

JOBSHEET 7
PHP - Form Proccessing



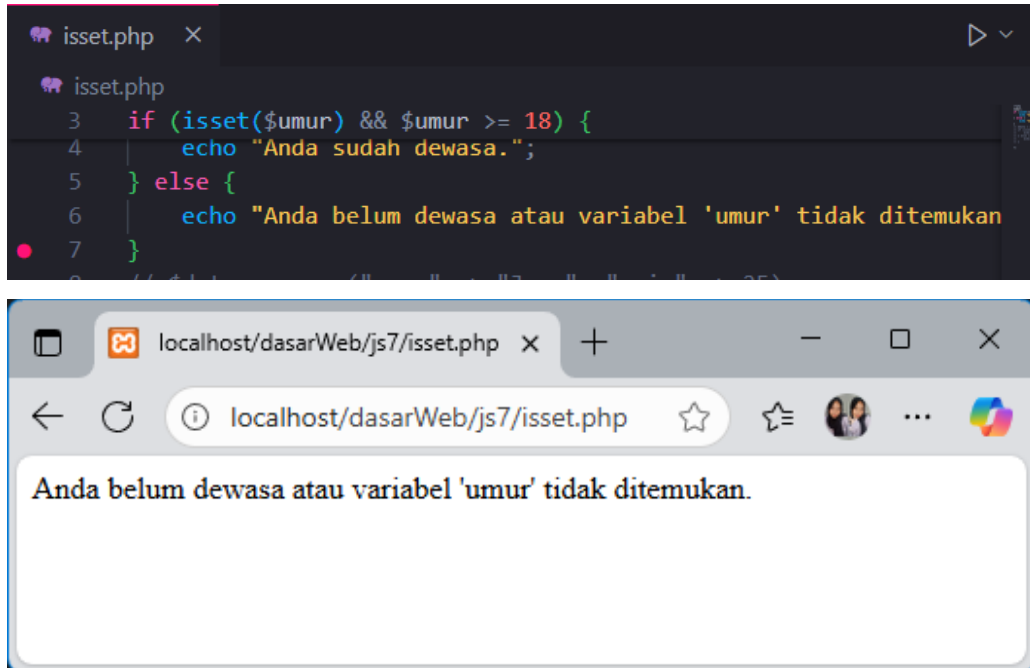
Oleh
Fina Ismatus Saniyah
D-IV Sistem Informasi Bisnis
244107060004

TEKNOLOGI INFORMASI
POLITEKNIK NEGERI MALANG
2025

PHP - Form Processing

Praktikum 1. Function Isset

Soal no 1.1



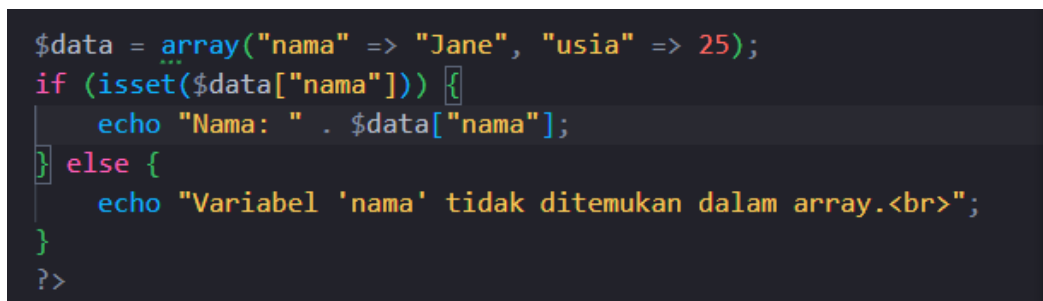
The image shows two parts: a code editor at the top and a web browser at the bottom. The code editor displays a PHP file named 'isset.php' with the following code:

```
3 if (isset($umur) && $umur >= 18) {  
4     echo "Anda sudah dewasa.";   
5 } else {  
6     echo "Anda belum dewasa atau variabel 'umur' tidak ditemukan  
7 }
```

The web browser shows the output of the script: "Anda belum dewasa atau variabel 'umur' tidak ditemukan."

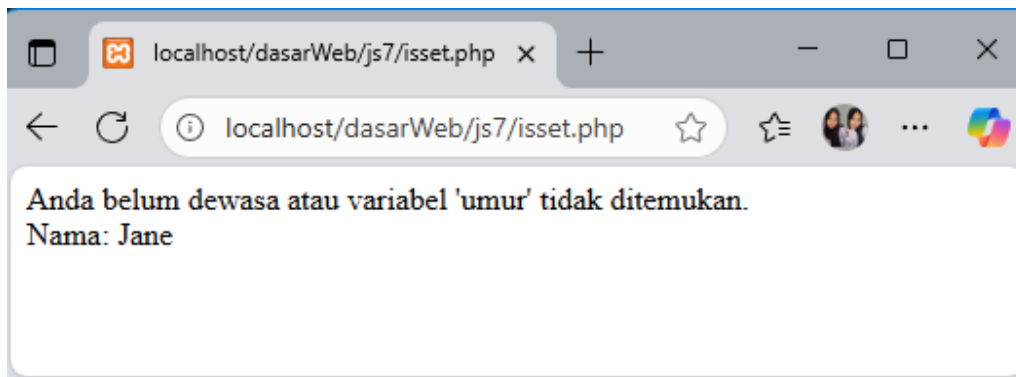
Fungsi `isset()` pada kode tersebut digunakan untuk memeriksa apakah variabel `\$umur` sudah didefinisikan atau belum. Jika variabel `\$umur` ada dan bernilai, maka program akan melanjutkan pemeriksaan usia untuk menentukan apakah seseorang sudah dewasa. Namun, jika variabel `\$umur` belum dibuat atau tidak memiliki nilai, maka bagian `else` akan dijalankan dan menampilkan pesan bahwa variabel tersebut tidak ditemukan. Dengan begitu, `isset()` membantu mencegah error saat variabel belum dideklarasikan.

Soal no 1.2



The image shows a code editor with the following PHP code:

```
$data = array("nama" => "Jane", "usia" => 25);  
if (isset($data["nama"])) {  
    echo "Nama: " . $data["nama"];  
} else {  
    echo "Variabel 'nama' tidak ditemukan dalam array.<br>";  
}  
?>
```

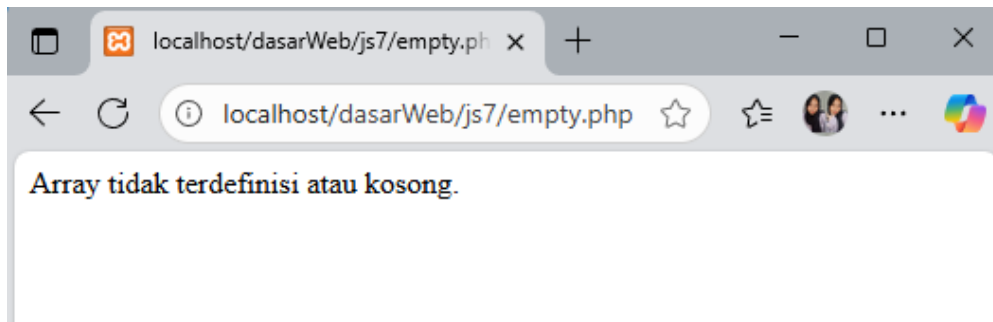


Fungsi `isset()` pada kode tersebut digunakan untuk mengecek apakah elemen dengan key "nama" ada di dalam array `$data`. Jika elemen tersebut ada, maka program akan menampilkan nilai dari "nama", yaitu "Jane". Jika tidak ada, maka program akan menampilkan pesan bahwa variabel 'nama' tidak ditemukan. Dengan `isset()`, kita bisa memastikan bahwa data yang ingin diakses benar-benar ada sehingga mencegah error saat program dijalankan.

Praktikum 2. Function empty

Soal no 2.1

```
empty.php > ...
1  <?php
2  $myArray = array();
3  if (empty($myArray)){
4      echo "Array tidak terdefinisi atau kosong.";
5  } else {
6      echo "Array terdefinisi dan tidak kosong.";
7  }
```

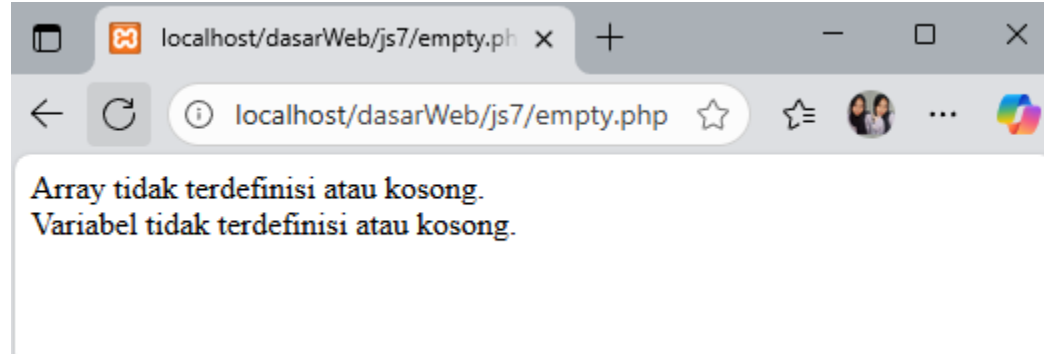


Fungsi `empty()` pada kode tersebut digunakan untuk memeriksa apakah variabel `$myArray` kosong atau belum memiliki nilai. Jika array tersebut kosong atau belum didefinisikan, maka program akan menampilkan pesan "Array tidak terdefinisi atau kosong." Sebaliknya, jika array berisi data,

maka akan ditampilkan pesan bahwa array terdefinisi dan tidak kosong. Dengan demikian, `empty()` berguna untuk memastikan suatu variabel memiliki isi sebelum digunakan.

Soal no 2.2

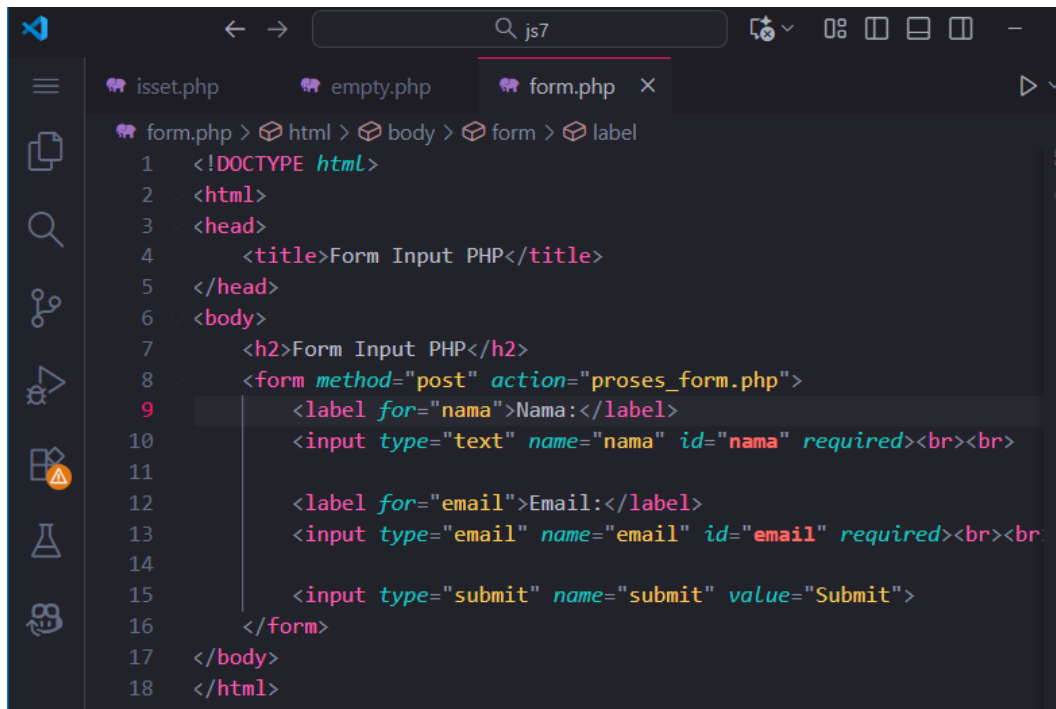
```
if (empty($nonExistentVar)){  
    echo "Variabel tidak terdefinisi atau kosong."  
} else {  
    echo "Variabel terdefinisi dan tidak kosong."  
}
```



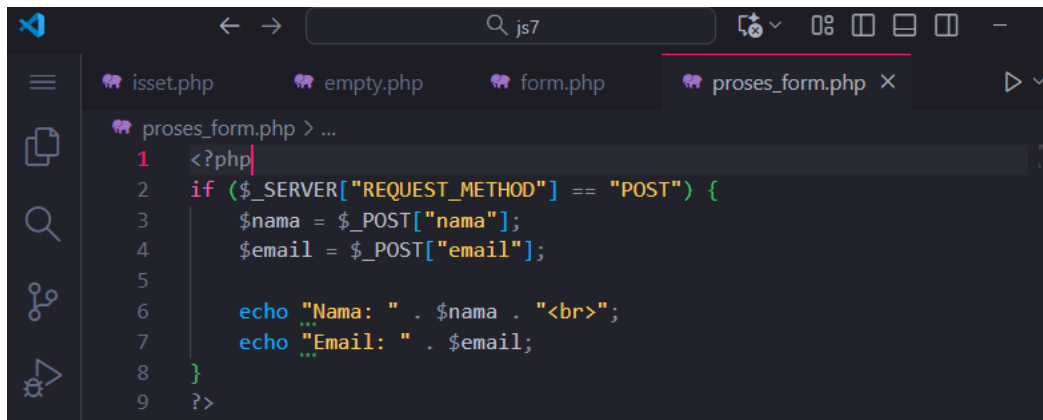
Fungsi `empty()` digunakan untuk mengecek apakah variabel tidak memiliki nilai atau belum didefinisikan. Jika variabel kosong atau belum dibuat, maka akan muncul pesan “Variabel tidak terdefinisi atau kosong.” Dengan begitu, `empty()` membantu mencegah error saat variabel belum diisi.

Praktikum Bagian 3 : Form Input PHP

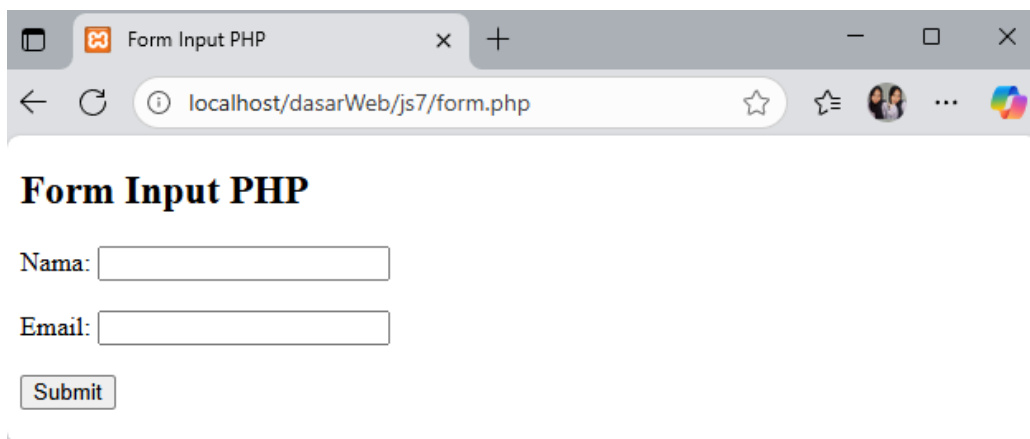
Soal no 3.1



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Form Input PHP</title>
5 </head>
6 <body>
7   <h2>Form Input PHP</h2>
8   <form method="post" action="proses_form.php">
9     <label for="nama">Nama:</label>
10    <input type="text" name="nama" id="nama" required><br><br>
11
12    <label for="email">Email:</label>
13    <input type="email" name="email" id="email" required><br><br>
14
15    <input type="submit" name="submit" value="Submit">
16  </form>
17 </body>
18 </html>
```



```
1 <?php
2 if ($_SERVER["REQUEST_METHOD"] == "POST") {
3   $nama = $_POST["nama"];
4   $email = $_POST["email"];
5
6   echo "Nama: " . $nama . "<br>";
7   echo "Email: " . $email;
8 }
9 ?>
```



Form Input PHP

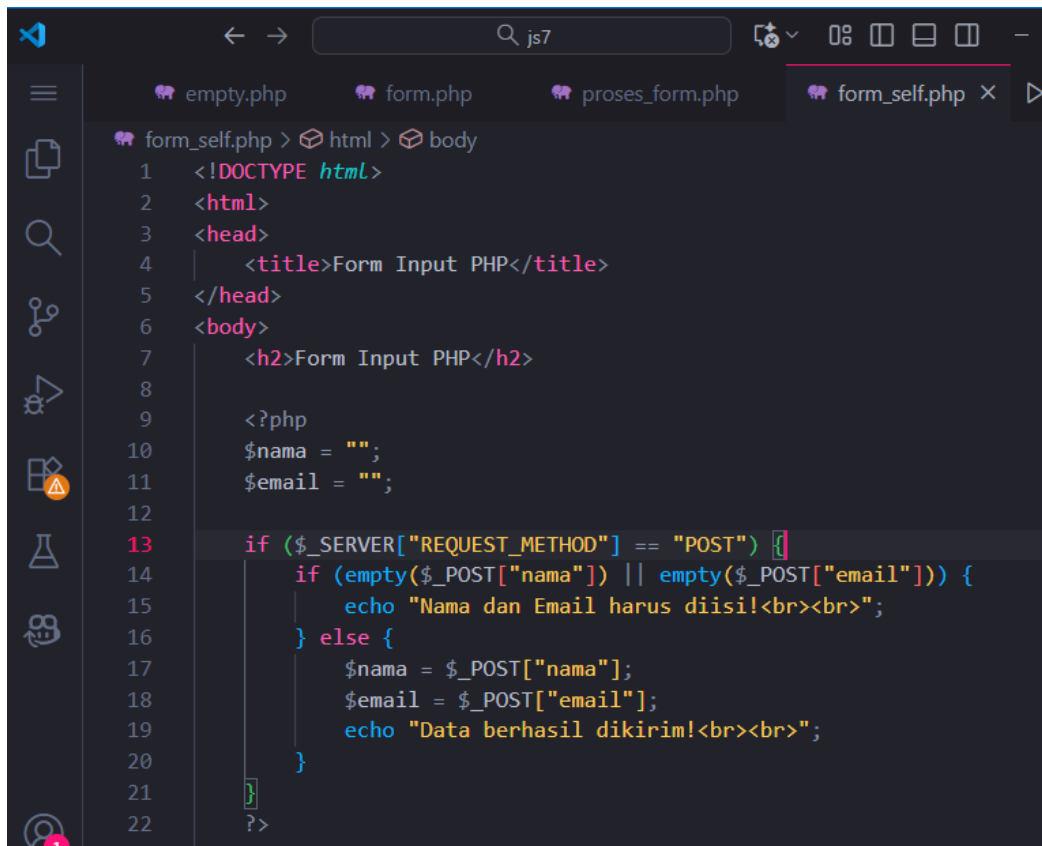
Nama:

Email:

Form pada kode tersebut digunakan untuk mengirim data dari pengguna, yaitu nama dan email, ke server menggunakan metode **POST**. Saat tombol **Submit** diklik, data akan dikirim ke file

proses_form.php untuk diproses lebih lanjut. Pada bagian PHP, data yang dikirim akan ditangkap menggunakan \$_POST dan kemudian ditampilkan kembali ke layar. Dengan form ini, pengguna bisa berinteraksi dan mengirim informasi ke server secara dinamis.

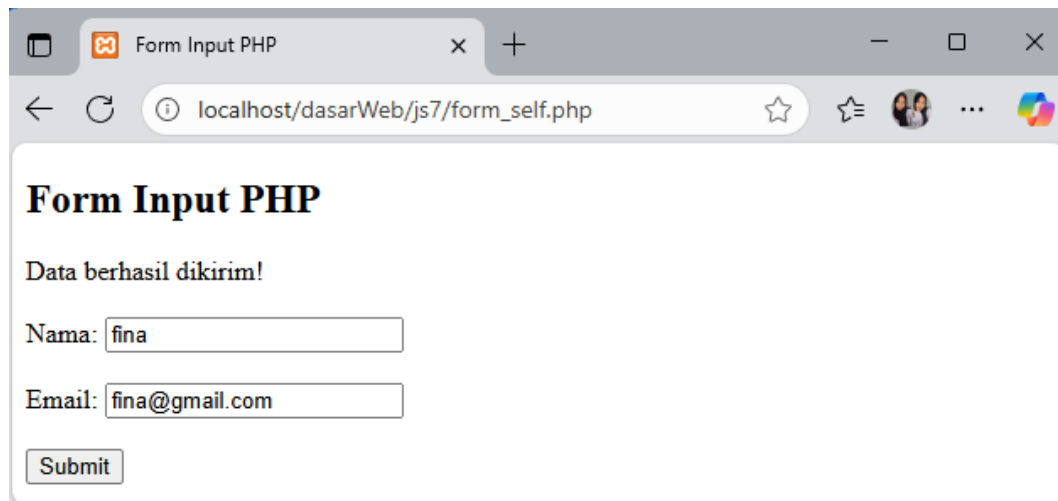
Soal no 3.2



```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Form Input PHP</title>
5 </head>
6 <body>
7   <h2>Form Input PHP</h2>
8
9   <?php
10     $nama = "";
11     $email = "";
12
13     if ($_SERVER["REQUEST_METHOD"] == "POST") {
14       if (empty($_POST["nama"]) || empty($_POST["email"])) {
15         echo "Nama dan Email harus diisi!<br><br>";
16       } else {
17         $nama = $_POST["nama"];
18         $email = $_POST["email"];
19         echo "Data berhasil dikirim!<br><br>";
20       }
21     }
22   ?>
```



```
23
24 <form method="post" action="<?php echo htmlspecialchars(string: $_SERVER["PHP_SELF"]); ?>
25   <label for="nama">Nama:</label>
26   <input type="text" name="nama" id="nama" value="<?php echo $nama; ?>"><br><br>
27
28   <label for="email">Email:</label>
29   <input type="email" name="email" id="email" value="<?php echo $email; ?>"><br><br>
30
31   <input type="submit" name="submit" value="Submit">
32 </form>
33 <?php
34 <?php
35
```



Form Input PHP

Data berhasil dikirim!

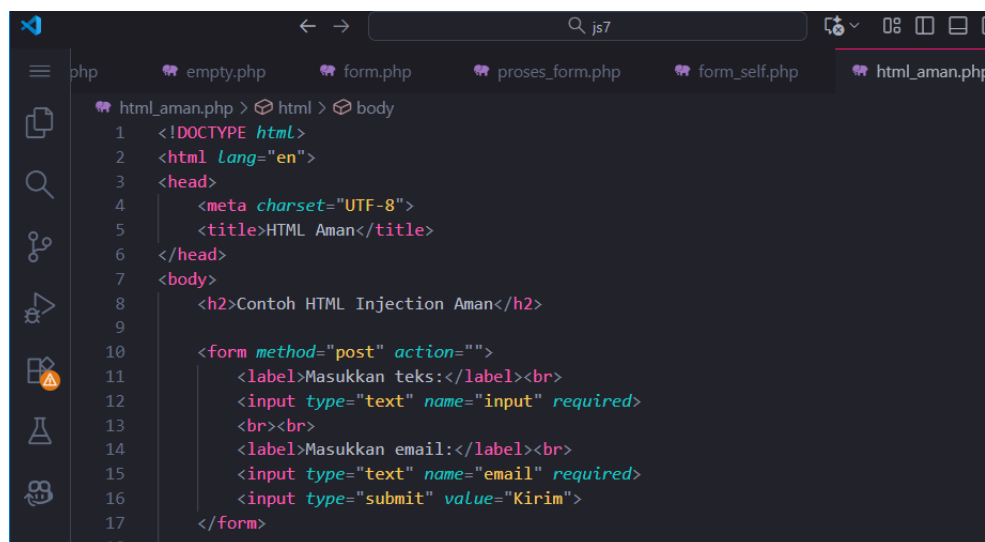
Nama:

Email:

Form pada kode tersebut digunakan untuk mengirim data ke halaman yang sama menggunakan metode POST. Program akan memeriksa apakah kolom nama dan email sudah diisi dengan menggunakan fungsi `empty()`. Jika salah satu kosong, muncul pesan peringatan, tetapi jika lengkap, akan muncul pesan “Data berhasil dikirim!”. Penggunaan `htmlspecialchars($_SERVER["PHP_SELF"])` juga berfungsi untuk mencegah serangan XSS dengan memastikan form aman dari input berbahaya.

Praktikum 4 : HTML Injection

Soal no 4.1



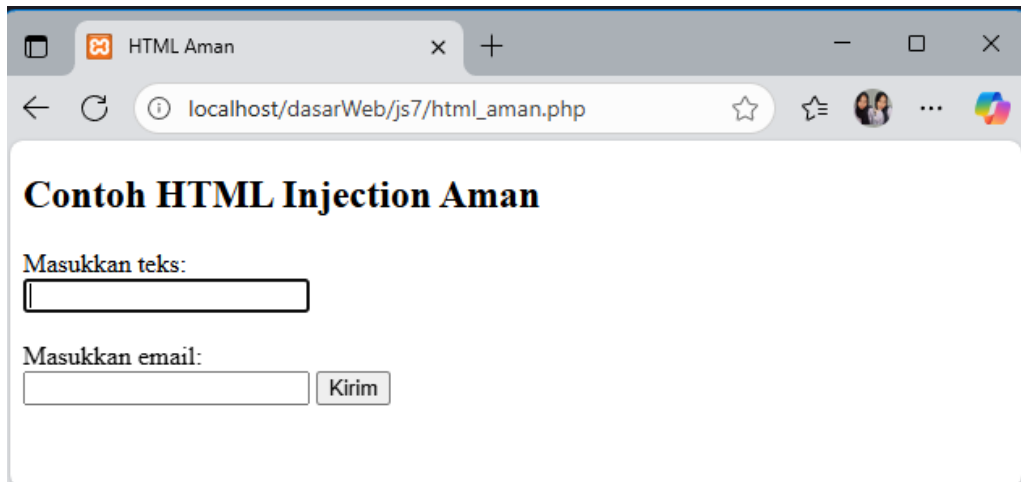
```
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <title>HTML Aman</title>
6 </head>
7 <body>
8   <h2>Contoh HTML Injection Aman</h2>
9
10  <form method="post" action="">
11    <label>Masukkan teks:</label><br>
12    <input type="text" name="input" required>
13    <br><br>
14    <label>Masukkan email:</label><br>
15    <input type="text" name="email" required>
16    <input type="submit" value="Kirim">
17  </form>
```



Form ini mengirimkan data teks dan email ke server menggunakan metode POST (yaitu action kosong dikirim ke halaman yang sama). Karena nilai input belum disanitasi di sisi server, ada risiko HTML injection/XSS jika langsung ditampilkan, jadi harus diproses dengan `htmlspecialchars()` dan validasi server-side.

Soal no 4.2

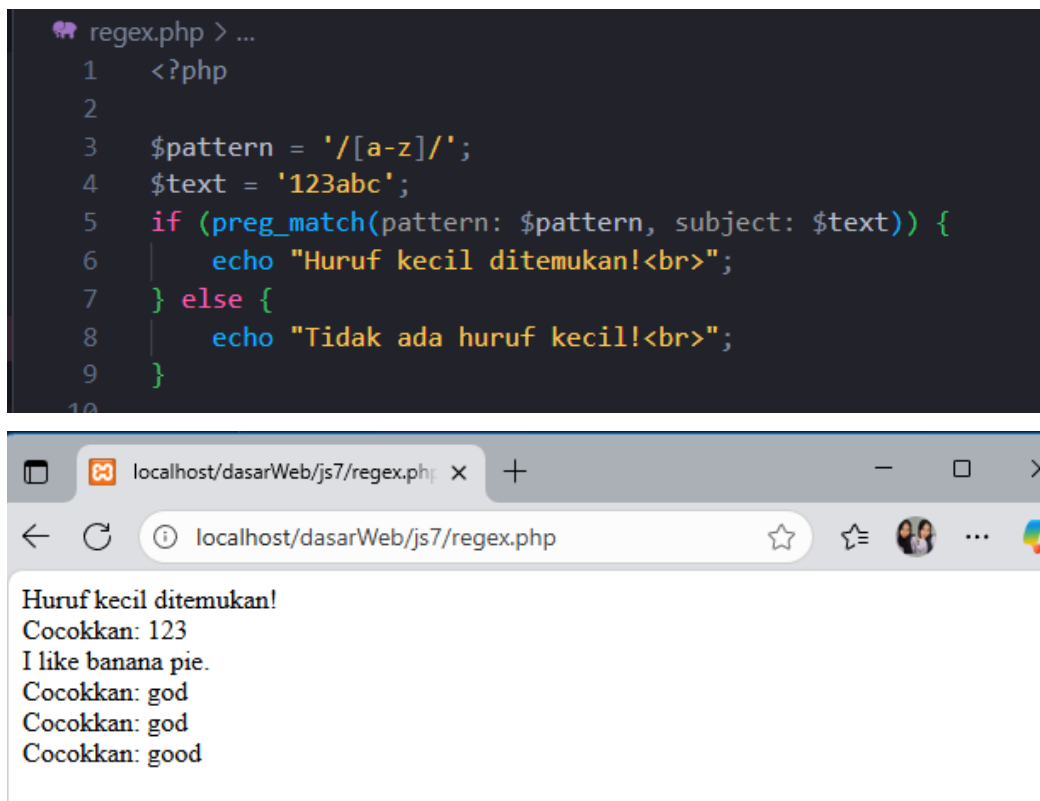
```
if ($_SERVER["REQUEST_METHOD"] == "POST") {  
    $input = isset($_POST['input']) ? $_POST['input'] : '';  
    $input = htmlspecialchars(string: $input, flags: ENT_QUOTES, encoding: 'UTF-8');  
    $email = isset($_POST['email']) ? trim(string: $_POST['email']) : '';  
    if ($email === '') {  
        echo "<p style='color:red;'>Email kosong.</p>";  
    } elseif (filter_var(value: $email, filter: FILTER_VALIDATE_EMAIL)) {  
        $email_safe = htmlspecialchars(string: $email, flags: ENT_QUOTES, encoding: 'UTF-8');  
        echo "<p>Alamat email valid: $email_safe</p>";  
    } else {  
        echo "<p style='color:red;'>Alamat email tidak valid!</p>";  
    }  
    echo "<p>Hasil input yang aman: $input</p>";  
}  
?>
```

Setelah ditambahkan kode tersebut, form menjadi lebih aman karena input pengguna difilter dan divalidasi sebelum ditampilkan. Fungsi `htmlspecialchars()` mencegah terjadinya HTML injection, sedangkan `filter_var()` memastikan format email yang dimasukkan benar. Selain itu, pengecekan dengan `isset()` dan `trim()` membantu mencegah error saat data kosong atau tidak diisi.

Praktikum 5 : Penggunaan Regex Pada PHP

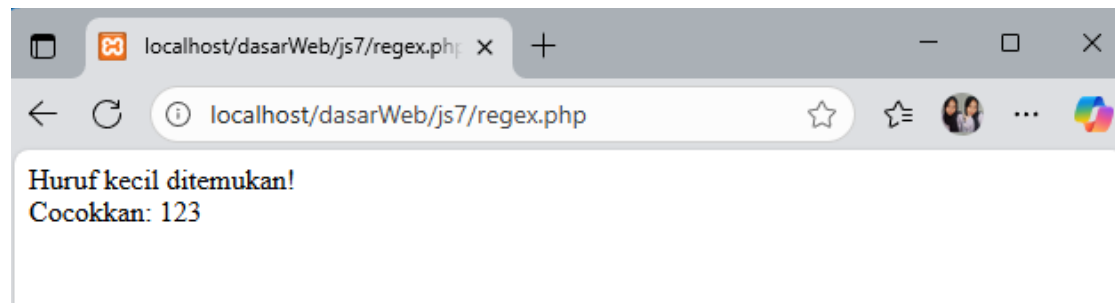
Soal no 5.1



Setelah ditambahkan kode tersebut, program berfungsi untuk memeriksa apakah terdapat huruf kecil pada teks menggunakan regex dengan `preg_match()`. Pola `/[a-z]/` berarti mencari minimal satu huruf kecil dalam string yang diberikan. Jika ditemukan, program menampilkan pesan “Huruf kecil ditemukan!”, dan jika tidak, akan menampilkan “Tidak ada huruf kecil!”.

Soal no 5.2

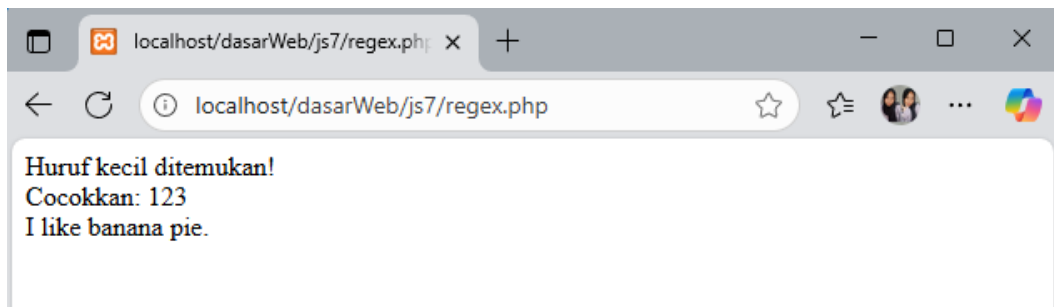
```
// (5.2)
$pattern = '/[0-9]+/';
$text = 'abc123xyz';
if (preg_match(pattern: $pattern, subject: $text, matches: &$matches)) {
    echo "Cocokkan: " . $matches[0] . "<br>";
} else {
    echo "Tidak ada angka yang cocok!<br>";
}
```



Kode ini mencari deretan angka (`[0-9]+`) di dalam teks `'abc123xyz'`. Karena ada angka **123**, maka akan ditampilkan “**Cocokkan: 123**”.

Soal no 5.3

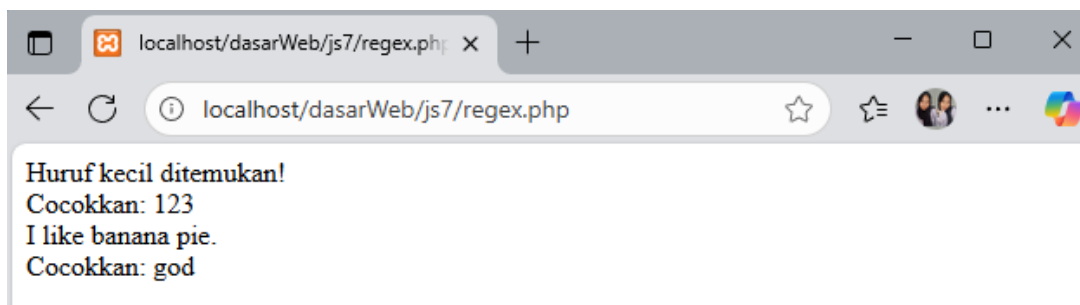
```
20 // (5.3)
21 $pattern = '/apple/';
22 $replacement = 'banana';
23 $text = 'I like apple pie.';
24 $new_text = preg_replace(pattern: $pattern, replacement: $replacement, subject: $text);
25 echo $new_text . "<br>";
26
```



Kode ini mengganti kata apple dengan banana dalam teks 'I like apple pie.'. Hasilnya menjadi "I like banana pie."

Soal no 5.4

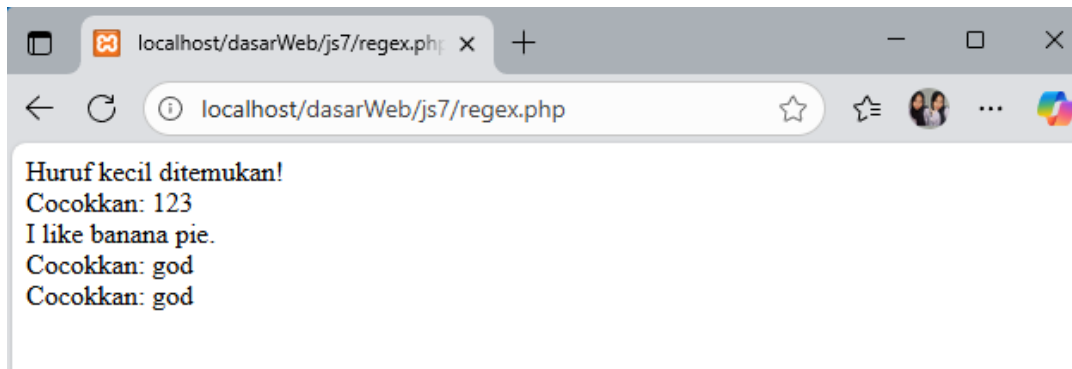
```
// (5.4)
$pattern = '/go*d/';
$text = 'god is good.';
if (preg_match(pattern: $pattern, subject: $text, matches: &$matches)) {
    echo "Cocokkan: " . $matches[0] . "<br>";
} else {
    echo "Tidak ada yang cocok!<br>";
}
```



Pola `/go*d/` berarti mencari kata yang diawali "g", diikuti nol atau lebih huruf "o", lalu "d". Dari teks 'god is good.', yang pertama cocok adalah "god", jadi itu yang ditampilkan.

Soal no 5.5

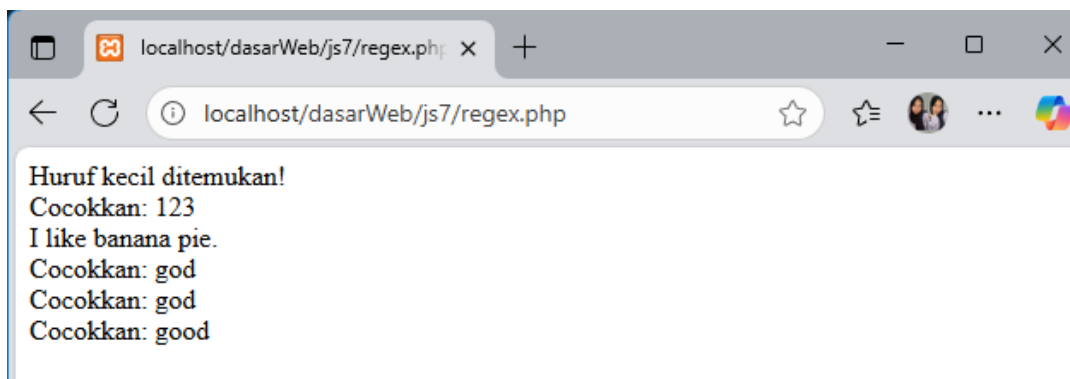
```
// (5.5)
$pattern = '/go?d/';
$text = 'god is good.';
if (preg_match(pattern: $pattern, subject: $text, matches: &$matches)) {
    echo "Cocokkan: " . $matches[0] . "<br>";
} else {
    echo "Tidak ada yang cocok!<br>";
}
```



Pola `/go?d/` berarti mencari kata “gd” atau “god” (karena `o?` artinya “o” boleh ada atau tidak). Hasilnya cocok dengan “god” dari teks 'god is good.'.

Soal no 5.6

```
//(5.6)
$pattern = '/go{2,4}d/';
$text = 'god good goood gooooood';
if (preg_match($pattern, $text, $matches)) {
    echo "Cocokkan: " . $matches[0] . "<br>";
} else {
    echo "Tidak ada yang cocok!<br>";
}
```



Pola `/go{2,4}d/` mencari kata “g” diikuti 2–4 huruf “o”, lalu “d”. Pada teks 'god good goood gooooood', yang pertama cocok adalah “good”, jadi itu yang ditampilkan.

Praktikum 6 : Form Lanjut

Soal no 6.1

```
php      proses_form.php      form_self.php      html_aman.php      regex.php      form_lanjut.php
form_lanjut.php > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <title>Contoh Form dengan PHP</title>
5  </head>
6  <body>
7  |   <h2>Form Contoh:</h2>
8  |   <form method="POST" action="proses_lanjut.php">
9  |       <label for="buah">Pilih Buah :</label><br>
10 |       <select name="buah" id="buah">
11 |           <option value="apel">Apel</option>
12 |           <option value="pisang">Pisang</option>
13 |           <option value="mangga">Mangga</option>
14 |           <option value="jeruk">Jeruk</option>
15 |       </select><br><br>
16 |
17 |       <label>Pilih Warna Favorit:</label><br>
18 |       <input type="checkbox" name="warna[]" value="merah"> Merah<br>
19 |       <input type="checkbox" name="warna[]" value="biru"> Biru<br>
20 |       <input type="checkbox" name="warna[]" value="hijau"> Hijau<br><br>
21 |
22 |       <label>Pilih Jenis Kelamin:</label><br>
23 |       <input type="radio" name="jenis_kelamin" value="laki-laki"> Laki-laki<br>
24 |       <input type="radio" name="jenis_kelamin" value="perempuan"> Perempuan<br><br>
25
```

```
25
26     <input type="submit" value="Submit">
27 </form>
28 </body>
29 </html>
30
31 <?php
32 if ($_SERVER["REQUEST_METHOD"] == "POST") {
33     $selectedBuah = $_POST['buah'];
34
35     if (isset($_POST['warna'])) {
36         $selectedWarna = $_POST['warna'];
37     } else {
38         $selectedWarna = [];
39     }
40
41     $selectedJenisKelamin = $_POST['jenis_kelamin'];
42
43     echo "Anda memilih buah : " . $selectedBuah . "<br>";
44
```

```
44
45     if (!empty($selectedWarna)) {
46         echo "Warna favorit Anda : " . implode(separator: ", ", array: $selectedWarna) . "<br>";
47     } else {
48         echo "Anda tidak memilih warna favorit.<br>";
49     }
50
51     echo "Jenis kelamin Anda : " . $selectedJenisKelamin;
52
53
54
```

Form Contoh:

Pilih Buah :
Mangga ▼

Pilih Warna Favorit:
☐ Merah
☒ Biru
☐ Hijau

Pilih Jenis Kelamin:
☐ Laki-laki
☒ Perempuan

Submit

Saat kode dijalankan, pengguna akan melihat form dengan pilihan buah, warna favorit, dan jenis kelamin. Setelah tombol Submit diklik, data yang dipilih dikirim ke server menggunakan metode POST dan ditampilkan kembali di halaman. Program menampilkan buah yang dipilih, daftar warna favorit (jika ada), serta jenis kelamin yang dipilih. Jika pengguna tidak memilih warna, maka akan muncul pesan “Anda tidak memilih warna favorit.”

Soal no 6.2

```

form.php  form_self.php  html_aman.php  regex.php  form_lanjut.php  form_ajax.php
form_ajax.php > html > body > form#myForm > select#buah
1  <!DOCTYPE html>
2  <html>
3  <head>
4    <title>Contoh Form dengan PHP dan jQuery</title>
5    <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6  </head>
7  <body>
8    <h2>Form Contoh</h2>
9    <form id="myForm">
10     <label for="buah">Pilih Buah:</label>
11     <select name="buah" id="buah">
12       <option value="apel">Apel</option>
13       <option value="pisang">Pisang</option>
14       <option value="mangga">Mangga</option>
15       <option value="jeruk">Jeruk</option>
16     </select>
17
18     <br><br>
19
20     <label>Pilih Warna Favorit:</label><br>
21     <input type="checkbox" name="warna[]" value="merah"> Merah<br>
22     <input type="checkbox" name="warna[]" value="biru"> Biru<br>
23     <input type="checkbox" name="warna[]" value="hijau"> Hijau<br>
24

```

```

24
25     <br>
26
27     <label>Pilih Jenis Kelamin:</label><br>
28     <input type="radio" name="jenis_kelamin" value="laki-laki"> Laki-laki<br>
29     <input type="radio" name="jenis_kelamin" value="perempuan"> Perempuan<br>
30
31     <br>
32
33     <input type="submit" value="Submit">
34 </form>
35
36 <div id="hasil">
37 </div>
38
39 <script>
40     $(document).ready(function () {
41         $("#myForm").submit(function (e) {
42             e.preventDefault();
43
44             var formData = $("#myForm").serialize();
45
46             $.ajax({
47                 url: "proses_lanjut.php",
48                 type: "POST",
49                 data: formData,
50                 success: function (response) {
51                     $("#hasil").html(response);
52                 }
53             });
54         });
55     });
56 </script>
57 </body>
58 </html>
59

```

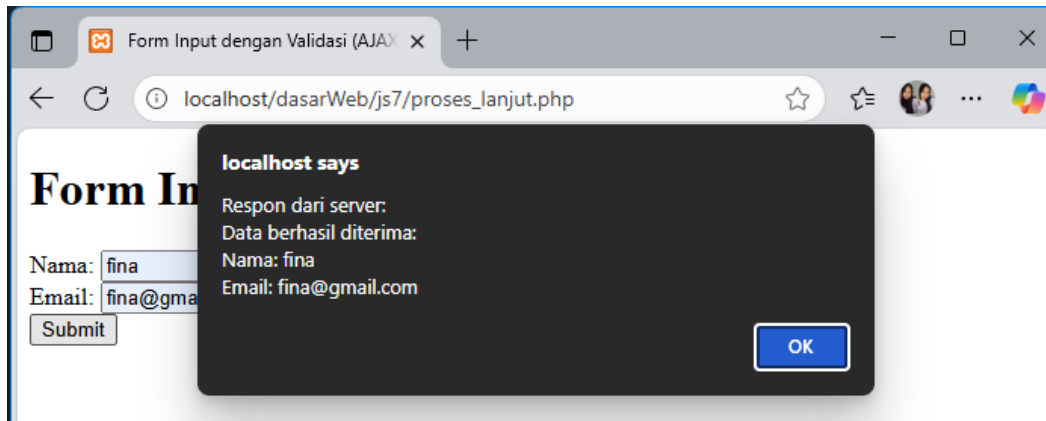
Form Input dengan Validasi (AJAX)

localhost/dasarWeb/js7/proses_lanjut.php

Form Input dengan Validasi (AJAX)

Nama:

Email:



Setelah kode dijalankan, pengguna akan mengisi form seperti memilih buah, warna favorit, dan jenis kelamin. Ketika tombol Submit ditekan, data form tidak langsung memuat ulang halaman karena dikendalikan oleh jQuery AJAX. Data dikirim ke file proses_lanjut.php, dan hasil respons dari file tersebut akan langsung ditampilkan di dalam elemen `

Praktikum 7 : Validasi Form

Soal no 7.1

```

form_validasi.php > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Form Input dengan Validasi</title>
5  </head>
6  <body>
7      <h1>Form Input dengan Validasi</h1>
8      <form method="post" action="proses_validasi.php">
9          <label for="nama">Nama:</label>
10         <input type="text" id="nama" name="nama">
11         <br>
12         <label for="email">Email:</label>
13         <input type="text" id="email" name="email">
14         <br>
15         <input type="submit" value="Submit">
16     </form>
17 </body>
18 </html>
19

```



```
20 <?php
21 if ($_SERVER["REQUEST_METHOD"] == "POST") {
22     $nama = $_POST["nama"];
23     $email = $_POST["email"];
24     $errors = array();
25
26     if (empty($nama)) {
27         $errors[] = "Nama harus diisi.";
28     }
29
30     if (empty($email)) {
31         $errors[] = "Email harus diisi.";
32     } elseif (!filter_var($email, filter: FILTER_VALIDATE_EMAIL)) {
33         $errors[] = "Format email tidak valid.";
34     }
35
36     if (count($errors) > 0) {
37         foreach ($errors as $error) {
38             echo "<p style='color:red;'>$error</p>";
39         }
40     } else {
41         echo "Data berhasil diterima dengan data: Nama = $nama, Email = $email";
42     }
43 }
44 ?>
```

Form Input dengan Validasi

localhost/dasarWeb/js7/form_validasi.php

Form Input dengan Validasi

Nama:

Email:

localhost/dasarWeb/js7/proses_val...

localhost/dasarWeb/js7/proses_validasi.php

Data berhasil diterima: Nama: fina Email: fina@gmail.com

Setelah kode dijalankan, pengguna dapat mengisi nama dan email pada form, lalu mengirimkannya ke server. Program akan melakukan validasi untuk memastikan kolom nama dan email tidak kosong serta format email sesuai. Jika ada kesalahan, pesan error berwarna merah akan ditampilkan di halaman. Namun, jika semua input valid, maka data akan ditampilkan dengan pesan bahwa input berhasil diterima.

Soal no 7.2

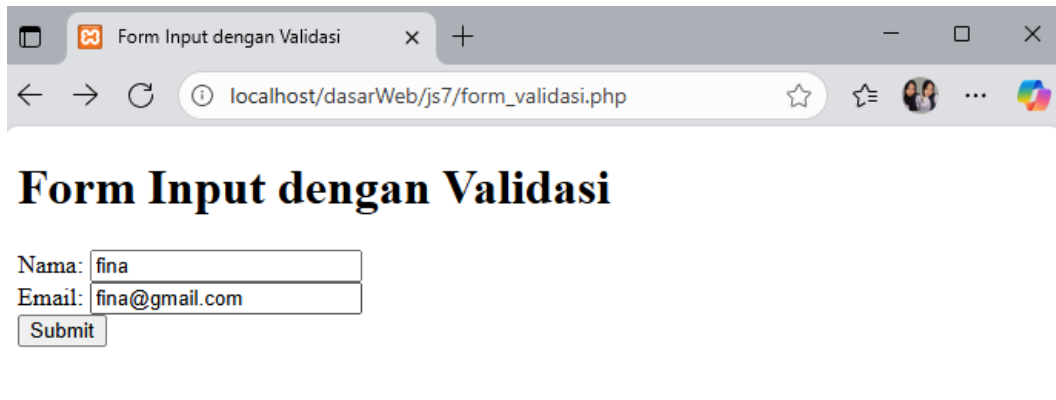
```
html_aman.php  regex.php  form_lanjut.php  form_ajax.php  form_validasi.php
form_validasi.php > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Form Input dengan Validasi</title>
5      <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6  </head>
7  <body>
8      <h1>Form Input dengan Validasi</h1>
9      <form id="myForm" method="post" action="proses_validasi.php">
10         <label for="nama">Nama:</label>
11         <input type="text" id="nama" name="nama">
12         <span id="nama-error" style="color: red;"></span><br>
13
14         <label for="email">Email:</label>
15         <input type="text" id="email" name="email">
16         <span id="email-error" style="color: red;"></span><br>
17
18         <input type="submit" value="Submit">
19     </form>
20
```

```
<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("");
            }
        });
    });
</script>
```

```
1
2  if (!valid) {
3      event.preventDefault(); // Menghentikan pengiriman form jika valida
4  }
5  });
6  });
7  </script>
8  </body>
9  </html>
```



Setelah kode dijalankan, pengguna dapat mengisi form nama dan email lalu menekan tombol Submit. Jika salah satu kolom kosong, pesan error berwarna merah akan muncul di bawah kolom tersebut tanpa memuat ulang halaman. Validasi ini dilakukan di sisi client menggunakan jQuery, sehingga lebih cepat dan interaktif. Form hanya akan dikirim ke server jika semua input sudah diisi dengan benar.

Soal no 7.3

```
form_validasi.php > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>Form Input dengan Validasi dan AJAX</title>
5      <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6  </head>
7  <body>
8      <h1>Form Input dengan Validasi dan AJAX</h1>
9      <form id="myForm" method="post" action="proses_validasi.php">
10         <label for="nama">Nama:</label>
11         <input type="text" id="nama" name="nama">
12         <span id="nama-error" style="color:red;"></span><br>
13
14         <label for="email">Email:</label>
15         <input type="text" id="email" name="email">
16         <span id="email-error" style="color:red;"></span><br>
17
18         <input type="submit" value="Submit">
19     </form>
20
21     <div id="hasil"></div>
22
```

```

<script>
$(document).ready(function () {
    $("#myForm").submit(function (event) {
        event.preventDefault(); // Mencegah form reload halaman

        var nama = $("#nama").val();
        var email = $("#email").val();
        var valid = true;

        // Validasi input kosong
        if (nama === "") {
            $("#nama-error").text("Nama harus diisi.");
            valid = false;
        } else {
            $("#nama-error").text("");
        }
    }

```

```

39
40         if (email === "") {
41             $("#email-error").text("Email harus diisi.");
42             valid = false;
43         } else {
44             $("#email-error").text("");
45         }
46
47         // Jika validasi lolos, kirim data via AJAX
48         if (valid) {
49             $.ajax({
50                 url: "proses_validasi.php",
51                 type: "POST",
52                 data: { nama: nama, email: email },
53                 success: function (response) {
54                     $("#hasil").html(response); // Tampilkan hasil dari PHP
55                 },
56                 error: function () {
57                     $("#hasil").html("<p style='color:red;'>Terjadi kesalahan s
58                 }

```

Form Input dengan Validasi dan AJAX

Nama:

Email:

Data berhasil diterima: Nama: fina Email: fina@gmai

Setelah kode AJAX ditambahkan, form tidak lagi melakukan reload halaman saat dikirim. Data dikirim ke file `proses_validasi.php` secara asinkron, dan hasilnya langsung ditampilkan pada

elemen `<div id="hasil">`. Hal ini membuat proses validasi dan pengiriman data terasa lebih cepat dan responsif tanpa berpindah halaman.

Soal no 7.4

```
html_aman.php  regex.php  form_lanjut.php  form_ajax.php  form_validasi.php
form_validasi.php > ...
1  <!DOCTYPE html>
2  <html>
3  <head>
4  <title>Form Input dengan Validasi</title>
5  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
6  </head>
7  <body>
8  <h1>Form Input dengan Validasi</h1>
9  <form id="myForm" method="post" action="proses_validasi.php">
10  <label for="nama">Nama:</label>
11  <input type="text" id="nama" name="nama">
12  <span id="nama-error" style="color: red"></span><br>
13
14  <label for="email">Email:</label>
15  <input type="text" id="email" name="email">
16  <span id="email-error" style="color: red"></span><br>
17
18  <label for="password">Password:</label>
19  <input type="password" id="password" name="password">
20  <span id="password-error" style="color: red"></span><br>
21
22  <input type="submit" value="Submit">
23  </form>
24
```

```
25  <script>
26  $(document).ready(function() {
27  $("#myForm").submit(function(event) {
28  var nama = $("#nama").val();
29  var email = $("#email").val();
30  var password = $("#password").val();
31  var valid = true;
32
33  if (nama === "") {
34  $("#nama-error").text("Nama harus diisi.");
35  valid = false;
36  } else {
37  $("#nama-error").text("");
38  }
39
40  if (email === "") {
41  $("#email-error").text("Email harus diisi.");
42  valid = false;
43  } else {
44  $("#email-error").text("");
45  }
46
```

```

46
47     if (password.length < 8) {
48         $("#password-error").text("Password minimal 8 karakter.");
49         valid = false;
50     } else {
51         $("#password-error").text("");
52     }
53
54     if (!valid) {
55         event.preventDefault(); // menghentikan pengiriman form jika validasi gagal
56     }
57     });
58 });
59 </script>
60 </body>
61 </html>

```

```

proses_validasi.php > ...
1  <?php
2  if ($_SERVER["REQUEST_METHOD"] == "POST") {
3      $nama = $_POST["nama"];
4      $email = $_POST["email"];
5
6      if (empty($nama) || empty($email)) {
7          echo "Data tidak lengkap!";
8      } else {
9          echo "Data berhasil diterima:\nNama: $nama\nEmail: $email";
10     }
11 }
12 ?>

```

Form Input dengan Validasi

Nama:

Email:

Password: Password minimal 8 karakter.

localhost/dasarWeb/js7/proses_va x

localhost/dasarWeb/js7/proses_validasi.php

Data berhasil diterima: Nama: fina Email: fina@gmail.com

Setelah ditambahkan kode validasi password, sistem kini memeriksa panjang karakter password sebelum form dikirim. Jika pengguna memasukkan password kurang dari 8 karakter, pesan kesalahan akan muncul di bawah kolom password dan form tidak akan dikirim. Validasi ini membantu mencegah pengiriman data yang tidak sesuai standar keamanan. Proses ini membuat form lebih interaktif dan memberikan umpan balik langsung kepada pengguna sebelum data dikirim ke server.