



Universitas Kristen Duta Wacana

Fakultas Teknologi Informasi

Prodi Informatika

Jln. Dr. Wahidin Sudiro Husodo No 5-25

Daerah Istimewa Yogyakarta 55224

Mata Kuliah	: Pemrograman Website
Kode Mata Kuliah	: TI0243
Periode	: Semester Genap 2020/2021
Dosen Pengampu	: Danny Sebastian S.Kom., M.M., M.T. : Maria Nila Anggia Rini S.T., M.TI. : Agata Filiana S.Kom., M.Cs.

BAB 8

PHP COOKIES & SESSION

A. CPL Pertemuan

- Membuat struktur HTML sederhana.
- Membuat CSS untuk memberikan styling pada dokumen HTML
- Mengatur layout dokumen HTML menggunakan CSS

B. Alat & Bahan

Kegiatan praktikum pada bagian ini membutuhkan perangkat keras (*hardware*) atau komputer dengan spesifikasi minimum:

- Processor : Intel Core i3
- Memory : 2 GB
- Hard disk : kapasitas kosong penyimpanan sebesar 3 GB

Selain perangkat keras, kegiatan praktikum bagian ini juga membutuhkan perangkat lunak (*software*):

- Sistem Operasi (Windows, Linux, atau MacOSX)
- Text editor (Atom, Visual Studio Code, atau editor lain)
- Browser (Google Chrome, Opera, atau Mozilla Firefox)
- Xampp (Web Server Apache & Database MariaDB)

C. Cookie & Session

C.1. Cookie

Cookie adalah file kecil yang disimpan di computer client sebagai salah satu media penyimpanan aplikasi website. Ukuran maksimum file cookie adalah 4KB. File cookie dibuat oleh aplikasi website.

Setelah file cookie diset/dibuat, maka setiap halaman yang di-request kepada aplikasi website tersebut akan mengirimkan nama cookie dan nilainya. Cookie hanya dapat diakses oleh domain aplikasi website yang membuatnya. Contoh, ketika cookie dibuat oleh aplikasi website dengan domain www.ukdw.ac.id, maka cookie tidak dapat diakses oleh aplikasi website dengan domain eclass.ukdw.ac.id. Contoh penggunaan cookie paling umum adalah untuk menampilkan advertising/iklan. Seperti iklan dari e-commerce yang sesuai dengan produk yang terakhir kali kita lihat pada e-commerce tersebut. Cookie yang dibuat oleh pengguna A, hanya dapat diakses oleh pengguna A, pengguna lain tidak dapat melihat nilai dari cookie tersebut.



Gambar 8.1 Cara kerja cookie (sumber: <https://www.guru99.com/cookies-and-sessions.html>)

Secara umum cara kerja cookie dapat dilihat pada Gambar 8.1. Pertama, pengguna mengakses sebuah halaman website yang meminta penyimpanan cookies. Kemudian, server mendapatkan informasi data pengguna. Terakhir, saat pengguna melakukan akses ke halaman lain pada aplikasi website tersebut, maka setiap request akan membawa data cookie yang sudah dibuat.

```

<?php
setcookie(cookie_name, cookie_value, [expiry_time], [cookie_path], [domain],
[secure], [httponly]);
?>
  
```

- Syntax PHP setcookie adalah nama fungsi yang akan digunakan untuk membuat cookie.
- Parameter “cookie_name” digunakan untuk mendefinisikan cookie. Parameter ini digunakan untuk mengakses index array \$_COOKIE.
- Parameter “cookie_value” digunakan untuk mendefinisikan isi cookie.
- Parameter “[expiry_time]” digunakan untuk mendefinisikan lama waktu hidup file cookie. Bersifat optional. Cara mendefinisikannya adalah menggunakan PHP time().

- Parameter “cookie_path” bersifat opsional. Digunakan untuk mendefinisikan path file cookie di server. Bila diset “/” maka file cookie akan dapat diakses pada semua domain. Sub-directory digunakan untuk limitasi akses cookie.
- Parameter “domain” adalah optional, dan digunakan untuk mendefinisikan cookie access.
- Parameter “secure” digunakan untuk mendefinisikan cookie dikirim melalui https atau http. Bersifat optional, secara default bernilai false.
- Parameter “httponly” bersifat optional. Bila diset menjadi true, maka hanya client side scripting (seperti javascript) yang dapat mengakses cookie.

Contoh dibawah ini adalah syntax membuat cookie pada PHP. Pada contoh dibawah ini, cookie akan hidup selama 60 detik. Cookie diakses dengan menggunakan variable \$_COOKIE['index']. Dengan 'index' adalah cookie_name. Cookie dapat dihapus dengan cara men-set cookie dengan time()-360 atau dengan nilai minus.

```
<?php
    setcookie("cookie_name", "cookie value", time()+ 60, '/');
    echo 'the cookie has been set for 60 seconds';
?>
```

C.2. Session

Session adalah cara yang digunakan untuk menyimpan informasi (dalam variabel) agar dapat digunakan di banyak halaman. Session merupakan variable global yang tersimpan di server. Setiap session memiliki ID unik yang digunakan untuk mengakses values. Setiap saat session dibuat, sebuah file cookie dibuat di computer client dengan menyimpan session ID milik computer client. Apabila client tidak meng-enable cookie, maka php session id dituliskan pada URL. Session dapat menyimpan data yang lebih besar daripada cookie. Nilai session akan otomatis terhapus apabila browser ditutup, sehingga bila ingin menyimpan value nya secara permanen harus disimpan di database.

Session cocok digunakan untuk menyimpan informasi yang lebih **privat**, seperti user_id dan password, karena disimpan di server. Session juga cocok digunakan untuk “melempar” nilai ke halaman lain. Session memiliki sifat yang lebih secure daripada melempar GET value melalui URL. Session yang akan dibuat pada modul ini belum menggunakan database. Session adalah cara yang digunakan untuk menyimpan informasi (dalam variabel) agar dapat digunakan di **banyak halaman**.

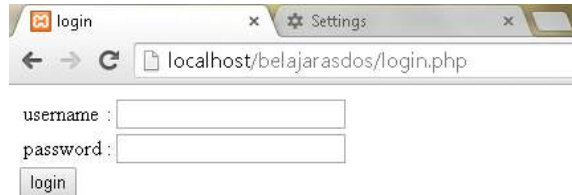
- Sebuah session dimulai dengan `session_start()` function. Dan harus diletakkan di bagian paling atas halaman sebelum ada HTML apapun.

- Variabel session di set menggunakan variabel global PHP yaitu `$_SESSION`
- Session diakhiri/ dihapus semua dengan `session_destroy()` function

D. Latihan Terbimbing

D.1. PHP login

Untuk lebih jelasnya, cobalah demo login dibawah ini. Terdapat sebuah form login seperti berikut.



Buat pada file login.php, dimana pada file ini dibuat form untuk login dan user dapat login apabila username dan passwordnya adalah admin.

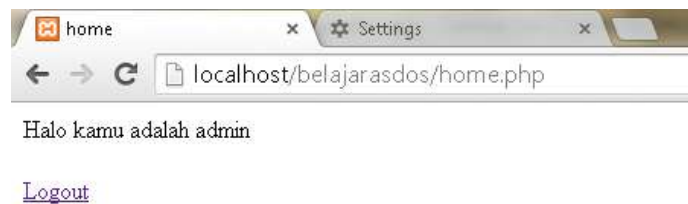
```
<html>
<head>
  <title>login</title>
</head>
<body>
  <div class="login">
    <form method="post" action="login.php" />
      <table>
        <tr>
          <td>username</td>
          <td> : </td>
          <td><input type="text" name="username" ></td>
        </tr>
        <tr>
          <td>password</td>
          <td> : </td>
          <td><input type="password" name="password" ></td>
        </tr>
      </table>
      <input type="submit" name="login" value="login" />
    </form>
  </div>
</body>
</html>
```

```

<?php
if(!empty($_POST['login']))
{
    if($_POST['username'] == "admin" AND $_POST['password'] == "admin")
    {
        session_start();
        $_SESSION['user']='admin';
        header("location:home.php");
    }
    else
        echo "user dan pass salah";
}
?>

```

Jika berhasil login, maka akan menampilkan halaman **home.php**



Buat pada file **home.php**:

```

<?php
session_start();
?>
<html>
<head>
    <title>home</title>
</head>
<body>
    <span>Halo kamu adalah <?php echo $_SESSION['user']; ?></span>
    <br><br>
    <a href=logout.php>Logout</a>
</body>
</html>

```

Dan kemudian pada file **logout.php** :

```

<?php
session_start();
session_destroy();
header("location:login.php");
?>

```

Keterangan:

- Pada login.php, apabila password dan username bukan 'admin', maka akan muncul tulisan password dan username salah tetapi apabila berhasil maka akan menuju ke halaman home.php
- Pada home.php, kita dapat mengetahui informasi mengenai siapa yang login
- Setelah logout maka halaman akan kembali lagi ke login.php