



Link Repository : [https://github.com/athfizh/AthaullaHafizh DESPROG TI2A 2025](https://github.com/athfizh/AthaullaHafizh_DESPROG_TI2A_2025)

Topik

- Pengenalan konsep login
- Multi user login
- Pengenalan konsep cookies dan session

Tujuan

Mahasiswa diharapkan mampu:

1. Memahami konsep login *single user* dan *multiuser*
2. Memahami konsep dasar *cookies* dan penggunaannya
3. Memahami konsep dasar *session* dan penggunaannya
4. Mengimplementasikan konsep PHP dan database MySQL ke dalam web dinamis

Pendahuluan

Pengenalan Login

Saat ini fitur login menjadi fitur yang banyak digunakan di dalam pengembangan aplikasi web modern. Login diartikan sebagai proses masuk ke jaringan komputer. Seorang *user* harus memasukkan identitas akun minimal terdiri dari *username*/akun pengguna dan *password* untuk mendapatkan hak akses. Kemudian sistem akan melakukan pencocokan *username* dan *password* yang diinput dengan data *username* dan *password* yang tersimpan di dalam *database*.

Di dalam praktikum ini akan dibahas mengenai login *single* dan *multiuser*. Login *multiuser* adalah proses login yang memiliki validasi level dari seorang user saat login. Pengaturan validasi level ini berkaitan dengan hak akses yang akan diberikan kepada user yang login ke dalam sistem. Di dalam login multiuser terdapat istilah level akses. Level inilah yang akan menentukan tampilan mana atau aksi apa saja yang bisa dilakukan oleh *user* tersebut.

Contoh penerapan login *multiuser* dapat kita lihat pada web lms.jti.polinema.ac.id. Terdapat beberapa level akses untuk *user* yang akan memberikan hak akses yang berbeda dan dapat melakukan aksi yang berbeda di dalam web tersebut. Berikut adalah level akses dan aksi yang dapat dilakukan setiap *user* di dalam web lms.jti.polinema.ac.id:

1. *Student*

Level akses *student* dapat melakukan beberapa aksi yaitu, mengedit profil, melihat mata kuliah, mengunduh (*download*) materi, mengumpulkan (*upload*) tugas, dan mengerjakan kuis.

2. *Teacher*

Level akses *teacher* dapat melakukan beberapa aksi yaitu, mengedit topik/bab pada mata kuliah, menambahkan aktivitas (*upload* materi, *jobsheet* praktikum, *submission* tugas, kuis), dan melakukan penilaian pada tugas.

3. *Administrator*

Level akses *administrator* dapat melakukan beberapa yaitu, mendaftarkan *user*, membuat dan mengatur mata kuliah, dan mengatur *grouping* tiap mata kuliah.

Pengenalan Cookies

Cookies adalah sebuah nilai yang dikirimkan dan ditanamkan *server* pada komputer *client*. *Cookies* berupa berkas berukuran kecil umumnya kurang dari 100 bytes, sehingga tidak akan mempengaruhi kecepatan *browsing*. *Cookies* menampung infomasi berkaitan dengan pengguna. *Cookies* berguna untuk memudahkan pengguna, seperti "mengingat" pengguna setiap kali mengunjungi website yang sama.

Informasi-informasi yang disimpan dalam *cookies* adalah data-data sederhana tentang kebiasaan pengguna. Contoh sederhananya saat pengguna mengunjungi sebuah website yang menawarkan tampilan dengan versi bahasa Inggris dan bahasa Indonesia. Apabila pengguna memilih bahasa Indonesia, *cookies* akan secara otomatis menyimpan konfigurasi tersebut. Konfigurasi tersebut akan diingat, sehingga apabila pengguna mengunjungi kembali website tersebut, pengguna tidak perlu memilih jenis bahasa lagi.

Cookies mempunyai umur, maksudnya setiap data yang disimpan dalam komputer pengguna suatu saat bisa hilang atau musnah. Berdasarkan umurnya *cookies* dibagi menjadi dua jenis, yaitu *session cookies* dan *persistent cookies*. *Session cookies* adalah jenis *cookies* yang tersimpan sementara dan akan hilang saat pengguna menutup *browser*. *Session cookies* ini biasanya digunakan pada fitur "keranjang belanja" di situs belanja online. *Persistent cookies* adalah jenis *cookies* yang tersimpan secara permanen di dalam komputer sampai pengguna memutuskan untuk menghapusnya. *Persistent cookies* umumnya digunakan pada fitur "remember me" saat login.

Pengenalan Session

Session adalah salah satu fasilitas yang ada pada PHP yang digunakan untuk menyimpan data sememntara ke dalam variabel, sehingga data tadi dapat diakses oleh pengguna selama variabel *session* tersebut tidak dikosongkan atau dihilangkan. *Session* pada prinsipnya hampir sama dengan *cookies*, yang membedakan adalah posisi penyimpanan data. Jika *cookies* menyimpan data pada *client*, data *session* akan disimpan pada sisi *server*. Jadi *session* relatif lebih aman digunakan untuk menyimpan variabel nilai yang bersifat rahasia seperti *username* dan *password* pada saat login. Contoh penggunaan *session* diantaranya untuk menyimpan informasi login yang berlaku hanya dalam satu sesi dan menyimpan catatan pemesanan barang dalam sistem *e-commerce/transaksi* online. Tabel berikut menjelaskan perbandingan *cookies* dan *session*:

Tabel 1 Perbandingan *Cookies* dan *Session*

Jenis	Menyimpan Informasi	Lokasi Penyimpanan	Dapat dipanggil	Bersifat Global	Dapat dipanggil Lain Waktu	Dibatasi Browser	Dapat diedit User
<i>Cookies</i>	Ya	<i>Client</i>	Ya	Ya	Ya	Ya	Ya
<i>Session</i>	Ya	<i>Server</i>	Ya	Ya	Tidak	Tidak	Tidak

Praktikum Bagian 1. Membuat Tabel User

Tahapan pertama dalam membuat fitur login adalah membuat database. Pada praktikum kali ini, kita akan menggunakan *database* yang telah dibuat pada praktikum sebelumnya, yaitu *database* yang bernama “prakwebdb”. Selanjutnya hal yang harus dilakukan adalah membuat tabel *user*. Tabel *user* minimal terdiri dari *username* dan *password*. Berikut adalah ketentuan *username* dan *password* yang dibuat di tabel user:

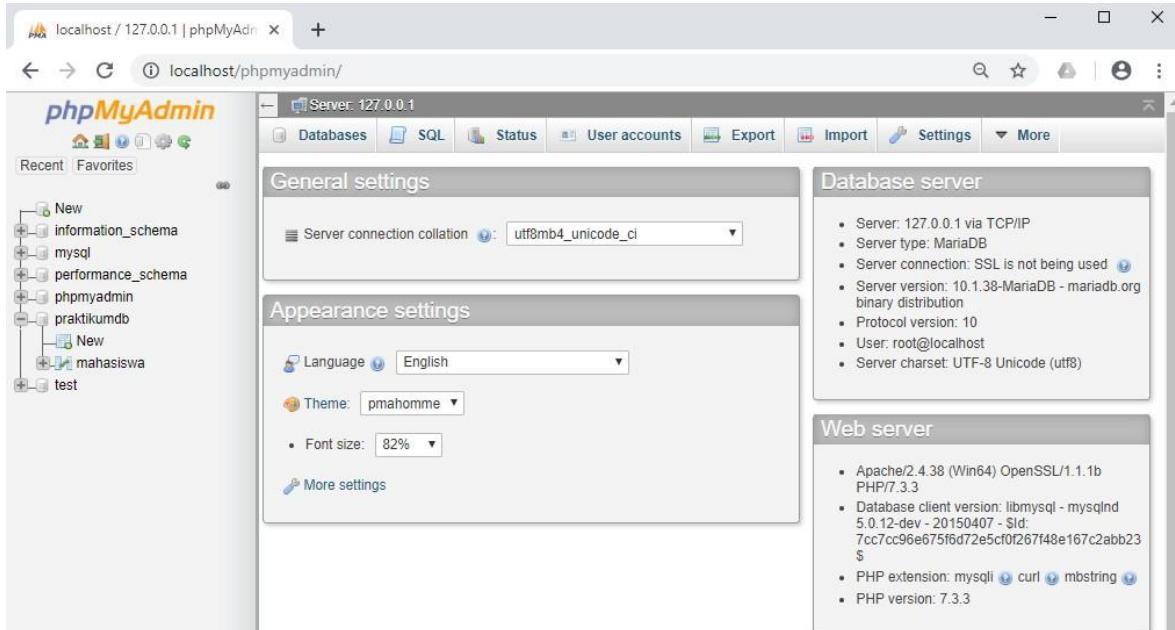
praktikumdb user	
id	: int(11)
username	: varchar(50)
password	: varchar(50)

- tabel bernama user
- kolom tabel terdiri dari id, *username*, dan *password*
- id memiliki tipe data integer dan di atur sebagai *primary key*
- *username* dan *password* memiliki tipe data varchar
- *password* akan dienkripsi menggunakan fungsi md5

MD5 adalah sebuah algoritma fungsi hash dari kriptografi yang digunakan dengan hash value 128-bit. MD5 telah digunakan untuk berbagai macam keperluan dalam membangun keamanan sebuah aplikasi, salah satunya digunakan untuk validasi login pada sistem. MD5 digunakan untuk mengubah nilai asli menjadi serangkaian nilai yang terenkripsi atau acak. Sintak MD5 di dalam PHP adalah:

```
<?php  
    echo md5('string');  
?>
```

Ikuti langkah-langkah berikut untuk melakukan persiapan dalam pembuatan tabel *user*:

Langkah	Keterangan
1	Buka XAMPP kemudian jalankan web server dan database dengan mengklik tombol start pada modul Apache dan MySQL
2	Buka browser kemudian ketikkan localhost/phpmyadmin
3	Berikut adalah tampilan halaman phpmyadmin 
4	Buka database “prakwebdb” kemudian buat tabel baru bernama “user” dengan kolom “3”

Server: 127.0.0.1 » Database: praktikumdb

Structure SQL Search Query More

Filters Containing the word:

Table Action

mahasiswa ★ Browse Structure Search Insert Empty 1 table Sum

Check all With selected:

Print Data dictionary

Create table

Name: user Number of columns: 3 Go

6 Berikut ini adalah kolom-kolom dalam tabel user

Server: 127.0.0.1 » Database: praktikumdb » Table: user

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Table name: user Add 1 column(s) Go

Struct

Name	Type	Length/Values	Default	Collation	Attributes	Null	Index	A_I	Com
id	INT		None			PRIMARY		✓	
username	VARCHAR	50	None			---			
password	VARCHAR	50	None			---			

7 Masukkan data sebagai berikut ke dalam tabel user

Server: 127.0.0.1 » Database: praktikumdb » Table: user

Browse Structure SQL Search Insert Export Import Privileges Operations

Column	Type	Function	Null	Value
id	int(11)		1	
username	varchar(50)			admin
password	varchar(50)	MD5		123

Go

8 Tunjukkan hasil screenshot isi tabel user (soal no 1)

← → id username password

Edit 1 admin 202cb962ac59075b964b07152d234b70

9 Tuliskan query untuk membuat tabel user menggunakan fungsi mysqli_query() (soal no 2)

```
create table user (
    id int(11) primary key,
    username varchar(50),
    password varchar(50)
);
```

10	Tuliskan <i>query</i> untuk memasukkan dan ke tabel user menggunakan fungsi <i>mysqli_query()</i> (soal no 3) <pre>insert into user (id, username, password) values (1, 'admin', md5('123'));</pre>
----	---

Praktikum Bagian 2. Login Single User

Pada praktikum ini akan dijelaskan langkah-langkah untuk membuat form login single user. Cara kerja login single user adalah dengan mencocokan *username* dan *password* yang diinputkan pada form login dengan data *username* dan *password* yang telah disimpan pada tabel user.

Ikuti langkah-langkah berikut untuk membuat login *single user*:

Langkah	Keterangan
1	Pastikan file <i>koneksi.php</i> dari jobsheet sebelumnya, tersimpan di dalam direktori <i>praktik_php</i> .
2	Buat file dengan nama <i>loginForm.html</i> di dalam direktori <i>praktik_php</i> , kemudian ketikkan kode program berikut ini:
	<pre> 1 <html> 2 <head> 3 </head> 4 <body> 5 <form action="loginProses.php" method="post"> 6 <table> 7 <tr> 8 <td>Username</td> 9 <td><input type="text" name="username" size="20"></td> 10 </tr> 11 <tr> 12 <td>Password</td> 13 <td><input type="password" name="password" size="20"></td> 14 </tr> 15 <tr> 16 <td>&nbsp;</td> 17 <td><input type="submit" name="Login" value="Proses"></td> 18 </tr> 19 </table> 20 </form> 21 </body> 22 </html> </pre>
3	Buat file dengan nama <i>loginProses.html</i> di dalam direktori <i>praktik_php</i> , kemudian ketikkan kode program berikut ini:

```

1  <?php
2      include "koneksi.php";
3
4      $username=$_POST['username'];
5      $password=md5($_POST['password']);
6
7      $query="SELECT * FROM user WHERE username='".$username"' and password='".$password."'";
8      $result=mysqli_query($connect, $query);
9      $cek=mysqli_num_rows($result);
10
11     if($cek){
12         echo "Anda berhasil login, silahkan menuju "; ?>
13         <a href="homeAdmin.html">Halaman HOME</a>
14     <?php
15     }else{
16         echo "Anda gagal login, silahkan ";?>
17         <a href="loginForm.html">Login kembali</a>
18     <?php
19         echo mysqli_error($connect);
20     }
21 ?>

```

- 4 Buat file dengan nama homeAdmin.html di dalam direktori praktik_php, kemudian ketikkan kode program berikut ini:

```

1  <html>
2      <head>
3          </head>
4      <body>
5          <h2> Ini adalah halaman admin </h2>
6      </body>
7  </html>

```

- 5 Simpan file dan jalankan kode program dengan mengetikkan
localhost/dasarWeb/praktik_php/loginForm.php

- 6 Lakukan login dengan mengetikkan username dan password yang datanya belum tersimpan di database. Jelaskan hasil pengamatanmu (**soal no 4**)

Anda gagal login. silahkan Login kembali

Gagal login karena jika login harus menggunakan username dan password yang sudah tersimpan dalam database.

- 7 Lakukan login dengan mengetikkan username dan password yang datanya sudah tersimpan di database. Jelaskan hasil pengamatanmu (**soal no 5**)

Anda berhasil login, silahkan menuju Halaman HOME

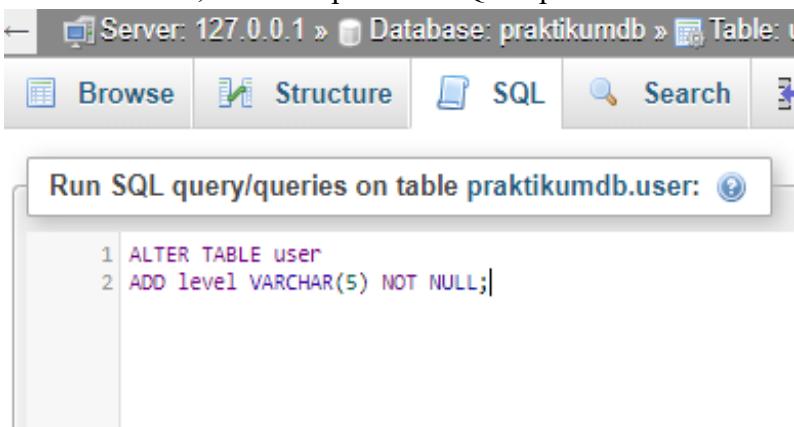
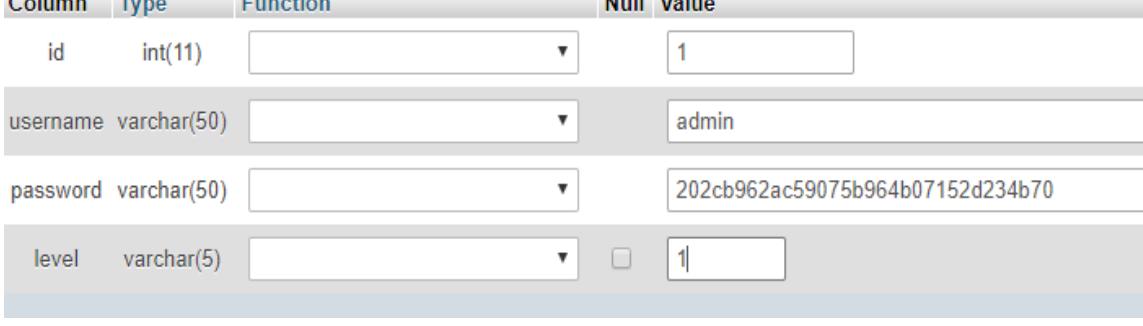
Login berhasil karena menggunakan username dan password yang sudah tersimpan dalam database.

Praktikum Bagian 3. Menambahkan Kolom pada Tabel

Setelah berhasil membuat login untuk *single user*, selanjutnya akan dibahas langkah-langkah untuk membuat login *multiuser*. Pertama yang perlu dipersiapkan adalah memberikan level akses di dalam *database*. Sebelumnya kita hanya memiliki kolom *id*, *username*, dan *password* di dalam tabel *user*. Untuk membuat login *multiuser* perlu ditambahkan kolom level akses. Untuk menambahkan kolom pada tabel digunakan perintah ALTER TABLE. Berikut ini adalah sintak dasar ALTER TABLE:

```
ALTER TABLE namaTabel  
ADD namaKolom tipeData
```

Ikuti langkah-langkah berikut untuk menambahkan kolom pada *database*

Langkah	Keterangan																									
1	Pada tabel <i>user</i> , ketikkan perintah SQL seperti berikut ini:  <pre>1 ALTER TABLE user 2 ADD level VARCHAR(5) NOT NULL;</pre>																									
2	Edit data admin, ubah level menjadi "1"  <table border="1"><thead><tr><th>Column</th><th>Type</th><th>Function</th><th>Null</th><th>Value</th></tr></thead><tbody><tr><td>id</td><td>int(11)</td><td></td><td></td><td>1</td></tr><tr><td>username</td><td>varchar(50)</td><td></td><td></td><td>admin</td></tr><tr><td>password</td><td>varchar(50)</td><td></td><td></td><td>202cb962ac59075b964b07152d234b70</td></tr><tr><td>level</td><td>varchar(5)</td><td></td><td><input type="checkbox"/></td><td>1</td></tr></tbody></table>	Column	Type	Function	Null	Value	id	int(11)			1	username	varchar(50)			admin	password	varchar(50)			202cb962ac59075b964b07152d234b70	level	varchar(5)		<input type="checkbox"/>	1
Column	Type	Function	Null	Value																						
id	int(11)			1																						
username	varchar(50)			admin																						
password	varchar(50)			202cb962ac59075b964b07152d234b70																						
level	varchar(5)		<input type="checkbox"/>	1																						
3	Tambahkan data guest pada tabel <i>user</i> , dengan rincian data sebagai berikut:																									

Server: 127.0.0.1 » Database: praktikumdb » Table: user				
	Browse Structure SQL Search Insert Export			
Column	Type	Function	Null	Value
id	int(11)		▼	
username	varchar(50)		▼	guest
password	varchar(50)	MD5	▼	234
level	varchar(5)		▼	<input type="checkbox"/> 2

4	Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 6)																									
	<table border="1"> <thead> <tr> <th>Column</th><th>Type</th><th>Function</th><th>Null</th><th>Value</th></tr> </thead> <tbody> <tr> <td>id</td><td>int(11)</td><td></td><td>▼</td><td>0</td></tr> <tr> <td>username</td><td>varchar(50)</td><td></td><td>▼</td><td>guest</td></tr> <tr> <td>password</td><td>varchar(50)</td><td></td><td>▼</td><td>234</td></tr> <tr> <td>level</td><td>varchar(5)</td><td></td><td>▼</td><td>2</td></tr> </tbody> </table> <p style="text-align: right;"><input type="button" value="Go"/></p> <p>Tabel user sekarang memiliki kolom baru bernama level. Data 'admin' yang sudah ada diubah levelnya menjadi "1". Selain itu, data baru ditambahkan: username: guest, password: [MD5 dari '234'], dan level: 2.</p>	Column	Type	Function	Null	Value	id	int(11)		▼	0	username	varchar(50)		▼	guest	password	varchar(50)		▼	234	level	varchar(5)		▼	2
Column	Type	Function	Null	Value																						
id	int(11)		▼	0																						
username	varchar(50)		▼	guest																						
password	varchar(50)		▼	234																						
level	varchar(5)		▼	2																						
5	Tunjukkan hasil <i>screenshot</i> isi tabel user (soal no 7) <table border="1"> <thead> <tr> <th></th> <th>id</th> <th>username</th> <th>password</th> <th>level</th> </tr> </thead> <tbody> <tr> <td><input type="checkbox"/></td> <td>0</td> <td>guest</td> <td>289dff07669d7a23de0ef88d2f7129e7</td> <td>2</td> </tr> <tr> <td><input type="checkbox"/></td> <td>1</td> <td>admin</td> <td>202cb962ac59075b964b07152d234b70</td> <td>1</td> </tr> </tbody> </table> <ul style="list-style-type: none"> - id: 1, username: admin, password: [hash MD5 '123'], level: 1. - id: [ID baru], username: guest, password: [hash MD5 '234'], level: 2. 		id	username	password	level	<input type="checkbox"/>	0	guest	289dff07669d7a23de0ef88d2f7129e7	2	<input type="checkbox"/>	1	admin	202cb962ac59075b964b07152d234b70	1										
	id	username	password	level																						
<input type="checkbox"/>	0	guest	289dff07669d7a23de0ef88d2f7129e7	2																						
<input type="checkbox"/>	1	admin	202cb962ac59075b964b07152d234b70	1																						

Praktikum Bagian 4. Login Multiuser

Pada praktikum ini, akan dijelaskan langkah-langkah untuk membuat login *multiuser*. Berbeda dengan login sebelumnya, selain melakukan pencocokan *username* dan *password* yang diinputkan melalui form dengan data *username* dan *password* yang disimpan di *database*, login *multiuser* akan mengenali *user* yang login dan menampilkan halaman yang sesuai dengan hak aksesnya. Pada praktikum ini terdapat 2 level login, yaitu level 1 untuk admin dan level 2 untuk guest.

Ikuti langkah-langkah berikut untuk memahami cara membuat login *multiuser*:

Langkah	Keterangan
1	Praktikum bagian 4 menggunakan file <code>koneksi.php</code> , sehingga pastikan file tersebut berada di dalam folder <code>praktik_php</code> .

2	<p>Buka file <code>loginForm.html</code> kemudian modifikasi kode program dengan mengubah nilai atribut <code>action</code> di baris ke-5 menjadi <code>loginMultiProses.php</code>, sehingga kode program menjadi seperti berikut ini:</p> <pre> 1 <html> 2 <head> 3 </head> 4 <body> 5 <form action="loginMultiProses.php" method="post"> 6 <table> 7 <tr> 8 <td>Username</td> 9 <td><input type="text" name="username" size="20"></td> 10 </tr> 11 <tr> 12 <td>Password</td> 13 <td><input type="password" name="password" size="20"></td> 14 </tr> 15 <tr> 16 <td>&nbsp;</td> 17 <td><input type="submit" name="Login" value="Proses"></td> 18 </tr> 19 </table> 20 </form> 21 </body> 22 </html> </pre>
3	<p>Buat file dengan nama <code>loginMultiProses.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode program berikut ini:</p> <pre> 1 <?php 2 include "koneksi.php"; 3 4 \$username=\$_POST['username']; 5 \$password=md5(\$_POST['password']); 6 7 \$query="SELECT * FROM user WHERE username='".\$username.' and password="'.\$password."'"; 8 \$result=mysqli_query(\$connect, \$query); 9 \$row = mysqli_fetch_assoc(\$result); 10 11 if(\$row['level'] == 1){ 12 echo "Anda berhasil login. silahkan menuju "; ?> 13 Halaman HOME 14 <?php 15 }else if(\$row['level'] == 2){ 16 echo "Anda berhasil login. silahkan menuju "; ?> 17 Halaman HOME 18 <?php 19 }else{ 20 echo "Anda gagal login. silahkan " ;?> 21 Login kembali 22 <?php 23 echo mysqli_error(\$connect); 24 } 25 ?> </pre>
4	<p>Buat file dengan nama <code>homeGuest.html</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode program berikut ini:</p> <pre> 1 <html> 2 <head> 3 </head> 4 <body> 5 <h2> Ini adalah halaman guest </h2> 6 </body> 7 </html> </pre>

5	<p>Lakukan login dengan mengetikkan data admin. Jelaskan hasil pengamatanmu (soal no 8)</p> <p><u>Halaman HOME</u></p> <h2 style="text-align: center;">Ini adalah halaman admin</h2> <p>Login berhasil. Kueri menemukan data 'admin'. Kondisi if (\$row['level'] == 1) bernilai benar, dan program menampilkan link ke homeAdmin.html.</p>
6	<p>Lakukan login dengan mengetikkan data guest. Jelaskan hasil pengamatanmu (soal no 9)</p> <p><u>Halaman HOME</u></p> <h2 style="text-align: center;">Ini adalah halaman guest</h2> <p>Login berhasil. Kueri menemukan data 'guest'. Kondisi if pertama salah, tetapi else if (\$row['level'] == 2) bernilai benar. Program menampilkan link ke homeGuest.html.</p>
7	<p>Lakukan login dengan mengetikkan username dan password yang tidak tersimpan di database. Jelaskan hasil pengamatanmu (soal no 10)</p> <p><u>Login kembali</u></p> <p>Gagal login karena jika login harus menggunakan username dan password yang sudah tersimpan dalam database.</p>

Praktikum Bagian 5. Membuat *Cookies*

PHP memiliki fungsi yang dapat digunakan untuk membuat dan mengambil nilai *cookies*. *Cookies* harus dideklarasikan sebelum halaman ditampilkan, yang artinya dituliskan sebelum tag <html>. *Cookies* dibuat menggunakan fungsi `setcookie()`. Berikut adalah sintaks dasar fungsi `setcookie()`:

```
<?php
    setcookie(name, value, expire);
?>
```

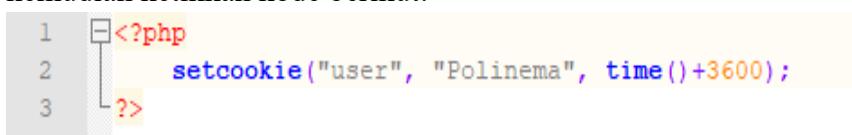
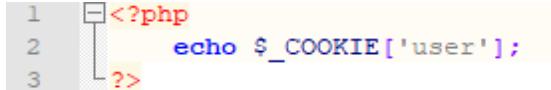
name, adalah nama *cookies* atau variabel yang menjadi pengenal *cookies*
 value, berisi nilai yang tersimpan dalam *cookies*

`expire`, adalah jangka waktu *cookies* akan disimpan di dalam komputer

Setelah mengetahui cara membuat *cookies*, selanjutnya nilai *cookies* yang telah dibuat dapat diambil dengan menggunakan variabel `$_COOKIE`. Berikut ini adalah sintaks untuk mengambil nilai *cookies*:

```
<?php  
    $_COOKIE['user'];  
?>
```

Ikuti langkah berikut untuk memahami cara kerja *cookies*:

Langkah	Keterangan
1	Jalankan Apache pada XAMPP.
2	Buat file baru dengan nama <code>cookiesCreate.php</code> di dalam direktori <code>praktik_php</code> , kemudian ketikkan kode berikut:  <pre>1 <?php 2 setcookie("user", "Polinema", time() + 3600); 3 ?></pre>
3	Buat file baru dengan nama <code>cookiesCall.php</code> di dalam direktori <code>praktik_php</code> , kemudian ketikkan kode berikut:  <pre>1 <?php 2 echo \$_COOKIE['user']; 3 ?></pre>
4	Buka <i>browser</i> dan jalankan kode program pada langkah ke-3 dengan mengetikkan <code>localhost/dasarWeb/praktik_php/cookiesCall.php</code>
5	Amati dan jelaskan hasil pengamatanmu (soal no 11) Warning: Undefined array key "user" in C:\laragon\www\pertemuan09\dasarWeb\praktik_php\cookiesCall.php on line 2 Halaman akan kosong atau menampilkan <i>warning</i> "Undefined index: user". Ini terjadi karena <code>cookiesCall.php</code> mencoba menampilkan <code>\$_COOKIE['user']</code> , tetapi cookie tersebut belum pernah dibuat oleh <code>cookiesCreate.php</code> .
6	Buka <i>browser</i> dan jalankan kode program langkah ke-2 dengan mengetikkan <code>localhost/dasarWeb/praktik_php/cookiesCreate.php</code>
7	Ulangi langkah ke-4
8	Amati dan jelaskan hasil yang ditampilkan (soal no 12) Polinema Halaman <code>cookiesCreate.php</code> akan kosong (karena hanya mengatur cookie). Setelah itu, saat menjalankan <code>cookiesCall.php</code> , halaman akan menampilkan "Polinema".
9	<i>Restart</i> komputer anda
10	Setelah komputer menyala, nyalakan kembali Apache pada XAMPP
11	Buka <i>browser</i> yang sama dengan sebelumnya kemudian ulangi langkah ke-4
12	Amati dan jelaskan hasil yang ditampilkan (soal no 13)

Polinema

Halaman akan tetap menampilkan "Polinema". Cookie yang dibuat di `cookiesCreate.php` diatur dengan waktu kedaluwarsa 1 jam (`time() + 3600`). Ini adalah *persistent cookie* yang disimpan di komputer client. Selama restart terjadi kurang dari 1 jam, cookie tersebut masih valid dan akan terbaca oleh browser.

Praktikum Bagian 6. Menghapus Nilai Cookies

Pada praktikum ini akan dibahas mengenai cara menghapus nilai *cookies*. Jika pada praktikum bagian sebelumnya *cookies* di set dengan waktu *expire time* ()+3600, maka untuk menghapus nilai *cookies* adalah dengan sebagai berikut:

Langkah	Keterangan
1	Buat file baru dengan nama cookiesDel.php di dalam direktori praktik_php, kemudian ketikkan kode berikut: <pre>1 <?php 2 setcookie("user", "Polinema", time()-3600); 3 ?></pre>
2	Buka <i>browser</i> dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/cookiesDel.php
3	Buka <i>browser</i> dan jalankan kode program dari praktikum bagian 1 dengan mengetikkan localhost/dasarWeb/praktik_php/cookiesCall.php
4	Amati dan jelaskan hasil yang dihasilkan dari langkah 2 dan 3, kemudian beri kesimpulan (soal no 14) Warning: Undefined array key "user" in C:\laragon\www\pertemuan09\dasarWeb\praktik_php\cookiesCall.php on line 2 Menjalankan cookiesDel.php tidak menampilkan output. Saat menjalankan cookiesCall.php setelahnya, halaman akan kembali kosong (atau warning "Undefined index: user"). Dikarenakan cookie dihapus dengan cara mengaturnya kembali (menggunakan setcookie) dengan waktu kedaluwarsa di masa lalu (misalnya time()-3600). Ini memerintahkan browser untuk segera menghapus cookie tersebut.

Praktikum Bagian 7. Penerapan Cookies pada Fitur Keranjang Belanja

Salah satu contoh pemanfaatan *cookies* adalah pada fitur “keranjang belanja” pada aplikasi web toko online. Keranjang belanja berisi barang-barang yang akan dibeli oleh user. *Cookies* digunakan untuk mengingat jumlah barang yang dipilih user. Berikut adalah contoh penggunaan *cookies* pada fitur keranjang belanja:

Langkah	Keterangan
1	Buat file baru dengan nama formBeli.html di dalam direktori praktik_php, kemudian ketikkan kode berikut: <pre>1 <html> 2 <head> 3 </head> 4 <body> 5 <form action="prosesBeli.php" method="POST"> 6 <p> Jumlah Novel yang dibeli : 7 <input type="text" name="beliNovel" value= "0" size="2"> </p> 8 <p> Jumlah Buku Teks yang dibeli : 9 <input type="text" name="beliBuku" value= "0" size="2"> </p> 10 <input type="submit"> 11 </form> 12 </body> 13 </html></pre>

2	<p>Buat file baru dengan nama prosesBeli.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <?php 2 if(isset(\$_POST["beliNovel"]) && isset(\$_POST["beliBuku"])){ 3 setcookie("beliNovel", \$_POST["beliNovel"]); 4 setcookie("beliBuku", \$_POST["beliBuku"]); 5 header("location:keranjangBelanja.php"); 6 } 7 ?> </pre>
3	<p>Buat file baru dengan nama prosesBeli.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <html> 2 <head> 3 </head> 4 <body> 5 <h2> Keranjang Belanja </h2> 6 7 <?php 8 \$beliNovel=\$_COOKIE['beliNovel']; 9 \$beliBuku = \$_COOKIE['beliBuku']; 10 11 echo "Jumlah Novel:" . \$beliNovel . "
"; 12 echo "Jumlah Buku :" . \$beliBuku ; 13 14 ?> 15 </body> 16 </html> </pre>
4	<p>Buka <i>browser</i> dan jalankan kode program langkah ke-3 dengan mengetikkan localhost/dasarWeb/praktik_php/keranjangBelanja.php</p>
5	<p>Amati dan jelaskan hasil yang ditampilkan (soal no 15)</p> <p>Keranjang Belanja</p> <p>Warning: Undefined array key "beliNovel" in C:\laragon\www\pertemuan09\dasarWeb\praktik_php\keranjangBelanja.php on line 7</p> <p>Warning: Undefined array key "beliBuku" in C:\laragon\www\pertemuan09\dasarWeb\praktik_php\keranjangBelanja.php on line 8</p> <p>Jumlah Novel: Jumlah Buku:</p> <p>Halaman akan menampilkan "Keranjang Belanja" tetapi diikuti dengan <i>warning</i> "Undefined index" untuk beliNovel dan beliBuku. Ini karena cookie tersebut belum dibuat.</p>
6	<p>Jalankan kode program langkah ke-1 dengan mengetikkan localhost/dasarWeb/praktik_php/formBeli.html</p>
7	<p>Isikan jumlah novel dan buku teks yang akan dibeli kemudian klik tombol "submit"</p>
8	<p>Amati dan jelaskan hasil yang ditampilkan (soal no 16)</p> <p>Keranjang Belanja</p> <p>Jumlah Novel: 5 Jumlah Buku: 2</p>

	Setelah mengisi form (misal Novel=5, Buku=2) dan submit, prosesBeli.php menyimpan nilai ke cookie dan mengarahkan ke keranjangBelanja.php.
9	Tutup <i>browser</i> kemudian buka kembali <i>browser</i> kemudian jalankan kembali kode program langkah ke-3 dengan mengetikkan localhost/dasarWeb/praktik_php/keranjangBelanja.php
10	Amati dan jelaskan hasil yang ditampilkan (soal no 17) Keranjang Belanja Warning: Undefined array key "beliNovel" in C:\laragon\www\pertemuan09\dasarWeb\praktik_php\keranjangBelanja.php on line 7 Warning: Undefined array key "beliBuku" in C:\laragon\www\pertemuan09\dasarWeb\praktik_php\keranjangBelanja.php on line 8 Jumlah Novel: Jumlah Buku: Halaman akan kembali menampilkan <i>warning</i> "Undefined index" (seperti Soal 15). Ini karena setcookie di prosesBeli.php tidak menyertakan parameter expire. Ini menjadikannya session cookies , yang otomatis terhapus saat browser ditutup.

Praktikum Bagian 8. Membuat Session

Untuk membuat *session* dalam PHP digunakan fungsi `session_start()`. Berikut adalah sintaks dasar untuk memulai *session*:

```
<?php
    session_start();
?>
```

Setelah *session* dimulai, maka variabel *session* sudah dapat mulai digunakan. Variabel yang dimaksud adalah PHP global variable `$_SESSION`. Berikut adalah sintaks penggunaan variable global `$_SESSION`:

```
<?php
    $_SESSION['namaVariabel'];
?>
```

Ikuti langkah berikut untuk memahami cara kerja *session*:

Langkah	Keterangan
1	Buat file baru dengan nama <code>sessionCreate.php</code> di dalam direktori <code>praktik_php</code> , kemudian ketikkan kode berikut:

```

1  <?php
2      session_start();
3  ?>
4
5  <!DOCTYPE html>
6  <html>
7  <body>
8      <?php
9          $_SESSION["favcolor"] = "green";
10         $_SESSION["favanimal"] = "cat";
11         echo "Session variables are set.";
12     ?>
13     </body>
14 </html>

```

- 2 Buat file baru dengan nama `sessionCall.php` di dalam direktori `praktik_php`, kemudian ketikkan kode berikut:

```

1  <?php
2      session_start();
3  ?>
4  <!DOCTYPE html>
5  <html>
6  <body>
7      <?php
8          echo "Favorite color is " . $_SESSION["favcolor"] . "<br>";
9          echo "Favorite animal is " . $_SESSION["favanimal"] . ".";
10     ?>
11     </body>
12 </html>

```

- 3 Buka *browser* dan jalankan kode program pada langkah ke-2 dengan mengetikkan `localhost/dasarWeb/praktik_php/sessionCall.php`

- 4 Buka *browser* dan jalankan kode program langkah ke-1 dengan mengetikkan `localhost/dasarWeb/praktik_php/sessionCreate.php`

- 5 Ulangi langkah ke-3

- 6 Amati dan jelaskan hasil yang ditampilkan (**soal no 18**)

**Favorite color is green.
Favorite animal is cat.**

Favorite animal is cat.". `session_start()` berhasil memuat data session yang telah di-set di langkah 4.

Praktikum Bagian 9. Menghapus Nilai Session

Pada praktikum ini akan dibahas mengenai cara menghapus *session*. PHP menyediakan fungsi `session_destroy()` yang dapat digunakan untuk menghapus *session*. Berikut adalah langkah-langkah untuk menghapus *session*:

Langkah	Keterangan
1	Buat file baru dengan nama sessionDel.php di dalam direktori praktik_php, kemudian ketikkan kode berikut: <pre> 1 <?php 2 session_start(); 3 ?> 4 5 <!DOCTYPE html> 6 <html> 7 <body> 8 <?php 9 session_unset(); 10 session_destroy(); 11 12 echo "All session variables are now removed, and the session is destroyed." 13 ?> 14 </body> 15 </html></pre>
2	Buka <i>browser</i> dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/sessionDel.php
3	Buka <i>browser</i> dan jalankan kode program dari praktikum bagian 1 dengan mengetikkan localhost/dasarWeb/praktik_php/sessionCall.php
4	Amati dan jelaskan hasil yang ditampilkan (soal no 19) <p>Warning: Undefined array key "favcolor" in C:\laragon\www\pertemuan09\dasarWeb\praktik_php\sessionCall.php on line 8 Favorite color is .</p> <p>Warning: Undefined array key "favanimal" in C:\laragon\www\pertemuan09\dasarWeb\praktik_php\sessionCall.php on line 9 Favorite animal is .</p> <p>Menampilkan <i>warning</i> "Undefined index" lagi. Ini membuktikan bahwa session_destroy() telah berhasil menghapus semua data dari session di server.</p>

Praktikum Bagian 10. Penerapan Session pada Fitur Login

Ikuti langkah-langkah berikut untuk mengetahui penerapan *session* pada fitur login:

Langkah	Keterangan
1	Jalankan MySQL pada XAMPP
2	Pastikan file koneksi.php yang telah dibuat pada praktikum jobsheet sebelumnya ada di dalam direktori praktik_php
3	Buat file baru dengan nama sessionLoginForm.html di dalam direktori praktik_php, kemudian ketikkan kode berikut:

```

1 <html>
2   <head>
3   </head>
4   <body>
5     <form action="sessionLoginProses.php" method="POST">
6       <table>
7         <tr>
8           <td>Username</td>
9           <td><input type="text" name="username" size="20"></td>
10      </tr>
11      <tr>
12        <td>Password</td>
13        <td><input type="password" name="password" size="20"></td>
14      </tr>
15      <tr>
16        <td>&nbsp;</td>
17        <td><input type="submit" name="Login" value="Log in"></td>
18      </tr>
19    </table>
20  </form>
21 </body>
22 </html>

```

- 4 Buat file baru dengan nama sessionLoginProses.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:

```

1 <?php
2   include "koneksi.php";
3
4   $username = $_POST['username'];
5   $password = md5($_POST['password']);
6
7   $sql = "SELECT * FROM user WHERE username='$username' and password='$password'";
8   $result = mysqli_query($connect, $sql);
9   $cek = mysqli_num_rows($result);
10
11  if($cek > 0){
12    session_start();
13    $_SESSION['username'] = $username;
14    $_SESSION['status'] = 'login';
15    ?>
16    Anda Berhasil Login, silahkan menuju
17    <a href = "homeSession.php"> Halaman Home </a> <?php
18  }else{
19    ?>
20    Gagal login, silahkan login lagi
21    <a href = "sessionLoginForm.html"> Halaman Login </a> <?php
22    echo mysqli_error($connect);
23  }
24 ?>

```

- 5 Buat file baru dengan nama homeSession.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:

```

1 <html>
2   <head>
3   </head>
4   <body>
5   <?php
6     session_start();
7
8     if($_SESSION['status'] == 'login'){
9       echo "selamat datang " . $_SESSION['username'];
10      ?>
11      <br> <a href = "sessionLogout.php"> Log Out </a>
12    <?php
13  }else{
14    echo "anda belum login. silahkan "?>
15    <a href = "sessionLoginForm.html"> Log In</a>
16  <?php
17  }
18  ?>
19  </body>
20 </html>

```

- 6 Buat file baru dengan nama sessionLogout.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:

```

1 <?php
2   session_start();
3   session_destroy();
4
5   echo "Anda berhasil logout";
6 ?>

```

- 7 Buka *browser* dan jalankan kode program dengan mengetikkan
localhost/dasarWeb/praktik_php/sessionLoginForm.html

- 8 Login menggunakan username admin.

- 9 Amati dan jelaskan hasil yang ditampilkan (**soal no 20**)

Anda Berhasil Login, silahkan menuju Halaman Home

selamat datang admin

[Log Out](#)

- sessionLoginProses.php akan memvalidasi data. Karena data 'admin' benar, session_start() dipanggil, \$_SESSION['username'] dan \$_SESSION['status'] di-set . Halaman menampilkan "Anda Berhasil Login..." dengan link ke homeSession.php .
- Saat mengklik link, homeSession.php akan memeriksa \$_SESSION['status']. Karena statusnya 'login', halaman akan menampilkan "selamat datang admin" dan link "Log Out".

10

Jelaskan urutan proses dari login hingga logout (sebutkan juga urutan file yang diproses) (**soal no 21**)

Urutan Proses:

1. **sessionLoginForm.html:** Pengguna memasukkan username/password dan submit.
2. **sessionLoginProses.php:** Menerima data, memvalidasinya ke database. Jika berhasil, file ini memulai session dan menyimpan username serta status 'login' ke `$_SESSION`. Lalu mengarahkan ke `homeSession.php`.
3. **homeSession.php:** Memeriksa apakah `$_SESSION['status']` adalah 'login'. Jika ya, halaman menyapa user dan menampilkan link "Log Out".
4. **sessionLogout.php:** Saat link "Log Out" diklik, file ini dipanggil. File ini menjalankan `session_destroy()` untuk menghapus semua data session, dan menampilkan "Anda berhasil logout".

Referensi :

- 1) Nixon, Robin. (2018). Learning PHP, MySQL, JavaScript, CSS & HTML: A Step-by-step Guide to Creating Dynamic Websites, 5th Edition. O'Reilly Media, Inc.
- 2) Forbes, Alan. (2012). The Joy of PHP: A Beginner's Guide to Programming Interactive Web Applications with PHP and MySQL, 5th Edition. Plum Island Publishing