

Nama : Jogja Tinna Annashobari

NIM : 10493

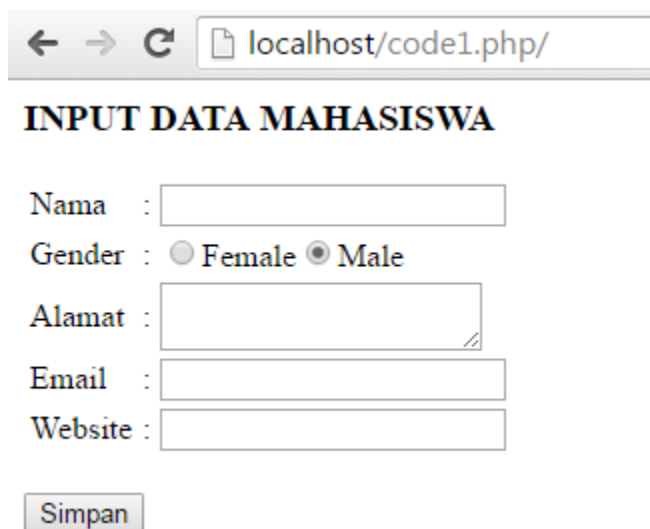
Kelas : Praktikum C2

TUGAS 08

1. Mencoba masing-masing kode (Code 1-3).
2. Capture hasil running masing-masing kode

- Code 1

Tampilan dari code 1 pada saat belum memasukkan data ke form



← → ↻

INPUT DATA MAHASISWA

Nama :

Gender : ☐ Female ☒ Male

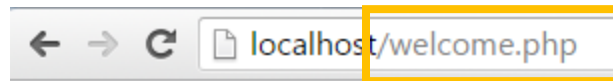
Alamat :

Email :

Website :

- Code 1

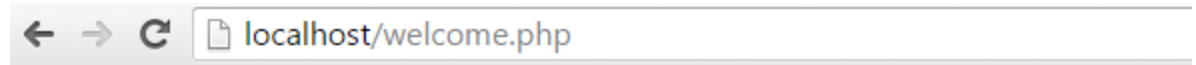
Tampilan dari code 1 pada saat sudah memasukkan data ke form, maka ke halaman welcome.php bukan lagi code1.php



Data Mahasiswa

Nama : oja
Gender : female
Alamat : dongkelan
Email : JTAnnashobari@gmail.com
Website : -

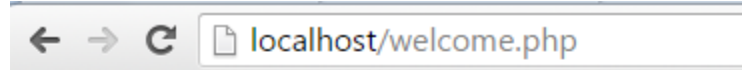
- welcome
Tampilan welcome, jika kita tidak memilih gender



Data Mahasiswa

Nama : oja
Gender : **Notice: Undefined index: gender in C:\xampp\htdocs\welcome.php on line 13**
Alamat : dongkelan
Email : JTAnnashobari@gmail.com
Website : -

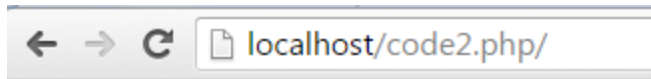
- welcome
Tampilan welcome jika menuliskan alamat email yang salah, tidak terdapat peringatan dan hanya memunculkan apa yang diinputkan user



Data Mahasiswa

Nama : oja
Gender : female
Alamat : dongkelan
Email : JTAnnashobari
Website : -

- Code 2
Tampilan code 2 sebelum



INPUT DATA MAHASISWA

Nama :

Gender : ☒ Female ☐ Male

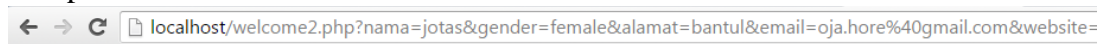
Alamat :

Email :

Website :

- Code 2

Tampilan code 2 sesudah memasukkan data



Data Mahasiswa

Nama : jotas
Gender : female
Alamat : bantul
Email : oja.hore@gmail.com
Website : -

Tampilan code 1 dan code 2 sama, jika tidak memilih gender maka ada notice seperti pada welcome (code1) dan jika salah menuliskan alamat email juga sama pada welcome(code1). Yang membedakan pada url websitenya sendiri, jika code 1 tercantum welcome.php maka code 2 menjadi welcome2.php

- Code 3

Tampilan code 3 sebelum user input data, dan terdapat notice jika user belum menginputkan data



INPUT DATA MAHASISWA

Nama : *

Jenis Kelamin : ☐ Perempuan ☐ Laki-laki *

Alamat :

Email : *

Website :

* harus diisi.

Data Mahasiswa

Nama : **Notice:** Undefined index: nama in C:\xampp\htdocs\code3.php on line 112

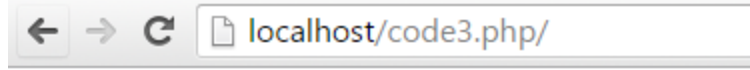
Jenis Kelamin : **Notice:** Undefined index: gender in C:\xampp\htdocs\code3.php on line 117

Alamat : **Notice:** Undefined index: alamat in C:\xampp\htdocs\code3.php on line 122

Email : **Notice:** Undefined index: email in C:\xampp\htdocs\code3.php on line 127

Website : **Notice:** Undefined index: website in C:\xampp\htdocs\code3.php on line 132

- Code 3
Setelah user menginputkan data



Warning: preg_match(): Compilation failed: range out of

INPUT DATA MAHASISWA

Nama	:	<input type="text" value="tinna"/>	*
Jenis Kelamin	:	<input checked="" type="radio"/> Perempuan <input type="radio"/> Laki-laki	*
Alamat	:	<input type="text" value="sewon"/>	
Email	:	<input type="text" value="JTAnnashobari@gmail.com"/>	*
Website	:	<input type="text" value="https://www.dumetschool.co"/>	URL salah
<input type="button" value="Simpan"/>			

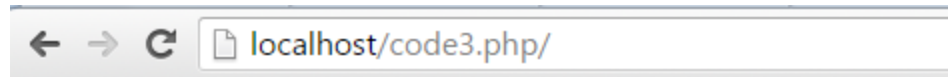
* harus diisi.

Data Mahasiswa

Nama	:	tinna
Jenis Kelamin	:	Perempuan
Alamat	:	sewon
Email	:	JTAnnashobari@gmail.com
Website	:	https://www.dumetschool.com

Terlihat perbedaan antara code 3 dan code 1 dan code 2 dimana pada code 1 dan code 2 jika user telah input data dan klik simpan maka akan ke halaman url yang berbeda. Sedangkan pada code3 data user yang diisikan akan muncul pada satu halaman yang sama dan berada di bawah form

- Code 3



Warning: preg_match(): Compilation failed: range out of order in character

INPUT DATA MAHASISWA

Nama : tinna *

Jenis Kelamin : ☒ Perempuan ☐ Laki-laki *

Alamat : sewon

Email : JTAnnashobari * Format email salah

Website : https://www.dumetschool.com URL salah

* harus diisi.

Data Mahasiswa

Nama : tinna

Jenis Kelamin : Perempuan

Alamat : sewon

Email : JTAnnashobari

Website : https://www.dumetschool.com

Pada code 3 jika user menginputkan format alamat email salah maka ada peringatan disebelahnya, dan jika user menginputkan format website salah maka akan muncul peringatan juga berbeda dengan code 1 dan code 2 tanpa peringatan

- Code 3
Tampilan code 3 jika tidak memilih gender

Warning: preg_match(): Compilation failed: range out of order in character class at offset 64 in C:\

INPUT DATA MAHASISWA

Nama : jeje *

Jenis Kelamin : ☐ Perempuan ☐ Laki-laki * Jenis Kelamin harus diisi

Alamat : jogja

Email : jeje@gmail.com *

Website : - URL salah

* harus diisi.

Data Mahasiswa

Nama : jeje

Jenis Kelamin : Notice: Undefined index: gender in C:\xampp\htdocs\code3.php on line 117

Alamat : jogja

Email : jeje@gmail.com

Website : -

Pada code 3 jika user tidak memilih gender pada form maka pada form terdapat peringatan disebelahnya dan pada data mahasiswa juga ada notice . berbeda dengan code 1 dan code 2, jika user tidak memilih gender pada form hanya ada notice di data mahasiswa tanpa ada peringatan pada formnya.

3. Analisis source code PHP

- Code 1

```
<html>
<body>
<form action="welcome.php" method="post">
<h3>INPUT DATA MAHASISWA</h3>
```

Tulisan `method="POST"` merupakan awal dari sebuah sistem penyimpanan data. Form `method="POST"` itu adalah nama method nya, sistem penyampaian datanya. Untuk yang tulisan `action="welcome.php"`, maksudnya adalah data yang telah diinputkan akan dikirim ke file yang bernama `welcome.php`. Jadi, membuat file dengan nama tujuan.php untuk menampung data yang tadi dimasukkan.

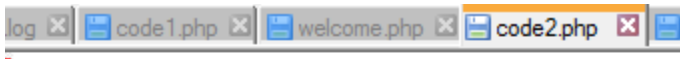
```

<td>Nama</td>
<td>:</td>
<td><?php echo $_POST["nama"]; ?></td>
</tr>
<tr>
<td>Gender</td>
<td>:</td>
<td><?php echo $_POST["gender"]; ?></td>
</tr>
<tr>
<td>Alamat</td>
<td>:</td>
<td><?php echo $_POST["alamat"]; ?></td>

```

String / karakter di dalam kurung siku harus sama dengan nilai (nama) sebuah data di file sebelumnya

- Code 2



```

</html>
<body>
<form action="welcome2.php" method="get">
<h3>INPUT DATA MAHASISWA</h3>
<table border="0">

```

Hampir sama dengan code 1, hanya saja method nya diganti dengan “GET”.
Pada

- Welcome 2

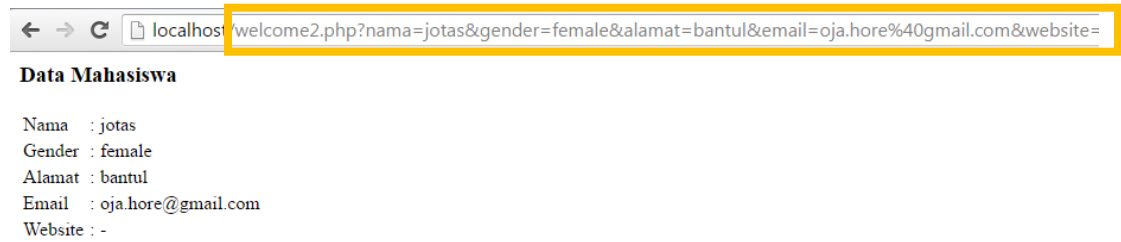
```

<td>:</td>
<td><?php echo $_GET["nama"]; ?></td>
</tr>
<tr>
<td>Gender</td>
<td>:</td>
<td><?php echo $_GET["gender"]; ?></td>
</tr>
<tr>
<td>Alamat</td>
<td>:</td>
<td><?php echo $_GET["alamat"]; ?></td>
</tr>

```

Pada code 2 menggunakan method “get”. \$_GET membuat data agar bisa ditampilkan dalam file action (file tujuan). Jika tidak ada \$_GET maka data tidak akan ditampilkan. Sama halnya dengan \$_POST. Yang membedakan code 1 dan code 2 adalah “POST” dan “GET” nya. Secara tampilan, hasilnya sama dengan

yang method POST, tapi pada URL nya berbeda. Untuk yang GET pasti akan menampilkan inputan user.



- Code 3

```
$namaErr = $genderErr = $alamatErr = $emailErr = $websiteErr = ""  
$nama = $gender = $alamat = $email = $website = "";  
if ($_SERVER["REQUEST_METHOD"] == "POST") {  
    if (empty($_POST["nama"])) {  
        $namaErr = "Nama harus diisi";  
    } else {  
        $nama = test_input($_POST["nama"]);  
    }  
}
```

\$namaErr = \$genderErr = \$alamatErr = \$emailErr = \$websiteErr = ""
Membuat variabel error dan diatur dalam kondisi kosong, sedangkan

\$nama = \$gender = \$alamat = \$email = \$website
Membuat variabel untuk menampung inputan user

Pada baris ke 4 pada gambar di atas(gbr 1) terdapat kondisi dimana jika form kosong/ user tidak menginputkan data maka akan memanggil variabel \$namaErr dan outputnya tulisan peringatan “Nama harus diisi” jika form terisi data user maka disimpan pada variabel \$nama

```

if (!preg_match("/^[a-zA-Z ]*$/", $nama)) {
$namaErr = "Inputan salah. Hanya dapat huruf dan spasi";
}
}
if (empty($_POST["gender"])) {
$genderErr = "Jenis Kelamin harus diisi";
} else {
$gender = test_input($_POST["gender"]);
}
if (empty($_POST["alamat"])) {
$alamat = "";
} else {
$alamat = test_input($_POST["alamat"]);
}
if (empty($_POST["email"])) {
$emailErr = "Email harus diisi";
} else {
$email = test_input($_POST["email"]);
// check if e-mail address is well-formed
if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
$emailErr = "Format email salah";
}
}
if (empty($_POST["website"])) {
$website = "";
} else {
$website = test_input($_POST["website"]);
// check if URL address syntax is valid (this regular expression also allows dashes in the URL

```

Pada baris pertama pada gambar di atas(gbr 2), jika user menginputkan data nama menggunakan karakter atau angka /tidak sesuai dengan ketentuan maka akan memanggil variabel \$namaErr dan memunculkan pesan error “Inputan salah. Hanya dapat huruf dan spasi”

Dan kondisi tersebut berlaku juga pada gender, alamat, email, dan website
 Pada email ada tambahan untuk mengecek format email user yang diinputkan
 Dan pada website juga ada pengecekan alamat URL apakah sudah sesuai atau belum

```

if (!preg_match("/\b(?:?:https?|ftp):\/\/|www\.)[-a-z0-9+&@#\/%?~_!|:,.;]*[-a-z0-9+&@#\/%?~_]/i", $website)) {
$websiteErr = "URL salah";
}
-}
-}
-}

```

ketentuan/format penulisan alamat URL website, preg_match itu untuk karakter yang tidak seharusnya ada pada format URL, jika ada karakter inputan user yang mengandung preg_match maka akan ada peringatan “URL salah”

```

function test_input($data) {
$data = trim($data);
$data = stripslashes($data);
$data = htmlspecialchars($data);
return $data;
}

```

Untuk membuat fungsi untuk validasi input form

4. Perbedaan GET dan POST

NO	POST	GET
1.	Tidak menampilkan nilai variabel pada URL	Menampilkan nilai variabel yang dikirimkan pada URL
2.	Tidak dibatasi	Dibatasi panjang string hingga 2047 karakter
3.	Biasanya digunakan untuk input dari FORM	Menggunakan input dari link atau akses menggunakan link.
4.	Digunakan untuk mengirimkan data rahasia seperti password	Digunakan untuk mengirimkan/mengambil data publik seperti id_user atau id_halaman.
5.	Mengirimkan data atau nilai langsung ke action untuk ditampung, tanpa menampilkan pada URL.	Menampilkan data/nilai pada URL, kemudian akan ditampung oleh action.