**JOBSHEET 7**

**DESAIN DAN PEMROGRAMAN WEB**

**“PHP dan jQuery”**

Disusun Untuk Memenuhi Tugas Mata Kuliah Desain dan Pemrograman Web

Yang dibimbing oleh Moch. Zawaruddin Abdullah S.ST., M.Kom.



Disusun oleh :

Andreas Gale Dwi Jaya

2B Sistem Informasi Bisnis

2241760033

**POLITEKNIK NEGERI MALANG**

**JURUSAN TEKNOLOGI INFORMASI**

**PROGRAM STUDI SISTEM INFORMASI BISNIS**

**2024**

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

**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

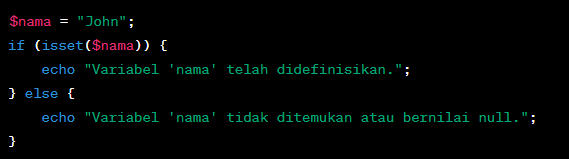
**Perhatian**

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()**:



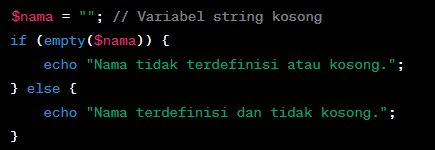
Kode di atas memeriksa apakah variabel **$nama** telah didefinisikan. Jika ya, maka pesan "Variabel 'nama' telah didefinisikan." akan ditampilkan, jika tidak, maka pesan "Variabel 'nama' tidak ditemukan atau bernilai null." akan ditampilkan.

**Praktikum 1. Function Isset**

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buatlah satu file baru di dalam direktori dasarWeb, beri nama isset.php. |
| 2 | Ketikkan ke dalam file isset.php tersebut kode di bawah ini. |
| 3 |  |
| 4 | 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)  Jawab : 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. Jadi, output diatas dikarenakan variable umur yang belum didefinisikan |
| 5 | Tambahkan isi dari file isset.php tersebut dengan kode di bawah ini. |
| 6 | 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)  Jawab : Variabel data telah didefinisikan, sehingga terdapat output Nama :Jane |

**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()**:



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

**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 |  |
| 3 | 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)  Jawab : Mengoutputkan kalimat di atas karena array tidak adda isinya. |
| 4 | Tambahkan isi dari file empty.php tersebut dengan kode di bawah ini. |
| 5 | Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/ empty.php buat hasil tidak menjadi 1 baris, jadi tampilan dari echo harus terpisah. Apa yang anda pahami dari penggunaan empty pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 2.2)  Jawab : Menampilkan output di atas karena variable nonExistenVar tidak didefinisikan. |

**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 |  |
| 3 | Buat satu file baru bernama proses\_form.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam proses\_form.php |
| 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)  Jawab : Kode ini memeriksa apakah halaman ini diakses dengan metode POST. Jika ya, maka ia mengambil data dari formulir (nama dan email), dan kemudian mencetaknya ke dalam halaman web. Jika halaman ini diakses dengan metode selain POST, maka kode di dalam blok if tidak akan dieksekusi.   * Tampilan awal      * Inputan      * Setelah di submit |
| 5 | Buat satu file baru bernama form\_self.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam form\_self.php |
| 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)  Jawab : pemrosesan form akan dijalankan oleh php di dalam form\_self.php itu sendiri   * Tampilan awal      * Input Data      * Setelah di submit |

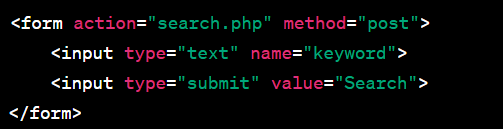
**HTML Injection**

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:



Sekarang, jika aplikasi web tidak memvalidasi input dengan benar, seorang penyerang dapat memasukkan input berbahaya seperti ini:



Jika aplikasi web tidak menghindari atau membersihkan input ini sebelum menampilkannya di halaman hasil pencarian, maka hasilnya akan menjadi seperti ini:



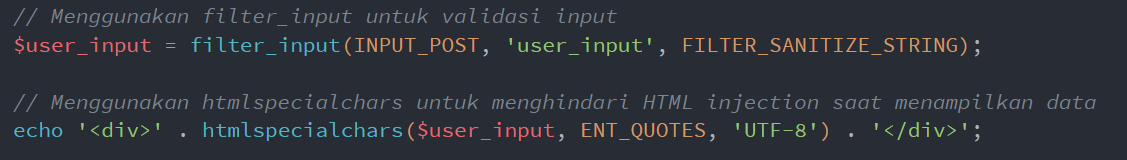
Kode JavaScript berbahaya tersebut akan dieksekusi oleh browser pengguna dan akan memunculkan kotak peringatan "You have been hacked!". Ini adalah contoh HTML injection yang sederhana. Sebagai pengembang web, penting untuk selalu melakukan sanitasi dan validasi input dari pengguna, serta menghindari 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:



Dengan menggunakan langkah-langkah ini, Anda dapat memitigasi risiko HTML injection dalam aplikasi web Anda. Selalu penting untuk memvalidasi input, membersihkan data yang masuk, dan menghindari menampilkan data pengguna tanpa escaping yang benar.

**Praktikum 4 : HTML Injection**

Langkah-langkah Praktikum :

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat satu file baru bernama html\_aman.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam html\_aman.php |
| 2 |  |
| 3 | Tambahkan script lainya 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)  Jawab : htmlspecialchars() digunakan untuk mengonversi karakter khusus yang ada di HTML seperti ,& dll menjadi entitas sehingga program tidak akan mengeksekusinya sebagai kode.   * Tampilan awal      * Input Data |
| 5 | Ketikkan kode tambahan pada langkah 6 di dalam html\_aman.php |
| 6 |  |
| 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)  Jawab : Dengan menggunakan FILTER\_VALIDATE\_EMAIL, program akan melakukan pengecekan apakah Alamat email yang dimasukkan oleh user sudah valid atau belum.   * Tampilan awal      * Input Data      * Setelah di input data |

**Regular Expression (Regex)**

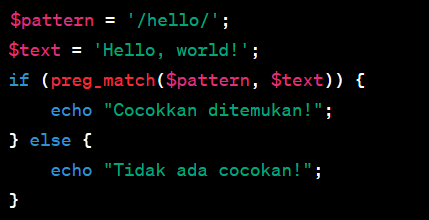
Regex adalah alat yang kuat untuk mencari, mencocokkan, atau memanipulasi teks berdasarkan pola tertentu. Anda dapat menggunakannya untuk berbagai tujuan seperti validasi input, pencarian teks, penggantian teks, dan banyak lagi.

**Basic Regex Patterns:**

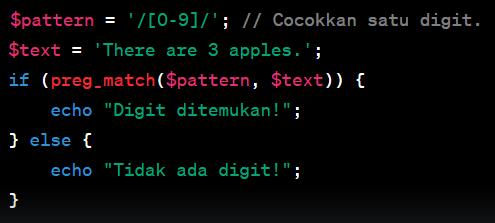
* **Karakter Tunggal**: Karakter apa pun akan cocok dengan dirinya sendiri dalam Regex, kecuali karakter khusus yang harus dihindari dengan karakter pelarian **\**.
  + Contoh: Pola **/a/** akan cocok dengan huruf "a" dalam teks.
* **Karakter Spesifik**: Anda dapat mencocokkan karakter tertentu dengan menyebutkannya dalam pola.
  + 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:
  + **\***: 0 atau lebih kali
  + **+**: 1 atau lebih kali
  + **?**: 0 atau 1 kali
  + **{n}**: Persis n kali
  + **{n,}**: Setidaknya n kali
  + **{n,m}**: Setidaknya n kali, maksimal m kali

**Contoh Penggunaan Regex:**

1. Mencocokan pola



1. Mencocokan karakter set



**Praktikum 5 : Penggunaan Regex Pada PHP**

Langkah-langkah Praktikum :

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat satu file baru bernama regex.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam regex.php |
| 2 |  |
| 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)  Jawab : Fungsi dari preg\_match() adalah untuk mencocokkan nilai dari variable $text dengan pola yang terdapat pada $pattern. Program akan menampilkan pesan “Huruf kecil ditemukan!” apabila menemukan kecocokan pada regex [a-z] dengan setidaknya satu huruf kecil pada text |
| 5 | Ketikkan kode tambahan pada langkah 6 di dalam regex.php |
| 6 |  |
| 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)  Jawab : Pada variable $pattern akan menyimpan pola regex yaitu /[0-9]+/ yang mana akan mencocokkan satu atau lebih karakter 0-9 secara berturut-turut. |
| 9 | Ketikkan kode tambahan pada langkah 10 di dalam regex.php |
| 10 |  |
| 11 | Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php |
| 12 | Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.3)  Jawab : Pada variable $pattern akan menyimpan pola regex yaitu “/apple/” yang mana akan mencocokkan kata “apple” dengan teks. Pada kode pola apple akan diganti menjadi banana. |
| 13 | Ketikkan kode tambahan pada langkah 14 di dalam regex.php |
| 14 |  |
| 15 | Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/regex.php |
| 16 | Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.4)  Jawab : Pada variable $pattern akan menyimpan pola regex yaitu /go\*d/ yang mana akan mencocokkan teks yang memiliki satu atau lebih huruf “o” (atau tidak ada sama sekali) di antara huruf “g” dan “d” |
| 17 | Soal 5.5: Buatlah script langkah 14 dengan variable pattern yang di ubah menggunakan `?` (0 atau 1 kali). Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.5)  Jawab : /go?d/ akan mencocokkan dengan kata “God” pada teks, ? berarti 0 atau 1 kali. |
| 18 | Soal 5.6: Buatlah script langkah 14 dengan variable pattern yang di ubah menggunakan `{n,m}`. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 5.6)  Jawab : Hasil dari kode program di atas akan mencocokkan kata-kata "god", "good", "gooood", atau kata-kata lain yang memiliki 2 atau 3 huruf 'o' di dalamnya. Namun, pada teks "god is good.", hanya kata "god" yang akan dicocokkan karena hanya itu yang memenuhi pola yang ditentukan |

**Praktikum 6 : Form Lanjut**

Langkah-langkah Praktikum :

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat satu file baru bernama form\_lanjut.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam form\_lanjut.php |
| 2 |  |
| 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)  Jawab : Pemrosesan form akan dikerjakan oleh proses\_lanjut.php yang akan mengecek buah yang dipilih, warna favorit, dan jenis kelamin yang dipilih. Seperti hasilnya dibawah ini :   * Tampilan awal      * Input data      * Setelah data di inputkan |
| 5 | Buat satu file baru bernama form\_ajax.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam form\_ajax.php |
| 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)  Jawab : Dengan menggunakan ajax, hasil inputan akan ditampilkan di halaman yang sama dengan halaman pengisian form, seperti dibawah ini :   * Tampilan awal      * Input Data      * Setelah di inputkan data |

**Praktikum 7 : Validasi Form**

Langkah-langkah Praktikum :

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat satu file baru bernama form\_validasi.php di dalam folder dasarWeb. Ketikkan kode pada langkah 2 di dalam form\_validasi.php |
| 2 |  |
| 3 | Simpan file , kemudian buka browser dan jalankan/refresh localhost/dasarWeb/ form\_validasi.php |
| 4 | Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 7.1)  Jawab : Proses validasi form dilakukan setelah data dikirim namun jika terdapat kesalahan validasi, pesan-pesan error akan ditampilkan di halaman. Jika tidak ada kesalahan validasi, data yang dimasukkan akan diproses di proses\_validasi. Dalam kode program di atas, hanya akan menampilkan kembali data yang dimasukkan. Berikut setelah di inputkan datanya dan disubmit. |
| 5 | Kembangkan file bernama form\_validasi.php di dalam folder dasarWeb. Ketikkan kode pada langkah 6 di dalam form\_validasi.php |
| 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)  Jawab : Dari penambahan kode program tersebut, kita melihat implementasi validasi form dengan menggunakan PHP untuk memastikan bahwa input nama dan email tidak kosong, dan bahwa email memiliki format yang valid. |
| 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)  Jawab : Dengan menggunakan jquery dan ajax, data formular akan ditampilkan secara langsung tanpa memerlukan proses refresh halaman. Seperti dibawah ini hasilnya :   * Tampilan awal      * Input data      * Setelah di input data |
| 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)  Jawab : Dengan tambahan kode validasi untuk password, form akan memeriksa apakah password memiliki minimal 8 karakter sebelum mengizinkan pengguna untuk mengirimkan data. Jika password tidak memenuhi persyaratan, pesan error akan ditampilkan dan data tidak akan dikirimkan ke server.   * Tampilan password kurang dari 8 karakter      * Tampilan password lebih atau pas 8 karakter |