/label&gt;         &lt;input type="text" id="nama" name="nama"&gt;         &lt;br&gt;           &lt;label for="email"&gt;Email:&lt;/label&gt;         &lt;input type="text" id="email" name="email"&gt;         &lt;br&gt;           &lt;input type="submit" value="Submit"&gt;     &lt;/form&gt; &lt;/body&gt; &lt;/html&gt; </pre> <pre> &lt;?php if (\$_SERVER["REQUEST_METHOD"] == "POST") {     \$nama = \$_POST["nama"];     \$email = \$_POST["email"];     \$errors = array();      // Validasi Nama     if (empty(\$nama)) {         \$errors[] = "Nama harus diisi.";     }      // Validasi Email     if (empty(\$email)) {         \$errors[] = "Email harus diisi.";     } elseif (filter_var(\$email, FILTER_VALIDATE_EMAIL)) {         \$errors[] = "Format email tidak valid.";     }      // Jika ada kesalahan validasi     if (empty(\$errors)) {         foreach (\$errors as \$error) {             echo \$error . "&lt;br&gt;";         }     } else {         // Lanjutkan dengan pemrosesan data jika semua validasi berhasil         // Misalnya, menyimpan data ke database atau mengirim email         echo "Data berhasil dikirim: Nama = \$nama, Email = \$email";     } } ?&gt; </pre>
3	Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/form_validasi.php</code>
4	Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.1)

	<div data-bbox="277 136 699 293"> <p><b>Form Input dengan Validasi</b></p> <p>Nama: <input type="text"/></p> <p>Email: <input type="text"/></p> <p><input type="submit" value="Submit"/></p> </div> <div data-bbox="722 208 952 280"> <p>Nama harus diisi. Email harus diisi.</p> </div> <div data-bbox="277 320 699 454"> <p><b>Form Input dengan Validasi</b></p> <p>Nama: <input type="text"/></p> <p>Email: <input type="text" value="Fatikah@gmail.com"/></p> <p><input type="submit" value="Submit"/></p> </div> <div data-bbox="722 407 948 441"> <p>Nama harus diisi.</p> </div> <div data-bbox="277 483 699 618"> <p><b>Form Input dengan Validasi</b></p> <p>Nama: <input type="text" value="Fatikah"/></p> <p>Email: <input type="text"/></p> <p><input type="submit" value="Submit"/></p> </div> <div data-bbox="722 573 952 607"> <p>Email harus diisi.</p> </div> <div data-bbox="277 638 699 772"> <p><b>Form Input dengan Validasi</b></p> <p>Nama: <input type="text" value="Fatikah"/></p> <p>Email: <input type="text" value="Fatik"/></p> <p><input type="submit" value="Submit"/></p> </div> <div data-bbox="683 725 967 759"> <p>Format email tidak valid.</p> </div> <div data-bbox="277 790 699 913"> <p><b>Form Input dengan Validasi</b></p> <p>Nama: <input type="text" value="Fatikah"/></p> <p>Email: <input type="text" value="Fatikah@gmail.com"/></p> <p><input type="submit" value="Submit"/></p> </div> <div data-bbox="689 869 1431 902"> <p>Data berhasil dikirim: Nama = Fatikah , Email = Fatikah@gmail.com</p> </div> <p>Kode program tersebut digunakan untuk melakukan validasi agar menginputkan data dengan sesuai.</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	Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/form_validasi.php
8	<p>Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.2)</p> <div data-bbox="272 1339 1460 1489"> <div> <h3>Form Input dengan Validasi</h3> <p>Nama: <input type="text"/> Nama harus diisi.</p> <p>Email: <input type="text"/> Email harus diisi.</p> <p><input type="submit" value="Submit"/></p> </div> <div> <h3>Form Input dengan Validasi</h3> <p>Nama: <input type="text"/> Nama harus diisi.</p> <p>Email: <input type="text" value="Fatimah@gmail.com"/> Email harus diisi.</p> <p><input type="submit" value="Submit"/></p> </div> <div> <p>Data berhasil dikirim:</p> <p>Nama = Fatimah</p> <p>Email = Fatimah@gmail.com</p> </div> </div> <p>Kode program tersebut digunakan untuk memastikan bahwa data yang dikirim valid.</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> <div data-bbox="272 1630 687 1843"> <h3>Form Input dengan Validasi</h3> <p>Nama: <input type="text" value="Fatimah"/></p> <p>Email: <input type="text" value="Fatimah@gmail.com"/></p> <p><input type="submit" value="Submit"/></p> <p>Data berhasil dikirim:</p> <p>Nama = Fatimah</p> <p>Email = Fatimah@gmail.com</p> </div> <p>Kode program tersebut menampilkan hasil input form di halaman web yang sama.</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>



## Form Input dengan Validasi

Nama:   
Email:   
Password:

Data berhasil dikirim:

Nama - Fatikah

Email - Fatikah@gmail.com

## Form Input dengan Validasi

Nama:   
Email:   
Password:  Password minimal 8 karakter.

Menambahkan input password dengan minimal 8 karakter, jika kurang dari 8 karakter maka tidak bisa disubmit.