



Jurusan Teknologi Informasi Politeknik Negeri Malang
Jobsheet-7: PHP - Form Processing
Mata Kuliah Desain dan Pemrograman Web
Pengampu: Tim Ajar Desain dan Pemrograman Web
Oktober 2023

Mata Kuliah : Desain Dan Pemrograman Web
Program Studi : D4 – Teknik Informatika
Semester : 3

Kelas : 2A
NIM : 244107020090
Nama : Fadhil Taufiqurrachman
Jobsheet Ke- : 7

Praktikum 1. Function Iset

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 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>Sintak isset() untuk memeriksa keberadaan variabel \$umur. Karena variabel \$umur tidak pernah dibuat dan didefinisikan dalam skrip, isset(\$umur) mengembalikan false. Akibatnya, program mengabaikan kondisi if dan langsung menjalankan blok else, mencetak pesan bahwa variabel tidak ditemukan</p>
5	<p>Tambahkan isi dari file isset.php 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> <hr/> <p>Nama : Jane</p> <p>Kode ini menunjukkan bahwa isset() tidak hanya untuk variabel, tapi juga sangat berguna untuk memeriksa apakah sebuah <i>key</i> ada di dalam <i>array</i>. Pengecekan isset(\$data["nama"]) mengembalikan nilai true karena <i>key</i> "nama" telah didefinisikan di dalam array \$data. Sehingga, program menjalankan blok if dan mencetak nilai yang terkait dengan <i>key</i> tersebut yakni ("Jane").</p>

Praktikum 2. Function empty

Langkah	Keterangan
1	Buat satu file baru bernama empty.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam empty.php
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 localhost/dasarWeb/empty.php. Apa yang anda pahami dari penggunaan empty pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 2.1)</p> <p>Array Tidak Terdefenisi Atau Kosong.</p> <p>Kode empty() akan memeriksa apakah suatu variabel "kosong". Dalam kode tersebut, \$myArray adalah sebuah array yang telah didefinisikan tetapi tidak memiliki elemen (array()). empty() menganggap array kosong sebagai true, sehingga kondisi if terpenuhi dan pesan "Array tidak terdefinisi atau kosong." Ditampilkan pada browser.</p>
4	<p>Tambahkan isi dari file empty.php 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 localhost/dasarWeb/empty.php buat hasil tidak menjadi 1 baris, jadi tampilan dari echo harus terpisah. Apa yang anda pahami dari penggunaan empty pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 2.2)</p> <p>Variabel Tidak Terdefenisi Atau Kosong.</p> <p>Perbedaan utama empty() dari isset() adalah empty() akan mengembalikan true untuk variabel yang tidak terdefinisi (dan juga untuk nilai seperti 0, "", false, null). Pada kode sebelumnya, \$myArray ada tapi kosong, jadi empty() = true. Pada contoh kedua, \$nonExistentVar tidak ada sama sekali, dan empty() juga menganggapnya true tanpa menyebabkan error.</p>

Praktikum Bagian 3 : Form Input PHP

Langkah	Keterangan
1	Buat satu file baru bernama form.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam form.php
2	<pre><!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	Buat satu file baru bernama proses_form.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam proses_form.php
	<pre><?php if (\$_SERVER["REQUEST_METHOD"] == "POST") { \$nama = \$_POST["nama"]; \$email = \$_POST["email"]; echo "Nama: " . \$nama . "
"; echo "Email: " . \$email; }</pre>
4	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 : Fadhil Taufiqurrachman

Email : fadhiltaufiqurrachman08@gmail.com

Submit

Nama : Fadhil Taufiqurrachman

Email : fadhiltaufiqurrachman08@gmail.com

File form.php bertugas menampilkan formulir kepada pengguna . Saat pengguna menekan "Submit" , browser mengemas data menggunakan atribut name sebagai kunci dan mengirimkannya ke proses_form.php untuk diproses. File proses_form.php kemudian menggunakan *array superglobal* \$_POST untuk "membuka" paket data tersebut dan menampilkannya.

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)

Form Input PHP

Nama :

Form Input PHP

Data Berhasil Disimpan!

Nama :

Kode ini menunjukkan *self-processing form*, di mana satu file menangani logika tampilan (GET) dan pemrosesan (POST). Ini dicapai dengan mengatur action formulir ke \$_SERVER["PHP_SELF"]. Kode ini juga mengenalkan validasi server sederhana, jika input nama kosong, formulir ditampilkan kembali dengan pesan kesalahan (\$namaErr).

Praktikum 4 : HTML Injection

Langkah	Keterangan
1	Buat satu file baru bernama html_aman.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam html_aman.php
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 localhost/dasarWeb/html_aman.php
4	Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 4.1) Fungsi htmlspecialchars() adalah fungsi keamanan yang menyeleksi input pengguna dengan mengubah karakter HTML khusus (seperti < dan >) menjadi entitas HTML (< dan >). Sehingga browser tidak lagi melihatnya sebagai tag <script> yang harus dieksekusi, melainkan hanya sebagai teks biasa yang ditampilkan apa adanya. Berikut adalah kodennya.
	<pre><!DOCTYPE html> <html> <head> <title>Uji Coba Keamanan HTML</title> </head> <body> <h2>Formulir Input</h2> <p>Coba Masukkan Teks Yang Mengandung Spesial Characters, Misalnya &lt;script&gt;alert('Tes')&lt;/script&gt;</p> <form method="POST" action="html_aman.php"> <label for="input">Masukkan Teks:</label> <input type="text" id="input" name="input" size="50"> <input type="submit" value="Kirim"> </form> <hr> <?php if (isset(\$_POST['input'])) { \$input = \$_POST['input']; \$input_aman = htmlspecialchars(\$input, ENT_QUOTES, 'UTF-8'); echo "<h3>Data Asli Yang Diterima : </h3>"; echo "<pre>" . \$input_aman . "</pre>"; echo "<h3>Data Ditampilkan Langsung (Tidak Aman) : </h3>"; echo \$input; echo "<h3>Data Ditampilkan Setelah 'htmlspecialchars' (Aman) : </h3>"; echo \$input_aman; } ?> </body> </html></pre>

Hasil Pada browser.

Formulir Input

Coba Masukkan Teks Yang Mengandung Spesial Characters, Misalnya <script>alert('Tes')</script>

Masukkan Teks:

Data Asli Yang Diterima :

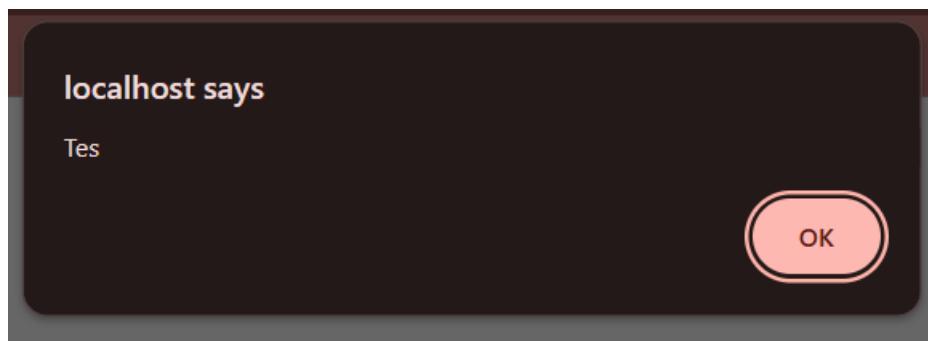
```
<script>alert('Tes')</script>
```

Data Ditampilkan Langsung (Tidak Aman) :

Data Ditampilkan Setelah 'htmlspecialchars' (Aman) :

```
<script>alert('Tes')</script>
```

- Di bagian "Data Ditampilkan Langsung (Tidak Aman)": Halaman akan menjalankan kode tersebut, dan sebuah kotak peringatan (alert) bertuliskan 'tes' akan muncul di browser Anda. Ini terjadi karena browser menerima teks <script>... dan menganggapnya sebagai perintah JavaScript yang harus dieksekusi.



- Di bagian "Data Ditampilkan Setelah 'htmlspecialchars' (Aman)": Halaman akan menampilkan teks persis seperti yang Anda ketik, yaitu <script>alert('tes')</script>. Tidak ada kotak peringatan yang muncul.

5 Ketikkan kode tambahan pada langkah 6 di dalam html_aman.php

```
// 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
}
```

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

8

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

Berikut adalah tambahan kode HTML dan PHPnya.

```
.....
<h2>Formulir Pengecekan Email</h2>
<p>Masukkan Email Untuk Divalidasi</p>

<form method="POST" action="html_aman.php">
    <label for="email">Email :</label>
    <input type="text" id="email" name="email" size="50">
    <input type="submit" value="Cek Validasi">
</form>

<hr>
.....
if (isset($_POST['email'])) {
    $email = $_POST['email'];
    $email_aman = htmlspecialchars($email, ENT_QUOTES, 'UTF-8');
    echo "<p>Email Yang Anda Masukkan : <strong>" .
    $email_aman . "</strong></p>";

    if (filter_var($email, FILTER_VALIDATE_EMAIL)) {
        echo "<p><strong>Format Email Tersebut
Valid.</strong></p>";

    } else {
        echo "<p><strong>Format Email Tersebut Tidak
Valid.</strong></p>";
    }
}
.....
```

Formulir Pengecekan Email

Masukkan Email Untuk Divalidasi

Email :

Email Yang Anda Masukkan : **fadhiltaufiqurrachman08@gmail.com**

Format Email Tersebut Valid.

Email Yang Anda Masukkan : **fadhiltaufiqurrachman08@gmail.com**

Format Email Tersebut Tidak Valid.

Kode secara keseluruhan akan menampilkan formulir untuk memasukkan email. Setelah pengguna mengirimkan email (via POST), halaman akan memuat ulang, lalu skrip PHP akan berjalan dan mengambil email yang dikirim, mengecek menggunakan fungsi filter_var() dengan FILTER_VALIDATE_EMAIL, dan akhirnya menampilkan pesan di bawah formulir yang menyatakan apakah format email tersebut valid atau tidak.

Praktikum 5 : Penggunaan Regex Pada PHP

Langkah	Keterangan
1	Buat satu file baru bernama regex.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam regex.php
2	<pre>\$pattern = '/[a-z]/'; // Cocokkan huruf kecil. \$text = 'This is a Sample Text.'; if (preg_match(\$pattern, \$text)) { echo "Huruf kecil ditemukan!"; } else { echo "Tidak ada huruf kecil!"; }</pre>
3	Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php
4	Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.1) <p><u>Huruf Kecil Ditemukan!</u></p> <p>Kode ini menggunakan preg_match untuk mencari apakah ada minimal satu huruf kecil pada pola /[a-z]/ di dalam string \$text. Karena string "This is a Sample Text." mengandung huruf kecil, preg_match menemukan kecocokan dan mengembalikan true, sehingga mencetak pesan "Huruf kecil ditemukan!".</p>
5	Ketikkan kode tambahan pada langkah 6 di dalam regex.php
6	<pre>\$pattern = '/[0-9]+/'; // Cocokkan satu atau lebih digit. \$text = 'There are 123 apples.'; if (preg_match(\$pattern, \$text, \$matches)) { echo "Cocokkan: " . \$matches[0]; } else { echo "Tidak ada yang cocok!"; }</pre>
7	Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php
8	Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.2) <p><u>Cocokkan : 123</u></p> <p>Sama seperti sebelumnya namun ini angka. Kuantifier + membuatnya cocok dengan "123" sekaligus (bukan hanya "1"). Parameter opsional ketiga \$matches pada preg_match sangat berguna untuk mengambil teks yang cocok dan \$matches[0] akan berisi teks lengkap yang cocok dengan pola yang telah ditentukan</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>Fungsi preg_replace berbeda dengan preg_match yang hanya mencari, preg_replace berfungsi untuk mencari dan mengganti. Ia mencari semua kemunculan pola (/apple/) dan menggantinya dengan string pengganti (banana), lalu mengembalikan string baru hasil penggantian tersebut.</p>
13	Ketikkan kode tambahan pada langkah 14 di dalam regex.php
14	<pre>\$pattern = '/go*d/'; // Cocokkan "god", "good", "gooood", dll. \$text = 'god is good.'; if (preg_match(\$pattern, \$text, \$matches)) { echo "Cocokkan: " . \$matches[0]; } else { echo "Tidak ada yang cocok!"; }</pre>
15	Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php
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>Pola '/go*d/' menggunakan kuantifier * , yang berarti nol atau lebih dari karakter 'o'. Dalam string "god is good.", preg_match akan menemukan kecocokan pertamanya, yaitu "god" (memiliki satu 'o', yang termasuk dalam kriteria "nol atau lebih"). Hasil "god" ini disimpan ke \$matches[0] dan dicetak ditampilkan pada browser.</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>Kode :</p> <pre>\$pattern = '/Go?d/'; \$text = 'God Is Good'; if (preg_match(\$pattern, \$text, \$matches)) { echo "Cocokkan : " . \$matches[0]; } else { echo "Tidak Ada Yang Cocok!"; }</pre>

	<p>Hasil :</p> <p>Cocokkan : God</p> <p>Quantifier ? merupakan nol atau satu. Pola '/go?d/' berarti 'g' diikuti oleh 'o' sebanyak nol atau satu kali, lalu 'd'. Pola ini cocok dengan "god" (satu 'o') pada string "god is good.". Jika stringnya adalah "good" (dua 'o'), pola ini tidak akan menemukan kecocokan.</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) Kita akan mengeceknya dengan 2 teks.</p> <p>Teks 1 :</p> <pre>\$pattern = '/Go{2,3}d/'; \$text1 = 'God Is Good.'; if (preg_match(\$pattern, \$text1, \$matches1)) { echo "Cocokkan : " . \$matches1[0]; } else { echo "Tidak Ada Yang Cocok!"; }</pre> <p>Cocokkan : Good</p> <p>Teks 2 :</p> <pre>echo "<hr>"; \$text2 = 'God Is Gooood.'; if (preg_match(\$pattern, \$text2, \$matches2)) { echo "Cocokkan : " . \$matches2[0]; } else { echo "Tidak Ada Yang Cocok!"; }</pre> <p>Tidak Ada Yang Cocok!</p> <p>Pola diubah menggunakan kuantifier {n,m}, yang artinya n adalah minimal dan m adalah maksimal. Pada kode saya menggunakan pola {2,3} berarti minimal 2 kali, maksimal 3 kali dari karakter sebelumnya yakni ('o').</p> <ul style="list-style-type: none"> • Pada \$text1 = 'god is good.': preg_match melewatkkan "god" (hanya 1 'o') tetapi menemukan "good" (2 'o'), yang sesuai dengan rentang {2,3}. • Pada \$text2 = 'god is gooood.': preg_match melewatkkan "god" (1 'o'). Ia juga melewatkkan "gooood" (4 'o'), karena 4 berada di luar rentang {2,3}.

Praktikum 6 : Form Lanjut

Langkah	Keterangan
1	Buat satu file baru bernama <code>form_lanjut.php</code> di dalam folder dasarWeb. 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	Simpan file , kemudian buka browser dan jalankan/refresh <code>localhost/dasarWeb/form_lanjut.php</code>

4

Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 6.1)

Form Contoh

Pilih Buah :

Pilih Warna Favorite :

- Merah
- Biru
- Hijau

Pilih Jenis Kelamin

- Laki-Laki
- Perempuan

Anda Memilih Buah : Apel

Warna Favorite Anda : Merah, Biru, Hijau

Jenis Kelamin Anda : Laki-laki

Pengguna akan menginput pada form_lanjut.php, kemudian data akan dikirim ke proses_lanjut.php. Yang dimana akan dilakukan pengecekan dari setiap kategori input. Jadi sangat ditekankan untuk menggunakan isset() dan empty() untuk mengecek apakah variabel tersebut masing-masing ada datanya, karena jika tidak ada yang dipilih, variabel tersebut tidak akan terkirim, yang dapat menyebabkan error di PHP.

5

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

```

<!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 (AJAX)

Pilih Buah :

Pilih Warna Favorit :

Merah

Biru

Hijau

Pilih Jenis Kelamin :

Laki-laki

Perempuan

Anda Memilih Buah : Pisang

Warna Favorite Anda : Biru

Jenis Kelamin Anda : Perempuan

Ketika saya mengisi formulir dan mengklik Submit, halaman tidak me-refresh atau berkedip., data langsung berubah menjadi output dari PHP dan muncul pada browser.

Ini merupakan formulir menggunakan konsep AJAX dengan bantuan jQuery. Perbedaan utamanya adalah pengalaman pengguna. Halaman tidak perlu dimuat ulang. Skrip jQuery mencegat pengiriman formulir , mengirim data di "latar belakang" ke server , lalu mengambil respons dari server dan menampilkannya di halaman yang sama.

Praktikum 7 : Validasi Form

Langkah	Keterangan
1	Buat satu file baru bernama <code>form_validasi.php</code> di dalam folder dasarWeb. 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></pre> <pre><?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>
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) Jika Input kosong :

Form Input Dengan Validasi

Nama :
Email :

Nama Harus Diisi.
Email Harus Diisi.

Jika input tidak kosong :

Form Input Dengan Validasi

Nama :
Email :

Data berhasil dikirim: Nama = Fadhil Taufiqurrachman, Email = fadhiltaufiqurrachman08@gmail.com

Ini adalah validasi sisi server (server-side validation). Pengguna mengirim data, lalu server (PHP) memeriksanya. Jika ada yang salah, server mengirimkan kembali halaman baru yang berisi daftar kesalahan. Jika benar, server mengirimkan pesan sukses. Proses ini memerlukan refresh halaman setiap kali submit, baik sukses maupun gagal.

5

Kembangkan file bernama `form_validasi.php` di dalam folder `dasarWeb`. Ketikkan kode pada langkah 6 di dalam `form_validasi.php`

```

<!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">
        <span id="nama-error" style="color: red;"></span><br>

        <label for="email">Email:</label>
        <input type="text" id="email" name="email">
        <span id="email-error" style="color: red;"></span><br>

        <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>

```

6

7

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

8

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

Form Input Dengan Validasi (jQuery)

Nama : Nama Harus Diisi.
 Email : Email Harus Diisi.

Jika input tidak kosong :

Form Input Dengan Validasi (jQuery)

Nama : Fadhil Taufiqurrachman
Email : fadhiltaufiqurrachman08@gmail.com

Submit

Data berhasil dikirim: Nama = Fadhil Taufiqurrachman, Email = fadhiltaufiqurrachman08@gmail.com

Jika klik Submit dengan formulir kosong, halaman tidak me-refresh. Sebaliknya, pesan "Nama harus diisi." dan "Email harus diisi." langsung muncul di sebelah kotak input. Jika saya mengisi data dengan benar dan mengklik "Submit", halaman akan me-refresh dan pindah ke proses_validasi.php yang menampilkan pesan sukses. Ini adalah validasi sisi klien (client-side validation) menggunakan jQuery. Validasi terjadi di browser pengguna *sebelum* data dikirim ke server.

- 9 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)

```
<!DOCTYPE html>
<html>
    <head>
        <title>Form Input dengan Validasi (jQuery + AJAX)</title>
        <script src="https://code.jquery.com/jquery-
3.7.1.min.js"></script>
    </head>
    <body>
        <h1>Form Input Dengan Validasi (jQuery + AJAX)</h1>
        <form id="myForm">
            <label for="nama">Nama : </label>
            <input type="text" id="nama" name="nama">
            <span id="nama-error" style="color: red;"></span><br>

            <label for="email">Email :</label>
            <input type="text" id="email" name="email">
            <span id="email-error" style="color: red;"></span><br>
            <br>

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

        <hr>
        <div id="hasil">
            <!-- Hasil Akan Ditempatkan Di Sini -->
        </div>
        <script>
            $(document).ready(function() {
                $("#myForm").submit(function(event) {
                    event.preventDefault();

                    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) {
            var formData = $("#myForm").serialize();

            $.ajax({
                url: "proses_validasi.php",
                type: "POST",
                data: formData,
                success: function (response) {
                    $("#hasil").html(response);
                }
            });
        });
    
```

Form Input Dengan Validasi (jQuery + AJAX)

Nama :

Email :

Data berhasil dikirim : Nama = Fadhil Taufiqurrachman, Email = fadhiltaufiqurrachman08@gmail.com

Dengan kode terebut, maka Jika validasi klien lolos (valid == true), data dikirim ke proses_validasi.php menggunakan \$.ajax . Halaman tidak me-refresh.

10

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

Kode program pada file proses_validasi.php :

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

        if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
            $errors[] = "Format Email Tidak Valid.";
        }

        if (empty($errors)) {
            echo "Data berhasil dikirim : <br>
                Nama = $nama <br>
                Email = $email <br>
                Password = $password";
        } else {
            foreach ($errors as $error) {
                echo $error . "<br>";
            }
        }
    }
?>
```

Kode program pada file proses_validasi.php :

```
<!DOCTYPE html>
<html>
    <head>
        <title>Form Input dengan Validasi (jQuery + AJAX)</title>
        <script src="https://code.jquery.com/jquery-
3.7.1.min.js"></script>
    </head>
    <body>
        <h1>Form Input Dengan Validasi (jQuery + AJAX)</h1>
        <form id="myForm">
            <label for="nama">Nama : </label>
            <input type="text" id="nama" name="nama">
            <span id="nama-error" style="color: red;"></span><br>

            <label for="email">Email :</label>
            <input type="text" id="email" name="email">
            <span id="email-error" style="color: red;"></span><br>

            <label for="password">Password : </label>
            <input type="password" id="password" name="password">
            <span id="password-error" style="color: red;"></span><br>
            <br>

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

        <hr>
        <div id="hasil">
            <!-- Hasil Akan Ditempatkan Di Sini -->
        </div>
    </body>
</html>
```

```

<script>
    $(document).ready(function() {
        $("#myForm").submit(function(event) {
            event.preventDefault();

            var nama = $("#nama").val();
            var email = $("#email").val();
            var password = $("#password").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 (password === "") {
                $("#password-error").text("Password Harus
Diisi.");
                valid = false;
            } else if (password.length < 8) {
                $("#password-error").text("Password Minimal 8
Karakter.");
                valid = false;
            } else {
                $("#password-error").text("");
            }

            if (valid) {
                var formData = $("#myForm").serialize();

                $.ajax({
                    url: "proses_validasi.php",
                    type: "POST",
                    data: formData,
                    success: function (response) {
                        $("#hasil").html(response);
                    }
                });
            }
        });
    });
</script>
</body>
</html>

```

Jika password kurang dari 8 karakter :

Form Input Dengan Validasi (jQuery + AJAX)

Nama :

Email :

Password :

Password Minimal 8 Karakter.

Jika semua benar :

Form Input Dengan Validasi (jQuery + AJAX)

Nama :

Email :

Password :

Data berhasil dikirim :

Nama = Fadhil Taufiqurrachman

Email = fadhiltaufiqurrachman08@gmail.com

Password = adfghjkmcb