



Modul-6 Session, Cookie, Encryption

Pemrograman Web
TEKNIK INFORMATIKA
UNIVERSITAS PASUNDAN BANDUNG

Pada modul kali ini, kita akan membuat halaman login yang digunakan untuk masuk ke halaman back-end (halaman admin.php) dari halaman front-end(index.php). Dalam halaman login kita akan menggunakan session, cookie dan encryption.

Pada latihan di modul kali ini kita juga akan menggunakan database yang sudah di buat pada modul sebelumnya dan kita masih menggunakan tabel yang sesuai dengan tugas besar masing-masing. **Jangan lupa juga untuk menyertakan file .sql yang sudah di ekspor pada saat pengumpulan, bukan hanya mengumpulkan file-file web-nya saja.**

Pendahuluan

Session

- Session adalah konsep abstrak untuk merepresentasikan interaksi antara browser / client dan server
- Session disimpan di server.
- Session dibangun diatas cookie.
- Setiap request, client mengirimkan cookie yang berisi session ID (PHPSESSID).
- Server menggunakan session ID tersebut untuk mencari data session.

Fungsi-fungsi session :

```
<?php
// fungsi untuk memulai session
session_start();

// mendefinisikan variabel super global session
// $_SESSION['nama key'] = 'value';
$_SESSION['nama'] = "Avip Syaifulloh";

// untuk menampilkan isi session
echo $_SESSION['nama'];

// fungsi untuk mematikan session
session_destroy();
?>
```

Cookie

Cookie adalah informasi yang dikirim oleh server kepada browser, yang kemudian dikirim kembali oleh browser ke server untuk request berikutnya. Cookie ini adalah sebuah Variable Global yang nilainya disimpan di Browser.

Fungsi-fungsi cookie :

```
<?php
// deklarasi cookie
// setcookie(name, value, expire, path, domain, secure);

// contoh :
// set cookie dengan nama 'nama', value 'avip' selama 24 jam
setcookie("nama", "avip", time() + 60 * 60 * 24);

// untuk menampilkan isi cookie
echo $_COOKIE['nama'];

// fungsi untuk mematikan cookie
// set ulang cookie dengan nama yang sama
// isi valuenya dengan string kosong dan set waktunya minus
setcookie("nama", "", -3600)
?>
```

Encryption

Enkripsi digunakan untuk mengamankan password pengguna pada sistem login, agar password pengguna tidak akan diketahui oleh oknum yang tidak bertanggung jawab.

Redirect Halaman

Redirect page atau pengalihan halaman adalah suatu cara yang digunakan untuk mengalihkan suatu halaman website menuju webpage lainnya.

```
<?php
header("Location: ....");

?>
```

Persiapan

Ada beberapa hal yang harus kalian siapkan.

1. Buatlah tabel user pada database yang sama dengan pertemuan sebelumnya dengan ketentuan seperti dibawah ini :

The screenshot shows the phpMyAdmin interface for creating a table named 'user'. The table has four columns: 'id' (INT, PRIMARY), 'username' (VARCHAR(64)), 'password' (VARCHAR(256)), and an unnamed column (INT). The storage engine is set to InnoDB.

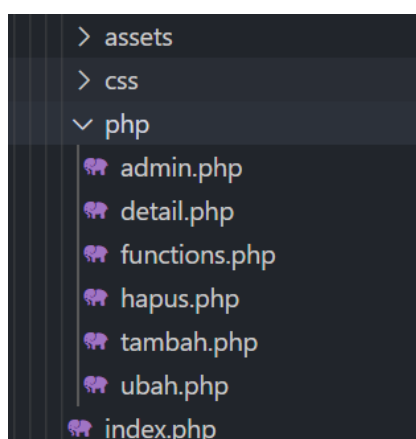
Name	Type	Length/Values	Default	Collation	Attributes	Null	Index
id	INT		None			<input type="checkbox"/>	PRIMARY
username	VARCHAR	64	None			<input type="checkbox"/>	---
password	VARCHAR	256	None			<input type="checkbox"/>	---
	INT		None			<input type="checkbox"/>	---

Table comments: Collation: Storage Engine: InnoDB

2. Tambahkan sebuah data seperti berikut ini :

	id	username	password
1	admin	admin	

3. Buatlah folder tugas6 didalam folder praktikum yang terhubung dengan repository github kalian(seperti pada modul sebelumnya).
4. Copy folder tugas modul sebelumnya kemudian rename menjadi latihan6a.



Latihan6a : Login dan session

1. Modifikasi link yang menuju ke halaman admin di halaman index.php agar tidak langsung ke halaman admin tapi melalui halaman login.php terlebih dahulu.

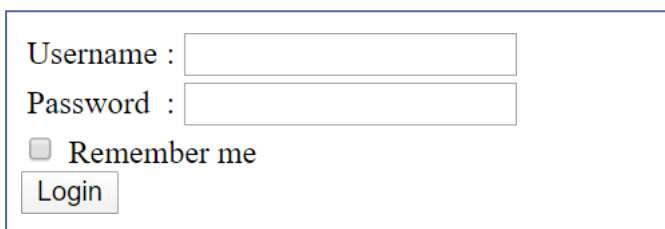
Contoh :

```
<a href="php/login.php">
  <button type="">Masuk ke halaman admin</button>
</a>
```

2. Buatlah halaman login.php didalam folder php kemudian buatlah tampilan login semenarik mungkin contoh kodingan :

```
<form action="" method="post">
  <?php if (isset($error)) : ?>
    <p style="color: red; font-style: italic;">Username atau Password salah</p>
  <?php endif; ?>
  <table>
    <tr>
      <td><label for="username">Username</label></td>
      <td><input type="text" name="username"></td>
    </tr>
    <tr>
      <td><label for="password">Password</label></td>
      <td><input type="password" name="password"></td>
    </tr>
  </table>
  <div class="remember">
    <input type="checkbox" name="remember">
    <label for="remember">Remember me</label>
  </div>
  <button type="submit" name="submit">Login</button>
</form>
```

Contoh Output :



Username :

Password :

☐ Remember me

Login

- * Gunakan style / framework css agar tampilan yang dibuat lebih bagus.
- * Style boleh berbeda namun harus ada dibagian `isset($error)` untuk menampilkan jika username/password salah
- * atribut name dan type dari tag input dan button harus sama.

3. Tambahkan kodingan berikut dibagian atas file login.php

```
<?php
session_start();
require 'functions.php';
// melakukan pengecekan apakah user sudah melakukan login jika sudah redirect ke halaman admin
if (isset($_SESSION['username'])) {
    header("Location: admin.php");
    exit;
}
// Login
if (isset($_POST['submit'])) {
    $username = $_POST['username'];
    $password = $_POST['password'];
    $cek_user = mysqli_query(koneksi(), "SELECT * FROM user WHERE username = '$username'");
    // mencocokkan USERNAME dan PASSWORD
    if (mysqli_num_rows($cek_user) > 0) {
        $row = mysqli_fetch_assoc($cek_user);
        if ($password == $row['password']) {
            $_SESSION['username'] = $_POST['username'];
            $_SESSION['hash'] = $row['id'];
        }
        if ($row['id'] == $_SESSION['hash']) {
            header("Location: admin.php");
            die;
        }
        header("Location: ../index.php");
        die;
    }
    $error = true;
}
?>
```

4. Buatlah file logout.php didalam folder php, kemudian copy kodingan berikut kedalamnya.

```
<?php
session_start();
session_destroy();
header("Location: ../index.php");
die;
?>
```

5. Modifikasi file admin.php(halaman backend), tambahkan button/link untuk logout yang jika di klik akan masuk ke logout.php . Contoh :

```
<div class="logout">
    <a href="logout.php">Logout</a>
</div>
```

Latihan6b : Registrasi dan enkripsi

1. Copy folder latihan6a kemudian rename menjadi latihan6b di folder tugas6.
2. Buatlah halaman registrasi.php di folder php, kemudian buatlah form input untuk menerima data username dan password yang akan dimasukkan ke dalam database.

Contoh :

```
<form action="" method="post">
  <table>
    <tr>
      <td><label for="username">username</label></td>
      <td><input type="text" name="username"></td>
    </tr>
    <tr>
      <td><label for="password">password</label></td>
      <td><input type="password" name="password"></td>
    </tr>
  </table>
  <button type="submit" name="register">REGISTER</button>
</form>
```

* namedan type dari input dan button samakan, untuk struktur dan style silahkan sesuaikan dengan keinginan kalian.

3. Tambahkan kodingan berikut dibagian atas registrasi.php :

```
<?php
require 'functions.php';

if (isset($_POST["register"])) {
  if (registrasi($_POST) > 0) {
    echo "<script>
      alert('Registrasi Berhasil');
      document.location.href = 'login.php';
    </script>";
  } else {
    echo "<script>
      alert('Registrasi Gagal');
    </script>";
  }
}
?>
```


Contoh output :

username :
 password :

* Buat tampilan semenarik mungkin, jangan sesimple seperti di modul.

4. Kemudian buat fungsi registrasi di file functions.php

```
function registrasi($data)
{
    $conn = koneksi();
    $username = strtolower(stripslashes($data["username"]));
    $password = mysqli_real_escape_string($conn, $data["password"]);

    // cek username sudah ada atau belum
    $result = mysqli_query($conn, "SELECT username FROM user WHERE username = '$username' ");
    if (mysqli_fetch_assoc($result)) {
        echo "<script>
            alert('username sudah digunakan');
        </script>";
        return false;
    }

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

    // tambah user baru
    $query_tambah = "INSERT INTO user VALUES(' ', '$username', '$password')";
    mysqli_query($conn, $query_tambah);

    return mysqli_affected_rows($conn);
}
```

5. Modifikasi halaman login.php tambahkan link menuju ke file registrasi.php. Contoh :

```
<div class="registrasi">
    <p>Belum punya akun ? Registrasi <a href="registrasi.php">Disini</a></p>
</div>
```

Contoh hasil:

Username :
 Password :
☐ Remember me

 Belum punya akun ? Registrasi [Disini](#)

* Buat tampilan semenarik mungkin, jangan sesimple seperti di modul.

6. Kemudian modifikasi kodingan php di file login.php menjadi seperti berikut :

```
<?php
session_start();
require 'functions.php';
// melakukan pengecekan apakah user sudah melakukan login jika sudah redirect ke halaman admin
if (isset($_SESSION['username'])) {
    header("Location: admin.php");
    exit;
}
// Login
if (isset($_POST['submit'])) {
    $username = $_POST['username'];
    $password = $_POST['password'];
    $cek_user = mysqli_query(koneksi(), "SELECT * FROM user WHERE username = '$username'");
    // mencocokkan USERNAME dan PASSWORD
    if (mysqli_num_rows($cek_user) > 0) {
        $row = mysqli_fetch_assoc($cek_user);
        if (password_verify($password, $row['password'])) {
            $_SESSION['username'] = $_POST['username'];
            $_SESSION['hash'] = hash('sha256', $row['id'], false);

            if (hash('sha256', $row['id']) == $_SESSION['hash']) {
                header("Location: admin.php");
                die;
            }
            header("Location: ../index.php");
            die;
        }
    }
    $error = true;
}
```

7. Coba untuk melakukan registrasi di file registrasi.php, kemudian cek tabel user di database dan lihat hasilnya.

Contoh (setelah dicoba memasukkan username : avip, dan password : avip):

id	username	password
1	admin	admin
2	avip	\$2y\$10\$SI/yfkgw.2dkpWpRAHON1.12cWICA5B13OXbeuZOPRq...

Bandingkan antara baris 1 dimana passwordnya itu sama dengan yang diinputkan oleh kita di form, sedangkan baris 2 password sudah mengalami enkripsi.

8. Coba lakukan login dengan username admin dan password admin, maka pasti akan gagal login, kemudian coba login dengan username dan password yang sudah kalian coba melalui registrasi, maka pasti akan berhasil karena sekarang sistem login hanya mengenali password yang sudah di enkripsi.

9. Hapus data user pertama(username : admin) dari database karena sudah tidak dapat digunakan kembali.

Latihan6c : Remember me (Cookie)

1. Copy folder latihan6b kemudian rename menjadi latihan6c di folder tugas6.
2. Modifikasi file login.php tambahkan kodingan berikut sebelum pengecekan login

```
// cek cookie
if (isset($_COOKIE['username']) && isset($_COOKIE['hash'])) {
    $username = $_COOKIE['username'];
    $hash = $_COOKIE['hash'];

    // ambil username berdasarkan id
    $result = mysqli_query(koneksi(), "SELECT * FROM user WHERE username = '$username'");
    $row = mysqli_fetch_assoc($result);

    // cek cookie dan username
    if ($hash === hash('sha256', $row['id'], false)) {
        $_SESSION['username'] = $row['username'];
        header("Location: admin.php");
        exit;
    }
}
```

3. Modifikasi file login.php tambahkan kodingan berikut dibagian pengecekan login

```
// Login
if (isset($_POST['submit'])) {
    $username = $_POST['username'];
    $password = $_POST['password'];
    $cek_user = mysqli_query(koneksi(), "SELECT * FROM user WHERE username = '$username'");
    // mencocokkan USERNAME dan PASSWORD
    if (mysqli_num_rows($cek_user) > 0) {
        $row = mysqli_fetch_assoc($cek_user);
        if (password_verify($password, $row['password'])) {
            $_SESSION['username'] = $_POST['username'];
            $_SESSION['hash'] = hash('sha256', $row['id'], false);

            // jika remember me dicentang
            if (isset($_POST['remember'])) {
                setcookie('username', $row['username'], time() + 60 * 60 * 24);
                $hash = hash('sha256', $row['id']);
                setcookie('hash', $hash, time() + 60 * 60 * 24);
            }

            if (hash('sha256', $row['id']) == $_SESSION['hash']) {
                header("Location: admin.php");
                die;
            }
            header("Location: ../index.php");
            die;
        }
    }
    $error = true;
}
```

4. Modifikasi file logout.php tambahkan kodingan berikut :

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

setcookie('username', '', time() - 3600);
setcookie('hash', '', time() - 3600);
header("Location: ../index.php");
die;
?>
```

5. Modifikasi file- file berikut :

- admin.php
- ubah.php
- tambah.php
- hapus.php

Tambahkan kodingan berikut dibagian paling atas kodingan php
(sebelum require 'functions.php') :

```
session_start();

if (!isset($_SESSION["username"])) {
    header("Location: login.php");
    exit;
}
```

Hal tersebut digunakan agar file-file tersebut tidak dapat diakses jika kita belum melakukan login.

Pengumpulan tugas

Silahkan masukkan kembali tugas pertemuan 6 ini ke dalam repositori github yang sama dengan pertemuan sebelumnya. Jangan lupa kirimkan juga databasenya yang sudah ditambah dengan tabel user. Kerjakan tugas dari jauh-jauh hari sebelum deadline agar anda memiliki waktu untuk bertanya kepada asisten.

Referensi

Youtube Web Programming UNPAS

<https://www.youtube.com/webprogrammingunpas>

Playlist PHP untuk pemula

<https://www.youtube.com/watch?v=L1W2OwV5rgY&list=PLFIM0718LjIUqXfmEIBE3-uzERZPh3vp6>

Video tutorial Registrasi

<https://www.youtube.com/watch?v=b2JUL5RDI18>

Video tutorial Login

<https://www.youtube.com/watch?v=2PAApp655es>

Video tutorial Session

<https://www.youtube.com/watch?v=PiYKZ65dkqc>

Video tutorial Cookie

<https://www.youtube.com/watch?v=maW4kzHrdKQ>

Website Pembelajaran

<http://www.webprogrammingunpas.com>

-- Jangan Lupa Titik Koma --