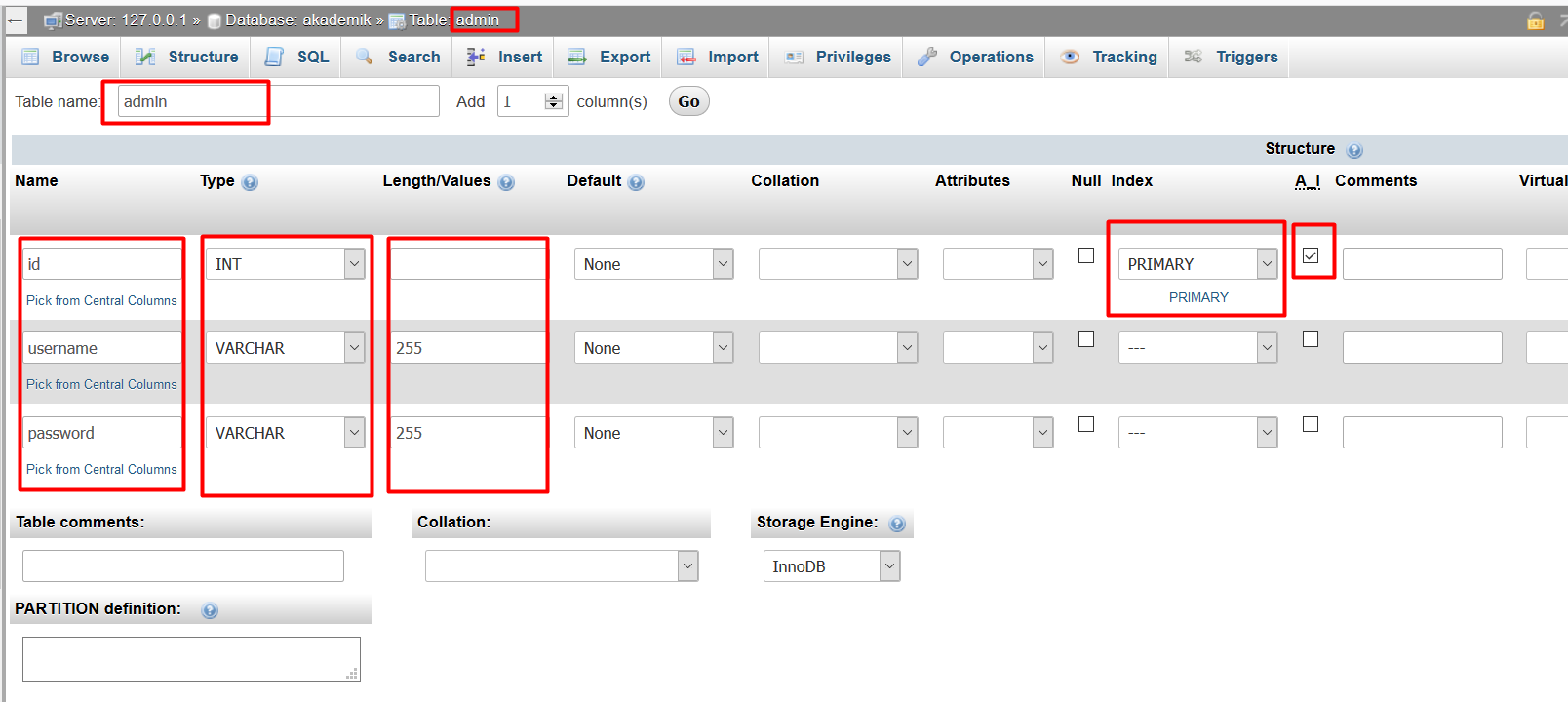
## **Membuat Login Dengan PHP dan MySQLi**

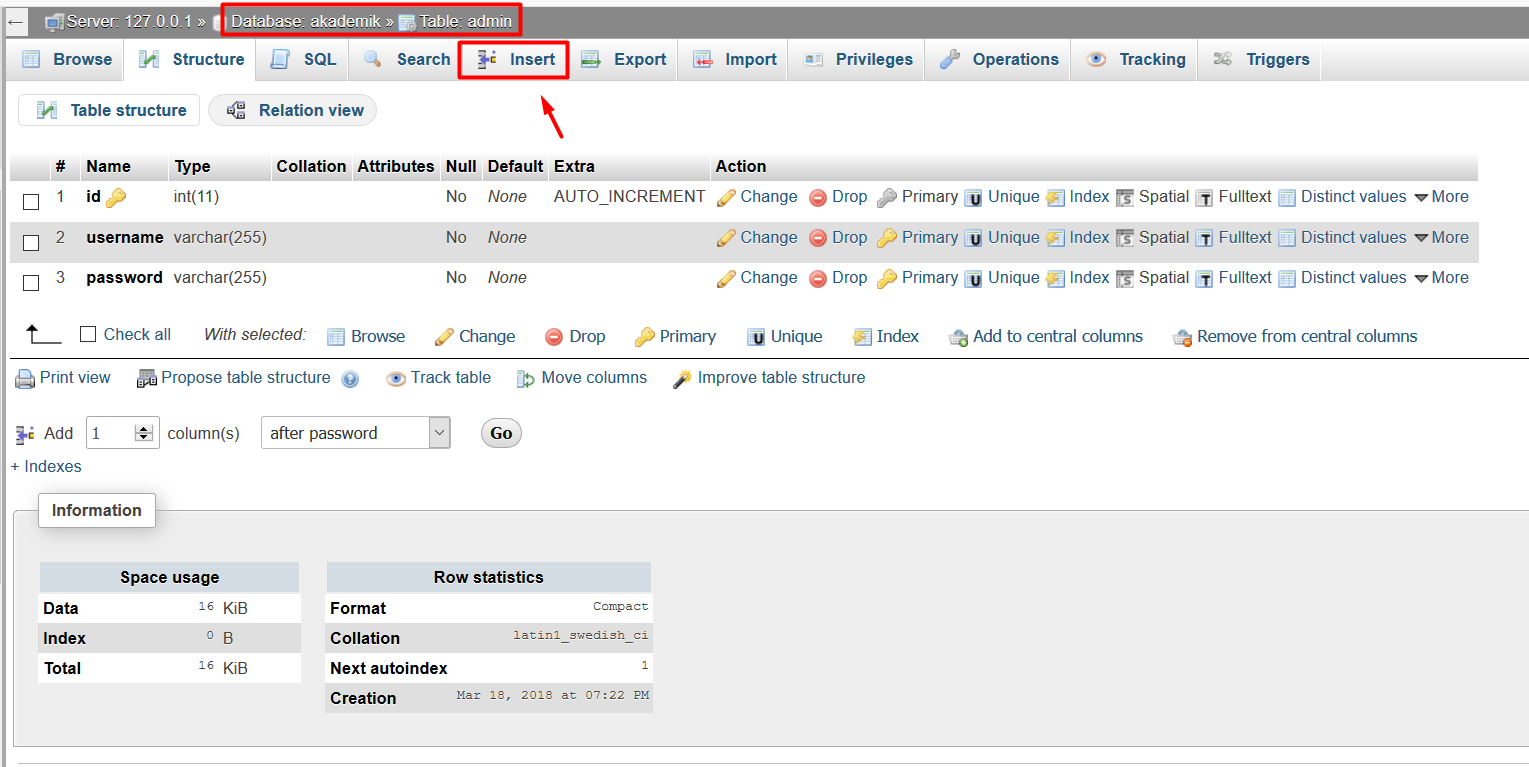
### **Persiapan**

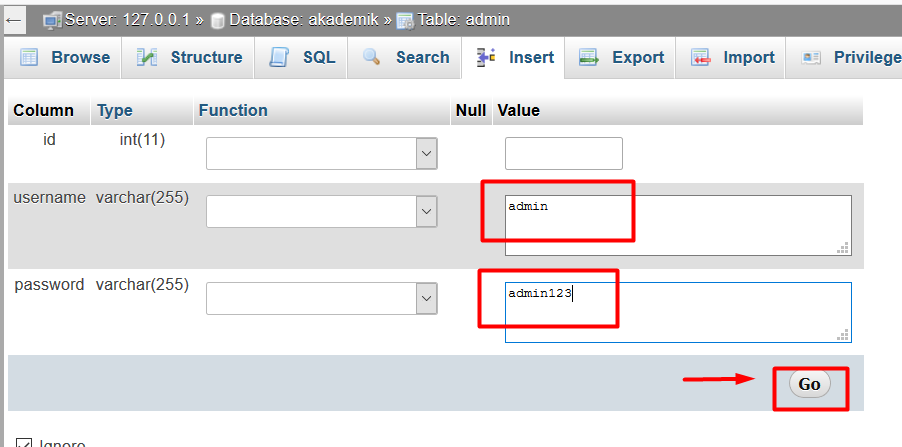
1 .  Untuk memulai tutorial dan project membuat login dengan PHP dan MySQLi seperti di contoh ini, silahkan buat sebuah folder baru dengan nama login pada direktori localhost .

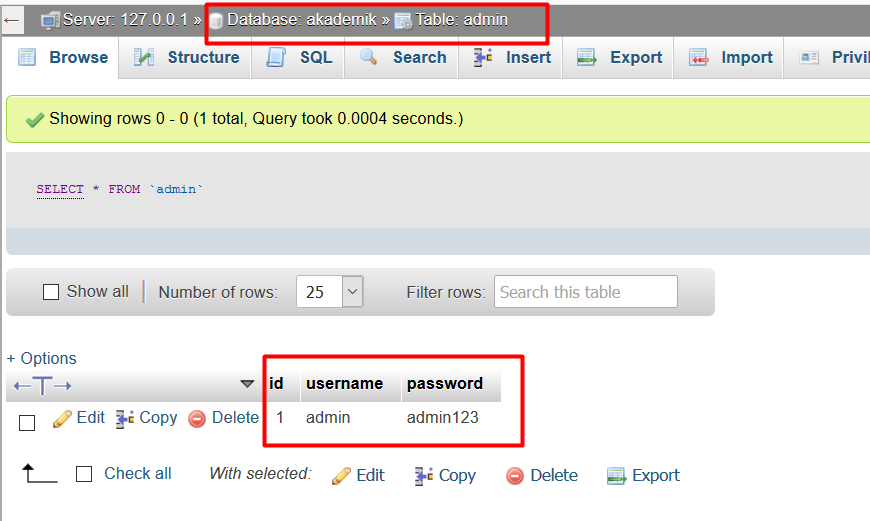
2 .  Buat sebuah database baru dengan nama **akademik**. kemudian buat sebuah tabel baru dengan nama **admin** dalam database akademik. isi dari tabel admin yaitu id, username dan password. dimana id bertipe int, pilih primary dan centang auto\_increment nya. username dan password sama-sama bertipe varchar dengan nilai 225.



selanjutnya isikan data admin. untuk data contoh saja. untuk contoh akun admin yang akan login. isikan username nya ‘admin’, dan password nya ‘admin123’.







#### **Membuat Koneksi Database**

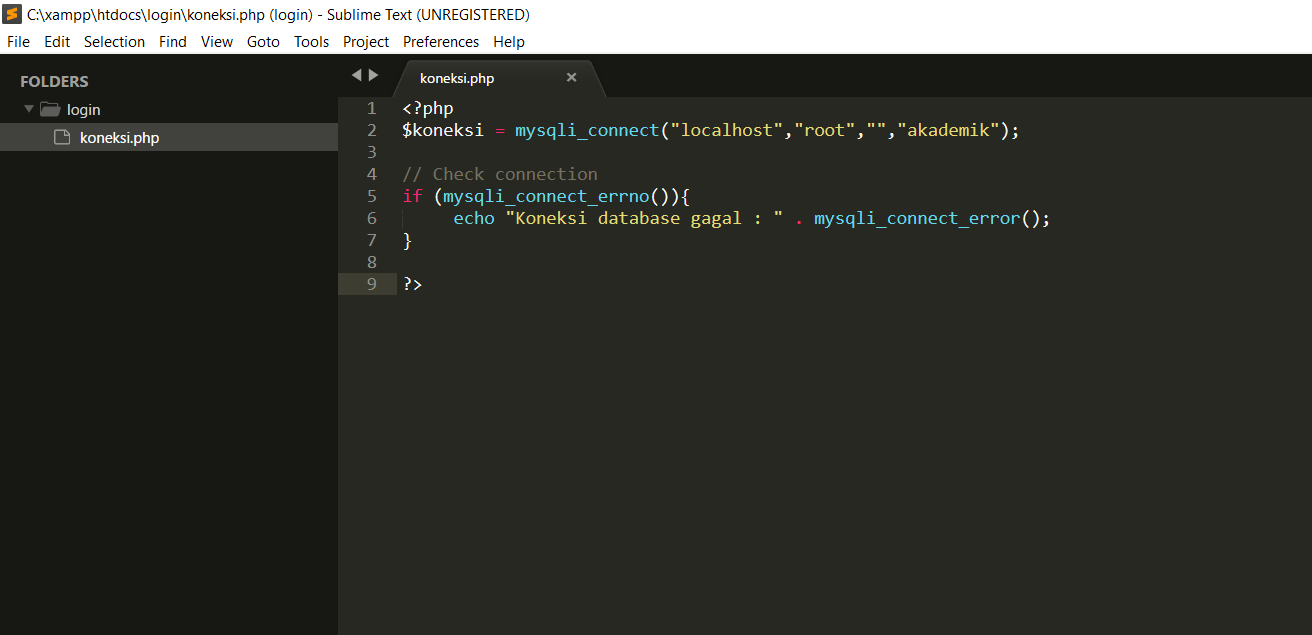
Buat sebuah file baru dengan nama **koneksi.php** dan simpan pada folder **login** yang telah kita buat.

koneksi.php

|  |  |
| --- | --- |
|  | $koneksi = mysqli\_connect("localhost","root","","akademik");  if (mysqli\_connect\_errno()){  echo "Koneksi database gagal : " . mysqli\_connect\_error(); |

silahkan teman-teman isi sendiri nama database nya sesuai nama database yang teman-teman buat, karena di sini kita akan menggunakan database akademik, maka nama databasenya **akademik**.

Silahkan klik di sini untuk membaca [tutorial cara membuat koneksi database dengan PHP dan MySQLi](https://www.malasngoding.com/membuat-koneksi-database-dengan-php-dan-mysqli/) dan penjelasan lengkapnya.



membuat koneksi database dengan php dan mysqli

sampai di sini **koneksi database php dan mysqli** telah selesai kita buat.

Selanjutnya akan kita buat halaman login sederhana untuk tempat admin memasukkan username dan password. kemudian di alihkan ke halaman admin.

Membuat sebuah file baru lagi dengan nama **index.php**.

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28  29  30  31  32  33  34  35  36  37  38  39  40  41  42  43 | <!DOCTYPE html>  <html>  <head>  <title>Membuat Login Dengan PHP dan MySQLi -WWW.MALASNGODING.COM</title>  </head>  <body>  <h2>Login - www.malasngoding.com</h2>  <br/>  <!-- cek pesan notifikasi -->  <?php  if(isset($\_GET['pesan'])){  if($\_GET['pesan'] == "gagal"){  echo "Login gagal! username dan password salah!";  }else if($\_GET['pesan'] == "logout"){  echo "Anda telah berhasil logout";  }else if($\_GET['pesan'] == "belum\_login"){  echo "Anda harus login untuk mengakses halaman admin";  }  }  ?>  <br/>  <br/>  <form method="post" action="cek\_login.php">  <table>  <tr>  <td>Username</td>  <td>:</td>  <td><input type="text" name="username" placeholder="Masukkan username"></td>  </tr>  <tr>  <td>Password</td>  <td>:</td>  <td><input type="password" name="password" placeholder="Masukkan password"></td>  </tr>  <tr>  <td></td>  <td></td>  <td><input type="submit" value="LOGIN"></td>  </tr>  </table>  </form>  </body>  </html> |

**index.php**

nah di file index.php ini kita membuat halaman login sederhana untuk admin. silahkan teman-teman baca lagi tutorial [form HTML](https://www.malasngoding.com/belajar-html-membuat-form-pada-html/) jika teman-teman belum membacanya. dan [penanganan atau pengiriman data dengan form](https://www.malasngoding.com/belajar-php-penanganan-form-pada-php/) yang sudah di bahas sebelumnya.

kita lanjutkan lagi dengan membuat file yang akan mengelola data yang di kirim dari form login dari index.php.

pada file index.php yang membuat halaman login di atas kita telah menetapkan aksi dari form ini adalah ‘cek\_login.php’ dengan method ‘post’ untuk jenis pengiriman datanya. silahkan baca lagi tentang GET dan POST jika teman-teman belum mempelajarinya. sudah di bahas juga di www.malasngoding.com.

buat sebuah file baru lagi dengan nama cek\_login.php.

**cek\_login.php**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25 | <?php  // mengaktifkan session php  session\_start();  // menghubungkan dengan koneksi  include 'koneksi.php';  // menangkap data yang dikirim dari form  $username = $\_POST['username'];  $password = $\_POST['password'];  // menyeleksi data admin dengan username dan password yang sesuai  $data = mysqli\_query($koneksi,"select \* from admin where username='$username' and password='$password'");  // menghitung jumlah data yang ditemukan  $cek = mysqli\_num\_rows($data);  if($cek > 0){  $\_SESSION['username'] = $username;  $\_SESSION['status'] = "login";  header("location:admin/index.php");  }else{  header("location:index.php?pesan=gagal");  }  ?> |

perhatikan, di sini kita aktifkan dulu session php. karena kita menggunakan session sebagai penanda status pengguna yang akan login. baca di sini tutorial [session php](https://www.w3schools.com/php/php_sessions.asp).

kita mengaktifkan session php dengan syntax berikut

kemudian jangan lupa juga untuk menghubungkannya dengan file koneksi database yang sudah kita buat.

kemudian menangkap data yang dikirim dari form login.

|  |
| --- |
| $username = $\_POST['username'];  $password = $\_POST['password']; |

silahkan baca di sini untuk tutorial [GET dan POST](https://www.malasngoding.com/belajar-php-penanganan-form-pada-php/) pada form.

kemudian kita seleksi data admin yang memiliki username dan password yang sesuai dengan yang di input pada form login.

|  |
| --- |
| $data = mysqli\_query($koneksi,"select \* from admin where username='$username' and password='$password'"); |

selanjutnya kita hitung jumlah record yang sesuai (yang sesuai dengan username dan password yang di input).

|  |
| --- |
| $cek = mysqli\_num\_rows($data); |

jika data username dan password admin di temukan di tabel ‘admin’, maka nilai dari variabel $cek adalah jumlah dari record yang di temukan. berarti variabel $cek bernilai 1 jika username dan password yang di input sesuai dengan yang ada di database. karena fungsi mysqli\_num\_rows() berfungsi untuk menghitung jumlah record hasil dari query yang ada dalam parameter si fungsi mysqli\_num\_rows() ini.

selanjutnya kita cek dengan if else jika variabel $cek lebih besar dari nol (berarti username dan password yang di input sesuai), maka kita buat session username dan session status seperti pada syntax berikut. kemudian kita alihkan ke folder admin/index.php.  
jika tidak maka kita alihkan halamannya kembali ke halaman login (index.php) sambil mengirimkan data pesan (?pesan=gagal).

|  |
| --- |
| if($cek > 0){  $\_SESSION['username'] = $username;  $\_SESSION['status'] = "login";  header("location:admin/index.php");  }else{  header("location:index.php?pesan=gagal");  } |

itu sebabnya di file index.php (halaman login) kita membuat pengecekan seperti berikut (perhatikan pada file index.php)

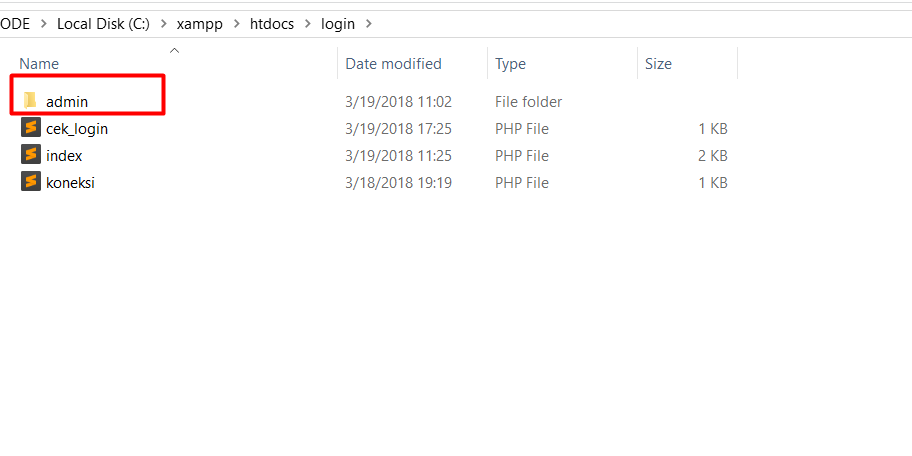
|  |
| --- |
| <?php  if(isset($\_GET['pesan'])){  if($\_GET['pesan'] == "gagal"){  echo "Login gagal! username dan password salah!";  }else if($\_GET['pesan'] == "logout"){  echo "Anda telah berhasil logout";  }else if($\_GET['pesan'] == "belum\_login"){  echo "Anda harus login untuk mengakses halaman admin";  }  }  ?> |

ini untuk memeriksa pesan yang di kirim. karena tadi jika username dan password tidak di temukan, akan dikirimkan pesan “gagal”. pada pengecekan notifikasi ini kita akan menampilkan pesan notifikasi “Login gagal! username dan password salah!” jika username dan password gagal. begitu juga seterusnya dengan pesan logout dan belum\_login yang akan kita buat pengalihannya sebentar lagi.

### **Membuat Halaman Admin**

Seperti yang teman-teman lihat pada syntax sebelumnya. saat melakukan pengecekan, jika username dan password benar, maka halaman akan kita alihkan ke halaman admin yang berada dalam folder **admin**.

jadi sekarang kita buat sebuah folder baru dengan nama ‘admin’.



setelah membuat folder admin, buat sebuah file baru lagi dengan nama **index.php**. pada file index.php ini kita akan membuat halaman sederhana untuk admin.

Membuat Login Dengan PHP dan MySQLi Part 2

**admin/index.php**

|  |  |
| --- | --- |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18  19  20  21  22  23  24  25  26  27  28 | <!DOCTYPE html>  <html>  <head>  <title>Membuat Login Dengan PHP dan MySQLi -WWW.MALASNGODING.COM</title>  </head>  <body>  <h2>Halaman Admin</h2>  <br/>  <!-- cek apakah sudah login -->  <?php  session\_start();  if($\_SESSION['status']!="login"){  header("location:../index.php?pesan=belum\_login");  }  ?>  <h4>Selamat datang, <?php echo $\_SESSION['username']; ?>! anda telah login.</h4>  <br/>  <br/>  <a href="logout.php">LOGOUT</a>  </body>  </html> |

coba perhatikan pada file index.php. di sini kita buat pengecekan dengan if else.

|  |  |
| --- | --- |
|  | <!-- cek apakah sudah login -->  <?php  session\_start();  if($\_SESSION['status']!="login"){  header("location:../index.php?pesan=belum\_login");  }  ?> |

jika session status tidak sama dengan “login” maka halaman akan di alihkan ke halaman login lagi sambil mengirim pesan “belum\_login”. pengecekan ini berfungsi untuk keamanan agar halaman admin tidak bisa di akses sebelum melakukan login (session login di buat saat login berhasil, perhatikan lagi pengecekan pada file cek\_login.php).

kita juga menampilkan notifikasi selamat datang untuk username yang sedang login dengan cara langsung menampilkan session username yang berhasil login.

jika yang login adalah pengguna dengan username “admin” maka yang tampil di sini adalah “selamat datang, admin”.  begitu juga selainnya.

terakhir kita buat tombol untuk link logout.

|  |
| --- |
| <a href="logout.php">LOGOUT</a> |

saat link logout di klik, maka akan di alihkan ke file logout.php.

buat file baru dengan nama **logout.php**.

admin/logout.php

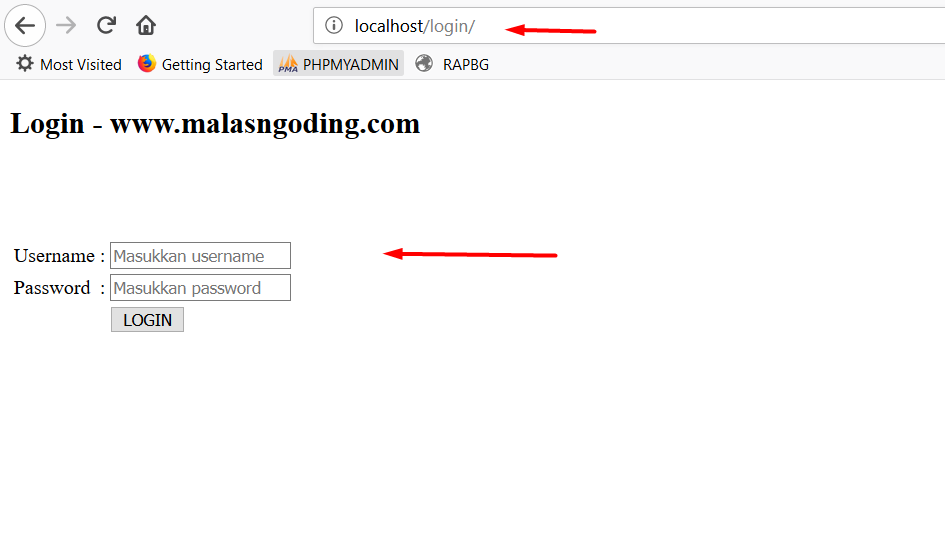
|  |
| --- |
| <?php  // mengaktifkan session  session\_start();  // menghapus semua session  session\_destroy();  // mengalihkan halaman sambil mengirim pesan logout  header("location:../index.php?pesan=logout");  ?> |

file logout berguna untuk menghapus semua session yang telah di buat saat login. dan mengalihkan halaman kembali ke halaman login sambil mengirim pesan ‘logout’.

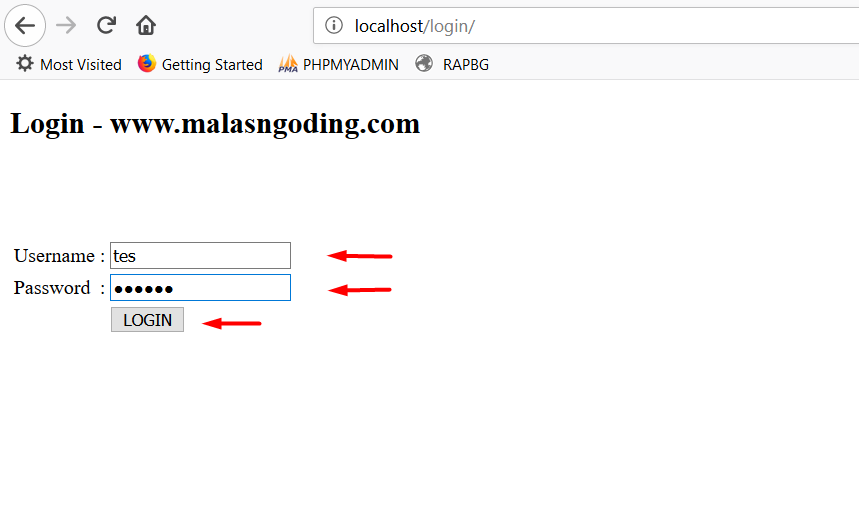
kita menghapus semua session dengan fungsi session\_destroy() seperti pada contoh di atas.

nah, semuaya selesai. waktunya kita coba.

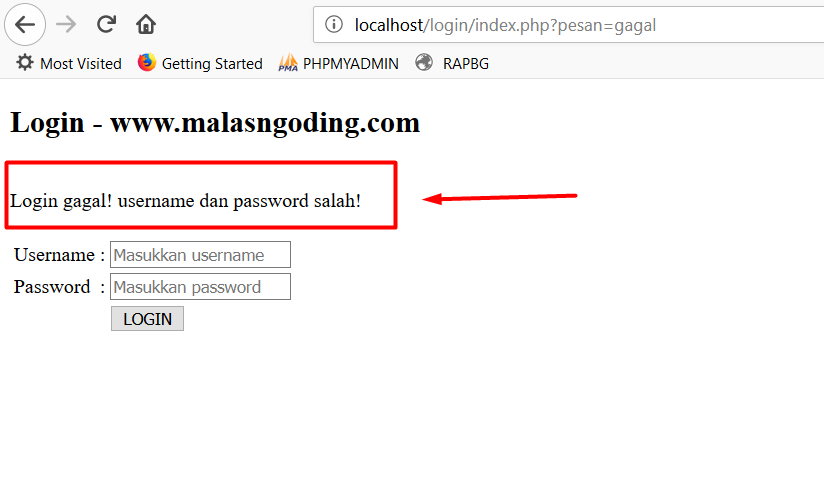
jalankan localhost/login pada browser.



selanjutnya untuk mencoba apakah form login ini sudah berjalan dengan sebagai mana mestinya, coba isikan username : tes, dan password tes123.

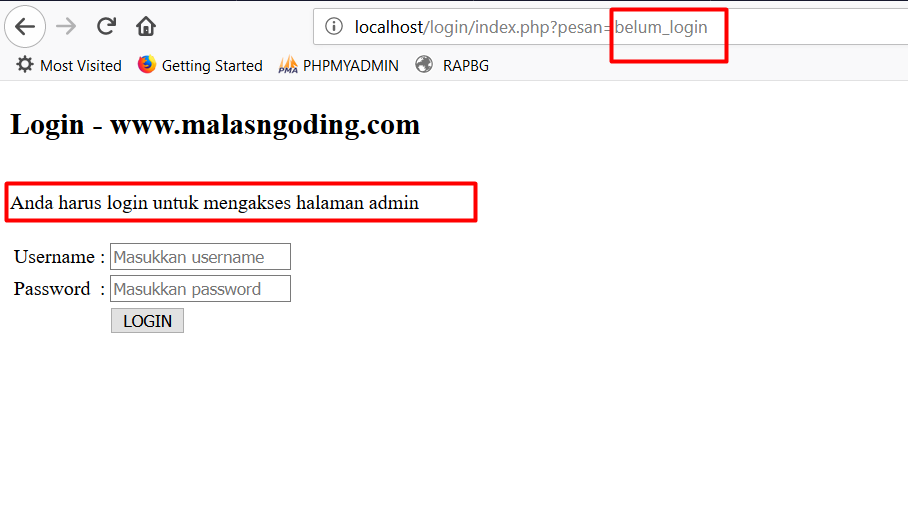


dan klik tombol LOGIN. maka akan muncul notifikasi kalau username dan password yang di masukkan salah.



login gagal

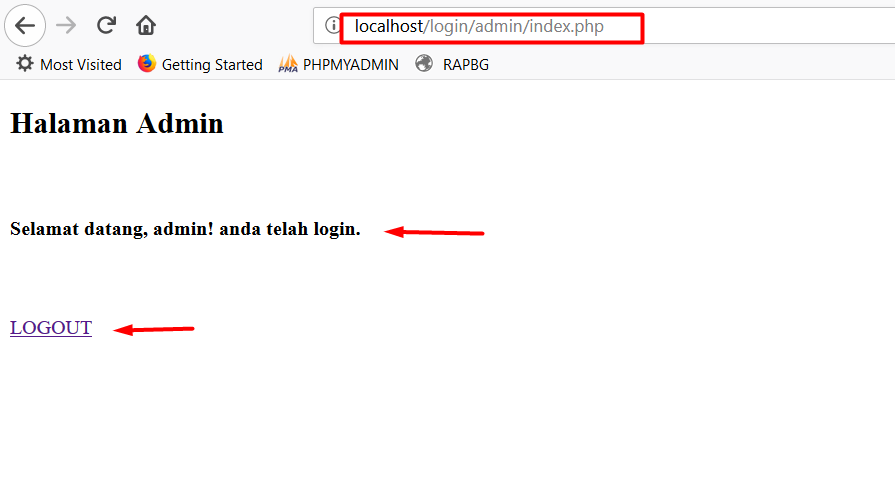
sekarang coba masuk langsung ke halaman admin tanpa melakukan login, akses “localhost/login/admin“.



keamanan login admin php dan mysqli

maka akan muncul notifikasi yang mengharuskan kita untuk login terlebih dulu.

mantap. sekarang coba masukkan username dan password yang benar. username : admin, password : admin123. dan klik LOGIN.  
baca [tutorial membuat login dengan php dan mysqli part 1](https://www.malasngoding.com/membuat-login-dengan-php-dan-mysqli/).



login pun berhasil. kita coba klik pada link logout, maka halaman akan di alihkan kembali ke halaman login sambil menampilkan pesan notifikasi bahwa telah berhasil logout.

