

## PERTEMUAN 15

### SESSION

#### A. CAPAIAN PEMBELAJARAN

Capaian pembelajaran dari pertemuan ini diharapkan mahasiswa mampu mendeskripsikan pengertian session, membuat session, memperbaiki session dan menghapus session.

#### B. URAIAN MATERI

##### 1. Session

Session biasa digunakan oleh programmer untuk menyimpan informasi pengguna. Informasi yang disimpan bersifat sementara sesuai dengan kebutuhan sistem. Biasanya programmer menggunakan session ini untuk menyimpan data login, setelah login data yang tersimpan pada session digunakan untuk validasi halaman yang dibuka oleh pengguna. Setelah pengguna keluar dari sistem atau aplikasi biasanya data session juga dihapus. Hal ini bertujuan untuk keamanan dari sistem atau aplikasi tersebut.

Session sering kali digunakan oleh programmer sebagai validasi pengguna pada saat login, dengan session programmer dapat mengelola siapa-siapa yang dapat membuka halaman tertentu. Misal, untuk dapat membuka halaman admin pada suatu situs web, pengguna harus terlebih dahulu melakukan login. Hal yang terjadi pada saat login, setelah login programmer membuat session yang didalamnya tersimpan data pengguna kemudian session tersebut akan selalu diperiksa ketika membuka halaman lain yang membutuhkan validasi. Pada halaman admin web, status session akan diperiksa isinya. Jika data yang terdapat pada session isinya sesuai maka pengguna dapat mengakses/membuka halaman Admin, namun jika tidak sesuai maka pengguna tidak dapat mengakses/membuka halaman Admin dan umumnya akan diarahkan ke halaman login agar pengguna melakukan login dahulu. Halaman admin tidak dapat diakses langsung tanpa melakukan login.

Berikut beberapa proses yang perlu diperhatikan pada saat menggunakan session:

a. Proses pembuatan session

Sebelum membuat session kita perlu mengetahui fungsi yang digunakan, fungsi yang digunakan untuk membuat session pada PHP adalah `session_start()`. Fungsi `session_start` perlu dipanggil terlebih dahulu sebelum membuat session. Sesion start juga dapat di jalan kan secara otomatis dengan mengubah konfigurasi pada file `php.ini`, pada bagian `session.auto_start` ubah nilai 0 menjadi 1. Cara lain yang dapat digunakan jika kita tidak memiliki akses ke dalam apache adalah dengan memodifikasi file `.htaccess`, tambahkan `php_value session.auto_start 1` pada file `.htaccess` aplikasi anda. Fungsi `session_start()` juga perlu dipanggil ketika memeriksa, memanipulasi variabel `$_sessio` dan jungan menghapus session.

Contoh pemangilan `session_start()` pada file php

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

Untuk membuat session kita menggunakan variabel super global dalam bentuk array yang sudah disiapkan ketika kita memnggil fungsi `session_start()`. Nama bariabel array tersebut adalah `$_SESSION`. `$_session` merupakan variabel array jadi ketiak akan digunakan kita bisa menuliskan nama elemnnya misa `$_SESSION['username']`, `$_SESSION['password']` dan lain-lain

Contoh pembuatan session

```
<?php
// start a session
session_start();

$username="admin";

$password="12345678";

$status=true;

// mengisi nilai session
```

```
$_SESSION['username'] = $username;  
$_SESSION['password'] = $password;  
$_SESSION['login'] = $status;  
?>
```

b. Proses pemeriksaan session

Untuk memeriksa session dapat dilakukan dengan cara memanggil variabel session menggunakan perintah `isset()`.

Contoh

```
<?php  
session_start();  
if (isset($_SESSION['login'])) {  
    echo $_SESSION['username'];  
    echo "<br> ".$_SESSION['password'];  
    echo "<br> ".$_SESSION['login'];  
  
} else {  
    echo "session belum dibuat";  
}  
?>
```

c. Proses penghapusan variabel session dan session

Untuk menghapus semua data terkait session yang sudah dibuat, dapat menggunakan fungsi `session_destroy`. Sedangkan untuk menghapus variabel tertentu dalam session fungsi yang digunakan adalah `unset`.

Contoh penghapusan beberapa variabel session

```
<?php  
session_start();  
unset($_SESSION['username']);  
unset($_SESSION['password']);
```

```
unset($_SESSION['login']);
```

```
?>
```

Contoh penghapusan session

```
<?php
```

```
session_start();
```

```
//penghapusan session
```

```
session_destroy();
```

```
?>
```

## 2. Membuat Session

Membuat Session : session1.php

```
<?php
```

```
/******
```

Membuat

session

```
*****/
```

```
session_start();
```

```
if (isset ($_POST['Login'])) {
```

```
$user1 = $_POST['user'];
```

```
$pass1 = $_POST['pass'];
```

```
//cek login
```

```
if ($user1== "rahadian" && $pass1 = "123") {
```

```
//membuat session
```

```
$_SESSION['login'] = $user1;
```

```
//menuju ke halaman pemeriksaan session
```

```
echo "<h1>Anda berhasil LOGIN</h1>";
```

```
echo "<h2>Klik <a href='session2.php'>di sini
```

```

(session2.php)</a>
untuk menuju ke halaman pemeriksaan session";
}
} else {
?>
<html>
<head>
<title>Login here...</title>
</head>
<body>
<form action="" method="post">
<h2>Login Here...</h2>
Username : <input type="text" name="user"><br>
Password : <input type="password" name="pass"><br>
<input type="submit" name="Login" value="Log In">
</form>
</body>
</html>

```

### 3. Update Session

Pemeriksaan Session : session2.php

```

<?php
/******
Pemeriksaan
Session

******/

```

```

session_start();

//cek session

if (isset($_SESSION['login'])) {

//jika sudah login

//menampilkan isi session

echo "<h1>Selamat Jumpa kembali ". $_SESSION['login'] . "</h1>";

echo "<h2>Jika sudah login Anda dapat mengakses halaman ini </h2>";

echo "<h2>Klik <a href='session3.php'>di sini

(session3.php)</a> untuk LOGOUT</h2>";

} else {

//jika belum login maka session tidak ada artinya

die ("Anda belum login! Anda tidak berhak masuk ke halaman

ini.Silahkan login <a href='session1.php'>di sini</a>");

}

?>

```

#### 4. Menghapus Session

Penghapusan Session : session3.php

```

<?php

/*****

Bagian ini adalah halaman logout, session yang ada akan dihapus

*****/

session_start();

if (isset($_SESSION['login'])) {

unset ($_SESSION);

session_destroy();

//

```

```

echo "<h1>Anda sudah berhasil LOGOUT</h1>";

echo "<h2>Klik <a href='session1.php'>di sini</a> untuk
LOGIN kembali</h2>";

echo "<h2>Anda sekarang tidak bisa masuk ke halaman
<a href='session2.php'>session2.php</a> lagi</h2>";
}
?>

```

### Login system dengan session

Siapkan database dan tabel user



#	Nama	Jenis	Penyortiran	Atribut	Tak Terilai	Bawaan	Komentar	Ekstra
1	id	int(11)			Tidak	Tidak ada		AUTO_INCREMENT
2	username	varchar(20)	utf8mb4_general_ci		Tidak	Tidak ada		
3	password	varchar(100)	utf8mb4_general_ci		Tidak	Tidak ada		

**Gambar 100** tabel user

Buat halaman login.php

```

<?php
include "koneksi.php";

?>

<!DOCTYPE html>

<html>

<head>

<title>Login</title>

</head>

<body>

<h2>Login</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";
}
else
echo "Anda berhasil Logout, Silahkan Login Kembali";

}
?>

<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>
```

Tampilan halaman login.php

## Login

Username :

Password :

**Gambar 101** tampilan halaman login

Buat koneksi.php

```
<?php

class koneksi

{
```

```
public function get_koneksi()
{
    $conn=mysqli_connect("localhost","root","","latihan");
    if(mysqli_connect_errno()){
        echo "Koneksi database gagal : " . mysqli_connect_error();
    }
    return $conn;
}
}

$konek=new koneksi();
$koneksi=$konek->get_koneksi();
?>
```

Buat cek\_login.php

```
<?php
session_start();
include 'koneksi.php';
$username=$_POST['username'];
$password=$_POST['password'];

$data = mysqli_query($koneksi,"select * from tabel_user where
username='$username' and password='$password'");

$cek=mysqli_num_rows($data);
if($cek>0){
```

```
$_SESSION['username']=$username;

$_SESSION['status']="login";

header("location:dashboard.php");

}else{

    header("location:login.php?pesan=gagal");

}

?>
```

Buat halaman dashboard.php

```
<!DOCTYPE html>

<html>

<head>

<title>Login</title>

</head>

<body>

<h2>Dashboard</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/>  
  
<h3> Halaman ini tampil karena anda berhasil login</h3>  
  
</body>  
  
</html>
```

Buat file logout.php

```
<?php  
  
session_start();  
  
session_destroy();  
  
header("location:login.php?pesan=ulang");  
  
?>
```

Uji Coba

Buka browser jalankan file login.php, Masukkan username dan password yang salah

## Login

Login gagal! username dan password salah!

Username :   
Password :

**Gambar 102** Tampilan jika gagal login

Masukkan username dan password yang benar

## Dashboard

Halaman ini tampil karena anda berhasil login

[logout](#)

**Gambar 103** tampilan halaman dashboard

Tampilan halaman ketika memilih logout

## Login

Anda berhasil Logout, Silahkan Login Kembali

Username :	<input type="text" value="Masukkan username"/>
Password :	<input type="password" value="Masukkan password"/>
	<input type="button" value="LOGIN"/>

**Gambar 104** tampilan setelah logout

Ketika memilih logout maka session akan dihapus dengan perintah `session_destroy()` kemudian menampilkan kembali halaman login dengan pesan “anda berhasil logout, silahkan login kembali”.

### C. LATIHAN SOAL

1. Apa gunanya session?
2. Buatlah program menggunakan session selain yang telah dicontohkan diatas, buat algoritmanya dan output yang dihasilkan

**D. DAFTAR PUSTAKA**

Abdul Kadir, 2002, Dasar Pemrograman Web Dinamis Menggunakan PHP, Andi Yogyakarta.

<http://www.w3schools.com>

<http://www.freewebmasterhelp.com/tutorials/phpmysql/>

<http://php.about.com/>

<http://www.howstuffworks.com/dns.htm>

<http://www.co.cc/>

<http://www.w3c.org>

[http://www.quackit.com/php/tutorial/php\\_mail\\_configuration.cfm](http://www.quackit.com/php/tutorial/php_mail_configuration.cfm)