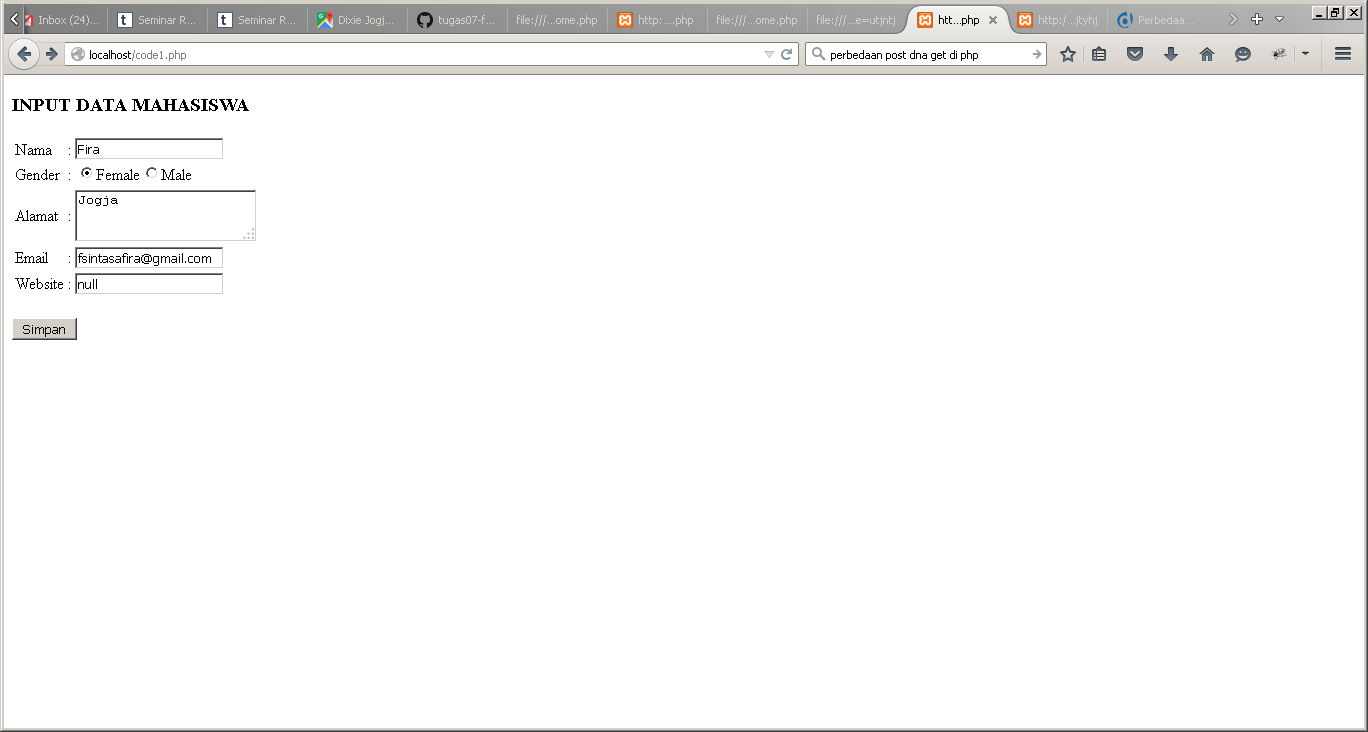
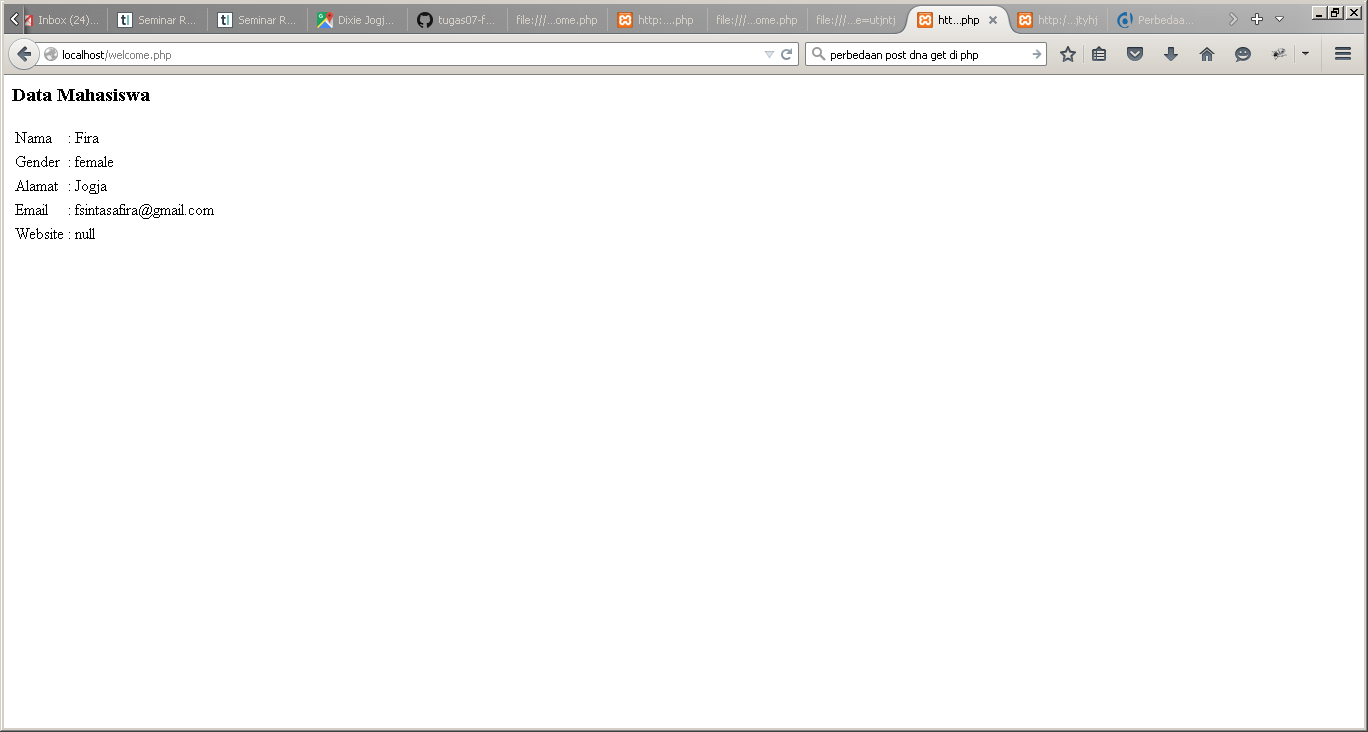
Fatihah Sinta Safira

16/401023/SV/11527

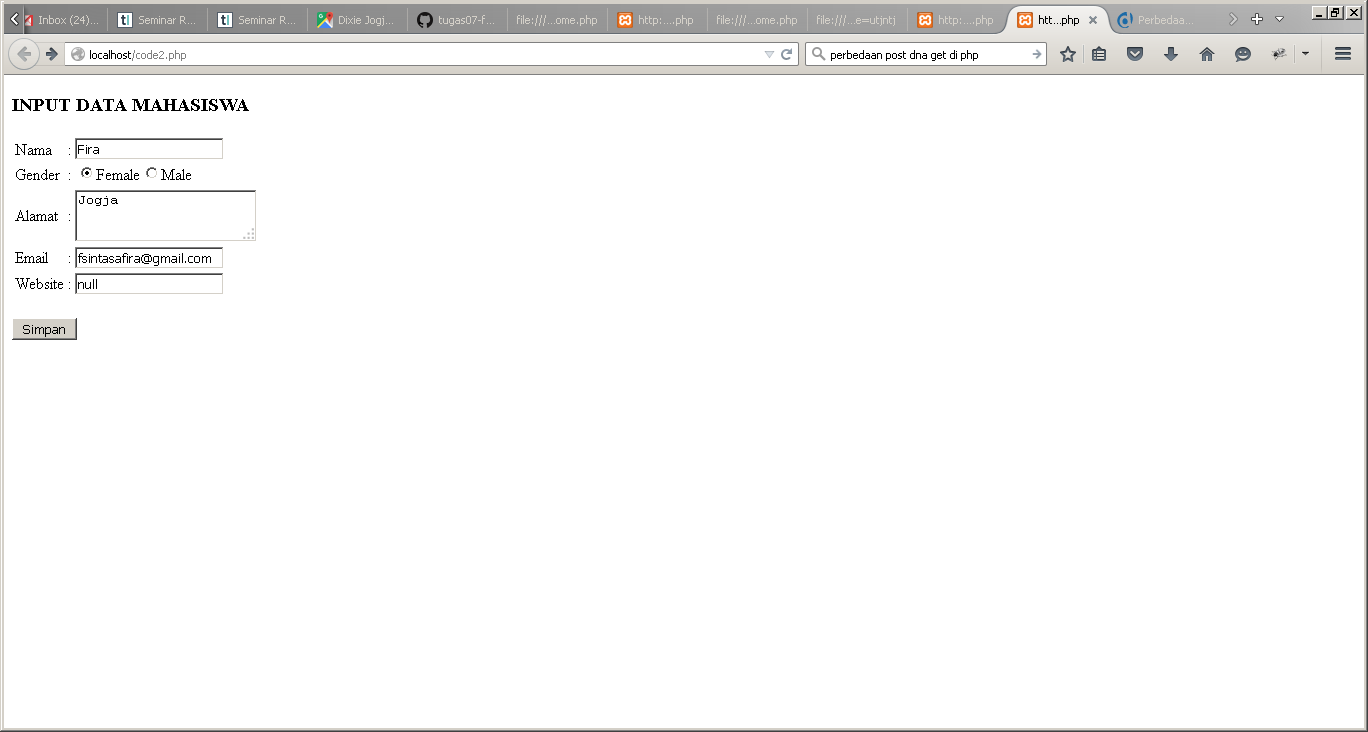
Kode 1

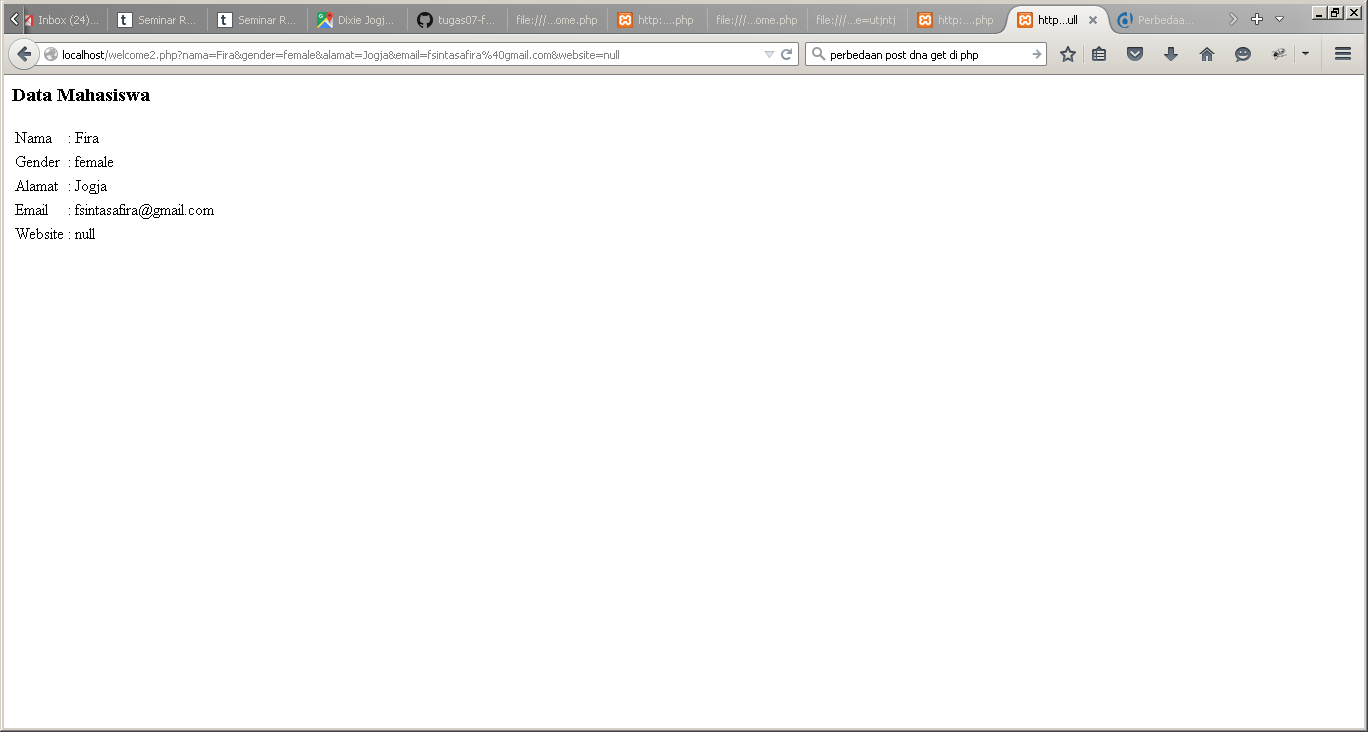




Pada percobaan kode 1, kita akan memasukkan data yang diminta dari halaman php code1. Ketika tombol ‘Simpan’ kita klik, maka akan terbuka halaman welcome.php berikut dengan data yang kita tulis di halaman sebelumnya (code1.php). Percobaan code1 menggunakan metode ‘post’ dimana ketika kita menekan tombol ‘Simpan’ maka metode ini akan langsung mengirimkan data ke action untuk ditampung dan halaman welcome.php langsung meng-“echo” atau menampilkan data yang tadi kita masukkan.

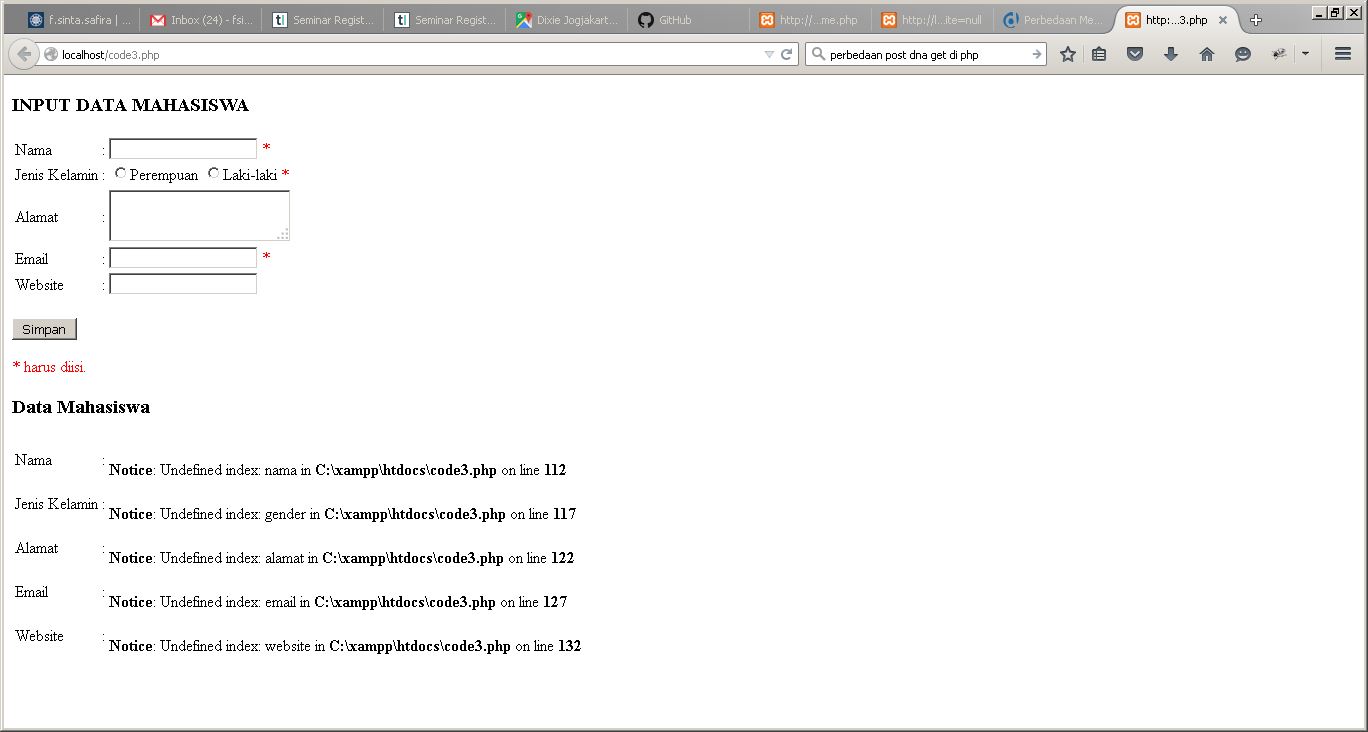
Kode 2





Pada percobaan kode 2, kita akan memasukkan data yang diminta dari halaman php code2. Ketika tombol ‘Simpan’ kita klik, maka akan terbuka halaman welcome2.php berikut dengan data yang kita tulis di halaman sebelumnya (code2.php) dan menampilkannya serta pada URL. Percobaan code1 menggunakan metode ‘get’ dimana ketika kita menekan tombol ‘Simpan’ maka metode ini akan menampilkan data pada URL lalu kemudian ditampung oleh action dan halaman welcome2.php menampilkan data yang tadi kita masukkan.

Kode 3



Analisis source code PHP

Kode1

***<form action="welcome.php" method="post">*** = untuk menampung data oleh action dengan method post

***<?php echo $\_POST["nama"]; ?>*** = untuk menjalankan method post dengan cara menampilkan data yang ditampung dalam value “nama”

Kode2

***<form action="welcome2.php" method="get">*** = untuk menampung data oleh action dengan method get

***<?php echo $\_GET["nama"]; ?>***  = untuk menjalankan method get dengan cara mengambil data yang ditampung dalam value “nama” dan menampilkannya.

Kode3

***$namaErr = $genderErr = $alamatErr = $emailErr = $websiteErr = "";***

***$nama = $gender = $alamat = $email = $website = "";*** = membuat variable dan mendefinisikannya

***if ($\_SERVER["REQUEST\_METHOD"] == "POST")***  = jika method == (sama dengan) post

***if (empty($\_POST["nama"])) {***

***$namaErr = "Nama harus diisi";*** = jika data “nama” yang diambil action dengan method post kosong, maka variable namaErr diisi dengan “Nama harus diisi”.

***else {***

***$nama = test\_input($\_POST["nama"]);*** = jika data “nama” yang diambil action dengan method post sudah sesuai dengan input yang diminta, maka variable nama diisi dengan inputan.

***else {***

***$nama = test\_input($\_POST["nama"]);***

***// check if name only contains letters and whitespace***

***if (!preg\_match("/^[a-zA-Z ]\*$/",$nama)) {***

***$namaErr = "Inputan salah. Hanya dapat huruf dan spasi";***

***}***

***}*** = berfungsi mengecek inputan sudah sesuai prosedur ata belum (inputan hanya boleh berisi huruf dan spasi). jika data “nama” yang diambil action dengan method post sudah sesuai dengan input yang diminta, maka variable nama diisi dengan inputan. Jika inputan masih salah, maka variable namaErr diisi dengan "Inputan salah. Hanya dapat huruf dan spasi".

***if (empty($\_POST["gender"])) {***

***$genderErr = "Jenis Kelamin harus diisi";***

***} else {***

***$gender = test\_input($\_POST["gender"]);***

***}***

jika data “gender” yang diambil action dengan method post kosong, maka variable genderErr diisi “Jenis Kelamin harus diisi”. Sedangkan jika input sudah sesuai maka variable gender diisi dengan inputan.

***if (empty($\_POST["alamat"])) {***

***$alamat = "";***

***} else {***

***$alamat = test\_input($\_POST["alamat"]);***

***}***

jika data “alamat” yang diambil action dengan method post kosong, maka variable alamat tidak diisi apa-apa. Sedangkan jika input sudah sesuai maka variable alamat diisi dengan inputan.

***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";***

***}***

***}***

jika data “email” yang diambil action dengan method post kosong, maka variable emailErr diisi “Email harus diisi”. Jika inputan tidak sesuai dengan format yang diminta, maka variable emailErr diisi “Format email salah”. Sedangkan jika input sudah sesuai maka variable email diisi dengan inputan.

***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)***

***if (!preg\_match("/\b(?:(?:https?|ftp):\/\/|www\.)[-a-z0-9+&@#\/%?=~\_|!:,.;]\*[-az0-***

***9+&@#\/%=~\_|]/i",$website)) {***

***$websiteErr = "URL salah";***

***}***

***}***

jika data “website” yang diambil action dengan method post kosong, maka variable website tidak diisi apa-apa (kosong). Jika inputan tidak sesuai dengan format yang diminta, maka variable websiteErr diisi “URL salah”. Sedangkan jika input sudah sesuai maka variable email diisi dengan inputan.

|  |  |
| --- | --- |
| Perbedaan method POST dan GET | |
| POST | GET |
| Mengirimkan data atau nilai / value langsung ke action untuk ditampung, tanpa menampilkan pada URL. Method POST digunakan untuk mengirimkan data yang penting / kredensial dan data yang orang lain tidak boleh tau / *secret data*, seperti *password*, dan sebagainya. | Mengirimkan Data Tidak Langsung. Method GET akan menampilkan data/nilai pada URL, kemudian akan ditampung oleh action. Method GET digunakan untuk mengirim data yang bersifat *public* seperti nama, alamat, *username*, dll. |
| Pemanggilan Method POST Menggunakan $\_POST. $\_POST berfungsi untuk memanggil data yang telah diinputkan agar bisa ditampilkan di file action. | Pemanggilan Method GET Menggunakan $\_GET. Fungsinya sama dengan $\_POST, yaitu membuat data agar bisa ditampilkan dalam file action (file tujuan). |
| Method POST Tidak Terbatas Data. Panjang karakter / data tidak terbatas. | Method GET Hanya Bisa Menampung 2047 Data. |