BELAJAR PHP

**Syarat-syarat menjalankan kode php**

1. Jalankan di localhost xampp
2. <?php ?>
3. Php harus menggunakan titik koma

**Kode Hello world**

echo "hai";

**PHP itu Bisa di HTML**

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <h1>Selamat Datang, <?php echo "halo"; ?></h1>

</body>

</html>

Kita bisa nyimpan variable juga looo……

**Variabel dan Tipe data di PHP**

Sama dengan javascript, ktia tidak perlu mendefisiikan tipe data…

$nama = "Manusia";

$nama adalah nama variabel dan “Manusia” adalah isi dari variabel nama bertipe string

**Operator matematika dari PHP**

**+ \* - / %**

**Operator Penggabung String menggunakan titik (.)**

$nama\_depan = "Maso";

$nama\_belakang = "Peso";

<h1>Selamat Datang, <?php echo $nama\_depan . $nama\_belakang; ?></h1>

Maka akan menghasilkan **MasoPeso**

**Operator Asignment**

**= , +=, -=, \*=, /=, %= , .=**

**Contoh:**

$angka1 = 2;

$angka1 \*= 2;

    <p><?php echo $angka1; ?></p>

**Output = 4**

**Mengecek Tipe data**

Untuk mengecek tipe data di PHP, kita bisa menggunakan var\_dump(isi);

var\_dump($nama);

$nama berisi **string dengan panjang 7**

**Operator Perbandingan**

**=, <=, >=, ==, != (tidak sama dengan)**

**Operator Logika**

&& = Harus bernilai keduanya benar

|| = bisa salah satu benar, atau keduanya benar

! = kebalikan, dari true ke false, false ke true.

**Struktur kendali PHP**

Berikut struktur kendali pada php:

1. Perulangan
2. For
3. While
4. Foreach
5. Do While
6. Pengkondisian
7. If
8. Else
9. Ternary
10. Switch (kalau banyak if else)

**For**

for ($i = 0 ; $i < 5; $i++){

    echo $i;

}

Pada bagian for, berisi parameter : **inisialisasi variabel, kondisi untuk stop, increment/decrement (nilai di tambah atau di kurang, bahkan bagi,kali, dll)** . Syarat berhenti nya For yakni sampai nilai tersebut berhenti di suatu kondisi. Misal di atas oputnya 01234. 4<5 ? ya… 5<5 ? tidak.

**While**

$i = 0;

while ( $i < 5){

    echo "<br>" . $i;

    $i++;

}

While dijalanknan **KETIKA KONDISI DI PARAMETERNYA NYA BERNILAI True yang akan menjalankan perulangan**. Sebagai contoh , ketika I yang awalnya 0, lalu di whilekan, apakah 0 < 5 ya, 1 < 5 ya … 5<5 tidak lalu akan stop. Di sini juga nilai akan bertambah ketika kondisi true

**Do While**

$do = 0;

do {

   echo "<br>". $do;

   $do++;

}

while ($do < 5);

**Do while pengulangannya di buat SAAT SEBELUM PENGECEKAN,** sedangkan while cek dulu baru kondisi..

**Implementasi Perulangan For di HTML**

    <table border="1" cellpading="10" cellspacing="0">

        <?php

        for($baris = 1; $baris <= 3 ; $baris++){

            echo "<tr>";

            for ($col = 1 ; $col <=5; $col++ ){

                echo "<td> $baris, $col";

            }

            echo "</tr>";

        }

        ?>

    </table>

**Penjelasan:**

Suatu tabel memiliki 3 baris (kebawah) dan masing masingnya memiliki 5 kolom (ke samping)

1. Membuat perulangan beris

Baris tag tr akan di ulangi sebanyak 3 kali.

1. Membuat kolom

Kolom akan buat 5 samping yang diulangi 5 kali, lalu akan cetak td.

**Implementasi Perulangan For di HTML dengan gaya *templeting.***

    <table border="1" cellpading="10" cellspacing="0">

        <?php

        for($baris = 1; $baris <= 3 ; $baris++){

        ?>

        <tr>

            <?php for($col =1 ; $col <=5; $col++ ){?>

            <td> <?php echo $baris. "," . $col; ?> </td>

            <?php }?>

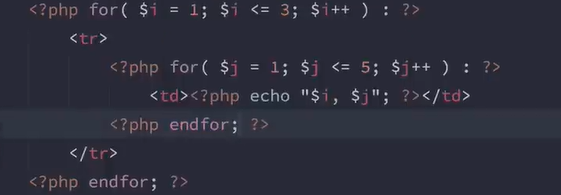
        </tr>

        <?php }?>

    </table>

PHP itu fleksibel boy 😊… Oke contohnya :

**Pada baris for (sintaksbaris) di buka dengan <?php ?>** di tutup dengan kurung kurawal buka lalu di isi kode perulangan baris lalu jangan lupa buat **<?php }?> .** Kalau semisal agak bingin dengan tutup nya dimana bisa aktifkan fitur color, atau bisa menggunakan **endfor, endif**



**IF**

$x = 10;

if($x < 20){

    echo "benar";

}

else{

    echo "salah";

}

IF artinya **JIKA**, berarti dapat disimpulkan **JIKA X < 20** tampilkan benar. Else artinya selainitu **SELAIN <20 Tampilkan SALAH**.

**ELSE IF**

$umur = 14;

if($umur <= 10){

    echo "anak anak";

}

else if ($umur <= 20){

    echo "remaja";

}

else{

    echo "Dewasa";

}

ELSE IF akan jalan ketika kondisi if sebelumnya bernilai salah. Dan Jika nila else if salah maka akan lanjut ke kode berikutnya.

**PENGKODISIAN PADA TABEL**

    <table border="1" cellpading="10" cellspacing="0">

        <?php

        for($baris = 1; $baris <= 5 ; $baris++){

        ?>

        <?php

            if($baris % 2 == 1){

            ?>

        <tr>

            <?php for($col =1 ; $col <=5; $col++ ){?>

            <td style="background-color: white;"> <?php echo $baris. "," . $col; ?> </td>

            <?php }?>

        </tr>

        <?php } else{?>

        <?php for($col =1 ; $col <=5; $col++ ){?>

        <td style="background-color: gray;"> <?php echo $baris. "," . $col; ?> </td>

        <?php }?>

        <?php }?>

        <?php }?>

    </table>

Ketika baris di moduluskan dengan 2 hasi nya 1 (ganjil) maka akan **BG WHITE** selain itu/genap MAKA akan **BG GRAY**

**SWITCH CASE**

switch ($umur){

    case $umur <= 10:

        echo "anak anak";

        break;

    case $umur <= 20:

        echo "remaja";

        break;

    default:

    echo "Dewasa";

}

**Ini mirip dengan if else**, Pada switch kita harus is apa yang mau kita bandingkan, di sini kita pakai $umur yang tadi. **CASE ITU MIRIP DENGAN IF/IFELSE** yang berjalan dari atas ke bawah. Ketika suatu **CASE BERNILAI TRUE** maka akan melakukan sesuatu dan **MENYELSESAIKANYA/BREAK** . Deafult ibaratnya **ELSE**.

**FUNCTION**

**Function adalah suatu kumpulan program yang di jadikan satu, berguna untuk pemangilan ulang.** Pada PHP ada dua fungsi yakni BuiltIn Function (fungsi yang sudah di sediakan di PHP), **User-Defined Function** (fungsi yang di buat pengguna).

**Bulit In Function**

**DATE**

    echo date("l, d-M-Y");

l = hari

d = Tanggal

M = Bulan

Y = Tahun

Masih banyak lagii……

**User Defiend Function.**

function  salam($nama){

    echo "Selamat Datang $nama";

}

Dalam Fungsi ada namanya Parameter yang berisi suatu nilai yang akan mengembalikan lagi

salam("orang");

Ketika ingin menulis lebih dari 1 parameter/argumen, jangan lupa harus berurutan

function  salam($waktu , $nama){

    echo "Selamat $waktu, $nama !";

}

salam("pagi" ,"Manusia")

**NUMERIC ARRAY**

Numeric Array adalah suatu variabel yang bisa menampung banyak nilai, nilai array di mulai dari index **0**

**Membuat array dengan cara LAMA**

<?php

// cara lama

$hari = array("Senin", "Selasa", "Rabu");

?>

**Membuat array dengan cara BARU**

$bulan = ["Januar", "Febuari", "Maret"];

**Ada 2 cara menampilkan array**

// Menampilkan array

// var dump, printf()

var\_dump($hari);

print\_r($bulan);

**Atau bisa menggunakan LOOP**

for ($i = 0 ; $i < count($hari); $i++){

    echo $hari[$i]. "<br>";

}

**Pertama** kita deklarasikan I nya, ketika I nya kurang dari **JUMLAH VALUE**maka I akan bertambah, sebagai contoh pada hari kita memilii 3 value, maka I akan menghitung dan membandingkan, **0 + 1 = 1, apakah 1<3 ? ya , 1+1 = 2, apakah 2<1 ? ya, 2+1 = 3, 3<3 ? tidak lalu stop.** Sembari itu **ARRAY AKAN DI PRINT DENGAN NAMA VARIABEL BERSERTA INDEX YANG DI ISI I** . Ini akan bertambah tambah: Ibartanya seperti ini

echo $hari[0];

echo $hari[1];

echo $hari[2];

**Penerapan HTML**

    <div style="display: flex; gap:10px;">

        <?php for($i = 0 ; $i < count($hari) ; $i++) {?>

        <div class="kotak">

            <?php echo $hari[$i] ?></div>

        <?php }?>

    </div>

**Kalau mau menampilkan 1 value, tulislah dengan indexnya**

echo $hari[0];

**Menambahkan value array**

$hari[] = "Kamis";

**Pengulangan khusus untuk aray/ For Each**

foreach ($hari as $harinew){

    echo $harinew;

}

Konsep ini lebih gampang, jadi , perulangan ini berarti **SETIAP ELEMEN/VALUE dari suatu Aray, sbg cnth: hari AKAN DI BUNGKUS DENGAN VARIABLE berikutnya yakni harinew lebih enak jika pakai b inggris, angap saja banyak buku = books, kita ambil satu persatu kita sebut book.**

**Penulisan HTML**

    <div style="display: flex; gap:10px;">

        <?php foreach($hari as $harinew) {?>

        <div class="kotak">

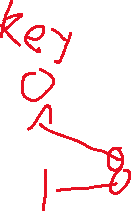
            <?php echo $harinew ?></div>

        <?php }?>

    </div>

**Multi dimensi Numeric Array**

Ibaratnya kita memiliki 2 siswa, yang memiliki data seperti , NIS, no telp, jurusan dll. Kita bisa gabungkan di suatu variabel saja (dengan berbagai key &value), tanpa kita harus buat variabel array yang lain, di sini di kita cari index (0,1) lalu di loopkan agar tidak mengulaingi kode yang sama. Setelah itu kita gunakan ul dan li yang diisikan value dalam index/key nya



$siswa = [



    ["Orang aja" , "01010101", "Teknik Abal Abalan", "Semester2"],



    ["Manusia" , "0132101", "Teknik Abal Abalan", "Semester1"]

];

    <?php foreach ($siswa as $siswatampil){?>

    <ul>

        <li>

            <?php echo $siswatampil[0]; ?>

        </li>

        <li>

            <?php echo $siswatampil[1]; ?>

        </li>

        <li>

            <?php echo $siswatampil[2]; ?>

        </li>

        <li>

            <?php echo $siswatampil[3]; ?>

        </li>

    </ul>

    <?php }?>

Jadi kita tampilkan per elemen dan di dalamnya beriisi value seperti nama, nis, jurusan, semester brp…

Atau bisa pakai

        <?php foreach ($siswatampil as $datasiswa) {?>

        <li>

            <?php echo $datasiswa; ?>

        </li>

        <?php }?>

$angka = [

    [1,2,3],

    [1,4,2]

];

echo $angka[1][1];

**Associateve Array**

Numeri array key nya itu berupa angka, sedangkan array asosiativ dapat berupa apa saja baik itu teks atau pun angka.

$siswa = [

    "nama" => "Manusia ujang",

    "nis" => "01012134",

    "jurusan" => "jurusan abal abal",

    "semester" => "semester 1"

];

Jadi key nya kita buat sendiri bukan berupa angka bawaan seperti 0 1 2 3 dst..

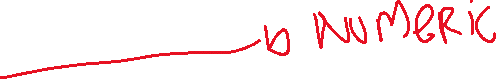
**Ketika kita ingin menampilkan, tulis aja key nya..**

echo $siswa["nama"];

maka akan tampil nama siswa



**Multi dimensi Associative Array & numeric**



$siswa = [

    [



    "nama" => "Manusia ujang",

    "nis" => "01012134",

    "jurusan" => "jurusan abal abal",

    "semester" => "semester 1"

    ],

    [



    "nama" => "Orang",

    "nis" => "01016572134",

    "jurusan" => "jurusan abal abal",

    "semester" => "semester 10"

    ]

];



**Nahh** di key pembukus nya itu adalah numeric, dikarena kan kita tidak menamakan pembukusnya, jadi dia punya index

echo $siswa[1]["nama"];

**Array di dalam array di dalam array di dalam array :v**

$siswa = [

    [

    "nama" => "Manusia ujang",

    "nis" => "01012134",

    "jurusan" => "jurusan abal abal",

    "semester" => "semester 1"

    ],

    [

    "nama" => "Orang",

    "nis" => "01016572134",

    "jurusan" => "jurusan abal abal",

    "semester" => "semester 10",

    "tugas" => [90, 80, 100]

    ]

];

echo $siswa[1]["tugas"][2];

**jadi ini akan mengambil siswa index ke 1, lalu ambil tugas, lalu di tugas ambil index ke 2 = 100.**

<?php

$siswa = [

    [

    "nama" => "Manusia ujang",

    "nis" => "01012134",

    "jurusan" => "jurusan abal abal",

    "semester" => "semester 1"

    ],

    [

    "nama" => "Orang",

    "nis" => "01016572134",

    "jurusan" => "jurusan abal abal",

    "semester" => "semester 10",

    ]

];

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>

<body>

    <h1>Daftar SIswa</h1>

    <?php foreach ($siswa as $siswatampil){?>

    <ul>

        <li>

            <?php echo $siswatampil["nama"]; ?>

        </li>

        <li>

            <?php echo $siswatampil["nis"]; ?>

        </li>

        <li>

            <?php echo $siswatampil["jurusan"]; ?>

        </li>

        <li>

            <?php echo $siswatampil["semester"]; ?>

        </li>

    </ul>

    <?php }?>

</body>

</html>

Ketika kita ingin panggil maka **pertama dia akan mengambil array terluarnya, yang berupa index di gunakan loop for each: setiap data siswa akan di namakan var baru yakni siswa tampil.** Lalu lalu di panggil lagi array yang di dalamnya berupa “nama”, “nis” dll…

Kita buat tampilkan lebih menantang….

**BUATLAH NILAI di dalam ARRAY PER SISWA, lalu tampilkan, dan jangan lupa JELASKAN!**

           <?php

                for ($i = 0 ; $i < count($siswatampil["nilai"] ) ; $i++ ){

                    echo $siswatampil["nilai"][$i];

                };

            ?>

Nah jadi

1. Inisialisasi $i
2. Mengambil panjang index dari $siswatampil = $siswatampil dari for each yang tadi yakni yang array terluar/merah merahnya. Tetapi ini di khuskan yang array asosiatif “nilai”. OUPUT = 0 1 2
3. Setelah mengambil di lakukan perbandingan dan pertambahan
4. Apakai 0 < dari 2? ya.. maka data akan ditambah = 1, lalu di print $siswatampil bedasarkan nilai dan di dalam nilai itu ada index yang diisi dengan 0
5. Apakai 1 < dari 2? ya.. maka data akan ditambah = 2, lalu di print $siswatampil bedasarkan nilai dan di dalam nilai itu ada index yang diisi dengan 2
6. Apakai 2 < dari 2? Tidak .. maka data akan ditambah, lalu di print $siswatampil bedasarkan nilai dan di dalam nilai itu ada index yang **tidak di isi apa lagi**

Atau juga bisa pakai

                foreach ($siswatampil["nilai"] as $siswatampilnilai){

                    echo $siswatampilnilai;

                }

**Jadii** di setiap $siswatampil yang berisi array asosiativ “nilai” akan di buatkan variabel baru agar bisa menampilkan isinya!

Sama halnya akan di tulis

                echo $siswatampil["nilai"][0];

                echo $siswatampil["nilai"][1];

                echo $siswatampil["nilai"][2];

**Menambahkan gambar**

    "gambar" => "/belajarphp/wpu/array/gambar/gambar2.png"

Sesuaikan lah path nya!!!

Lalu

        <li>

            <img style="height: 90px;" src="<?php echo $siswatampil["gambar"];  ?>" alt="">

        </li>

**Metod GET dan POST**

Sebelum itu , kita harus belajar Variabel Scope/Lingkup Variabel

**Variabel Local/lingkup terkecil dari suatu variabel**

// Variabel local

$x = 10;

echo $x;

function tampilx(){

    $x = 20;

    echo $x;

}

tampilx();

variabel local yang ada di function **tidak sama dengn** varaibel local di luar fungsi.

**Variabel Global/bisa di fungsi**

<?php

// Variabel local

$x = 10;

echo $x;

function tampilx(){

    global $x;

    echo $x;

}

tampilx();

?>

Variabel global di luar function **sama denan** variabel di dalam fungsi.

**Superglobals variabel/kapan pun dan di manapun di pakai dan tipenya array asosiatif**

var\_dump($\_POST);

post dan get itu awalya kosong **array(0) { }, kalau $\_SERVER** akan menampilkan tentang info server kita.

**$\_GET**

**Get adalah** metode superglobal yang berguna untuk mengirimkan lalu menangkap suatu data, seperti di link, ini bersifat asosiatif array, yang di mana di setiap data harus ada **key** dan **value, GET AKAN DIKIRIMKAN MELALUI LINK**

Mengisi get bisa menggunakan array asisoatif dengan suatu key yang bernilai biasanya string dan beriisi value

$\_GET["nama"] = "Ujang";

$\_GET["nis"] = "0021321343";

**CARA MENGIRIM DATA GET**

[**http://localhost/belajarphp/wpu/metod/get.php?nama=ujang**](http://localhost/belajarphp/wpu/metod/get.php?nama=ujang)

di akhirnya , tambahkan

**? artinya** saya akan memasukan data ke $\_GET

nama= **artinya** key

ujang **artinya** value

Saya akan memasukan data ke halaman ini dengan metode GET dengan key nama dan value ujang, lalu di tangkap oleh variabel super global GET

Jika kita ingin menambahkan data lain kita pakai simbol &

**http://localhost/belajarphp/wpu/metod/get.php?nama=ujang&nis=2313213**

**Praktik Request Metode Get**

<?php

$siswa = [

    [

    "nama" => "Manusia ujang",



    "nis" => "01012134",



    "jurusan" => "jurusan abal abal",

    "semester" => "semester 21",

    "nilai" => [80, 90 , 87],

    "gambar" => "/belajarphp/wpu/array/gambar/gambar1.png"

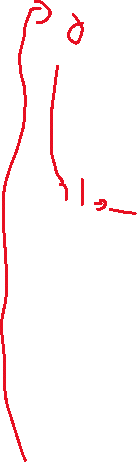
    ],

    [

    "nama" => "Orang",



    "nis" => "01016572134",



    "jurusan" => "jurusan abal abal",

    "semester" => "semester 10",

    "nilai" => [88, 92 , 94],

    "gambar" => "/belajarphp/wpu/array/gambar/gambar2.png"

    ]

];

?>

<!DOCTYPE html>

<html lang="en">

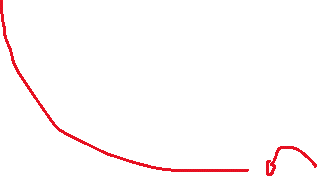
<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Document</title>

</head>



<body>

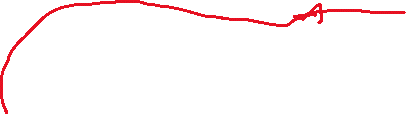
    <h1>Daftar Siswa</h1>

    <ul>

        <?php foreach ($siswa as $siswatampil){?>

        <li>

            <a href="latihan2get.php?nama=<?php echo $siswatampil["nama"]  ?>">



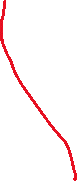
                <?php echo $siswatampil["nama"]  ?>

            </a>

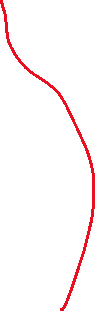
        </li>

        <?php }?>

    </ul>



</body>



</html>

**Jadi**

            <a href="latihan2get.php?nama=<?php echo $siswatampil["nama"]  ?>">

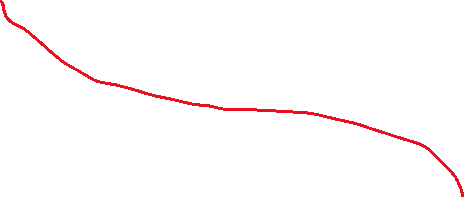


                <?php echo $siswatampil["nama"]  ?>

            </a>

Setiap nama siswa akan di ceta menjadi a href, yang akan di beritauhukan dengan **latihan2get.php?nama=<?php echo $siswatampil["nama"] , pada setiap nama siswa.**

Jadi



**“nama=”**sebagai key yang akan nanti di panggil

$siswatampil[“nama”] diambil dari array asosiatif dari siswa dengan key nama

**MENERIMA DATA GET**

**http://localhost/belajarphp/wpu/metod/latihan2get.php?nama=Manusia%20ujang**



Setelah itu di file **latihan2get.php**



        <li>



            <?php echo $\_GET["nama"]; ?>

        </li>

$\_GET berutgas untuk menangkpa data dari link tersebut kususnya nama

**Keamanan tambahan**

<?php

if (!isset($\_GET["nama"]) || !isset($\_GET["nis"])){

    header("Location: latihan1get.php");

    exit;

}

?>

**Di sini** ada validasi **isset,** isset bekerja **untuk mengecek data di isi.** Kita tambahkan logic NOT (!) . Yang berarti **jika data nama atau data nis..dst yang harus di isi itu TIDAK ADA maka jalankan balikan ke lokasi latihan1get.php, lalu keluar.**

**Tantangan !**

**Coba tampilkan NILAI!!!**

**Jawab:**

**url**

&nilai=<?php foreach ($siswatampil["nilai"] as $siswatampilnilai) {echo $siswatampilnilai . " ";} ?>"

        <li>

            <?php echo $\_GET["nilai"]; ?>

        </li>

**$\_POST**

**Post** mengirimkan data tanpa melihatkan di url, ini akan berguna saat login/register. POST hanya bisa di kirim menggunakan form. Ini juga menggunakan prinsip asosiatif

    <form action="postdasar1.php" method="post">

        <label for="nama">Nama:</label>

        <input type="text" name="nama">

        <br>

        <button type="submit">Kirim!</button>

    </form>

Untuk menggunakan metode ini **input** nya kita harus isi name=””. Ini berfungsi sebagai key dari assosiatif. Nanti value dari inputnya akan menjadi value dari key tersebut. Misal kasus di atas menggunakan key name=”nama”, ketika kita isi “ujang”, maka akan menjadi nama=>”ujang”.

Untuk menampilkanya gunnakan $\_POST[“nama”] ini akan menampilkan asosiatif yang tadi. Contoh: nama=>”ujang”

    <h1>Selamat datang, <?= $\_POST["nama"] ?>! </h1>

**Buat halaman login sederhana dengan $\_POST**

<?php

// Mengecek apakah tombol submit sudah ditekan?

if(isset($\_POST["submit"])){

    // Mengecek apakah username = admin dan password = 123 ?

    if($\_POST["username"] == "admin" && $\_POST["password"] == "123"){

        // jika benar ke admin.php

        header("Location: admin.php");

        exit;

    }

    else{

        // jika salah tampilkan eror

        $error = true;

    }

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Login</title>

</head>

<body>

    <h1>Halaman Login</h1>

    <?php

    if(isset($error)){  ?>

    <p>SALAH OI PASSWORDNYA</p>

    <?php    } ?>

    <form action="" method="post">

        <label for="username">Username:</label>

        <input type="text" name="username">

        <br>

        <label for="password">Password:</label>

        <input type="password" name="password">

        <br>

        <button type="submit" name="submit">Submit</button>

    </form>

</body>

</html>

**Jadi pada formulir tersebut** akan mengirimkan key dan value dari Username, password, submit yang dikirim dengan metod POST. Setelah itu, kode ini di tangkap OLEH $\_POST lalu mengecek kondisinya. **KONDISI array asosiatif di harapkan dengan if else yang tadi** :

submit=>true **(saat di tekan)**

username=>admin

password=>123

**Tantangan !**

**Buatlah lebih dari 1 pengguna dengan data dari ARRAY ASOSIATIF!!!**

<?php

$user = [

    [

        "nama" => "admin",

        "password"=>"123"

    ],

    [

        "nama" => "ujang",

        "password"=> "321"

    ]

];

// Mengecek apakah tombol submit sudah ditekan?

if(isset($\_POST["submit"])){

    // Mengecek apakah username = admin dan password = 123 ?

    foreach ($user as $akun){

        if($\_POST["username"] == $akun["nama"]  && $\_POST["password"] == $akun["password"]){

            // jika benar ke admin.php

            header("Location: admin.php");

            exit;

        }

        else{

            // jika salah tampilkan eror

            $error = true;

        }

    }

}

?>

**ALUR**

Saat submit di tekan, lalu melakukan for each dari array asosiatif user yang di namakan AKUN yang berisi **nama** dan **password yang key perentnya nya itu tergantung dengan array numeric** ini akan menjadi satu kesatuan. Lalu melakukan logic if yang di harapkan sebagai contoh berikut:

Foreach akan melakukukan LOOP terus menerus sampai hasil **BENAR!**

($\_POST["username"] == $akun["nama"] && $\_POST["password"] == $akun["password"])

Contoh 1 input username = admin, dan input password = 123 :

**username=>admin == nama=>admin && password=>123 == password=>123 “BENAR”**

Contoh 2 input username = ujang, dan input password = 321 :

**username=>ujang == nama=>admin && password=>321 == password=>123 “SALAH”**

**username=>ujang == nama=>ujang && password=>321 == password=>321 “BENAR”**

**INTINYAAA, METOD GET MENGIRIM & MENGAMBIL DATA DAN *PERLIHATKAN DI LINK* (COCOK UNTUK MENCARI DATA TERTENTU MELALUI LINK), METOD POST MENGIRIM DAN MENGAMBIL DATA BIASANYA DI FORM DAN *TIDAK DI PERLIHATKAN DI LINK* (COCOK UNTUK FORM SENSITIF)**

**DATABASE & MySQL**

Sebelum itu, kita harus tau apa itu **data?** Data adalah representasi fakta di dunia nyata. Contoh : nama, dokumen, video musik, dll.. Database adalah suatu sistem untuk menyimpan data. Contoh software : Mysql, MongoSql dll.

Konsep per-tabelan pada database:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |

**Warna Oren = Kolom (field). Warna hijau = Baris (record)**

**Aturan Field : harus isi dengan tipe datanya, misal : string, integer dll.**

**Konsep Primary KEY**

Sebuah data yang dapat merepresntasikan data secara unik (tidak ada data yang sama) per barisnya. Contoh Nomor Absen/NIS yang digunakan dengan **ID** , di database juga ada namanya AUTO INCREMENT / penambahan setiap baris (record)

Disini kita pakai Mysql….

Ketik di terminalnya laragon

**mysql -u root -p**

**Command Melihat database:**

show databases;

**Praktik Membuat database:**

**Nama database : phpdasar**

**Tabel : siswa**

**id int primary key auto\_increment**

**nama varchar(100) => panjang ada 100 karakter**

**nis varchar(10)**

**jurusan varchar(100)**

**semester varchar(100)**

**gambar varchar(100)**

1. **Comand untuk membuat database:**

**create database namadatabase;**

create database phpdasar;

**Jika :**

**Query OK, 1 row affected (0.01 sec), AMANN!**

1. **Comand untuk menggunakan suatu database (untuk prompt selanjtunya)**

**use namadatabase;**

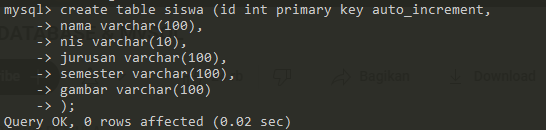
use phpdasar;

Jika

**Database changed, AMANNN!!!**

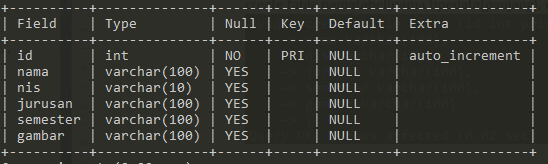
**3. Comand Untuk membuat table berserta isinya.**

**create table namatabel(nama tipedata(panjang), nama2 tipedata(panjang));**

****

**4. Melihat struktur tabel**

**describe namatabel;**



**CRUD (CREATE, READ, UPDATE, DELETE)**

**5. Menambahkan data siswa (INSERT)**

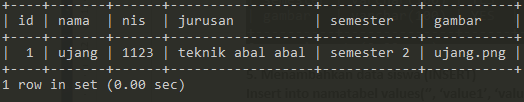
**Insert into namatabel values(‘’, ‘value1’, ‘value2’, ‘value3’);**

**HARUS BERURUTAN DENGAN FIELD**

INSERT INTO siswa VALUES (NULL, 'ujang', '1123', 'teknik abal abal', 'semester 2', 'ujang.png');

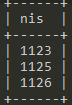
**6. Melihat data tabel (SELECT)**

**Select \* from siswa;**

****

**a. menampilkan data lebih spesifik (misal nis)**

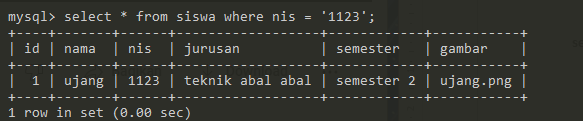
**select nis form siswa;**

****

**Select nama,nis form siswa;**

**b. mencari data kesluruhan data spesifik dari satu data (mencari datanya si ujang dengan menggunakan nis)**

**select \* form siswa where nis = ‘1123’**

****

**7. Mengubah data/update data (UPDATE)**

**update namatabel set apayangdiganti = ‘isiyangdiganti’ where dataspesifik = valuedaridataspesifik;**

menganti jurusan si ujang

update siswa set jurusan = ‘teknik kuli’ where nama = 1;

**8. Menghapus data (DELETE)**

**delete from namatabel where dataspesifik = valuedaridataspesifik;**

menghapus data SI MUMANG

delete from siswa where id = 3;

untuk mastering, di harapkan berlatihan di https://sqlbolt.com/

**atau bisa menggunanak phpmy admin…….**

**PEGABUNGAN PHP DAN SQL**

ada banyak cara menggunakan pegabunganya ini, ada Mysql (caralama), mysqli (lebih baru dan aman), PDO (PHP Data Object)

**HAL HAL WAJIB**

**Koneksi ke database**

$koneksi = mysqli\_connect("localhost", "root", "" , "phpdasar");

Fungsi sqli\_connect memiliki parameter secara turut menurut yakni : **host, namauser, pasword, namadatabse.**

**Mengquery/mencari/melakukan data di php**

mysqli\_query($koneksi, $sqlsintaks);

Fungsi mysqli\_query memiliki 2 parameter yakni: **pengkoneksian terhadap database , sintaks eksekusi dari sql**

**Peneksekusi sintaks**

$sqlsintaks = "SELECT \* FROM siswa ";

Ini harus ditambahkan di atas mysql query

// Ambil data siswa menggunakan query

$sqlsintaks = "SELECT \* FROM siswa ";

$hasil = mysqli\_query($koneksi, $sqlsintaks);

**Jika mau mengecek apakah sudah terkoneksi dan terjalankan dengan baik maka**

if ( !$hasil) {

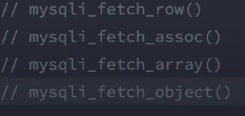
    mysqli\_error($koneksi);

}

Mysqli\_error berisi 1 parameter yang di isi dengan koneksian database

**Mengambil data dari sql (FETCH)**

Ada 4 cara di php untuk mengefetch data



Mysqli\_fetch\_row = memakai array numberic (angka)

**Mysqli\_fetch\_assoc = memakai array asosiatif (teks) yang paling bagus di pakai**

Mysqli\_fetch\_array = memakai array assoiatif dan numeric (fleksibel)

Mysqli\_fetch\_object = menggunakan objek.

**Menamilkan data**

while ($siswa = mysqli\_fetch\_assoc($hasil)){

    var\_dump($siswa["nama"]);

}

**Perulangan ini di jalankan selama ada data di database (dari hasil yang dikoneksikan ke database, dan menjalankan sql), untuk menampilkan permasing data gunakan metode assoiatif**

**TANTANGAN !!! , COBA PERLIHATKAN DATANYA DI TABELLLLL!!!!!!**

        <?php while($siswa = mysqli\_fetch\_assoc($hasil)){ ?>

        <tr>

            <td><?php echo $siswa["id"] ?></td>

            <td>

                <a href="">Ubah</a> |

                <a href="">Hapus</a>

            </td>

            <td><?php echo $siswa["nama"] ?></td>

            <td><?php echo $siswa["nis"] ?></td>

            <td><?php echo $siswa["jurusan"] ?></td>

            <td><?php echo $siswa["semester"] ?></td>

            <td><img src="" alt=""></td>

        </tr>

        <?php } ?>

**Marapihkan Kode PHP menggunakan fungsi**

Memisahkan mysqli\_connect ke file koneksi.php

Gunakaan require untuk mengimportnya

require\_once 'koneksi.php';

Menggunakan select di fungsi.php

<?php

require\_once 'koneksi.php';

function query($query){

    global $koneksi;

    $hasil = mysqli\_query($koneksi, $query);

    $rows = [];

    while ($row = mysqli\_fetch\_assoc($hasil)){

        $rows[] = $row;

    }

    return $rows;

}

?>

Index2.php

require\_once 'koneksi.php';

require\_once 'fungsi.php';

$siswadata = query("SELECT \* FROM siswa ");

       <?php foreach($siswadata as $siswatampil){ ?>

        <tr>

            <td><?php echo $siswatampil["id"] ?></td>

            <td>

                <a href="">Ubah</a> |

                <a href="">Hapus</a>

            </td>

            <td><?php echo $siswatampil["nama"] ?></td>

            <td><?php echo $siswatampil["nis"] ?></td>

            <td><?php echo $siswatampil["jurusan"] ?></td>

            <td><?php echo $siswatampil["semester"] ?></td>

            <td><img src="" alt=""></td>

        </tr>

        <?php } ?>

**PENJELASAN!!!!**

1. Kode index2.php, dan fungsi.php akan terkoneksi ke koneksi php yang berisi dengan mysqli\_query
2. **Kode index2.php ,** ada variabel yang menyimpan fungsi berisikan parameter SELECT \* FROM siswa.
3. Parameter itu akan di jalankan ke **fungsi.php**
4. Di dalam **fungsi.php** terdapat sebuah fungsi yang bernama query yang berisi $query (parameter yang tadi).
5. Di dalam fungsi query terdapat
6. Global $koneksi, **ini bertujuan supaya dapat mengabil variabel $koneksi di koneksi.php**
7. Terdapat variabel $hasil yang menajalankan $mysqli\_query yang mengambil data dari database lalu di jadikan sebagai array
8. Terdapat **$rows** yang akan menyimpan array asosiatif (perulangan di bawah)
9. Terdapat perulangan yang di mana **mewakilakan $row , lalu melakukan pengambilan array secara assosiatif bedasarkan mysqli\_query**
10. Setelah row di wakilkan, **$row akan di masukan ke $rows**
11. Mengembalikan nilai $rows, agar nanti dipanggil lagi di index2.php
12. **Pada index2.php** , setelah melakukan perulangan fungsi tersebut akan tersimpan di fungsi query tersebut. Fungsi query ini di simpan ke $siswadata
13. **$siswadata** yang memiliki lebih dari 1 data akan melalukan perulangan foreach asosiatif. Lalu mengatasnamakan $siswatampil lalu di tampilkan.

**INSERT PHP SQL**

Kita harus buat dulu form yang berisi dengan suatu indentitas.

    <form action="" method="post">

        <label for="nama">Nama:</label>

        <input type="text" id="nama" name="nama"><br><br>

        <label for="nis">Nis:</label>

        <input type="text" id="nis" name="nis"><br><br>

        <label for="jurusan">Jurusan:</label>

        <input type="text" id="jurusan" name="jurusan"><br><br>

        <label for="semester">Semester:</label>

        <input type="text" id="semester" name="semester"><br><br>

        <br>

        <button type="submit" name="submit">Tambah data</button>

    </form>

Ditabel ini pastikan setiap inputnya sudah berisi **name dengan metod post.** Ini akan di tangkap oleh kode php. Ingat POST tidak akan menampilkan data di link.

**Kode PHP**

Pertama kita include dulu koneksi.php

require\_once 'koneksi.php';

Ketika ada data ditekan dari button , akan menjalankan kode di bawah ini

if(isset($\_POST["submit"])){

    $nama = $\_POST["nama"];

    $nis = $\_POST["nis"];

    $jurusan = $\_POST["jurusan"];

    $semester = $\_POST["jurusan"];

    $query = "INSERT INTO siswa VALUES (NULL, '$nama', '$nis', '$jurusan', '$semester', NULL) ";

    $hasil = mysqli\_query($koneksi, $query);

}

**Kode ini terdri dari 3 bagian:**

1. Pengambilan data dari form menggunakan $\_POST, ini bedasarkan data dari **name di html**

    $nama = $\_POST["nama"];

    $nis = $\_POST["nis"];

    $jurusan = $\_POST["jurusan"];

    $semester = $\_POST["jurusan"];

1. Sintaks SQL. Mengisi tabel siswa dengan value yang bedasarkan data yang di ambil dari form

    $query = "INSERT INTO siswa VALUES (NULL, '$nama', '$nis', '$jurusan', '$semester', NULL) ";

1. Eksekusi Koneksi dan sql

    $hasil = mysqli\_query($koneksi, $query);

**Pengecekan EROR**

    if(mysqli\_affected\_rows($koneksi) >0){

        echo "berhasil";

    }

    else{

        echo "gagal". mysqli\_error($koneksi);

    }

Fungsi dari mysqli\_affected\_rows yakni mengecek setiap perubahan di database. **Ketika Nilainya 1 = berhasil dan ada perubahan, 0 = tidak ada perubahan, -1 akan menjalankan eror.**

**Penyempurnaan dengan fungsi.**

**Fungsi.php**

function tambah($data){

    global $koneksi;

    $nama = $data["nama"];

    $nis = $data["nis"];

    $jurusan = $data["jurusan"];

    $semester = $data["jurusan"];

    $query = "INSERT INTO siswa VALUES (NULL, '$nama', '$nis', '$jurusan', '$semester', NULL) ";

    $hasil = mysqli\_query($koneksi, $query);

    return $hasil;

}

**Tambah.php**

if(isset($\_POST["submit"])){

    if(tambah($\_POST)){

        echo"oke";

    }

    else{

        echo"error";

    }

}

**PENJELASAN!!!**

**Fungsi.php**

1. Di sediakan sebuah fungsi yang berisi parameter **$data**. Parameter $data ini berasal dari tambah.php. **INI SAMA AJAH DENGAN POST**
2. Dikarenakan **$koneksi** berada di luar fungsi maka harus menggunakan global variabels.
3. $nama = $data[“nama”], itu sama aja dengan $nama=$\_post[“nama”], dikarenakan data parameter nya itu $\_post
4. $query berisi sintaks sql
5. Setelah itu mysqli\_query akan mengeksekusi koneksi dan query
6. Merturnkan $hasil alias mysqli\_query.

**Tambah.php**

1. Melakukan pengecekan apakah beneran sudah di submit
2. Memangil fungsi tambah dengan isi parameter $\_POST yang akan mereplace $data
3. Ketika data di return berhasil maka akan mengluarkan oke
4. Ketika Data di return gagal maka akan mengluarkan eror

**Penambahan pengecekan error lebih lanjut & script otamatisasi agar balik kehalaman**

function tambah($data){

    global $koneksi;

    $nama = $data["nama"];

    $nis = $data["nis"];

    $jurusan = $data["jurusan"];

    $semester = $data["jurusan"];

    $query = "INSERT INTO siswa VALUES (NULL, '$nama', '$nis', '$jurusan', '$semester', NULL) ";

    mysqli\_query($koneksi, $query);

    return mysqli\_affected\_rows($koneksi);

}

Pada fungsi.php, perubahan nya di return dan $hasil di hapus. Di sini kita pakai mysqli\_affected\_rows yang berisi $koneksi. Fungsi ini akan megecek setiap perubahan data. Ketika **data bernilai 0 artinya tidak ada perubahan, ketika nilainya 1 ada perubahan, dan ketika nilainya -1 artinya eror!**

if(isset($\_POST["submit"])){

    if(tambah($\_POST)>0){

        echo"

        <script>

        alert('data berhasil ditambahkan');

        document.location.href = 'index2.php'

        </script>

        ";

    }

    else{

        echo "

                <script>

        alert('data gagal ditambahkan');

        document.location.href = 'index2.php'

        </script>

        ";

    }

}

Pada tambah.php, di tambahakan sedikit validasi di if (tambah($\_POST) > 0), artinya ketika nilainya lebih besar dari 0 , yang berarti fungsi mysqli\_affected\_rows nya 1 **artinya berhasil!**, selain itu eror, atau tidak di tambahkan datanya.

Terdapat juga script javascript yang akan menampilkan alert. Ketika **tombol oke di alert di pencet** maka akan mengarahkan ke index.php

**Penambahan KEAMANAN DASAR !!**

1. Gunakan lah requierd di html
2. Html special chars, ini digunakan agar tidak ada orang iseng memasukin script html di sql, yang menyebabkan menggangu tampilan admin

    $nama = htmlspecialchars($data["nama"]);

    $nis = htmlspecialchars($data["nis"]);

    $jurusan = htmlspecialchars($data["jurusan"]);

    $semester = htmlspecialchars($data["jurusan"]);

**DELETE PHP SQL**

Masih ingat dengan bagaimana cara menghapus data di mysql?. Kita perlu data kolom berapa dihapus. Ini kita tandai dengan kunci yakni ID, untuk mengetahui id setiap elemen kita harus tulis kode ini

                <a href="hapus.php?id=<?php echo $siswatampil["id"]?>">Hapus</a>

Fungsi.php

function hapus($id){

    global $koneksi;

    $query = "DELETE FROM siswa WHERE id = '$id'";

    mysqli\_query($koneksi, $query);

    return mysqli\_affected\_rows($koneksi);

}

Penjelasan:

1. Terdapat fungsi hapus yang berparameter $id, ini didapatkan dari hapus.php yang dimana ada $\_GET[“id”] dari url.
2. Terdapat sintaks sql yang mengahpus data tabel siswa yang bedasarkan id
3. Eksekusi query
4. Validasi, 1= berhasil, 0, gagal, -1 eror

Hapus.php

require 'fungsi.php';

$id = $\_GET["id"];

if(hapus($id) > 0){

            echo"

        <script>

        alert('data berhasil dihapus');

        document.location.href = 'index2.php'

        </script>

        ";

}

else{

            echo "

                <script>

        alert('data gagal ditambahkan');

        document.location.href = 'index2.php'

        </script>

        ";

}

Ini lah yang akan mengirim parameter $id bedasarkan $\_GET[“id”] dari link…..

**UPDATE DATA**

Agar tahu data mana yang akan di ubah bedasarkan link, maka kita harus begini. Ini akan menyimpan array “id” dengan data dari mysql yakni id. Ini juga akan diambil oleh metod GET

            <td>

                <a href="ubah.php?id=<?php echo $siswatampil["id"]; ?>">Ubah</a> |

Oke pada update.php . Ketika kita salah input data/typo. Maka kita tampilkan lagi namanya per input.

Pertama tama, kita wajib mengambil id dari link tersebut menggunakan $\_GET

$id = $\_GET['id'];

Dikarenakan kita sudah membuat query select, kita tinggal panggil fungsi nya saja, dengan parameter yang berbeda dengan sebelumnya

$datasiswa = query("SELECT \* FROM siswa WHERE id = $id");

Notes: ini berarti : ambil data semua data dari siswa, tetapi dengan id = dengan $id (didapatkan dari GET).

Karena elemen paling luarnya menggunakan ARRAY NUMERIC, kita harus ambil luaranya menggunakan foreach. Di setiap input bervalue dengan $siswatampil.

    <form action="" method="post">

        <?php foreach ($datasiswa as $siswatampil){ ?>

        <input type="hidden" name="id" value="<?php echo $siswatampil["id"] ?>">

        <label for="nama">Nama:</label>

        <input type="text" id="nama" name="nama" required value="<?php echo $siswatampil["nama"]  ?>"><br><br>

        <label for="nis">Nis:</label>

        <input type="text" id="nis" name="nis" required value="<?php echo $siswatampil["nis"]  ?>"><br><br>

        <label for="jurusan">Jurusan:</label>

        <input type="text" id="jurusan" name="jurusan" required value="<?php echo $siswatampil["jurusan"]  ?>"><br><br>

        <label for="semester">Semester:</label>

        <input type="text" id="semester" name="semester" required value="<?php echo $siswatampil["nama"]  ?>"> <br><br>

        <br>

        <button type="submit" name="submit">Ubah data</button>

        <?php } ?>

    </form>

Di fungsi.php, kita buat fungsi ubah.

Pertama kita deklarasikan $koneksi sebagai global variabel agar bisa di baca dalam fungsi

    global $koneksi;

lalu kita ambil data dengan $\_POST

    $nama = htmlspecialchars($data["nama"]);

    $nis = htmlspecialchars($data["nis"]);

    $jurusan = htmlspecialchars($data["jurusan"]);

    $semester = htmlspecialchars($data["jurusan"]);

    $id = $data["id"];

NOTES : $data itu adalah $\_POST. Ini di dapat dari pemangilan fungsi di ubah.php

Laksanakan dan eksekusi query

    $query = "UPDATE siswa SET

    nama = '$nama',

    nis = '$nis',

    jurusan = '$jurusan',

    semester = '$semester'

    where id = $id

     ";

    mysqli\_query($koneksi, $query);

Sintaks query ini berarti : Update tabel siswa, isikan nama (data dari sql) dengan $nama (data dari $\_POST yang dikirim dari form) yang dimana id (dari sql) = $id (data dari form $\_POST)

Pengecekan perubahan

    return mysqli\_affected\_rows($koneksi);

Jika berhasil terubah maka akan menghasilkan 1, tidak ada yang terjadi 0, eror -1.

Ketika value nya lebih dari 0 alias 1. Maka berhasil lalu di alertkan.

if(isset($\_POST["submit"])){

    if(ubah($\_POST)>0){

        echo"

        <script>

        alert('data berhasil diubah');

        document.location.href = 'index2.php'

        </script>

        ";

    }

    else{

        echo "

                <script>

        alert('data gagal diubah');

        document.location.href = 'index2.php'

        </script>

        ";

    }

}

**SEARCHING**

Searching menggunakan metode read/select dengan kriteria tertentu, untuk membuat kondisi tersebut, kita menggunakan **WHERE** yang dilanjutkan oleh suatu nama.

**Index2.php**

if( isset($\_POST["cari"])){

    $siswadata = cari($\_POST["keyoword"]);

}

Kalau tombol cari di pencet, ambil yang diketikan ke user, lalu jalankan fungsi cari. Dengan menimpa hasil dari siswadata.

**Fungsi.php**

function cari($keyword){

    $query = "SELECT \* FROM siswa WHERE nama = '$keyword'";

    return query($query);

}

Ini akan mengembalikan hasil fungsi query yang dimana mencari satu persatu nama.

Jika mau pencarian fleksibel, misal ada nama orang “manusia ujang” jika kita tulis manusia maka akan di jari yang semua nama yang namanya manusia, ataupun ketika kita typo.

    $query = "SELECT \* FROM siswa WHERE nama LIKE '%$keyword%'";

**LIKE** di gunakan untuk mencari data “seperti”, dan % itu apapun namanya. Ataupun kita ingin mencari sekaligus nis, jurusan, kita bisa menggunakan **OR** di sql.

    $query = "SELECT \* FROM siswa WHERE nama LIKE '%$keyword%' OR nis LIKE '%$keyword%' OR jurusan LIKE '%$keyword%'";

**UPLOAD FILE PHP**

Pada materi ini kita menggunakan **file handling** yang dimana sintaks php menangani suatu file tersebut. Yang akan kita pelajari:

<input type=’file>

enctype (encoding type)

$\_FILES (super globals variabel khusus untuk menangani data file).

move\_uploaded\_file (fungsi ini untuk mengupload file di komputer ke file itu sendiri).

Dikarenakan pada **tambah.php** , form nya bersifat post, maka tambahkan ini :

enctype="multipart/form-data"

enctype ini digunakan agar form data alias file nya di tangkap oleh **$\_FILES.** Untuk memastikan apakah file nya sudah kebaca oleh $\_FILES, maka coba lah vardump seperti kode yang ada di bawah ini :

    var\_dump($\_FILES['gambar']); die;

Jika terlihat ada berarti, file sudah berhasilo dibaca oleh superglobal variabel $\_FILES

|  |
| --- |
| [“gambar”]  ["name"]=> |
| string(32) "Screenshot 2025-09-03 200207.png" |
| ["full\_path"]=> |
| string(32) "Screenshot 2025-09-03 200207.png" |
| ["type"]=> |
| string(9) "image/png" |
| ["tmp\_name"]=> |
| string(45) "C:\Users\cokvr\AppData\Local\Temp\phpE25D.tmp" |
| ["error"]=> |
| int(0) |
| ["size"]=> |
| int(31777) |

Pada tambah.php di fungsi tambah, tambahkan :

    $gambar = upload();

    if (!$gambar){

        return false;

    }

$ gambar akan di ambil dari fungsi upload. Ketika $gambar **tidak** bernilai true, maka batalkan kode nya alias retrun false.

Pada tambah.php di fungsi upload

function upload(){

    $namafile = $\_FILES['gambar']['name'];

    $ukuranfile = $\_FILES['gambar']['size'];

    $eror = $\_FILES['gambar']['error'];

    $tmpName = $\_FILES['gambar']['tmp\_name'];

    // cek apakah ada gambar di upload menggunakan 4

    if($eror === 4){

        echo "

        <script>

        alert('pilih gambar dulu');

        </script>

        ";

        return false;

    }

}

Kita mengambil data bedasarkan name gambar, **yang di mana name gambar ini adalah array multidimensi.** Jadi kita bisa ambil sesuatu di dalam array yang di sebut array gambar.

Bagian ceking :

**Pengecekan apakah ada file di upload?**

    if($eror === 4){

        echo "

        <script>

        alert('pilih gambar dulu');

        </script>

        ";

        return false;

    }

Dikarenakan secara bawaan nilai eror kalau ga ada gambar itu adalah **4** maka buat seperti itu.

**Pengecekan apakah file tersebut gambar?**

    // cek apakah file di upload itu adalah gambar.

    $ekstensiGambardibolehkan = ['jpg', 'jpeg', 'png'];

    $ekstensiGambar = explode('.', $namafile);

    $ekstensiGambar = strtolower(end($ekstensiGambar));

    if(!in\_array($ekstensiGambar, $ekstensiGambardibolehkan)){

        echo "

        <script>

        alert('ini bukan gambar coy!!!!');

        </script>

        ";

        return false;

    }

**Pertama tama,** kita buat sesuka kita, ekstensi apa saya yang diperbolehkan. Di sini ada ekstensi umum yakni jpg, jpeg, png.

**Kedua**, kita bedah string dari nama filenya menjadi aray dengan patokan “.” . Semisal :

manusia.jpg **akan menjadi** [“manusia”, “jpg”];

**Ketiga** , kita ubah semua katanya itu menjadi huruf kecil, ini digunakan ketika user menggunakan file ekstensi huruf besar. Setelah itu mencari **ARRAY TERKAHIR DARI KUMPULAN ARRAY.**

Manusia.JPG **akan menjadi** [“manusia”, “jpg”]; yang lalu **dicari array paling belakan yakni “jpg”**

**Keempat,** mencocokan apakah ekstensi gambar yang diupload oleh **user** diperbolehkan oleh **developer** bedasarkan **array $ekstensiGambardiperbolehkan.** Ini menggunakan konsep menemukan jarum di dalam jerami.

Manusia.pdf file ini tidak ada di $ekstensiGambaryangdibolehkan.

**Membatasi Ukuran File**

Karena di dalam php menghitung sesuatu itu dengan byte, maka kita konversikan suatu satuan ukuran ke byte, jika ingin cepat kita dapat menggunakan converter di beberapa webiste seperti : <https://www.gbmb.org/mb-to-bytes>

    if($ukuranfile > 4194304){

        echo "

        <script>

        alert('gambar terlalu besar');

        </script>

        ";

        return false;

    }

**Siap di upload dengan memindahkan temp file ke file asli di server 😊**

    move\_uploaded\_file($tmpName, 'img/'. $namafile);

    return $namafile;

}

Di dalam fungsi move\_uploaded\_file yang berarti memindahkan file dari **tmp**, **lalu ke file server dengan** **path** dan **nama file.**

**Kalau ada user yang mengupload file yang sama?**

Kita bisa menggunakan fungsi uniq id() yang akan mengrandom angka dan tulisan..

    $namafileBaru = uniqid();

    $namafileBaru .= '.';

    $namafileBaru .= $ekstensiGambar;

**Pertama tama**, nama file akan di random namnaya, lalu digabungkan dengan titik, lalu digabungkan dengan ekstensi gambar contoh : fjklsdj2130ssf.jpg

**Hapus Gambar pada Upload**

    // cari nama gambar di database

    $gambar = query("SELECT gambar FROM siswa WHERE id = $id")[0]['gambar'];

    // hapus file kalau ada

    if ($gambar && file\_exists("img/$gambar")) {

        unlink("img/$gambar");

    }

Pertama, kita menjalankan query sql yang adai di fungsi query(). Digunakan untuk mengambil gambar. Gambar ini merupakan array multi dimensi dimana ada array numeric dan juga array asosiatif berupa gambar. **Pengahpusan file utuh menggunakan unlink** yang beriisi **path** dan **nama gambar**.

**Update Data pada Upload.**

Saat update data, pengguna kan ga mesti juga update gambar dong?, oleh karen itu, kita harus mengambil data gambar sebelum di update alias data lama.

$sqlGambarLama = mysqli\_query($koneksi, "SELECT gambar FROM stok WHERE id = '$id'");

$dataLama = mysqli\_fetch\_assoc($sqlGambarLama);

$gambarLama = $dataLama['gambar'];

**LOGIN SYSTEM**

**REGISTRASI**

**Front End**

<head>

    <meta charset="UTF-8">

    <meta name="viewport" content="width=device-width, initial-scale=1.0">

    <title>Halaman Registrasi</title>

    <style>

    label {

        display: block;

    }

    </style>

</head>

<body>

    <h1>Halaman Registrasi</h1>

    <form action="" method="post">

        <label for="nama" id="nama">Nama :</label>

        <input type="text" name="nama" id="nama">

        <br>

        <label for="password" id="password">Passoword :</label>

        <input type="password" name="password" id="password">

        <br>

        <label for="password2" id="password2">Konfirmasi Pw :</label>

        <input type="password" name="password2" id="password2">

        <br>

        <button type="submit" name="registrasi">Registrasi !</button>

    </form>

</body>

</html>

Pada bagian ini, menggunakan form yang bertipe post agar tidak terlihat di link.

require\_once 'koneksi.php';

require\_once 'fungsi.php';

if (isset($\_POST["registrasi"])){

    if(registrasi($\_POST) > 0){

        echo "

        <script>

        alert('akun berhasil ditambahkan!');

        </script>

        ";

    }

    else{

        echo mysqli\_error($koneksi);

    }

}

Jika tombol registrasi di tekan maka, jika registrasi bernilai true alias 1 maka akun berhasil di tambahkan selain itu, eror.

**BACK END**

function registrasi ($data){

    global $koneksi;

    $nama = strtolower(stripslashes($data["nama"]));

    $password = $data["password"];

    $password2 = $data["password2"];

    // cek apakah ada user yang sama namanya?

    $hasil = mysqli\_query($koneksi, "SELECT nama FROM user WHERE nama = '$nama'");

    if (mysqli\_fetch\_assoc($hasil)){

        echo "

        <script>

        alert('Username sudah pernah di pakai!');

        </script>

        ";

        return false;

    }

    // cek konfirmasi password.

    if($password !== $password2){

        echo "

        <script>

        alert('Password konfirmasi tidak sama dengan password!');

        </script>

        ";

        return false;

    }

    // mengenkripsi password

    $password = password\_hash($password, PASSWORD\_DEFAULT);

    // menambahkan data

    $sqlsintaks = "INSERT INTO user VALUES (NULL, '$nama', '$password')";

    mysqli\_query($koneksi, $sqlsintaks);

    return mysqli\_affected\_rows($koneksi);

}

Pada segment ini terdapat beberapa bagian kode yakni sebagai berikut:

**1. Pengambilan data dari front end**

    global $koneksi;

    $nama = strtolower(stripslashes($data["nama"]));

    $password = $data["password"];

    $password2 = $data["password2"];

Kode di atas mengambil $nama yang akan dijadikan **huruf kecil, menghapus semua simbol “/”** . Lalu mengambil passoword dan data konfirmasi passoword.

**2. Mengecek ke database tabel user.**

    // cek apakah ada user yang sama namanya?

    $hasil = mysqli\_query($koneksi, "SELECT nama FROM user WHERE nama = '$nama'");

    if (mysqli\_fetch\_assoc($hasil)){

        echo "

        <script>

        alert('Username sudah pernah di pakai!');

        </script>

        ";

        return false;

    }

Kode tersebut berguna untuk mengecek apakah ada user yang sama namanya?

1. Mengambil data dengan mysqli\_query dengan parameter $koneksi dan pengambilan data melalui kode SQL yakni bermakna “pilih nama dari tabel user yang dimana nama sama dengan nama yang di dapat dari user di frontend ($\_POST).
2. Lalu mysqli\_fetch\_assoc mengambil hasilnya. Jika nilainya sama alias TRUE , maka tampilkan eror menggunakan alert dan berhentikan kode.

**3. Pengecekan konfrimasi password**

    if($password !== $password2){

        echo "

        <script>

        alert('Password konfirmasi tidak sama dengan password!');

        </script>

        ";

        return false;

    }

Pada kode diatas berbunyi : jika passoword itu ***tidak sama dengan***  password yang dikonfirmasi, maka tampilkan alert gagal, dan hentikan kode.

**4. Enkripsi passoword**

    $password = password\_hash($password, PASSWORD\_DEFAULT);

Sesunguhnya banyak metode enskirpsi password seperti menggunakan fungsi md5(), tetapi md5() masih bisa di jebol pakai metode pengdeskripsian. Oleh karena itu metode pengenikrisian password pada php adalah password\_hash yang **berparemeter nilai password serta menabahkan metode yang dinginkan, disini kita pakai yang deafult.**

**5. Penambahan data**

    $sqlsintaks = "INSERT INTO user VALUES (NULL, '$nama', '$password')";

    mysqli\_query($koneksi, $sqlsintaks);

Di sintaksi ini menambahkan data pada tabel user yang bearada di bawah pengoneksian database yang berisi sql sintaks yang berbunyi “isi user dengan value id = otomatis, nama = “$nama (didapatkan dari input user)”, “$password” (yang di dptkan dari input user dan sudah di enskripsi) )”.

**6. Pengecekan EROR**

    return mysqli\_affected\_rows($koneksi);

jika hasil 1 = berhasil, 0 tidak ada data yang berubah, -1 eror.