



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 informasi 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

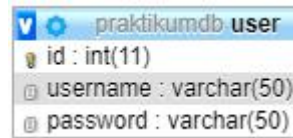
Session adalah salah satu fasilitas yang ada pada PHP yang digunakan untuk menyimpan data sementara ke dalam variabel, sehingga data tadi dapat diakses oleh pengguna selama variabel *session* tersebut tidak dikosongkan atau dihilangkan. *Session* pada prinsipnya hampir sama dengan *cookies*, yang membedakan 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*:

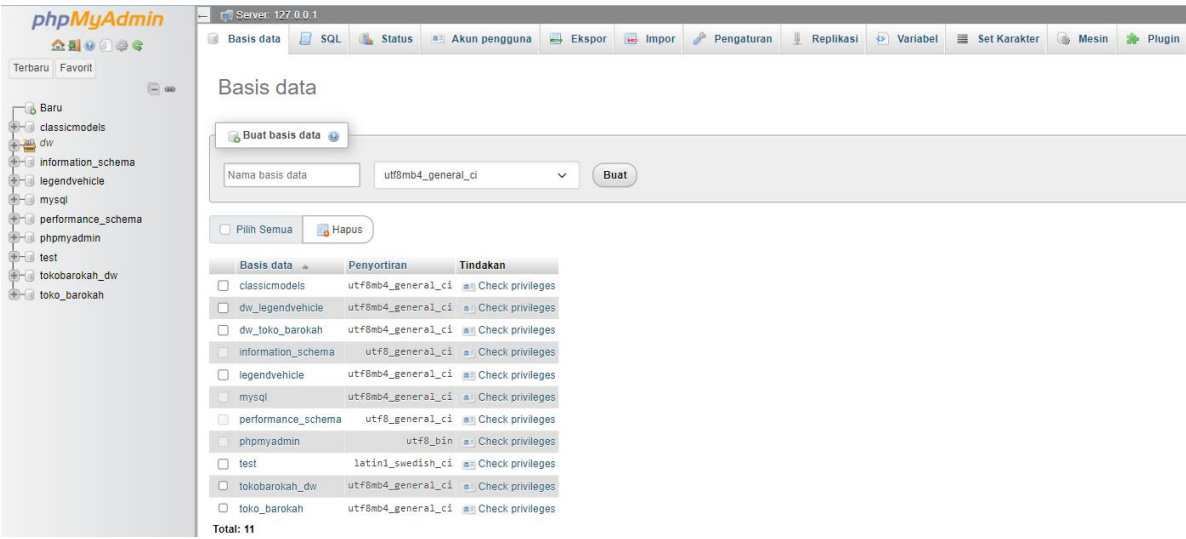


- tabel bernama *user*
- kolom tabel terdiri dari *id*, *username*, dan *password*
- *id* memiliki tipe data integer dan di atur sebagai *primary key*
- *username* dan *password* memiliki tipe data *varchar*
- *password* akan dienkripsi menggunakan fungsi *md5*

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”
6	Berikut ini adalah kolom-kolom dalam tabel <i>user</i>

Server: 127.0.0.1 Database: praktikwebdb

Nama tabel: user

Nama	Jenis	Panjang/Nilai	Bawaan	Penyortiran	Atribut	Tak Ternilai	Indeks	A.J	Komentar	Virtuality	Move column
id	INT	11	Tidak ada				PRIMARY	PRIMARY			
username	VARCHAR	50	Tidak ada	utf8mb4_general							
password	VARCHAR	50	Tidak ada	utf8mb4_general							

Komentar tabel: Collation: Storage Engine: InnoDB

PARTITION definition:

Partitions:

7 Masukkan data sebagai berikut ke dalam tabel user

Server: 127.0.0.1 Database: praktikwebdb Tabel: user

Kolom	Jenis	Fungsi	Tak Ternilai	Nilai
id	int(11)			1
username	varchar(50)			admin
password	varchar(50)	MD5		123

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

Server: 127.0.0.1 Database: praktikwebdb Tabel: user

Menampilkan baris 0 - 0 (total 1, Pencarian dilakukan dalam 0,0002 detik.)

`SELECT * FROM `user``

☐ Profil

☐ Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini

Extra options

	id	username	password
<input type="checkbox"/> Ubah <input type="button" value="Salin"/> <input type="button" value="Hapus"/>	1	admin	202cb962ac59075b964b07152d234b70

☐ Pilih Semua Dengan pilihan:

☐ Tampilkan semua | Jumlah baris: 25 | Saring baris: Cari di tabel ini




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

Run SQL query/queries on table praktikwebdb.user:

```

1 CREATE TABLE 'user' (
2   id int PRIMARY KEY,
3   username VARCHAR(50),
4   password VARCHAR(50)
5 );

```

10	<p>Tuliskan <i>query</i> untuk memasukkan dan ke tabel user menggunakan fungsi <code>mysqli_query()</code> (soal no 3)</p> <div><div>Run SQL query/queries on table prakwebdb.user: ⓘ</div><div><pre>1 INSERT INTO user (id, username, password) 2 VALUES (2, "user", "user");</pre></div></div> <div><div><div>↩ T →</div><div>▼</div><div>id</div><div>username</div><div>password</div></div><table><tr><td><input type="checkbox"/></td><td> Ubah</td><td> Salin</td><td> Hapus</td><td>1</td><td>admin</td><td>202cb962ac59075b964b07152d234b70</td></tr><tr><td><input type="checkbox"/></td><td> Ubah</td><td> Salin</td><td> Hapus</td><td>2</td><td>user</td><td>user</td></tr></table></div>	<input type="checkbox"/>	 Ubah	 Salin	 Hapus	1	admin	202cb962ac59075b964b07152d234b70	<input type="checkbox"/>	 Ubah	 Salin	 Hapus	2	user	user
<input type="checkbox"/>	 Ubah	 Salin	 Hapus	1	admin	202cb962ac59075b964b07152d234b70									
<input type="checkbox"/>	 Ubah	 Salin	 Hapus	2	user	user									



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 mencocokkan *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	<p>Pastikan file <code>koneksi.php</code> dari jobsheet sebelumnya, tersimpan di dalam direktori <code>praktik_php</code>.</p>  <p>The screenshot shows the content of the <code>koneksi.php</code> file: <pre>1 <?php 2 \$namaHost = "localhost"; 3 \$username = "naufal123"; 4 \$password = "123"; 5 \$database = "prakwebdb"; 6 7 try { 8 \$connect = mysqli_connect(\$namaHost, \$username, \$password, \$database); 9 } catch (exception \$e) { 10 echo \$e->getMessage(); 11 }</pre> </p>
2	<p>Buat file dengan nama <code>loginForm.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 <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	<p>Buat file dengan nama loginProses.html di dalam direktori praktik_php, 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 \$cek=mysqli_num_rows(\$result); 10 11 if(\$cek){ 12 echo "Anda berhasil login. silahkan menuju "; 13 Halaman HOME 14 } 15 else{ 16 echo "Anda gagal login. silahkan "; 17 Login kembali 18 } 19 <?php 20 echo mysqli_error(\$connect); 21 } 22 <?> </pre>
4	<p>Buat file dengan nama homeAdmin.html di dalam direktori praktik_php, kemudian ketikkan kode program berikut ini:</p> <pre> 1 <html> 2 <head> 3 </head> 4 <body> 5 <h2> Ini adalah halaman admin </h2> 6 </body> 7 </html> </pre>
5	<p>Simpan file dan jalankan kode program dengan mengetikkan localhost/dasarWeb/praktik_php/loginForm.php</p>
6	<p>Lakukan login dengan mengetikkan username dan password yang datanya belum tersimpan di database. Jelaskan hasil pengamatanmu (soal no 4)</p>

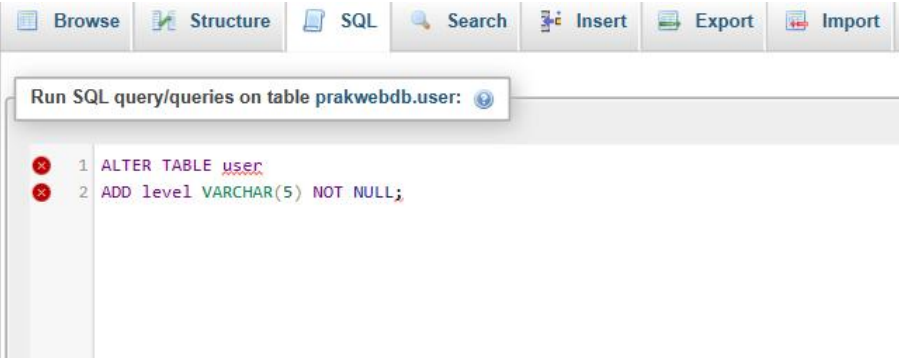
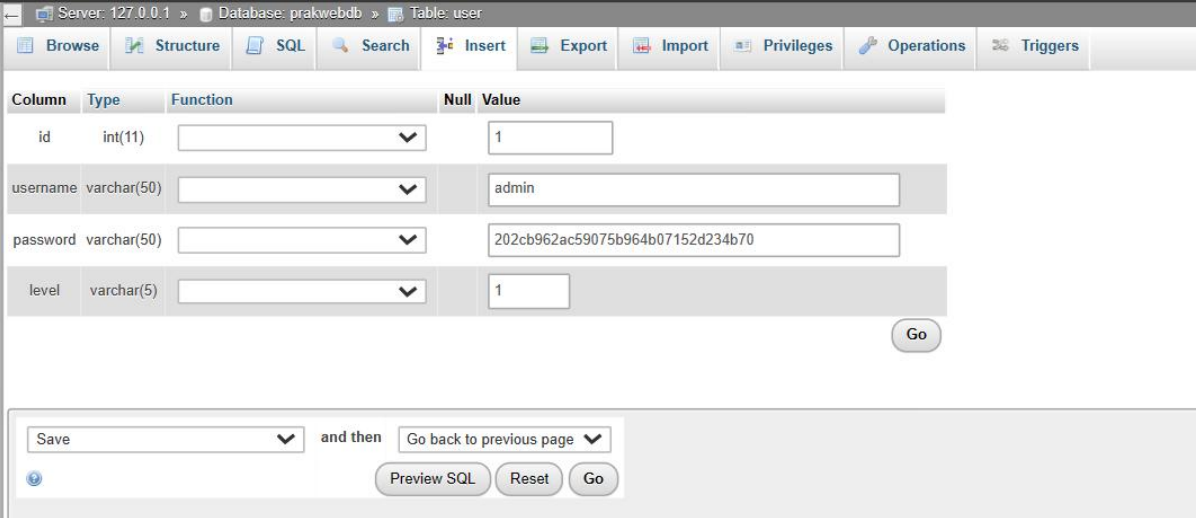
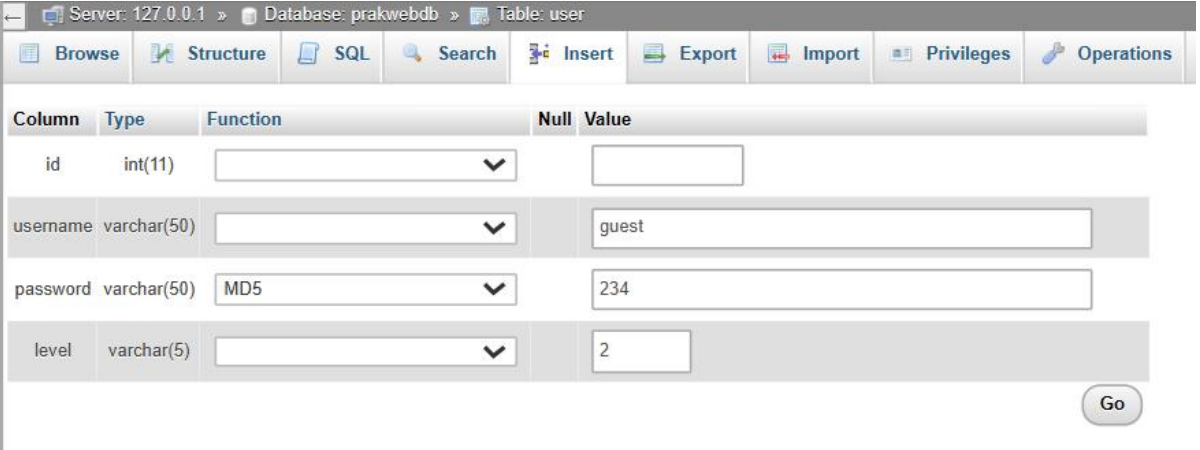
	 <p>Anda gagal login. silahkan Login kembali</p> <p>Terdapat tulisan Anda gagal login, silahkan Login Kembali, apabila login kembali nya di klik maka kembali ke tampilan pertama LoginForm.html karena menggunakan echo "Anda gagal login. silahkan "; ?> Login Kembali</p>
7	<p>Lakukan login dengan mengetikkan username dan password yang datanya sudah tersimpan di database. Jelaskan hasil pengamatanmu (soal no 5)</p>  <p>Anda berhasil login. silahkan menuju Halaman HOME</p> <p>Jadi pada login pertama menggunakan username dan password yang ada di database kemudian apabila username dan password benar akan lanjut ke halaman LoginProses.php, dan diarahkan menuju halaman admin, pada database menggunakan password md5 untuk supaya password nya acak di database nya.</p>

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	<p>Pada tabel user, ketikkan perintah SQL seperti berikut ini:</p> 
2	<p>Edit data admin, ubah level menjadi “1”</p> 
3	<p>Tambahkan data guest pada tabel user, dengan rincian data sebagai berikut:</p> 
4	<p>Amati hasil yang ditampilkan dan jelaskan hasil pengamatanmu (soal no 6)</p>

			id	username	password	level
<input type="checkbox"/>	Edit	Copy	Delete	1	admin	202cb962ac59075b964b07152d234b70 1
<input type="checkbox"/>	Edit	Copy	Delete	2	user	user
<input type="checkbox"/>	Edit	Copy	Delete	3	guest	289dff07669d7a23de0ef88d2f7129e7 2

- **Penjelasan:** Pada database terdapat penambahan data pada tabel user yaitu data guest dengan level 2, pada kolom id menjadi 0 karena otomatis dari mysql nya.

5

Tunjukkan hasil *screenshot* isi tabel user (soal no 7)

				id	username	password	level
<input type="checkbox"/>	Edit	Copy	Delete	1	admin	202cb962ac59075b964b07152d234b70	1
<input type="checkbox"/>	Edit	Copy	Delete	2	user	user	
<input type="checkbox"/>	Edit	Copy	Delete	3	guest	289dff07669d7a23de0ef88d2f7129e7	2

- **Penjelasan:** Pada database terdapat penambahan data pada tabel user yaitu data guest dengan level 2, pada kolom id menjadi 0 karena otomatis dari mysql nya

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 praktik_php.
2	<p>Buka file loginForm.html kemudian modifikasi kode program dengan mengubah nilai atribut action di baris ke-5 menjadi loginMultiProses.php, 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 loginMultiProses.php di dalam direktori praktik_php, 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 homeGuest.html di dalam direktori praktik_php, 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	Lakukan login dengan mengetikkan data admin. Jelaskan hasil pengamatanmu (soal no 8)

Username > admin

Password > ...

\$nbsp; > proses

Anda berhasil login. silahkan menuju [Halaman HOME](#)

Ini adalah halaman admin

- **Penjelasan:** Jadi pada login pertama menggunakan username dan password yang ada di database kemudian apabila username dan password benar akan lanjut ke halaman LoginProses.php, dan diarahkan menuju halaman admin, pada database menggunakan password md5 untuk supaya password nya acak di database nya.

6

Lakukan login dengan mengetikkan data guest. Jelaskan hasil pengamatanmu (**soal no 9**)

- Tampilan Login :

Username > guest

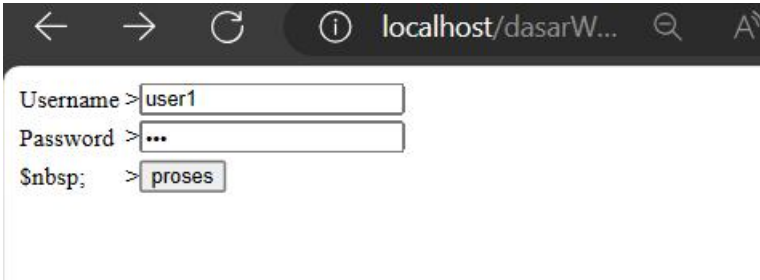
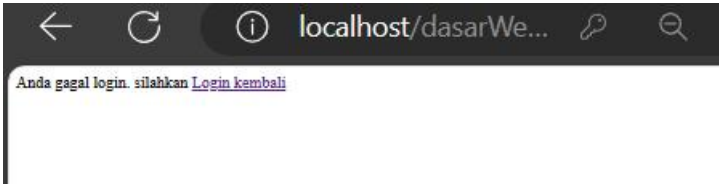
Password > ...

\$nbsp; > proses

Anda berhasil login. silahkan menuju [Halaman HOME](#)

Ini adalah halaman guest

- **Penjelasan:** Jadi pada halaman pertama login menggunakan username guest dan

	<p>password 234, kemudian jika berhasil login akan menuju ke halaman home yang ada di file loginProses.php dan apabila di klik Halaman Home maka lanjut ke halaman guest yang ada di file 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>Jawaban: Terdapat tulisan Anda gagal login, silahkan Login Kembali, apabila login kembali nya di klik maka kembali ke tampilan pertama LoginForm.html karena menggunakan echo "Anda gagal login. silahkan "; ?> Login Kembali.</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 pengenalan *cookies*

value, berisi nilai yang tersimpan dalam *cookies*

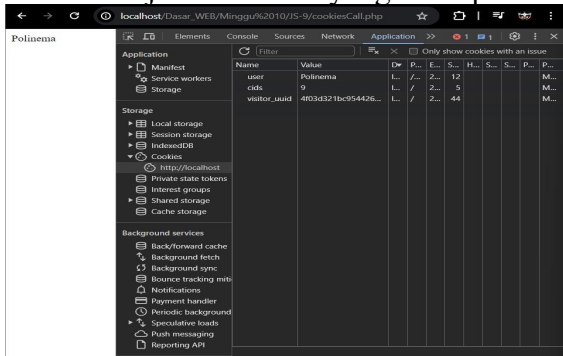
expire, adalah jangka waktu *cookies* akan disimpan di dalam komputer

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

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

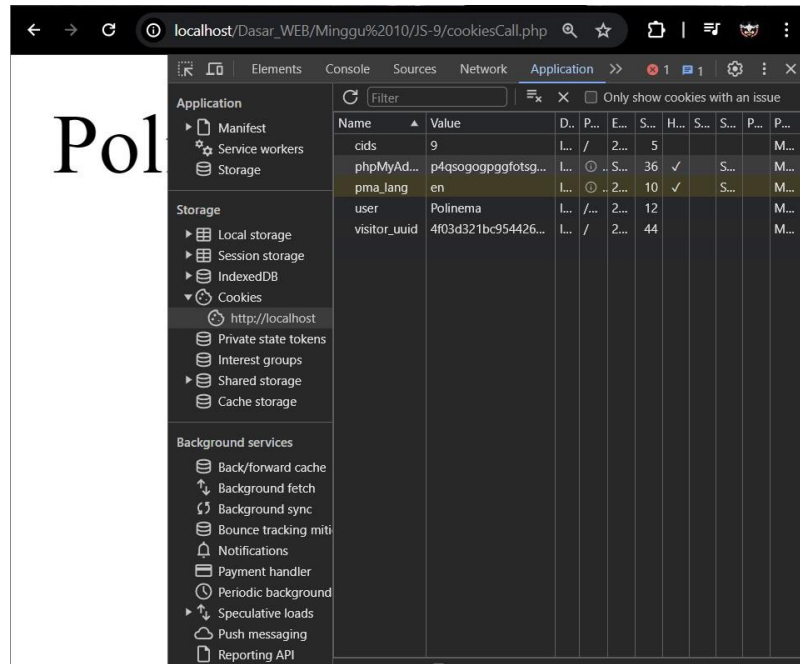
Ikuti langkah berikut untuk memahami cara kerja *cookies*:

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) 

	<p>Penjelasan: Jadi dalam output nya terdapat cookies yang jika di ctrl+shift+i / F12 dan masuk ke area application maka ada cookies nya yang memiliki value Polinema dan Name nya user, dan cookies tersebut akan berlaku selama 60 menit.</p>
6	Buka <i>browser</i> dan jalankan kode program langkah ke-2 dengan mengetikkan localhost/dasarWeb/praktik_php/cookiesCreate.php
7	Ulangi langkah ke-4
8	<p>Amati dan jelaskan hasil yang ditampilkan (soal no 12)</p>  <p>Penjelasan: Tetap sama dengan gambar di atas karena cookies di atur dengan waktu 60 menit, jadi jika melebihi 60 menit maka cookies tidak ada di dalam output tersebut.</p>
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

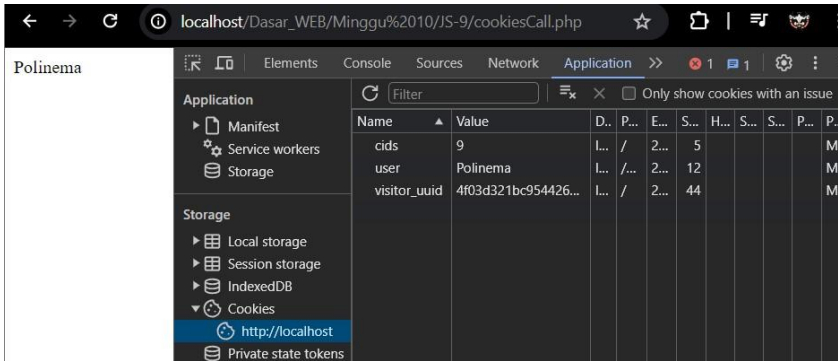
12

Amati dan jelaskan hasil yang ditampilkan (soal no 13)



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:


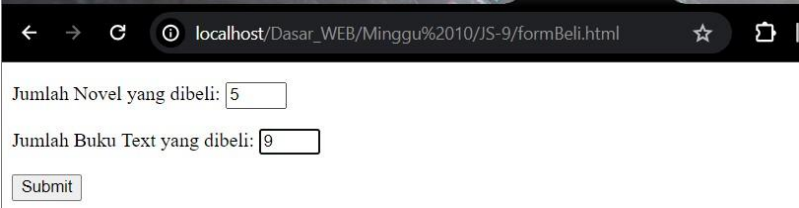

Langkah	Keterangan
1	Buat file baru dengan nama <code>cookiesDel.php</code> di dalam direktori <code>praktik_php</code> , kemudian ketikkan kode berikut: <pre>1 <?php 2 setcookie("user", "Polinema", time()-3600); 3 ?></pre>
2	Buka <i>browser</i> dan jalankan kode program dengan mengetikkan <code>localhost/dasarWeb/praktik_php/cookiesDel.php</code>
3	Buka <i>browser</i> dan jalankan kode program dari praktikum bagian 1 dengan mengetikkan <code>localhost/dasarWeb/praktik_php/cookiesCall.php</code>
4	Amati dan jelaskan hasil yang dihasilkan dari langkah 2 dan 3, kemudian beri kesimpulan (soal no 14)  Penjelasan: Jika diubah menjadi <code>-3600</code> , maka cookie "user" akan dihapus dari browser user karena waktu kadaluwarsanya yaitu 3600 detik (1 jam) yang lalu.

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:

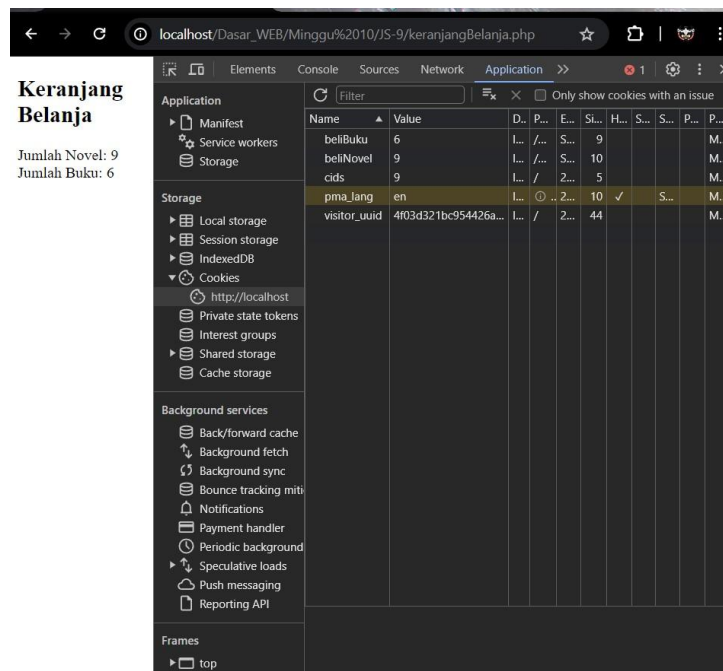
Langkah	Keterangan
1	Buat file baru dengan nama <code>formBeli.html</code> di dalam direktori <code>praktik_php</code> , kemudian ketikkan kode berikut:

	<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 </body> 15 </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>Jumlah Novel: 5 Jumlah Buku: 9</p> <p>Penjelasan: Pada program tersebut pertama menampilkan formBeli.html yang merupakan inputan menggunakan html, dan ketika sudah menginputkan dengan sesuai maka lanjut ke halaman keranjang belanja menggunakan cookies.</p>
6	Jalankan kode program langkah ke-1 dengan mengetikkan <code>localhost/dasarWeb/praktik_php/formBeli.html</code>
7	Isikan jumlah novel dan buku teks yang akan dibeli kemudian klik tombol “submit”
8	<p>Amati dan jelaskan hasil yang ditampilkan (soal no 16)</p> <div style="display: flex; justify-content: space-around;">   </div> <p>Penjelasan: Pada program tersebut pertama menampilkan formBeli.html yang merupakan inputan menggunakan html, dan ketika sudah menginputkan dengan sesuai maka lanjut ke halaman keranjang belanja menggunakan cookies.</p>
9	Tutup <i>browser</i> kemudian buka kembali <i>browser</i> kemudian jalankan kembali kode program langkah ke-3 dengan mengetikkan <code>localhost/dasarWeb/praktik_php/keranjangBelanja.php</code>

10

Amati dan jelaskan hasil yang ditampilkan (soal no 17)

➤ **Penjelasan:** Tidak ada perubahan dan tetap sama.

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 `$_SESSION`. Berikut adalah sintaks menggunakan variable global `$_SESSION`:

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

Ikuti langkah berikut untuk memahami cara kerja *session*:

Langkah	Keterangan
1	<p>Buat file baru dengan nama <code>sessionCreate.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre> 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> </pre>
2	<p>Buat file baru dengan nama <code>sessionCall.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <pre> 1 <?php 2 session_start(); 3 ?> 4 <!DOCTYPE html> 5 <html> 6 <body> 7 <?php 8 echo "Favorite color is " . \$_SESSION["favcolor"] . "
"; 9 echo "Favorite animal is " . \$_SESSION["favanimal"] . "."; 10 ?> 11 </body> 12 </html> </pre>
3	<p>Buka <i>browser</i> dan jalankan kode program pada langkah ke-2 dengan mengetikkan <code>localhost/dasarWeb/praktik_php/sessionCall.php</code></p>
4	<p>Buka <i>browser</i> dan jalankan kode program langkah ke-1 dengan mengetikkan <code>localhost/dasarWeb/praktik_php/sessionCreate.php</code></p>
5	<p>Ulangi langkah ke-3</p>
6	<p>Amati dan jelaskan hasil yang ditampilkan (soal no 18)</p>  <p>Langkah ke 2:</p>



Penjelasan: Pada output di atas menampilkan variabel session di tetapkan yang artinya tidak dapat berubah.

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	<p>Buat file baru dengan nama <code>sessionDel.php</code> di dalam direktori <code>praktik_php</code>, kemudian ketikkan kode berikut:</p> <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 <code>localhost/dasarWeb/praktik_php/sessionDel.php</code>
3	Buka <i>browser</i> dan jalankan kode program dari praktikum bagian 1 dengan mengetikkan <code>localhost/dasarWeb/praktik_php/sessionCall.php</code>
4	<p>Amati dan jelaskan hasil yang ditampilkan (soal no 19)</p>  <p>All session variables are now removed, and the session is destroyed.</p> <p>➤ Output pada sessionDel.php</p>  <p>Warning: Undefined array key "favcolor" in D:\XAMPP\htdocs\Dasar_WEB\Minggu 10\JS-9\sessionCall.php on line 8 Favorite color is .</p> <p>Warning: Undefined array key "favanimal" in D:\XAMPP\htdocs\Dasar_WEB\Minggu 10\JS-9\sessionCall.php on line 9 Favorite animal is .</p>

	<p>➤ Output pada sessionCall.php</p> <p>➤ Penjelasan: Jadi pada tampilan tersebut semua variabel session di hapus dan di hilangkan, maka pada sessionCall.php terjadi eror.</p>
--	---

Praktikum Bagian 10. Penerapan *Session* pada Fitur Login

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

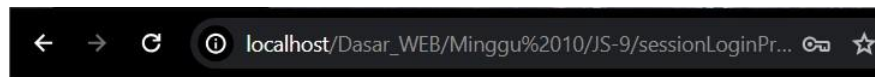
Langkah	Keterangan
1	Jalankan MySQL pada XAMPP
2	Pastikan file koneksi.php yang telah dibuat pada praktikum jobsheet sebelumnya ada di dalam direktori praktik_php
3	<p>Buat file baru dengan nama sessionLoginForm.html di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <html> 2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

	<pre> 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 Halaman Home <?php 18 }else{ 19 ?> 20 Gagal login, silahkan login lagi 21 Halaman Login <?php 22 echo mysqli_error(\$connect); 23 } 24 ?> </pre>
5	<p>Buat file baru dengan nama homeSession.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 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
 Log Out 12 } 13 <?php 14 }else{ 15 echo "anda belum login. silahkan "; 16 Log In 17 } 18 ?> 19 </body> 20 </html> </pre>
6	<p>Buat file baru dengan nama sessionLogout.php di dalam direktori praktik_php, kemudian ketikkan kode berikut:</p> <pre> 1 <?php 2 session_start(); 3 session_destroy(); 4 5 echo "Anda berhasil logout"; 6 ?> </pre>
7	Buka <i>browser</i> 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)

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



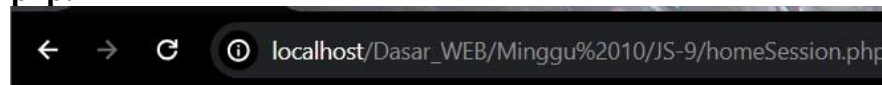
Output pada file sessionLoginForm.html:



Anda berhasil Login, silahkan menuju [Halaman Home](#)

sessionLoginProses.

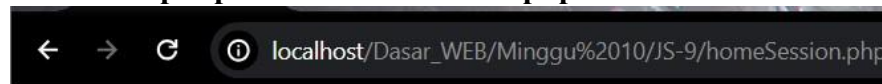
php:



selamat datang admin

[Log Out](#)

➤ **Output pada file homeSession.php:**



selamat datang admin

[Log Out](#)

➤ **Output pada file sessionLogout.php**



Anda berhasil logout

- **Penjelasan:** Ketika login menggunakan username dan password yang ada di database dan berhasil maka lanjut ke halaman sessionLoginProses.php dan di beri session, kemudian jika di klik Halaman Home akan menuju ke homeSession.php dan ada session nya juga, dan jika di klik logout maka lanjut ke halaman sessionLogout.php untuk halaman terakhir nya, dan apabila tidak menginputkan username dan password yang ada di dalam database maka akan menampilkan pesan Gagal login dan di arahkan untuk login kembali, hal tersebut terdapat di file sessionLoginProses.php.

10

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

1. Menginputkan username dan password yang ada di database, jika salah maka diarahkan untuk login kembali.
2. Jika login berhasil maka menuju halaman pada file sessionLoginProses.php dan diarahkan menuju ke halaman Home.

	<ol style="list-style-type: none"> 3. Setelah di klik link ke Halaman Home maka lanjut ke halaman admin yang ada di homeSession.php, dan ada link untuk Logout, 4. Setelah diklik logout maka akan menuju halaman terakhir yang ada di dalam file sessionLogout.php dan terdapat pesan 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 Beginners's Guide to Programming Interactive Web Applications with PHP and MySQL, 5th Edition. Plum Island Publishing