

File Upload

Pada bab ini kita akan membuat proses upload. Untuk lanjut lebih dalam maka kita harus mengerti terlebih dahulu file *handling*. File *handling* adalah suatu cara untuk menangani file dengan menggunakan sintak php. Ada beberapa hal yang harus kita ketahui dari file *handling* ini, antara lain

1. **<input type="file".....>**

Ini adalah sebuah **input** pada php dengan **atribut type-nya** adalah **file**. Jika kita lihat *code* sebelum-sebelumnya **type** dari **input** adalah **text**, untuk kali ini atau untuk proses *upload* ini kita membutuhkan **type file** untuk mengolah file kita.

2. **enctype (encoding type) pada form**

Ini merupakan salah satu atribut pada *form* untuk menentukan *encoding* apa yang akan kita gunakan nanti.

3. **\$_FILES**

Ini adalah salah satu variabel superglobal. Superglobal ini digunakan khusus untuk menangani files nantinya

4. **move_uploaded_file**

Ini adalah sebuah *function* pada php yang digunakan untuk memindahkan file ke dalam *computer* ke *server*

Pada database kita dapat memasukkan data berupa gambar dengan menggunakan **type BLOB (Binary Large Object)**. Tetapi jika kita menggunakan cara ini maka ukuran *database* akan menjadi lebih besar dan membuat *database* akan menjadi berat dalam proses pengolahan data. **Jadi nanti kita akan melakukan cara lain untuk proses upload, yaitu dengan cara kita hanya menyimpan nama gambar dari file gambar yang akan kita upload, dimana gambar tersebut disimpan di sebuah direktori.** Setelah kita mengetahui tentang file *handling* dan cara untuk melakukan proses penyimpanan pada database maka langkah selanjutnya adalah kita mulai dengan proses *coding*. Untuk langkah-langkah proses *coding*-nya sebagai berikut:

- 1) Kita akan memodifikasi **tambah.php (Gambar 7.1)**. Hal ini kita lakukan karena pada proses-proses sebelumnya kita, pada proses tambah data pada *field* gambar kita hanya memasukkan nama dari gambar tersebut dan dimana gambar sudah ada di *folder image*.

```

<h1>Tambah Data Mahasiswa</h1>
<form action="" method="post" enctype="multipart/form-data">
  <ul>
    <li>
      <!-- for pada label terhubung dengan id jadi jika label nama diklik maka textfield nama akan aktif juga-->
      <label for="Nama">Nama :</label>
      <!-- untuk pengisian name besar kecilnya harus sesuai dengan fieldnya -->
      <input type="text" name="Nama" id="Nama" >
    </li>
    <li>
      <label for="Nim">Nim :</label>
      <input type="text" name="Nim" id="Nim" required >
    </li>
    <li>
      <label for="Email">Email :</label>
      <input type="text" name="Email" id="Email" required>
    </li>
    <li>
      <label for="Jurusan">Jurusan :</label>
      <input type="text" name="Jurusan" id="Jurusan" required>
    </li>
    <li>
      <label for="Gambar">Gambar :</label>
      <input type="file" name="Gambar" id="Gambar" required>
    </li>
    <li>
      <button type="submit" name="submit"> Tambah </button>
    </li>
  </ul>

```

Gambar 7.1 tambah_data.php

Keterangan Gambar 7.1

- Pada *line 67* kita rubah **type** yang sebelumnya adalah **text** kita rubah menjadi **file**. Maksud dari kita merubah nilai dari atribut *type* ini adalah karena kita akan mengolah file maka nilai dari *type* bukan lagi *text* tetapi file.
- Pada *line 45* pada **form** kita tambahkan atribut **enctype="multipart/form-data"**. Maksud dari kita tambahkan atribut ini adalah untuk melakukan proses *encoding* file (untuk lebih jelasnya ada pada langkah berikutnya).

2) Untuk lebih jelas fungsi dari enctype maka kita lakukan var_dump pada tambah_data.php

```

tambah_data.php x
1  <?php
2      require 'functions.php';
3      // cek apakah button submit sudah di tekan atau belum
4      if(isset($_POST['submit']))
5      {
6
7          // cek isi dari post menggunakan vardump
8          var_dump($_POST);
9          var_dump($_FILES);
10         die();

```

Gambar 7.2 var_dump

3) Setelah kita tambahkan data pada tambah_data.php dan klik view_page_source

```

3 array(5) {
4   ["Nama"]=>
5   string(4) "dita"
6   ["Nim"]=>
7   string(4) "1234"
8   ["Email"]=>
9   string(14) "dita@gmail.com"
10  ["Jurusan"]=>
11  string(6) "bisnis"
12  ["submit"]=>
13  string(0) ""
14 }
15 array(1) {
16  ["Gambar"]=>
17  array(5) {
18    ["name"]=>
19    string(8) "dina.jpg"
20    ["type"]=>
21    string(10) "image/jpeg"
22    ["tmp_name"]=>
23    string(45) "/Applications/XAMPP/xamppfiles/temp/phpw6EP2J"
24    ["error"]=>
25    int(0)
26    ["size"]=>
27    int(295527)
28  }
29 }

```

Gambar 7.3 Hasil view_page_source

Jika kita lihat hasil dari gambar 7.3 maka kita dapat lihat bahwa ada 2 *array associative* disana. *Line* 3 sampai *line* 14 itu merupakan *array* pertama yang merupakan hasil `var_dump($_POST)` sedangkan *line* 15 sampai *line* 29 merupakan hasil dari `var_dump($_FILES)`. Jika kita lihat hasil dari dump files merupakan *array* 2 dimensi (["Gambar"]["name"] dst) sedangkan pada hasil dump post merupakan *array* 1 dimensi ["Nama"]. Keterangan Gambar 7.3 pada *line* 18 sampai 27 adalah sebagai berikut:

- a. *Line* 18 menunjukkan bahwa nama filenya adalah `dina.jpg`
- b. *Line* 20 menunjukkan *type* dari file tersebut adalah `image/jpeg`
- c. *Line* 22 menunjukkan dimana file tersebut disimpan sementara
- d. *Line* 24 menunjukkan jika terdapat *error*
- e. *Line* 26 menunjukkan ukuran dari file tersebut

4) Memodifikasi *function* tambah pada `functions.php` (Gambar 7.4)

```

26 function tambah ($data)
27 {
28     global $conn;
29
30     $nama=htmlspecialchars($data["Nama"]);
31     $nim=htmlspecialchars($data["Nim"]);
32     $email=htmlspecialchars($data["Email"]);
33     $jurusan=htmlspecialchars($data["Jurusan"]);
34     // $gambar=htmlspecialchars($data["Gambar"]);
35
36     $gambar=upload();
37     if(!$gambar)
38     {
39         return false;
40     }
41
42     $query= " INSERT INTO mahasiswa
43             VALUES
44             ('','$nama','$nim','$email','$jurusan','$gambar')";
45     mysqli_query($conn,$query);
46
47     return mysqli_affected_rows($conn);
48 }
49

```

Gambar 7.4 functions.php

Keterangan gambar 7.4

- a. Pada *line* 36 \$gambar adalah variabel yang digunakan untuk menampung hasil dari *function* *upload*
- b. *Line* 37 sampai *line* 40 adalah sebuah kondisi jika tidak ditemukan \$gambar (!\$gambar) maka akan dikembalikan ke nilai *false*
- c. Pada *line* 34 comment saja karena sudah tidak digunakan lagi

5) Buat *function* *upload* pada functions.php (Gambar 7.5)

```

51 function upload()
52 {
53     $nama_file = $_FILES['Gambar']['name'];
54     $ukuran_file = $_FILES['Gambar']['size'];
55     $error = $_FILES['Gambar']['error'];
56     $tmpfile = $_FILES['Gambar']['tmp_name'];
57
58     if($error===4)
59     {
60         //pastikan pada inputan gambar tidak terdapat atribut required
61         echo "
62             <script>
63                 alert('Tidak ada gambar yang diupload');
64             </script>
65         ";
66         return false;
67     }
68
69     $jenis_gambar=['jpg','jpeg','gif'];
70     $specah_gambar=explode('.', $nama_file);
71     $specah_gambar=strtolower(end($specah_gambar));
72     if(!in_array($specah_gambar,$jenis_gambar))
73     {
74         echo "
75             <script>
76                 alert('yang anda upload bukan file gambar');
77             </script>
78         ";
79         return false;
80     }
81
82
83     // cek kapasitas gambar dalam byte kalau 1000000 byte = 1 Megabyte
84     if($ukuran_file > 1000000)
85     {
86         echo "
87             <script>
88                 alert('ukuran gambar terlalu besar');
89             </script>
90         ";
91         return false;
92     }
93
94     move_uploaded_file($tmpfile,'image/'.$nama_file);
95
96     // kita return nama file nya agar dapat masuk ke $gambar=$upload() pada function tambah
97     return $nama_file;
98 }

```

Gambar 7.5 *Function update*

Keterangan gambar 7.5

1. Line 53 sampai 56 adalah variabel baru yang gunanya adalah menampung isi data *array* (**data hasil array pada gambar 7.3**). Tulisan ['Gambar'] adalah nama dari inputan gambar (**gambar 14.1 line 67**), sedangkan ['name'], ['size'] dstnya adalah nama dari *array* (**gambar 7.3 line 18 dst**).
2. Line 58 sampai 67 adalah sebuah kondisi untuk mengecek *error*. jika terjadi *error* kategori 4 (tidak ada data masukan) (<http://php.net/manual/en/features.file-upload.errors.php>) maka terdapat pesan *error* gambar belum di-*upload*
3. Line 70 pada line ini kita tentukan ekstensi gambar apa saja yang boleh di *upload* oleh *user*.
4. Line 71 kita akan memecah string menjadi *array* dengan menggunakan fungsi *explode* (***explode (delimiter,string)***). Di dalam *function explode* terdapat parameter dengan nama *delimiter* yang nanti digunakan untuk memecah string ke *array* (kita set string apa yang digunakan sebagai pemisah, sebagai contoh titik maka ***delimiter*** akan memecah *string* ke *array* jika terdapat *string* titik). Sebagai contoh **andi.jpeg** maka ***delimiter*** akan memecahnya menjadi **['andi','jpeg']**.

5. *Line 72* artinya Setelah kita pecah menjadi *array* maka selanjutnya adalah mengambil string paling akhir menggunakan ***end*** sebagai contoh

a. *andi.jpeg* setelah dipecah menjadi ['*andi*', '*jpeg*'] maka akan diambil *string* terakhir yaitu '*jpeg*'

b. *andi.boy.jpeg* setelah dipecah menjadi ['*andi*', '*boy*', '*jpeg*'] maka *string* yang akan diambil adalah '*jpeg*'

Setelah kita ambil *string* yang paling belakang yaitu ekstensinya maka langkah berikutnya adalah mengkonversi huruf besar menjadi huruf kecil menggunakan ***strtolower***, agar kita tidak perlu lagi menulis 2 kali pada *line 70 (step)*

6. *Line 73* kita akan mengecek apakah ada *string* didalam *array* menggunakan *function in_array(\$needle, \$haystack) -> in_array(\$pecah_gambar, \$jenis_gambar)* maksudnya isi dari *\$pecah_gambar* semisal *jpeg* dicek apakah ada pada *array* pada *\$jenis_gambar* jika tidak ada maka akan muncul pesan ***error***.

7. *Line 84* sampai *line 92* ini digunakan untuk mengecek ukuran file jika lebih dari 1 Mb maka akan muncul pesan ***error***

8. *Line 94* jika data sudah dicek mulai dari ada tidaknya file yang diupload, jenis filenya dan ukuran filenya maka langkah berikutnya adalah memindahkan dengan fungsi

move_uploaded_file ('tempat penyimpanan sementara', 'tempat penyimpanan sesungguhnya') -> move_uploaded_file(\$tmpfile, 'img/'. \$nama_file) ingat tanda titik digunakan untuk menggabungkan string. (CATATAN: *folder img* sesuaikan dengan nama *folder* kalian dan sesuaikan dengan *path folder image* anda)

9. *Line 97 return \$nama_file* langkah terakhir adalah kita kembalikan nilai dari *\$nama_file*. Kenapa harus dikembalikan ke *\$nama_file* -> coba kita lihat kembali pada *function* tambah terdapat *code \$gambar=upload()* jadi kita kembalikan nilai dari *\$nama_file* agar dapat disimpan sementara pada *\$gambar* pada *function* tambah

6) Jika berhasil maka image yang di-*upload user* akan berpindah ke *folder image*

7) Jika kita jalankan masih ada beberapa kekurangan seperti

a. Jika kita tambahkan data dengan nama gambar sama dengan yang sudah ada maka akan terjadi penumpukan (Gambar 7.6)

1. Tampilan awal

Daftar Mahasiswa

[Tambah Data Mahasiswa](#)

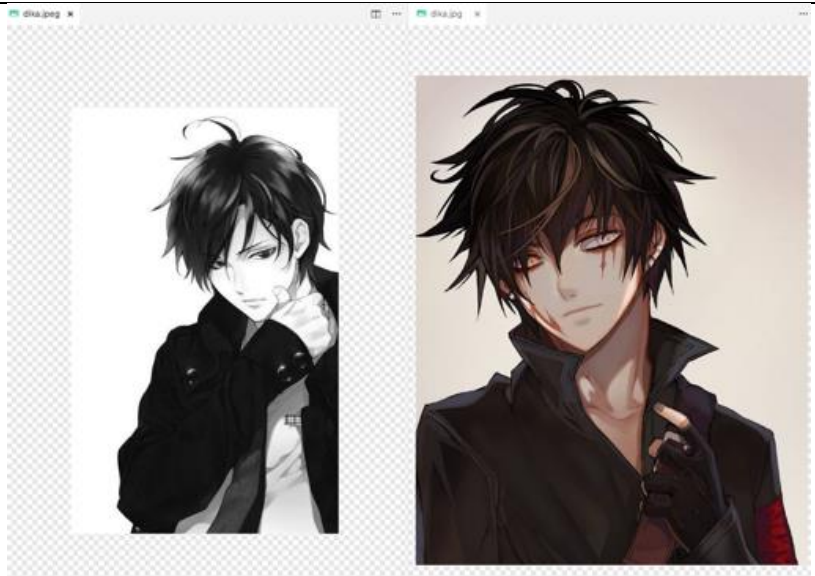
No.	Nama	Nim	Email	Jurusan	Gambar	Aksi
1	Dinos	1234568	dino@gmail.com	Ekonomi		Edit Hapus
2	Dito	1234	dito@gmail.com	mesin		Edit Hapus
3	dito bagus	123455	bagas@gmail.com	akutansi		Edit Hapus
4	dito bagus saputra	123455	bagas@gmail.com	akutansi		Edit Hapus
5	dika	1234	dika@gmail.com	mesin		Edit Hapus

2. Tambah data

Tambah Data Mahasiswa

- Nama :
- Nim :
- Email :
- Jurusan :
- Gambar :
-

3. Perhatikan gambar dika (saya punya 2 gambar dika dengan nama dan extensi yang sama



4. Hasil

Daftar Mahasiswa

[Tambah Data Mahasiswa](#)

cari

No.	Nama	Nim	Email	Jurusan	Gambar	Aksi
1	Dinos	1234568	dino@gmail.com	Ekonomi		Edit Hapus
2	Dito	1234	dito@gmail.com	mesin		Edit Hapus
3	dito bagus	123455	bagas@gmail.com	akutansi		Edit Hapus
4	dito bagus saputra	123455	bagas@gmail.com	akutansi		Edit Hapus
5	dika	1234	dika@gmail.com	mesin		Edit Hapus
6	dika dito	1234	dika@gmail.com	mesin		Edit Hapus

Gambar 7.6 Penumpukan gambar

Jika kita lihat pada gambar 7.6(4) terlihat gambar dika yang semula tertumpuk oleh gambar dika yang terbaru.

- b. Untuk mengatasi hal berikut maka diperlukan generate id pada function update (Gambar 7.7) dan ada beberapa *code* yang harus dirubah


```
//generate id untuk penamaan gambar dengan function uniqid()
$namafilebaru=uniqid();
$namafilebaru .= '.';
$namafilebaru .= $pecah_gambar;
// var_dump ($namafilebaru);die();

move_uploaded_file($tmpfile,'image/'.$namafilebaru);






















// kita return nama file nya agar dapat masuk ke $gambar=$upload() pada function tambah
return $namafilebaru;
```

Gambar 7.7 uniqid

Keterangan pada gambar 7.7

1. Pada *line 95* merupakan variabel baru untuk menampung **uniqid** (untuk menggenerate id).
2. *Line 96* gabungkan uniqid dengan titik (**contohnya:1312341.**)
3. *Line 97* gabungkan uniqid titik dan extensi gambarnya (**contohnya: 1312341.jpg**)
4. *Line 100* pindah file **\$tmpfile** ke **\$namafilebaru** (yang sebelumnya **\$nama_file**)
5. *Line 103* kita kembalikan nilai dari **\$namafilebaru** yang sebelumnya **\$nama_file**
6. Pada *line 96 dan 97* jika ingin mengetahuinya tinggal anda **var_dump**

c. Hasilnya jika kita lihat di database maka akan seperti pada gambar 7.8

+ Options									
				id	Nama	Nim	Email	Jurusan	Gambar
<input type="checkbox"/>	 Edit	 Copy	 Delete	1	Dinos	1234568	dino@gmail.com	Ekonomi	dina.jpg
<input type="checkbox"/>	 Edit	 Copy	 Delete	39	Dito	1234	dito@gmail.com	mesin	dito.jpeg
<input type="checkbox"/>	 Edit	 Copy	 Delete	40	dito bagas	123455	bagas@gmail.com	akutansi	dito.jpeg
<input type="checkbox"/>	 Edit	 Copy	 Delete	41	dito bagas saputra	123455	bagas@gmail.com	akutansi	dina.jpeg
<input type="checkbox"/>	 Edit	 Copy	 Delete	45	dika	1234	dika@gmail.com	mesin	dika.jpg
<input type="checkbox"/>	 Edit	 Copy	 Delete	67	dika dito	1234	dika@gmail.com	mesin	dika.jpg
<input type="checkbox"/>	 Edit	 Copy	 Delete	69	santi	1213	santi@gmail.com	matematika	5b0d6934964fd.jpeg

Gambar 7.8 Isi dari *database*

- 8) Jika kita melakukan proses *edit* maka gambar tidak akan muncul dihalaman *update* sehingga kita tidak bisa melakukan *update* pada gambar (Gambar 7.9)

Update Data Mahasiswa

- Nama :
- Nim :
- Email :
- Jurusan :
- Gambar :
-

Gambar 7.9 Halaman *update* mahasiswa

9) Untuk mengatasi hal diatas maka kita harus melakukan perubahan *code* pada halaman edit.php (Gambar 7.10)

```
<h1>Update Data Mahasiswa</h1>
<!-- tambahkan atribut enctype -->
<form action="" method="post" enctype="multipart/form-data">
  <li>
    <input type="hidden" name="id" value="<?= $mhs[id] ?>">
    <!-- untuk mengirimkan gambar lama -->
    <input type="hidden" name="GambarLama" value="<?= $mhs[Gambar]; ?>">
  </li>
  <ul>
    <li>
      <!-- for pada label terhubung dengan id jadi jika label nama diklik maka textfield nama akan aktif juga-->
      <label for="Nama">Nama :</label>
      <!-- untuk pengisian name besar kecilnya harus sesuai dengan fieldnya -->
      <input type="text" name="Nama" id="Nama" value="<?= $mhs[Nama]; ?>" >
    </li>
    <li>
      <label for="Nim">Nim :</label>
      <input type="text" name="Nim" id="Nim" required value="<?= $mhs[Nim]; ?>">
    </li>
    <li>
      <label for="Email">Email :</label>
      <input type="text" name="Email" id="Email" required value="<?= $mhs[Email]; ?>" >
    </li>
    <li>
      <label for="Jurusan">Jurusan :</label>
      <input type="text" name="Jurusan" id="Jurusan" required value="<?= $mhs[Jurusan]; ?>">
    </li>
    <li>
      <label for="Gambar">Gambar :</label><br>
      <!-- tambahkan image source agar gambar dapat muncul -->
      <br>
      <input type="file" name="Gambar" id="Gambar" >
    </li>
    <li>
      <button type="submit" name="submit"> Update </button>
    </li>
  </ul>
</form>
```

Gambar 7.10 Halaman edit.php

Keterangan Gambar 7.10

1. *Line 52* tambahkan atribut **enctype**
2. *Line 55* digunakan untuk mengirimkan gambar yang lama. Hal ini dilakukan jika *user* tidak melakukan perubahan data (**beri name="GambarLama"**)
3. *Line 80* kita tambahkan **img source** agar gambar dapat tampil dihalaman *update*

10) Setelah kita melakukan modifikasi pada **edit.php** maka langkah berikutnya adalah memodifikasi **function edit** pada **functions.php** (Gambar 7.11)

```

112 function edit ($data){
113     global $conn;
114
115     $id      = $data["id"];
116     $nama    = htmlspecialchars($data["Nama"]);
117     $nim     = htmlspecialchars($data["Nim"]);
118     $email   = htmlspecialchars($data["Email"]);
119     $jurusan = htmlspecialchars($data["Jurusan"]);
120     $GambarLama = htmlspecialchars($data["GambarLama"]);
121
122     // cek apakah user menekan button browse
123     if($_FILES['Gambar']['error']===4)
124     {
125         $gambar=$GambarLama;
126     }else
127     {
128         $gambar=upload();
129     }
130
131     $query= " UPDATE mahasiswa SET
132             Nama = '$nama',
133             Nim = '$nim',
134             Email = '$email',
135             Jurusan = '$jurusan',
136             Gambar = '$gambar'
137             WHERE id= $id ";
138     mysqli_query($conn,$query);
139
140     return mysqli_affected_rows($conn);
141 }

```

Gambar 7.11 *Function edit*

Keterangan gambar 7.11

1. Tambahkan variabel baru dengan nama \$GambarLama seperti pada line 120
2. Line 122 sampai line 129 digunakan untuk mengecek apakah *user* menekan tombol *browser* dengan cara kita cek *error* nya apakah bernilai 4 artinya tidak ada gambar yang akan di-*upload* jika iya maka \$gambar=\$GambarLama(line 125) jika tidak maka \$gambar=upload() artinya lakukan proses *function* upload

11) Selesai silakan jalankan

Registrasi

Pada bab ini kita akan membuat fitur registrasi atau fitur *sign up*. Untuk membuat fitur registrasi ini kita membutuhkan **tabel user**. Sebelum kita membuat tabel *user* kita membuat terlebih dulu *form* registrasi. Untuk memudahkan kita dalam membuat fitur registrasi ini maka ikutilah langkah-langkah membuat fitur registrasi sebagai berikut :

1. Buat file baru dengan nama *registrasi.php* dimana isinya adalah *form* registrasi (Gambar 8.1)

```
registrasi.php x
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <meta http-equiv="X-UA-Compatible" content="ie=edge">
7      <title>Form Registrasi</title>
8      <style>
9          label {
10             display: block;
11         }
12     </style>
13 </head>
14 <body>
15     <h1> Halaman Registrasi</h1>
16     <form action="" method="post">
17         <ul>
18             <li>
19                 <label for="username">Username :</label>
20                 <input type="text" name="username" id="username">
21             </li>
22             <li>
23                 <label for="password">Password :</label>
24                 <input type="password" name="password" id="password">
25             </li>
26             <li>
27                 <label for="password2">Konfirmasi Password :</label>
28                 <input type="password" name="password2" id="password2 ">
29             </li>
30             <li>
31                 <button type="submit" name="register">Registrasi</button>
32             </li>
33         </ul>
34     </form>
35 </body>
36 </html>
```

Gambar 8.1 Code registrasi.php

Halaman Registrasi

- Username :
- Password :
- Konfirmasi Password :
-

Gambar 8.2 Hasil tampilan registrasi.php

2. Setelah kita membuat form registrasi langkah berikutnya adalah membuat **tabel** dengan nama **user** (Gambar 8.3)

Server: localhost Database: phpdatabase Table: users

Table structure

#	Name	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(11)			No	None	AUTO_INCREMENT	Change Drop Primary Unique Index More
2	username	varchar(50)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index More
3	password	varchar(255)	latin1_swedish_ci		No	None		Change Drop Primary Unique Index More

Check all With selected: Browse Change Drop Primary Unique Index Add to central columns

Gambar 8.3 Tabel users

- Kita tambahkan *code* php pada halaman register.php (Gambar 8.4)

```

registrasi.php x
1  <?php
2      require 'functions.php';
3
4      if(isset($_POST['register']))
5      {
6          if(registrasi($_POST)>0)
7          {
8              echo "
9                  <style>
10                     alert('user berhasil ditambahkan');
11                 </style>
12             ";
13          }else
14          {
15              echo mysqli_error($conn);
16          }
17      }
18
19  ?>

```

Gambar 8.4 register.php

- Setelah kita tambahkan *code* php pada registrasi.php maka langkah berikutnya adalah kita membuat *function* registrasi pada functions.php

```

function registrasi($data)
{
    global $conn;

    //stripslashes digunakan untuk menghilangkan backslashes
    $username=strtolower(stripslashes($data['username']));

    //mysqli_real_escape_string untuk memberikan perlindungan terhadap karakter-karakter unik (sql_injection)
    //pada mysqli_real_escape_string terdapat 2 parameter
    $password=mysqli_real_escape_string($conn,$data['password']);
    $password2=mysqli_real_escape_string($conn,$data['password2']);

    // cek username sudah ada apa belum
    $result=mysqli_query($conn,"SELECT username FROM users WHERE username='$username'");

    //cek kondisi jika line 175 bernilai true maka cetak echo
    if(mysqli_fetch_assoc($result))
    {
        echo "
        <script>
        | alert('username sudah ada');
        </script>
        ";
        // agar proses insertnya gagal
        return false;
    }

    // cek konfirmasi password
    if($password!=$password2)
    {
        echo"
        <script>
        | alert('password anda tidak sama');
        </script>
        ";
        return false;
    }

    // enkripsi password
    //$password=md5($password);
    $password=password_hash($password,PASSWORD_DEFAULT);
    //var_dump($password);

    // tambahkan user baru ke database
    mysqli_query($conn,"INSERT INTO users VALUES ('','$username','$password')");

    return mysqli_affected_rows($conn);
}

```

Gambar 8.5 *Function* registrasi

Keterangan gambar 8.5

- 1) Line 164 terdapat variable baru **\$username**. Pada variabel username ini terdapat **function stripslashes** digunakan untuk menghilangkan **backslashes**
- 2) Line 168 dan 169 terdapat variabel **password** dan **password2**. password2 nanti digunakan sebagai pembanding dari *password* (Untuk mengecek inputan *user* berupa *password* apakah sama dengan password2). Di-line ini terdapat *function* **mysqli_real_escape_string** (parameter1, parameter2) yang fungsinya adalah untuk memberikan perlindungan terhadap karakter-karakter unik(menghindari serangan *sql_injection*).
- 3) Line 172 adalah *query* untuk mengecek apakah isi dari **username** yang diisikan oleh *user* ada pada *database* (**WHERE username='Susername'**) dan hasilnya dari *query*-nya disimpan pada variabel baru **\$result**
- 4) Line 175 sampai 184 adalah untuk mengecek kondisi dari *result*. Jika nilai dari *result* bernilai *true* (line 175) maka akan mencetak **alert("username sudah ada")**. Dan pada akhir line 183 terdapat **return false;** (ini digunakan agar proses berhenti tidak dilanjutkan). Jika ingin tahu kalian *comment* returnnya (line183) dan buat inputan *username* yang sudah ada pada *database* dan lihat hasilnya pada databasenya-> data akan tetap masuk walaupun ada *warning*-nya

- 5) *Line 187 sampai line 195* digunakan untuk mengecek apakah **password** yang diinputkan oleh *user* tidak sama dengan **password2** (konfirmasi *password*).
- 6) *Line 199* untuk mengenkripsi *password*. Untuk enkripsi *password* ada 2 cara yaitu
- a. Menggunakan **md5**
- Jika kita menggunakan metode **md5** maka hasil enkripsi dapat mudah diketahui oleh orang lain yaitu dengan cara copy paste kode enkripsi dan search di google.(Gambar 8.6 (4))

1. Ubah *code* sebagai berikut

```
// enkripsi password
$password=md5($password);
// $password=password_hash($password,PASSWORD_DEFAULT);
var_dump($password);
```

2. Masukkan data

Halaman Registrasi

- Username :
- Password :
- Konfirmasi Password :
-

3. *Searching* di google nilai enkripsinya
string(32) "21232f297a57a5a743894a0c4a801fc3"

Halaman Registrasi

- Username :
- Password :
- Konfirmasi Password :
-

4. Hasil

Decoded value:

Select Decoded Value

admin

Original Hash (Md5):

Select Original Hash

21232f297a57a5a743894a0c4a801fc3

<div>Decoded value:</div> <div>Select Decoded Value</div> <div>admin</div>	<div>Original Hash (Md5):</div> <div>Select Original Hash</div> <div>21232f297a57a5a743894a0c4a801fc3</div>
--	---

Gambar 8.6 Menggunakan md5

b. Menggunakan *Hash*

- Dengan menggunakan **metode hash** dirasa lebih aman. maka dari itu kita akan menggunakan **metode hash**
5. *Line 203 query* yang digunakan untuk menambahkan ke **tabel user**
 6. *Line 205* merupakan nilai kembalian ke parameter **\$conn(koneksi)**

