

**MODUL PRATIUM – 14
PEMROGRAMAN BERBASIS WEB
(CCP119)**

Universitas
Esa Unggul

Universitas
Esa Unggul

Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul

**Disusun oleh :
KARTINI, SKom.,MMSI**



Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul

**LABORATORIUM KOMPUTER FAKULTAS ILMU
KOMPUTER**

UNIVERSITAS ESA UNGGUL



Universitas
Esa Unggul



Universitas
Esa Unggul



Universitas
Esa Unggul

MODUL PRATIKUM PBW 14

Cookies, session, upload-file, send mail

A. TUJUAN

1. Praktikan mengetahui kegunaan Cookies, session, upload-file, send mail dalam halaman web
2. Praktikan mengetahui penulisan Cookies, session, upload-file, send mail
3. Praktikan dapat mengaplikasikan Cookies, session, upload-file, send mail dalam tiap halaman web

B. DASAR TEORI

PHP-Email :

1. PHP Email - FORM SEND E-MAIL
2. PHP UPLOAD _ FORM UPLOAD
3. PROSES SIMPAN FILE YG DIUPLOAD
4. Pemrosesan Upload File Lebih Lanjut

Email dengan interface web dan pengiriman otomatis tentang informasi suatu artikel atau situs web dapat dibuat dengan mudah. PHP menyediakan fasilitas ini.

- ▶ Untuk dapat melakukan pengiriman email dengan menggunakan PHP maka harus dipastikan terlebih dahulu bahwa anda mempunyai server yang berfungsi untuk mengirimkan email disebut dengan server SMTP (Simple Mail Transfer Protocol).
- ▶ To send email in a PHP script use the mail function.
Mail(string to, string subject, string message,[,string additional_headers [,string additional_parameters]])

KETERANGAN

string to, = Kepada : Alamat yang dituju.
string subject : Subject/Judul dari pesan yang akan ditampilkan/dimunculkan dalam E-mail penerima
string message = Isi pesan, bila diclick subject/judul pesan, oleh penerima akan muncul ini pesan
[,string additional_headers [,string additional_parameters]] : Bersifat optional, bila ada data lain, bias berupa file, dll

ATAU, Bisa ditulis :

Mail("kepada", "subejk", "isipesan", "datalain")

This sends an email to the email address in to, with subject subject and message message. OR You can also set additional mail headers and parameters to mail :

```
$Result = mail(steve@ispname.com, "Web mail", $_REQUEST["message"]);
```

OR : Can also Writted with attachments

```
$result = mail("steve@chooseanisp.com", "Web mail", $_REQUEST["message"], $headers);
```

Message is retrieved from html form

Additional email headers like the following may be specified.

[,string additional_headers [,string additional_parameters]] ="DataLain" : Bersifat optional, bila ada data lain, bias berupa file, dll

```
$result = mail("steve@chooseanisp.com", "Web mail", $_REQUEST["message"], DataLain);
```

Examp : cc ("carbon copy")
 bcc ("blind carbon copy")

These are set with the additional_headers parameter.

```
$headers .= "cc:" . $_REQUEST["cc"] . "\r\n";
```

```
$headers .= "bcc:" . $_REQUEST["bcc"] . "\r\n";
```

```
$result = mail("steve@chooseanisp.com", "Web mail", $_REQUEST["message"], $headers);
```

Kirim Email dengan SMTP *nix

Pada lingkungan *nix maka server SMTP menggunakan software sendmail. Software yang akan digunakan ini telah didefinisikan pada php

```
Mail(string to, string subject, string message,[,string additional_headers  
[,string additional_parameters]])
```

OR : Can also Writted with attachments

```
Mail("kepada", "subejk", "isipesan", "datalain")
```

Contoh :

```
ini_set("SMTP", "172.16.1.1");  
ini_set("sendmail_form", "$dari");  
mail("$kepada", "$subjek", "$pesan");
```

PRAKTEK I :

Perhatikan script PHP dibawah ini yg bernama : mail_01.php hanya menun jukan kemampuan mengirimkan email dari PHP dengan menggunakan server SMTP pada Windows NT4 Server dan Windows 2000 Server.

Nama script : mail01.php

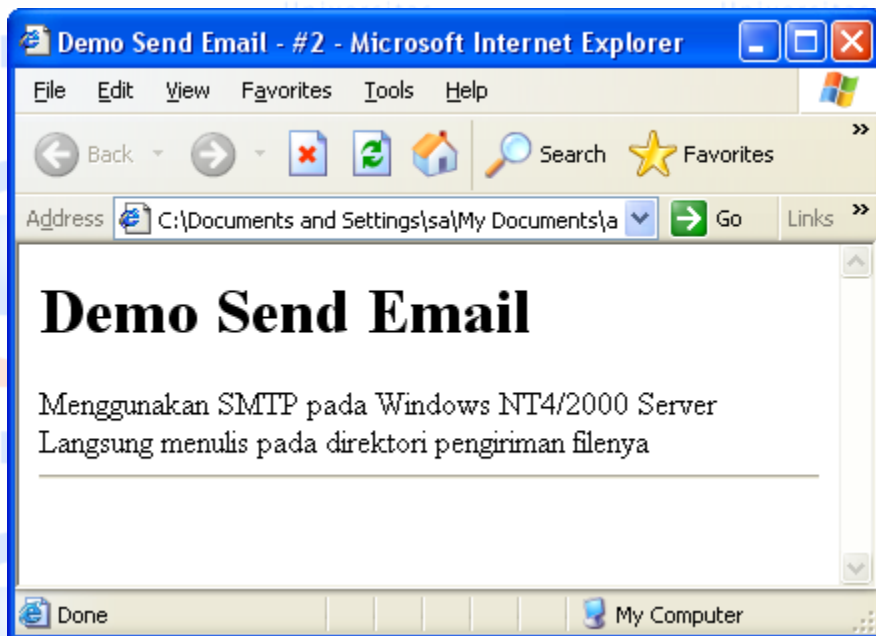
```
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">  
<html>  
<head>  
    <title>Demo Send Email - #2</title>  
</head>  
<body>  
<h1>Demo Send Email</h1>  
Menggunakan SMTP pada Windows NT4/2000 Server  
<br>
```

```

Langsung menulis pada direktori pengiriman filenya
<hr>
<?
$file = fopen("c:\inetput\mailroot\pickup\message.eml", "w");
fwrite( $file, "To:betha@hotmail.com\r\n");
fwrite( $file, "From:Who Am I me@localhost.com\r\n");
fwrite( $file, "Subject: Email testing ... \r\n");
fwrite( $file, "\r\n");
fwrite( $file, "Here is the message\r\n");
fwrite( $file, "\r\n");
fclose( $file );
?>
</html>
<body>

```

Dan lihat hasilnya dibawah ini, sbb :



PRAKTEK FORM SEND E-MAIL

CONTOH : Kirim Email dengan Form Web

Menggabungkan *form isian* dengan *dokumen dan PHP* untuk mengirimkan masukan dari *form isian* dengan menggunakan *mail()*.

Nama script : form_email.html

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head><title>Demo Form untuk Send Email</title></head>
<body>
<h1>Demo Form untuk Send Email</h1>
<hr>
<form action="send_email.php" method="get">

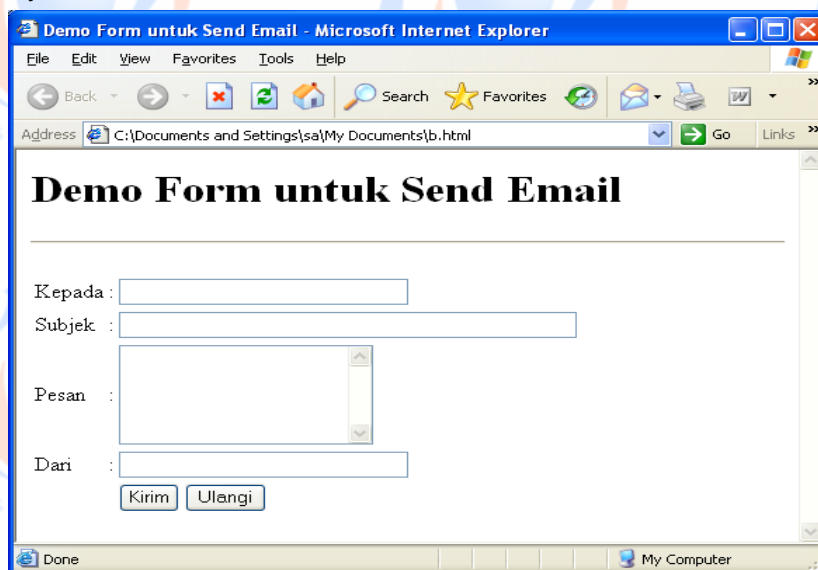
```

```

<table>
<tr><td>Kepada</td>
      <td>:</td>
      <td><input type="text" name="kepada" Size="30"></td>
</tr>
<tr>
      <td>Subjek</td>
      <td>:</td>
      <td><input type="text" name="subjek" Size="50"></td>
</tr>
<tr>
      <td>Pesan</td>
      <td>:</td>
      <td><textarea name="pesan" rows="5" Size="45"></td>
</tr>
<tr>
      <td>Dari</td>
      <td>:</td>
      <td><input type="text" name="dari" Size="30"></td><textarea></td>
</tr>
<tr>
      <td>&nbsp;</td>
      <td></td>
      <td>
      <input type="submit" value="Kirim">
      <input type="reset" value="Ulangi">
      <td>
</tr>
</tr>
</table>
</form></body></html>

```

Dan hasilnya sbb :



Nama script : send_email.php

```

<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0
Transitional//EN">
<html>
<head>
  <title>Demo Send Email</title>
</head>
<body>

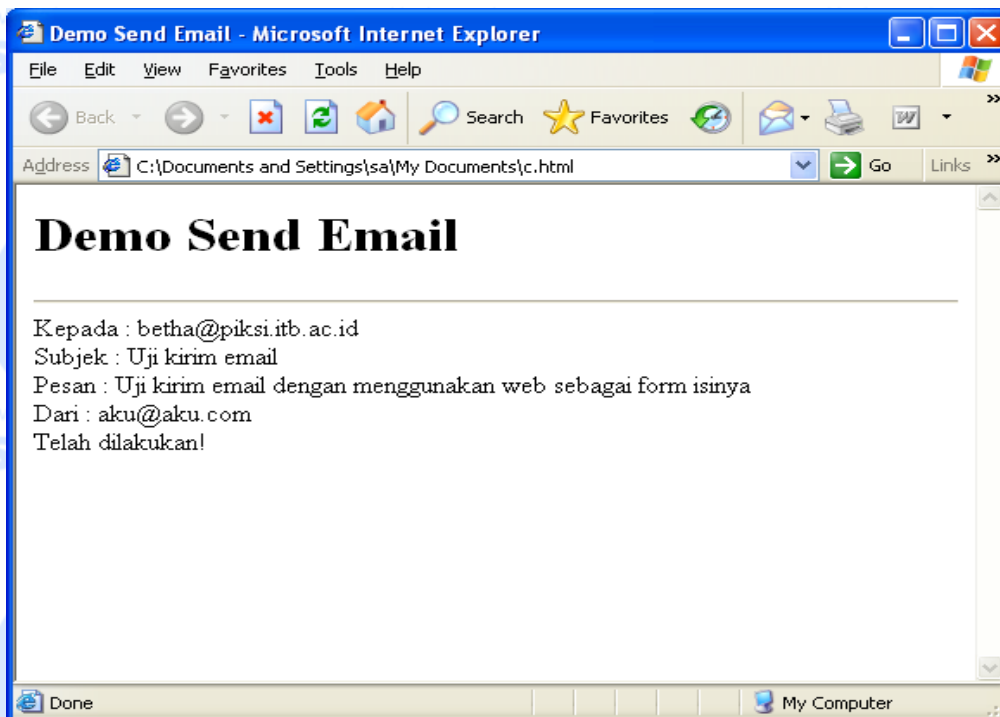
```

```

<h1>Demo Send Email</h1>
<hr>
Kepada : <?echo $kepada;?><br>
Subjek : <?echo $subjek;?><br>
Pesan : <?echo $pesan;?><br>
Dari : <?echo $dari;?><br>
<?
ini_set("SMTP", "172.16.1.1");
ini_set("sendmail_form", "$dari");
mail("$kepada", "$subjek", "$pesan");
?>
Telah dilakukan!
</body>
</html>

```

Dan Tampilan send_email.php



PRAKTEK a : PHP UPLOAD

Elemen-elemen yang dibutuhkan untuk dapat melakukan upload file via browser maka harus disiapkan :

- Sebuah form untuk menyajikan tampilan agar pemakai memilih file yang akan diupload
- File yang akan diupload (dan tentu saja browser yang mendukung fungsi upload)
- Tempat dalam server yang digunakan untuk menyimpan file yang akan diupload
- Script untuk menempatkan file ke tempat yang telah ditentukan di dalam server

Nama script formt_upload.php

```

<html>
<head>

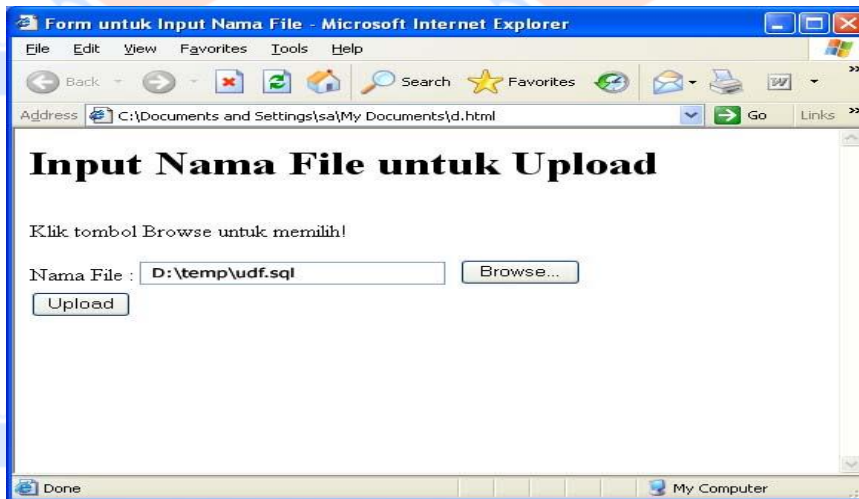
```

```

<title>Form untuk Input Nama File</title>
</head>
<body>
<h1>Input Nama File untuk Upload</h1>
<br> Klik tombol Browse untuk memilih!
<form enctype="multipart/form-data" method="post" action="do_upload.php">
<input type="hidden" name="MAX_FILE_SIZE" Value="10000">
Nama File : <input type="file" name="file1" size="30">
<br>
<input type="submit" value="Upload">
</form>
</body>
</html>

```

Dan hasilnya sbb :



PROSES SIMPAN FILE YG DIUPLOAD 1

Nama script : do_upload.php

```

<html>
<head>
<title></title>
</head>
<body>

```

```

<h1>Simpan file yang diupload!</h1>
<!-- <p>Nama File : <?echo $file1;?></p> -->
<br>
<?
if ($file1!="none"){
    copy("$file1","hasilupload.txt") or die ("No files!"); }
else {
    die ("Tidak ada file yang akan diupload!"); }
?>
</body>
</html>

```

Dan hasilnya : Tampilan : do_upload.php



PROSES SIMPAN FILE YG DIUPLOAD 1

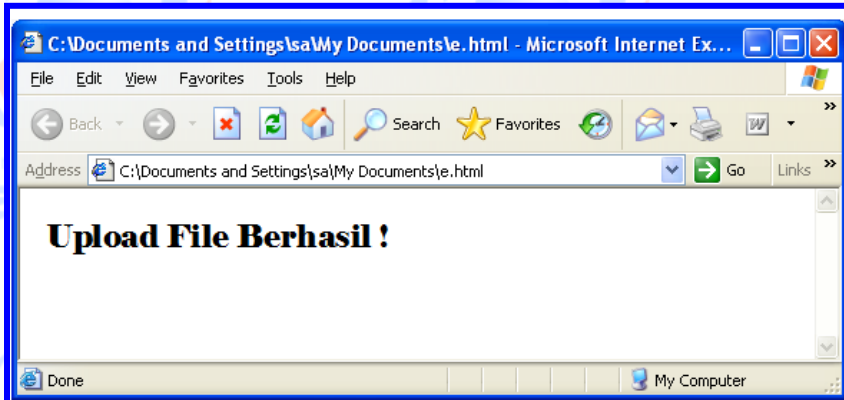
Buatlah Script web untuk menampilkan form Upload File dibawah ini



Klik Tombol Browse untuk memilih file, Bila sudah di pilih file yang akan di upload, seperti terlihat pada tampilan dibawah ini :



Dan click Upload bila berhasil akan tampil output seperti terlihat dibawah ini :



Dan bila gagal, silahkan anda Kreasi sendiri bentuk tampilan nya ?

PRAKTEK 3 :

PHP SESSION dan COOKIES

PHP SESSION

Sesion memungkinkan pelacakan akses pemakai.

Untuk memulai session perintah yang digunakan adalah **session_start()**. Dan untuk mengakhiri session menggunakan perintah **session_destroy()**. Terdapat banyak fungsi yang berhubungan dengan session, yaitu:

Fungsi session_start()

Berfungsi untuk memulai session. Sintaksnya adalah sebagai berikut:

session_start()

Fungsi session_destroy()

Berfungsi untuk mengakhiri session. Sintaksnya adalah sebagai berikut:

session_destroy()

Fungsi session_name()

Digunakan untuk mengambil atau menentukan nama sebuah session. Sintaksnya adalah sebagai berikut:

session_name([nama])

Jika argumen nama tidak disertakan, maka fungsi ini digunakan untuk mengambil nama sebuah session dan jika tidak disertakan digunakan untuk memberi nama pada session.

Fungsi session_module_name()

Digunakan untuk mengambil atau menentukan nama sebuah modul session. Sintaksnya adalah sebagai berikut:

session_module_name([modul])

Jika argumen modul tidak disertakan, fungsi ini digunakan untuk mengambil nama sebuah modul session dan jika tidak disertakan digunakan untuk memberi nama pada modul session.

Fungsi session_save_path()

Digunakan untuk mengambil atau menentukan path dari direktori yang digunakan untuk menyimpan data-data sebuah session. Sintaksnya adalah sebagai berikut:

session_save_path([path])

Fungsi session_id()

Digunakan untuk mengambil atau menentukan identitas sebuah session. Sintaksnya adalah sebagai berikut:

`session_id([id])`

Fungsi `session_register()`

Digunakan untuk mendaftarkan variabel ke dalam sebuah session. Sintaksnya adalah sebagai berikut:

`session_register([namavar1],[namavar2...])`

Argumen `namavar1`, `namavar2` dan seterusnya berupa string yang menampung nama variabel (tanpa tanda \$ didepannya).

Fungsi `session_unregister()`

Digunakan untuk menghilangkan sebuah variabel dari sebuah session, namun session tersebut tetap ada. Sintaksnya adalah sebagai berikut:

`session_unregister(namavar)`

Fungsi `session_unset()`

Digunakan untuk menghilangkan nilai semua variabel yang ada dalam sebuah session. Sintaksnya adalah sebagai berikut:

`session_unset()`

Fungsi `session_is_registered()`

Digunakan untuk memeriksa apakah sebuah variabel telah didaftarkan pada sebuah session. Sintaksnya adalah sebagai berikut:

`session_isregistered(namavar)`

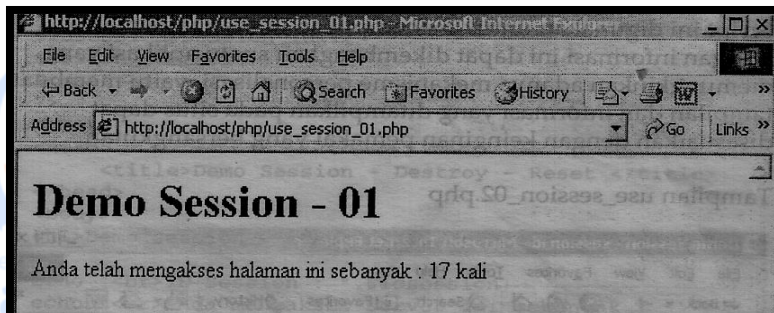
Argumen `namavar` berupa string yang merujuk nama variabel (tanpa tanda \$ didepannya).

CONTOH 1 : Nama script: `use_session_01.php`

```
<?php
session_start()
session_register ("count")
$count++;
?>
<html>
<head><title>Demo Session - 01</title> < head>
<body>
<h1>Demo Session - 01</h1>
<?
echo "Anda telah mengakses halaman ini sebanyak : $count Kali";
?>
</body>
</html>
```

OUTPUT :

Berikut tampilan `use_session_01.php` yang direfresh hingga 17 kali pada saat melakukan uji coba.



Dilakukan refresh hingga 17 kali

CONTOH 2 :

personalisasi. Yaitu :

“Membuat tampilan dan informasi yang ditampilkan pada situs dapat disesuaikan dengan keinginan pemakai yang bersangkutan”.

Nama script : Use_Session_02.php

```
<?php
session_start();
session_register ("count")
$count++;
$idsession=session_id();
?>
<html>
<head><title>Demo Session — session id</title></head>
<body>
<h1>Demo session — session id</h1>
<?
echo "<br>Anda user ke : ".$count;
echo "<br>ID Session anda ".$idsession;
?>
</body> </html>
```

OUTPUT :



Tampilan use_session_02.php

PRAKTEK FUNGSI HAPUS SEMUA ISI VARIABEL YANG ADA DLM SESSION

session_destroy[]

session_destroy() merupakan fungsi yang digunakan utk melakukan reset terhadap semua nilai variabel yg ada dlm suatu session.

Id_session untuk koneksi suatu klien (komputer) akan tetap sampai dengan browser di close.

CONTOH 1 : Nama script use_session_out.php

```
<?php
session_start();
$idsession=session_id();
session_destroy ();
session_register("count")
```

```

$count++;
?>
<html>
<head><title>Demo Session — Destroy — Reset</title>
</head>
<body>
<h1>Demo session — reset nilai counter</h1>
<?
echo "<br>ID Session ".$sidsession;
echo "<br>Anda mengakses server ini sebanyak :".$count;
?>
</body>
</html>

```

OUTPUT :



Tampilan use_session_out.php

CONTOH 2 :

Script berikut ini menunjukkan bahwa variabel \$count yang dihasilkan oleh suatu session_register() dikenal oleh session dan klien pada setiap halaman, jika session_start() selalu dijalankan di awal script.

Nama script : use_session_03.php

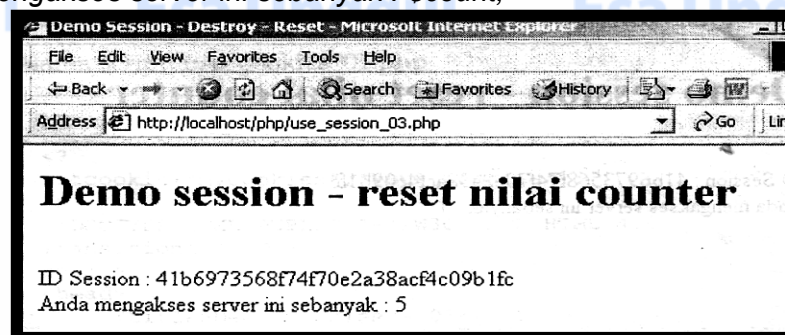
```

<?php
session_start();
$sidsession=session_id();
session_register("count")
$count++;
?>
<html>
<head><title>Demo Session - Destroy - Reset c</title></head>
<body>
<h1>Demo session - reset nilai counter</h1>
<?
echo "<br>ID Session ".$sidsession;
echo "<br>Anda mengakses server ini sebanyak :".$count;
?>
</body>
</html>

```

OUTPUT :

use_session_03.php



PRAKTEK : Contoh Penggunaan

Untuk memahami session, kita akan latihan membuat session dengan aplikasi *login*. Yang perlu dibuat pertama kali adalah file *admin.html*.

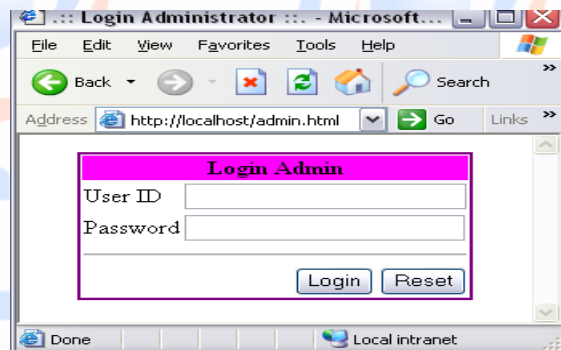
File *admin.html*

```
<html>
<head> <title>... Login Administrator ...</title></head>
<body bgcolor="#FFFFFF">
<form name="form1" action=login.php method=POST>
<table width="62%" border=1 align=center
cellpadding=0 cellspacing=0 bgcolor=#fffff bordercolor=purple>
<td>
<table width="100%" border=0 align=center cellpadding=1 cellspacing=1
bgcolor=#fffff >
<tr bgcolor=magenta>
<td colspan=2><div align=center><strong>Login Admin</strong></div></td>
</tr>
<tr bgcolor=#fffff>
<td width=31%>User ID</td>
<td width=69%><input type=text name=user id=admin size=25
maxlength=25></td>
</tr>
<tr bgcolor=#fffff>
<td width=31%>Password</td>
<td width=69%><input type=password name=password
id=pass size=25 maxlength=25></td>
</tr>
<tr>
<td colspan=2 align=center><hr></td></tr>
<tr><td colspan=2 align=right><input type=submit value="Login">
<input type=reset value="Reset"></td>
</tr>
</table>
</td>
</table>
</form></body></html>
```

Script tersebut jika dijalankan akan tampak seperti gambar disamping
Selanjutnya buat file berikut:

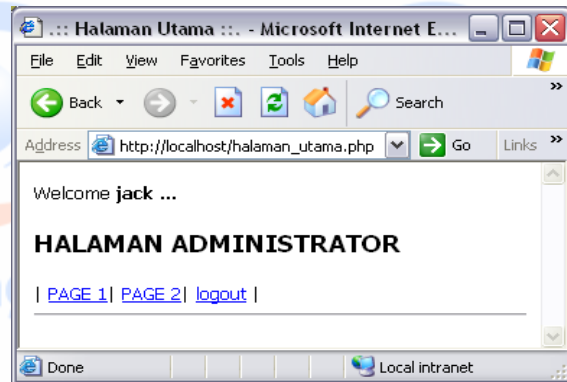
File *login.php*

```
<?
session_start();
if (empty($user))
    echo "Nama User Belum Di Isi";
elseif (empty($password))
    echo "Password Belum Di Isi";
elseif ($user=="jack" &
    $password=="onlyme")
{
    $master=$user;
    session_register("master");
    header("location:halaman_utama.php");
    exit();
}
else echo "Anda Tidak Terdaftar";
?>
```



File halaman_utama.php

```
<?
session_start();
?>
<html>
<head><title>::: Halaman Utama :::</title></head>
<body>
<font face=tahoma size=2>
<?
if(!session_is_registered("master"))
echo "<h3>Akses Ditolak...</h3>";
else
{
echo "Welcome <b>
    $master ... </b><br>";
echo "<h3>HALAMAN ADMINISTRATOR</h3>";
echo "| <a href=hal1.php>PAGE 1</a>";
echo "| <a href=hal2.php>PAGE 2</a>";
echo "| <a href=logout.php>logout</a> |";
echo "<hr>";
}
?>
</body></html>
```



File hal1.php

```
<?
    session_start();
?>
<html>
<head>
    <title>::: Halaman 1 :::</title></head>
<body>
<font face=tahoma size=2>
<?
    if(!session_is_registered
        ("master"))
        echo "<h3>Akses Ditolak...</h3>";
    else
    {
        echo "<font face=tahoma size=2>";
        echo "<h3>PAGE 1</h3>";
        echo "<p align=right> | <a href=logout.php>logout</a> |";
        echo "<hr>";
        echo "Nama User <b><u>$master</b></u>";
    }
?>
</body></html>
```



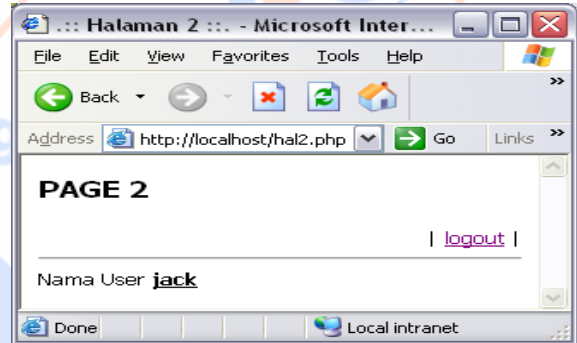
File hal2.php

```
<?
    session_start();
?>
<html>
<head> <title>::: Halaman 2 :::</title></head>
<body>
```

```

<font face=tahoma size=2>
<?
  if(!session_is_registered("master"))
    echo "<h3>Akses Ditolak...</h3>";
  else
  {
echo "<font face=tahoma size=2>";
echo "<h3>PAGE 2</h3>";
echo "<p align=right> | <a href=logout.php>logout</a> |";
echo "<hr>";echo "Nama User <b><u>$master</u></b>";
  }
?>
</body></html>

```



File hal2.php

```

<?
  session_start();
  session_destroy();
  header("location:admin.html");
?>

```

PRAKTEK 4. PHP Cookies

Seperti halnya session maka perintah untuk membuat cookies harus dilakukan pertama kali, sebagai perintah pertama dalam script kita.

Teknik mengirimkan cookies dengan mengikutsertakan fungsi setcookies ("namacookies", "nilaicookies")

Nama cookie akan dikenal sebagai nama variabel secara otomatis apabila browser mendukung cookies.

Life-time atau masa berlaku cookies jika tidak ada informasi tambahan sama seperti halnya session, sampai dengan browser web ditutup (close).

CONTOH 1 : Nama Script : use_sookie_06.php

```

<?
  Set cookie ("mycookies","Kuekecil",time() + 300);
<?
<DOCTYPE HTML PUBLIC "-//DTD HTML 4.0 Transitional//EN">
<html>
</head><title>Demo Cookies – 06</title></head>
<body>
<h1>Democookies — 06</h1>
echo "Cookies anda : $mycookies";
</body></html>

```

CONTOH 1 : Data berupa integer : Nama Script : use_sookie_06.php

```

<?
  Set cookie ("mycookies","Kuekecil",time() + 300);
<?
<DOCTYPE HTML PUBLIC "-//DTD HTML 4.0 Transitional//EN">

```

```

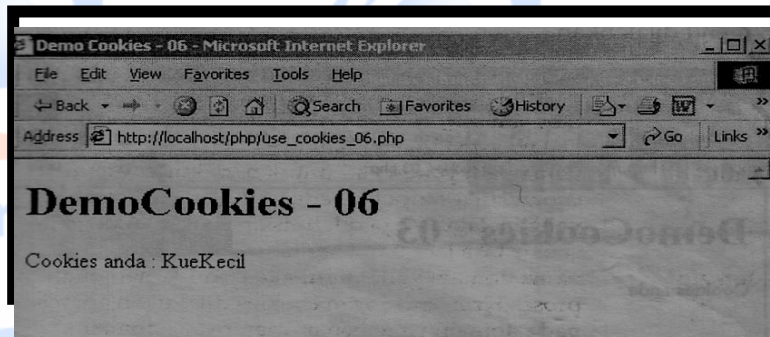
</html>
</head><title>Demo Cookies – 06</title></head>
<body>
<h1>Democookies — 06</h1>
echo "Cookies anda : $mycookies";
</body></html>

```

KETERANGAN

Script diatas memperlihatkan masa berlaku script yang hanya selama 5 menit. Fungsi time() digunakan untuk mendapatkan waktu saat script mengirimkan cookies ditambah dengan jumlah detik masa berlakunya.

OUTPUT 1 :



CONTOH 2 :

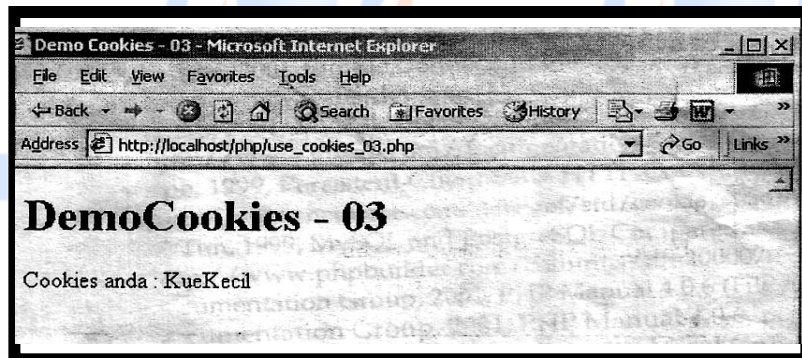
Nama script: use_cookies_03.php

```

<?
setcookie("mycookies", "KueKecil")
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<html>
<head><title>Demo Cookies – 03</title></head>
<body>
<h1>DemoCookies – 03</h1>
<?
echo "Cookies anda : $mycookies";
?>
</body>
</html>

```

OUTPUT 2 :



Cookies untuk Counter

- ▶ Kita dapat menggunakan cookies untuk dijadikan sebagai parameter counter untuk menghitung jumlah pengunjung situs kita. Jumlah pengunjung bisa diidentifikasi,
- ▶ Bisa saja seorang pengunjung melakukan akses dokumen lebih dari satu dokumen pada saat yang bersamaan, karenanya kita harus mencatat siapa saja yang tengah melakukan akses kepada server kita.
- ▶ Cara sederhana dengan menggunakan cookies adalah dengan memberikan sebuah cookies berupa nomor kepada setiap pengunjung. Katakan cookies ini diberi nama "tamu", setiap pengunjung akan diberi nomor unit tamu.
- ▶ Nomor urut akan disimpan kepada browser, sehingga setiap kali seorang pengunjung mengakses situs kita, nilai cookies tamunya diperiksa, apabila belum mempunyai nomor maka pengunjung diberi cookies dengan menggunakan setcookie("tamu", nomor).

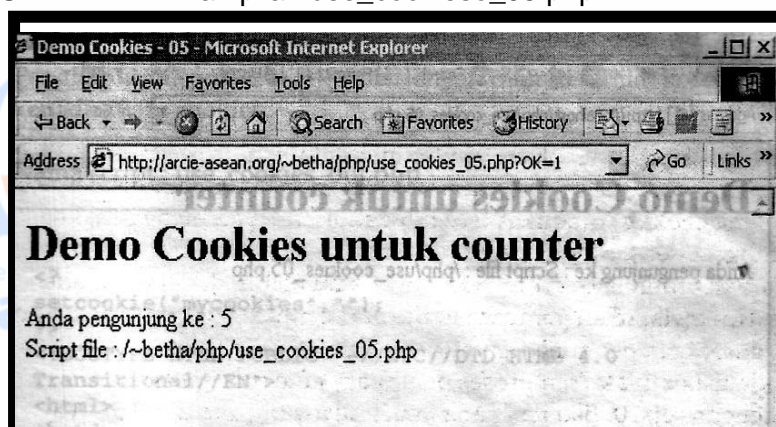
CONTOH : Nama script : use_cookies_05.php

```

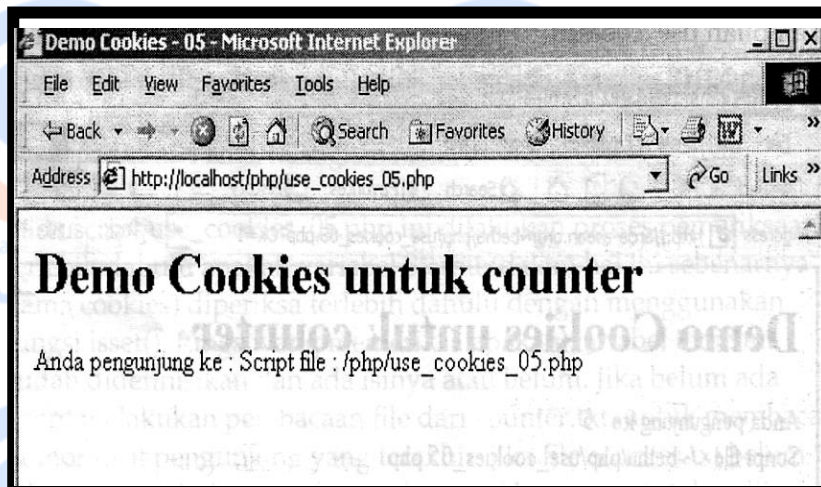
<?
if (! Isset($tamu))
{ $namafile="/tmp/counter.txt"; // tmp/... Nama folder
if (!file_exists($namafile)) {
$fp=fopen($namafile,"w")
$write($fp,"0");
fclose($fp); }
$fp=fopen($namafile,r);
$urut=fread($fp,5);
fclose($fp);
$urut++;
$fp=fopen($namafile,"w");
$write($fp,$urut);
fclose($fp);
setcookie("tamu",$urut);
if (!isset($OK))
{ header("Location: $PHP_SELF.">OK=1");
exit; }
}
?>
<!DOCTYPE HTML PUBLIC "-//W3C//DTD HTML 4.0 Transitional//EN">
<head><html> <title>Demo Cookies – 05</title></head>
<body>
<h1>Demo Cookies untuk counter</h1>
<?
echo "Anda pengunjung ke :$tamu";
echo "Script file : ".$PHP_SELF";
?>
</body></html>

```

OUTPUT : Tampilan use_cookies_05.php

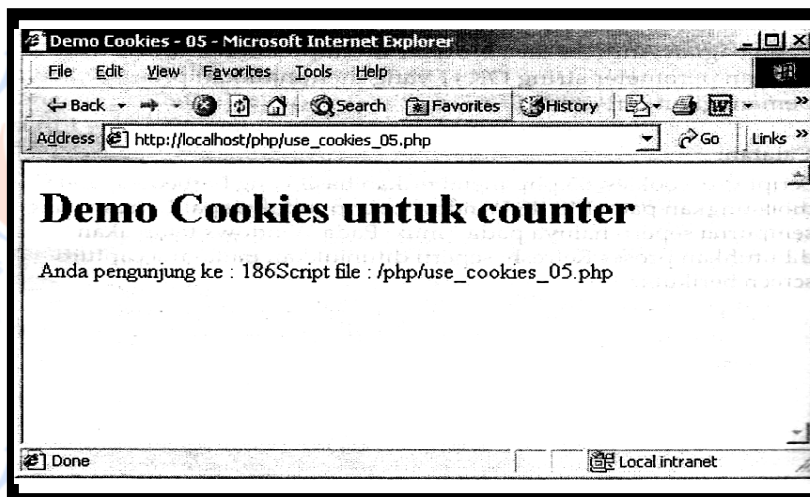


- ▶ Script use_cookies_05.php memberikan hasil yang berbeda apabila dihostingkan pada Linux script tidak dapat di Refresh dgn sempurna. seperti halnya pada MS ITS Windows memberikan hasil yang berbeda seperti terlihat di bwh ini,



Tampilan use_cookieso_05.php (No Refresh)

- ▶ Pada Windows tetap akan dibutuhkan proses Refresh, seperti ditunjukkan pada dua screen berikut:



Tampilan use_cookieso_05.php (bisa refresh)