|  |  |
| --- | --- |
| Materi | Nilai |
| PHP-UTK-SMK -VID 12-16 WEEK 9 | 96 |

**Saya Sudah Belajar dan Mengerti dan Saya BISA**

1. Saya tahu bahwa POST dan GET itu sangat penting dalam pembuatan website, terutama pada bagian form.
2. Saya sudah bisa membuat file yang berekstensi PHP di VSCode.
3. Saya mengerti bahwa POST ini digunakan untuk mengirim data, untuk penggunaanya biasanya dipadukan dengan form, contohnya :

<form action="nav.php" method="post">

1. Di dalam form terdapat berbagai jenis tipe input seperti :

* ‘text’ untuk memasukkan data berupa teks atau tulisan.
* ‘email’ untuk memasukkan data dalam bentuk email, dan tak bisa dimasukkan jenis teks biasa.
* ‘password’ untuk memasukkan data dalam bentuk password, maksudnya data yang dimasukkan akan berubah menjadi titik-titik hitam sesuai jumlah digit yang dimasukkan.
* ‘submit’ untuk memanggil url setelah input dimasukkan, dsb.

1. Saya sudah bisa membuat form didalam file PHP dengan menggunakan tag dari HTML.
2. Saya sudah mengerti bahwa program itu dibaca dari atas kebawah, sehingga dalam membuat program harus runtut.
3. Untuk memanggil data yang sudah dimasukkan dan dikirim di dalam form tadi, kita bisa menggunakan variabel, contohnya :

$email = $\_POST['email'];

echo $email;

1. Untuk $\_POST dapat dilihat dari method di dalam form (html), jadi bisa berubah tergantung isi dari methodnya, kemudian untuk ‘email’ setelah post itu diambil dari isi name di dalam tag input di dalam form htmlnya, jika namanya tidak sama maka data yang dimasukkan tak akan masuk.
2. Saya sudah bisa membuat dan menggunakan tag <br> dalam php supaya enter kebawah. Contoh penggunaanya yaitu echo ‘<br>’;
3. Jika kita tak memasukkan data apapun dalam form, maka akan menghasilkan error berupa undefined index…
4. Saya sudah bisa menggunakan if dan isset untuk menguji apakah nilai / data dalam form sudah dimasukkan atau belum, sehingga tak akan muncul error. Penggunaanya seperti :

if ( isset($\_POST['submit']) ) { }

1. Jika menggunakan isset, maka form yang tak diisi tak akan ditampilkan (halaman kosong, bukan error).
2. Saya sudah tahu bahwa elemen action dalam tag form berguna untuk memanggil file setelah tombol submit ditekan, jika dikosongi atau tak diisi maka akan memanggil dirinya sendiri. Contoh :

<form action="navbar.php" method="post">

1. Jika menggunakan GET maka data yang dimasukkan akan tampil diatas alias sejajar dengan alamat url . Penggunaanya yaitu :

<form action="" method="get">

1. Saya sudah mengerti bahwa dikanan samadengan (=) merupakan isi variabel dan yang dikiri samadengan merupakan nama variabel.
2. Untuk memanggil nilai ‘get’ yang ada di dalam url kita bisa menggunakan :

$alamat = $\_GET['alamat'];

echo $alamat;

1. Saya sudah tau bahwa untuk menggunakan href dalam html kita tak hanya bisa menggunakan tanda pagar (#), tapi bisa menggunakan ?menu=nama file. Penggunaanya seperti :

<li><a href="?menu=kontak">kontak</a></li>

1. Jika menggunakan tanda pagar (#) pada href dan file yang dituju tak ada atau salah dalam penulisan, maka browser akan menampilkan error berupa “object not found”.
2. Jika menggunakan ?menu pada href dan file yang dituju tak ada atau salah dalam penulisan, maka browser akan menampilkan halaman kosong, dan nama fle tersebut akan ada didalam url.
3. Saya mengerti bahwa untuk memanggil nama file dalam url kedalam halaman web kita bisa mengunakan :

 if (isset($\_GET['menu'])) {

        $menu = $\_GET['menu'];

}

Maka hasilnya adalah nama file di url tersebut dipanggil ke dalam halaman web.

1. Saya mengerti bahwa ?menu dalam href digunakan untuk mengisi url.
2. Saya sudah bisa menggunakan require once untuk memanggil isi file dan menggabungkannya dengan concat, contoh penggunaanya yaitu :

require\_once $menu . '.php';

1. Perbedaan antara method get dan post yaitu jika method POST akan mengirimkan data langsung ke action untuk ditampung tanpa menampilkannya di url, sedangkan untuk method GET akan menampilkan data pada url kemudian ditampung oleh action.
2. Saya tahu bahwa session itu sering digunakan untuk menyimpan variabel.
3. Saya tahu bahwa variabel yang disimpan di dalam session tak akan hilang dan akan tersimpan terus selama browsernya menyala atau belum dihapus.
4. Saya mengerti bahwa session sebelum digunakan harus dinyalakan atau diaktifkan terlebih dahulu dengan session start, sebagai berikut :

session\_start();

1. Untuk memanggil session kita harus menggunakan $\_SESSION diikuti nama kemudian isinya.
2. Saya sudah bisa memanggil atau membuat session, contohnya :

$\_SESSION['user'] = 'joni';

echo $\_SESSION['user'];

1. Saya bisa mellihat session dalam program yang dibuat dengan menggunakan elemen var\_dump.
2. Jika dilihat dari bentuknya, session ini seperti array asosiatif dimana terdapat bagian key dan value nya.
3. Saya bisa menampilkan seluruh isi session dengan menggunakan foreach, contoh penggunaannya adalah :

foreach ($\_SESSION as $key => $value) {

echo $value . ' => ' . $value . '<br>';

}

1. Session bisa ditampilkan dengan menggunakan foreach karena session itu sendiri sama dengan array asosiatif.
2. Saya bisa menggunakan function pada session sebagai wadah, contohnya :

function isiSession(){

        $\_SESSION['user'] = 'joni'; }

1. Untuk menghapus session kita bisa menggunakan switch case, yaitu :

case 'hapus':

                unset($\_SESSION['user']);

                break; }

1. Untuk menghancurkan atau memusnahkan session kita dapat menggunakan switch case juga, contonya :

case 'destroy':

                session\_destroy();

                break;

1. Jika session sudah berhasil dihapus atau sudah dimusnahkan, maka akan memunculkan error berupa *Undefined index* pada halaman sessionnya, karena sudah kosong. Untuk mengatasinya yaitu hanya perlu diisi lagi.
2. Saya tahu perbedaan antara unset dan destroy, jika unset maka akan hanya menghapus session yang sudah ditentukan, sedangkan jika destroy akan menghancurkan atau memusnahkan semua session tanpa pandang bulu.
3. Saya mengerti bahwa Cookie digunakan untuk menyimpan informasi dari user didalam browser, misal jika kita mengklik suatu ads atau iklan maka iklan tersebut akan muncul lagi setelah beberapa saat karena informasinya sudah disimpan.
4. Untuk membuat cookie kita bisa menggunakan :

$cookie\_name = 'user';

$cookie\_value = 'joni';

setcookie($cookie\_name, $cookie\_value);

1. Untuk memanggil cookie yang sudah diisi kita dapat menggunakan :

echo $\_COOKIE[$cookie\_name];

1. Saya tahu bahwa kita bisa mengatur kapan cookie itu berhenti atau expire dengan cara mengetikkan *time() - waktu yang diinginkan(berupa detik)* dalam setcookie, kita juga bisa menggunakan ini untuk menghapus cookie. contohnya :

setcookie("user", "", time() - 3600);

1. Saya sudah mengerti bahwa untuk mengubah isi dari cookie kita hanya tinggal meng set cookienya lagi dengan menggunakan function *setcookie().*
2. Kita bisa mengecek apakah cookienya ada atu tidak dengan menggunakan *var\_dump($\_COOKIE);*
3. Jika sebelumnya kita belum menyimpan file cookie dalam browser maka var\_dump hanya menampilkan tulisan empty.
4. Saya sudah bisa membuat form dengan input type ‘file’. Contohnya :

<input type="file" name="upload">

1. Jika kita menggunakan input type ‘file’ , maka saat diklik otomatis akan mengarahkan kita ke dalam File Explorer untuk memilih file.
2. Untuk menampilkan seluruh isi dalam variabel file kita dapat menggunakan foreach, sebagai berikut :

$file = $\_FILES['upload'];

foreach ($file as $key => $value) {

            echo $key . ' = ' . $value; }

1. Saya sudah mengerti bahwa file yang diupload dengan input type ‘file’ akan disimpan dalam bentuk tmp atau temporary file.
2. Saya bisa mengambil nama dan lokasi dari tmp file dari file yang diupload dengan menggunakan :

$name = $\_FILES['upload']['name'];

$temp = $\_FILES['upload']['tmp\_name'];

        echo $name . ' = ' . $temp;

1. Untuk memasukkan file yang sudah diupload kedalam folder yang sudah ditentukan kita dapat menggunakan function *move\_uploaded\_file(alamat folder).* Contohnya :

move\_uploaded\_file($temp,'gambar/' . $name);

1. Yang harus diperhatikan untuk mengambil nama file dan tmp file adalah kita harus mengambilnya dengan $\_FILES kemudian diikuti nama dari input dalam form di kode HTML nya.

**Saya kurang mengerti :**

1. -