**LAPORAN PRAKTIKUM**

**DESAIN DAN PEMROGRAMAN WEB**

**JOBSHEET 7**

****

**Oleh :**

Chiko Abilla Basya 2341720005

**PROGRAM STUDI TEKNIK INFORMATIKA**

**JURUSAN TEKNOLOGI INFORMASI**

**POLITEKNIK NEGERI MALANG**

**2024/2025**

Jurusan Teknologi Informasi Politeknik Negeri Malang

Jobsheet-7: PHP - Form Proccessing

Mata Kuliah Desain dan Pemrograman Web

Tim Ajar Desain dan Pemrograman Web

Oktober 2024

|  |  |
| --- | --- |
| **Praktikum Bagian 1. Function Isset** | |
| **Langkah** | **Keterangan** |
| 1 | Buatlah satu file baru di dalam direktori **minggu7**, beri nama **isset.php.** |
| 2 | Ketikkan ke dalam file **isset.php** kode di bawah ini. |
| 3 |  |
| 4 | Simpan file tersebut, kemudian buka browser dan jalankan **localhost/minggu7/ isset.php**. |
|  | Apa yang anda pahami dari penggunaan isset pada file tersebut. Catat di bawah ini pemahaman anda. (Pertanyaan No. 1)      isset($umur) memeriksa apakah variabel $umur sudah di-set atau belum. Jika variabel $umur sudah di-set dan lebih besar dari 18, maka program mencetak "Anda sudah dewasa". Jika tidak, maka program akan mencetak "Anda belum dewasa atau variabel 'umur' tidak ditemukan." |
| 5 | Tambahkan isi dari file isset.php tersebut dengan kode di bawah ini. |
| 6 | Simpan file, kemudian buka browser dan jalankan **localhost/minggu7/isset.php**. Buat hasil tidak menjadi 1 baris, jadi tampilan dari echo harus terpisah. Apa yang anda pahami dari penggunaan isset pada file tersebut? Catat pemahaman anda di bawah ini.  (Pertanyaan No. 2)      isset($data["nama"]) memeriksa apakah kunci 'nama' ada di dalam array $data dan nilainya bukan null. Jika kunci 'nama' ada, maka program mencetak "Nama: Jane" (atau nama yang terdapat dalam array). Jika kunci 'nama' tidak ada atau bernilai null, program akan mencetak "Variabel 'nama' tidak ditemukan dalam array." |

# Praktikum Bagian 2. Function empty

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat satu file baru bernama **empty.php** di dalam folder minggu7. Ketikkan kode pada langkah 2 di dalam **empty.php** |
| 2 |  |
| 3 | Simpan file tersebut, kemudian buka browser dan jalankan **localhost/minggu7/ empty.php**. Apa yang anda pahami dari penggunaan empty pada file tersebut? Catat pemahaman anda di bawah ini.  (Pertanyaan No. 3)      $myArray = array(); mendefinisikan array kosong. empty($myArray) memeriksa apakah array tersebut kosong atau tidak terdefinisi. Jika array kosong, program akan mencetak "Array tidak terdefinisi atau kosong". Jika array berisi elemen, program akan mencetak "Array terdefinisi dan tidak kosong". |
| 4 | Tambahkan isi dari file empty.php tersebut dengan kode di bawah ini. |
| 5 | Simpan file tersebut, kemudian buka browser dan jalankan **localhost/minggu7/ 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 pemahaman anda di bawah ini. (Pertanyaan No. 4)      empty($nonExistentvar) memeriksa apakah variabel $nonExistentvar tidak terdefinisi atau memiliki nilai yang dianggap kosong (seperti null, 0, string kosong, dll). Jika variabel tersebut tidak ada atau kosong, program akan mencetak "Variabel tidak terdefinisi atau kosong." Jika variabel tersebut ada dan memiliki nilai, program akan mencetak "variabel terdefinisi dan tidak kosong." |

**Praktikum Bagian 3. Form Input PHP**

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat satu file baru bernama **form.php**. Ketikkan kode pada langkah 2. |
| 2 |  |
| 3 | Buat satu file baru bernama **proses\_form.php**. Ketikkan kode berikut dalam  **proses\_form.php** |
| 4 | Simpan file tersebut, kemudian buka browser dan jalankan **localhost/minggu7/ proses\_form.php**. Apa yang anda pahami dari penggunaan form pada file tersebut? Catat pemahaman anda di bawah ini. Selanjutnya jalankan **localhost/minggu7/form.php**  Apa yang terjadi? Catat pemahaman anda di bawah ini. (Pertanyaan No. 5)          Form menggunakan metode **POST** (method="post") untuk mengirim data ke file lain yang bernama proses\_form.php melalui atribut action="proses\_form.php". Data ini kemudian dikirim ke server ketika tombol "Submit" ditekan. PHP kemudian mengakses nilai yang dimasukkan dalam form menggunakan variabel **superglobal** $\_POST. Misalnya, $\_POST["nama"] mengambil data yang dimasukkan pada input "Nama" dan $\_POST["email"] untuk data email. Nilai yang diterima dari form kemudian ditampilkan kembali di browser menggunakan fungsi echo. |
| 5 | Buat satu file baru bernama **form\_self.php** di dalam folder **minggu7**. Ketikkan kode di bawah ini. |
|  |  |
| 6 | Simpan file tersebut, kemudian buka browser dan jalankan **localhost/minggu7/ form\_self.php**. Apa yang anda pahami dari penggunaan form pada file tersebut? Catat pemahaman anda di bawah ini. (Pertanyaan No. 6)        Kode ini menciptakan form input dengan validasi sederhana untuk memastikan bahwa tidak mengirimkan nama kosong. Jika nama diisi, data dianggap berhasil disimpan. |

# Praktikum Bagian 4. HTML Injection

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat file baru bernama **html\_aman.php** di dalam folder **minggu7**. Ketikkan kode berikut |
| 2 |  |
| 3 | Buat file dengan nama **form1.php**, isi dengan kode untuk membuat form dengan 1 input teks yang akan diproses pada **html\_aman.php**. |
| 4 | Simpan file, buka browser dan jalankan/refresh **localhost/minggu7/form1.php**  Isikan input teks yang memuat tag html. |
| 5 | Selanjutnya hapus kode **$input = htmlspecialchars($input, ENT\_QUOTES, 'UTF-8');** |
|  | Simpan file, buka browser dan jalankan/refresh **localhost/minggu7/form1.php**  Isikan kembali input teks yang memuat tag html. |
| 6 | Catat di sini apa yang anda amati, berikan penjelasan anda. (Pertanyaan No. 7)          Fungsi htmlspecialchars() berfungsi untuk mengubah karakter khusus menjadi entitas HTML. Misalnya, < menjadi &lt;, > menjadi &gt;, dan & menjadi &amp;. Ini mencegah karakter-karakter tersebut diperlakukan sebagai kode HTML. menghilangkan penggunaan htmlspecialchars() dapat mengakibatkan potensi risiko keamanan yang serius bagi aplikasi web. Oleh karena itu, sangat disarankan untuk selalu melakukan sanitasi input. |
| 7 | Ketikkan kode di bawah ini pada **html\_aman.php** |
| 8 |  |
| 9 | Lengkapi kode program pada bagian if dan else sehingga dapat dijalankan dengan benar. |
| 10 | Pada **form1.php**, tambahkan kode untuk membuat input betupa email yang akan diproses pada  **html\_aman.php**.  Simpan file, buka browser dan jalankan/refresh **localhost/minggu7/form1.php** |
| 11 | Catat di sini apa yang anda amati dari penambahan kode program di atas. (Pertanyaan No. 8)          Kode ini membuat sebuah form yang memungkinkan untuk memasukkan teks dan email. Setelah mengirimkan form, server akan memproses input tersebut. Jika email valid, hasil input ditampilkan. Jika tidak valid atau jika form belum disubmit, akan ditampilkan pesan kesalahan atau informasi sesuai dengan situasi yang terjadi. |

# Praktikum Bagian 5 : Penggunaan Regex Pada PHP

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat satu file baru bernama **regex.php** di dalam folder **minggu7**. Ketikkan kode berikut |
| 2 |  |
| 3 | Simpan file , kemudian buka browser dan jalankan/refresh **localhost/minggu7/regex.php** |
| 4 | Catat di sini apa yang anda amati dari penambahan kode program di atas. (Pertanyaan No. 9)      Kode ini berfungsi untuk memeriksa apakah ada huruf kecil dalam teks. Jika ada, pesan "Huruf kecil ditemukan." akan ditampilkan. Jika tidak ada huruf kecil sama sekali, pesan "Tidak ada huruf kecil!" akan muncul. |
| 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/minggu7/regex.php** |
| 8 | Catat di sini apa yang anda amati dari penambahan kode program di atas. (Pertanyaan No. 10)      Kode ini mencari satu atau lebih digit angka dalam teks 'There are 123 apples.'. Jika ditemukan, angka tersebut akan ditampilkan. Dalam contoh ini, angka **123** ditemukan dan ditampilkan sebagai "Cocokkan: 123". Jika tidak ada angka, pesan "Tidak ada yang cocok!" akan muncul. |
| 9 | Ketikkan kode tambahan berikut di dalam **regex.php** |
| 10 |  |
| 11 | Lengkapi kode program di atas sehingga hasilnya rapi. Simpan file , kemudian buka browser dan jalankan/refresh localhost/minggu7/regex.php |
| 12 | Catat di sini apa yang anda amati dari penambahan kode program di atas. (Pertanyaan No. 11)      Kode ini mencari kata **"apple"** dalam teks dan menggantinya dengan **"banana"**. Setelah itu, hasil penggantian ditampilkan sebagai **"I like banana pie."**. |
| 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/minggu7/regex.php** |
| 16 | Catat di sini apa yang anda amati dari penambahan kode program di atas. (Pertanyaan No. 12)      Kode ini mencari kata-kata seperti **"god"**, **"good"**, atau variasi lain yang memiliki pola "g" diikuti oleh nol atau lebih huruf "o" dan diakhiri dengan "d". Jika ditemukan, kata tersebut ditampilkan, jika tidak, akan ada pesan bahwa tidak ada yang cocok. |
| 17 | Pada script di langkah 14, ubah isi variable pattern dari ‘\*’ menjadi ‘?’.  Simpan file , kemudian buka browser dan jalankan/refresh **localhost/minggu7/regex.php**  Catat di sini apa yang anda amati dari penambahan kode program di atas. (Pertanyaan No. 13)      Dengan **?** pada pola **/go?d/**, kode akan mencocokkan kata **"gd"** (tanpa "o") atau **"god"** (dengan satu "o"). Hasilnya, dalam teks **'god is good.'**, kata **"god"** akan cocok dan ditampilkan. |
| 18 | Ubah script langkah 14 dengan variable pattern **'/[o]{1,3}/'**.  Simpan file , kemudian buka browser dan jalankan/refresh **localhost/minggu7/regex.php**  Catat di sini apa yang anda amati dari penambahan kode program di atas. (Pertanyaan No. 14)      Kode ini mencari satu hingga tiga huruf "o" dalam string **"god is good."**. Dengan pola **/[o]{1,3}/**, kode akan menemukan dan menampilkan kecocokan yang ditemukan. Hasil yang ditampilkan tergantung pada huruf "o" yang ada dalam teks, hasilnya bisa berupa **"Cocokkan: o"** atau **"Cocokkan: oo"**. |

**Praktikum Bagian 6 : Form Lanjut**

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat satu file baru bernama **form\_lanjut.php** di dalam folder **minggu7**. Ketikkan kode pada langkah 2 di dalam **form\_lanjut.php** |
| 2 |  |
|  | Buat satu file baru bernama **proses\_lanjut.php** di dalam folder **minggu7**. Ketikkan kode berikut |
| 4 | Simpan file, kemudian buka browser dan jalankan/refresh  **localhost/minggu7/form\_lanjut.php** |
| 5 | Catat di sini apa yang anda amati dari penambahan kode program di atas. (Pertanyaan No. 15)          Kode ini membuat sebuah form HTML yang memungkinkan untuk memilih buah, warna favorit, dan jenis kelamin. Setelah form disubmit, PHP memproses data yang dikirimkan, menampilkan pilihan di web, dan menangani situasi di mana tidak ada warna favorit yang dipilih. |
| 6 | Buat satu file baru bernama **form\_ajax.php**. Ketikkan kode berikut. |
|  |  |
| 7 | Simpan file, buka browser dan jalankan/refresh **localhost/minggu7/form\_ajax.php**.  Catat di sini apa yang anda amati dari penambahan kode program di atas. (Pertanyaan No. 16)      Kode ini membuat form interaktif yang memungkinkan untuk memilih buah, warna favorit, dan jenis kelamin. Dengan menggunakan jQuery dan AJAX, saat pengguna mengirimkan form, data dikirim ke server tanpa me-refresh halaman, dan hasilnya ditampilkan secara dinamis di div yang disediakan. |

# Praktikum Bagian 7 : Validasi Form

|  |  |
| --- | --- |
| **Langkah** | **Keterangan** |
| 1 | Buat satu file baru bernama **form\_validasi.php** di dalam folder **minggu7**. Ketikkan kode berikut. |
| 2 | Buat satu file baru bernama **proses\_validasi.php** di dalam folder **minggu7**. Ketikkan kode berikut. |
| 3 | Simpan file, kemudian buka browser dan jalankan/refresh  **localhost/minggu7/form\_validasi.php** |
| 4 | Catat di sini apa yang anda amati dari penambahan kode program di atas. (Pertanyaan No. 17)          Kode ini membuat form input dengan validasi sederhana. diminta untuk mengisi nama dan email, dan ketika form dikirim, PHP akan memvalidasi input tersebut. Jika validasi gagal, pesan kesalahan akan ditampilkan; jika berhasil, data berhasil dikirim dan ditampilkan. |
| 5 | Ubah file **form\_validasi.php** menjadi seperti berikut. |
| 6 |  |
| 7 | Simpan file, buka browser dan jalankan/refresh **localhost/minggu7/form\_validasi.php** |
| 8 | Catat di sini apa yang anda amati dari penambahan kode program di atas. (Pertanyaan No. 18) |
|  | Kode ini membuat form dengan validasi input menggunakan jQuery. Ketika form dikirim, JavaScript memeriksa apakah nama dan email diisi. Jika ada yang kosong, pesan kesalahan ditampilkan dan pengiriman form dihentikan. Jika validasi berhasil, form akan diteruskan ke **proses\_validasi.php**. |
| 9 | Buatlah script langkah 6 dengan menggunakan ajax. Screenshoot kode yang anda buat. Catat di sini apa yang anda amati dari penambahan kode program.  (Pertanyaan No. 19) |
|  | Kode ini menciptakan form interaktif dengan validasi, di mana data yang dimasukkan tidak hanya divalidasi tetapi juga dikirim ke server menggunakan AJAX. Hal ini memungkinkan untuk mendapatkan respons tanpa perlu memuat ulang halaman. |
| 10 | Tambahkan kode untuk validasi password dengan ketentuan minimal 8 karakter dengan menggunakan jQuery dan PHP.  Screen shoot kode dan catat di sini apa yang anda amati dari penambahan kode program tersebut.  (Pertanyaan No. 20)          Kode ini memungkinkan untuk memasukkan nama, email, dan password dengan validasi yang dilakukan sebelum mengirimkan data ke server. Jika data valid, maka pengiriman dilakukan secara asinkron (tanpa me-refresh halaman) menggunakan AJAX. Jika tidak valid, diberi tahu tentang kesalahan yang perlu diperbaiki. |

Link Github : <https://github.com/ChikoAbillaBasya/Desain-Dan-Pemrograman-Web/tree/main/Minggu7>