

Pengembangan Aplikasi Web

Pertemuan Ke-4 (PHP Lanjut 2)

Noor Ifada

`noor.ifada@{trunojoyo.ac.id, if.trunojoyo.ac.id}`

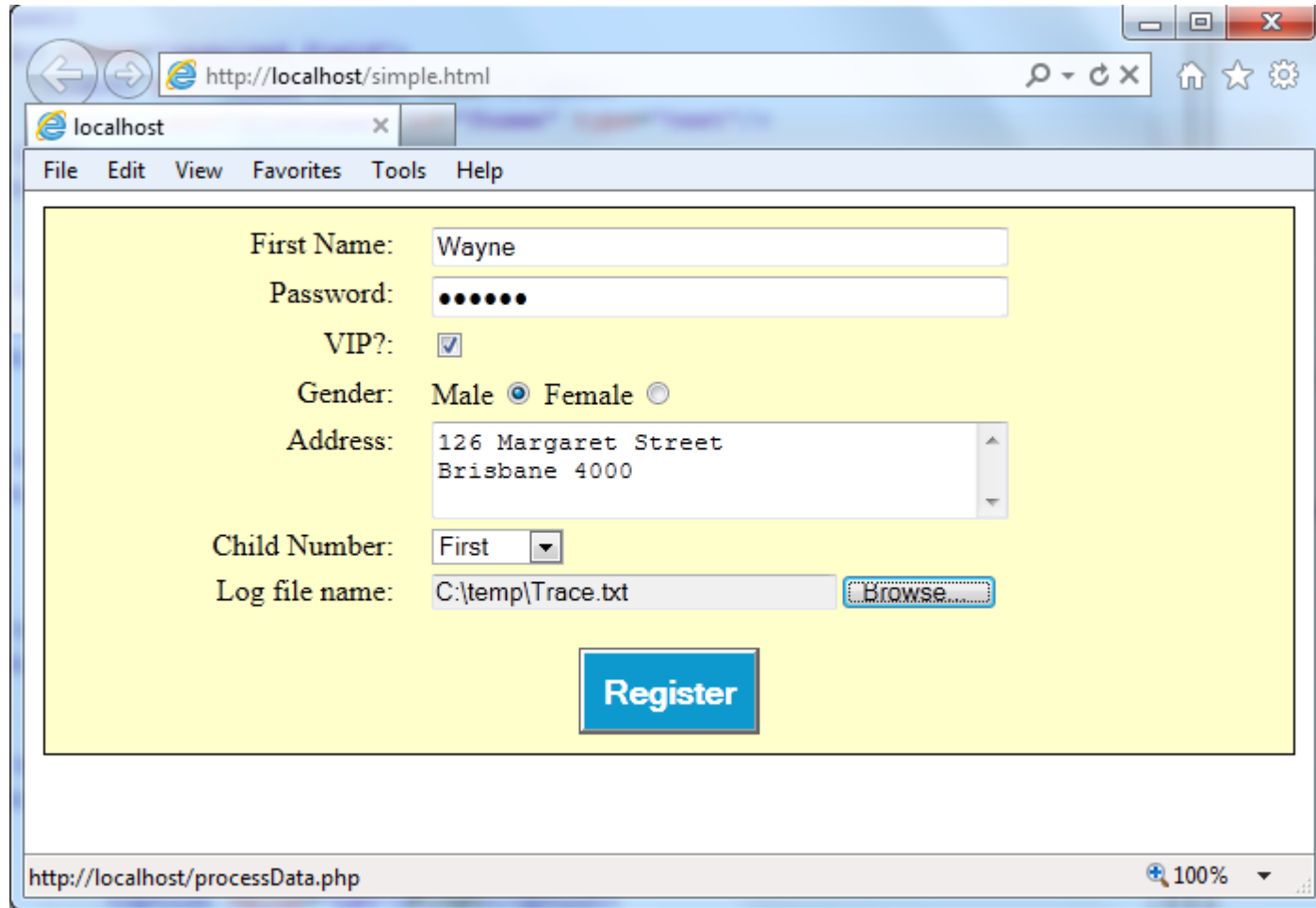
Sub Pokok Bahasan

- Form HTML dan PHP
- Pengiriman *Form*
- Mengakses Data *Form*
- Validasi *Server-side*
- Fungsi (*Function*) untuk Validasi PHP
- Enkripsi dan Integritas Data
- Pemrosesan File
- *Cookie* dan *Session*
- Contoh Aplikasi PHP

Form HTML dan PHP

```
69 <form action="processData.php" method="POST" onsubmit="validate()">
70   <fieldset>
71     <div class="required_field">
72       <label for="fname">First Name:</label>
73       <input name="firstname" id="fname" type="text"/>
74     </div>
75     <div class="required_field">
76       <label for="pass">Password:</label>
77       <input name="passwd" id="pass" type="password"/>
78     </div>
79     <input name="surname" type="hidden" value="Kelly"/>
80     <div class="required_field">
81       <label for="VIP">VIP?:</label>
82       <input name="VIP" id="vip" type="checkbox"/>
83     </div>
84     <div class="required_field">
85       <label>Gender:</label>
86       Male <input type="radio" name="sex" id="male" value="Male"/>
87       Female <input type="radio" name="sex" id="femle" value="Female"/>
88     </div>
89     <div class="required_field">
90       <label for="address">Address:</label>
91       <textarea name="address" id="address" rows="3" cols="40"></textarea>
92     </div>
93     <div class="required_field">
94       <label for="number">Child Number:</label>
95       <select name="child" id="number">
96         <option value="1st">First</option>
97         <option value="2nd">Second</option>
98       </select>
99     </div>
100    <div class="required_field">
101      <label for="log">Log file name:</label>
102      <input name="logfile" id="log" type="file"/>
103    </div>
104    <input name="register" type="submit" value="Register"/>
105  </fieldset>
106 </form>
```

Form HTML dan PHP [2]



The screenshot shows a web browser window with the address bar displaying `http://localhost/simple.html`. The browser's address bar also shows `localhost` in the dropdown menu. The browser's menu bar includes `File`, `Edit`, `View`, `Favorites`, `Tools`, and `Help`. The main content area displays a registration form with the following fields and controls:

- First Name:** A text input field containing the value `Wayne`.
- Password:** A text input field with masked characters (dots).
- VIP?:** A checkbox that is checked.
- Gender:** Two radio buttons labeled `Male` and `Female`, with `Male` selected.
- Address:** A text area containing the text `126 Margaret Street` and `Brisbane 4000`.
- Child Number:** A dropdown menu with the value `First` selected.
- Log file name:** A text input field containing the path `C:\temp\Trace.txt`, followed by a `Browse...` button.
- Register:** A large blue button with the text `Register`.

The status bar at the bottom of the browser window displays `http://localhost/processData.php` and a zoom level of `100%`.

Pengiriman *Form*

Client/Browser

MyForm.html

```
<html>
  <form action="processData.php"
        method="POST">
    ...
  </form>
</html>
```



message

Submit



HTTP Post



Web Server



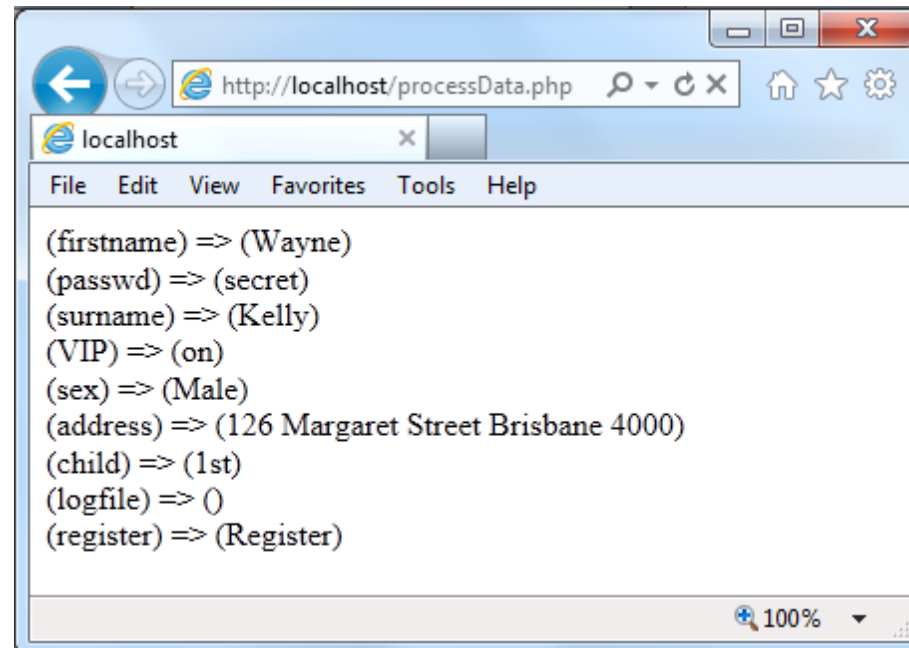
processData.php

```
<?php
...
?>
```

NOTE: pemrosesan server-side berdasarkan atribut `name`, bukan atribut `id` (atribut `id` hanya digunakan pada pemrosesan client-side)

Mengakses Data *Form*

```
processData.php
1 <html>
2     <?php
3         foreach ($_POST as $key => $value)
4             echo "($key) => ($value)<br/>";
5
6         $firstname = $_POST['firstname'];
7         // now do something with $firstname ...
8     ?>
9 </html>
```



Validasi Server-Side

- Jangan hanya mengandalkan validasi *client-side*
- Karena:
 1. *Client* mungkin tidak mengaktifkan JavaScript pada *web browser* yang digunakan
 2. Hacker dapat menuliskan program untuk mengirimkan pesan HTTP request ke halaman PHP

Validasi Server-Side [2]

```
basicValidation.php
4  <?php
5      require_once 'validate.inc';
6
7      $errors = array();
8      validatePattern($errors, $_POST, 'name', '/^[a-zA-Z]+$/',);
9      validatePattern($errors, $_POST, 'phone', '/^((\d{4}[ ]?\d{4})|(04\d{2}[ ]?\d{3}[ ]?\d{3}))$/');
10     validateDate($errors, $_POST, 'dob');
11     // ...
12
13     if (count($errors) == 0)
14     {
15         // Data validates so continue with processing
16         // ...
17     }
18     else
19     {
20         // Data doesn't validate, so display error message
21         // ...
22     }
23  ?>
```

```
11  function validatePattern(&$errors, $field_list, $field_name, $pattern)
12  {
13      if (!isset($field_list[$field_name]) || $field_list[$field_name] == '')
14          $errors[$field_name] = 'Required';
15      else if (!preg_match($pattern, $field_list[$field_name]))
16          $errors[$field_name] = 'Invalid';
17  }
```


Fungsi (*Function*) untuk Validasi PHP

- Regular expressions: `pregmatch`
- string functions: `trim`, `strtolower`, `strtoupper`
- Filter functions:
 - `filter_var`, `filter_input`
 - `FILTER_VALIDATE_EMAIL`,
 - `FILTER_VALIDATE_URL`,
 - `FILTER_VALIDATE_FLOAT`
 - `FILTER_VALIDATE_IP`,
 - ...
- Type testing functions: `is_float`, `is_int`, `is_numeric`, `is_string`
- Date functions: `checkdate`

Enkripsi dan Integritas Data

- Enkripsi data diperlukan dalam aplikasi yang berhubungan dengan *password*. Data *user* beserta *password*nya harus disimpan sehingga yang tidak mungkin dibaca oleh orang lain bahkan oleh seorang administrator sekalipun
- Enkripsi merupakan proses pengaburan data password semula ke dalam bentuk lain yang tidak mudah ditebak, namun dapat dikembalikan lagi ke bentuk semula jika diperlukan
- Karena alasan keamananlah, maka enkripsi banyak digunakan pada proses autentikasi di dalam basisdata

Enkripsi dan Integritas Data [2]

Beberapa fungsi untuk enkripsi data:

- Fungsi **password('kata_yang_dienkripsi')**
 - memperhitungkan bahwa sebuah *password* hasil enkripsi berasal dari sebuah *plain text string*
 - biasanya dipakai pada pemasukan data ke *database* MySQL. Hasil tersebut harus digunakan sebagai *password* yang disimpan di dalam 'user' *grant table*
- Fungsi **crypt('kata_yang_dienkripsi','password_pengenkripsi')**
 - merupakan perintah enkripsi standar pada UNIX yang memanfaatkan *password* perng-enkripsi yang disebut 'salt' yang terdiri dari dua buah karakter
- Fungsi **md5('kata_yang_dienkripsi')**
 - menghasilkan enkripsi berupa deretan *character* dengan jumlah digit 32
- Fungsi **crc32(string)**
 - untuk menghitung nilai polinomial *crc32* terhadap suatu *string*
 - biasanya digunakan untuk memvalidasi integritas data yang dikirim dari suatu lokasi ke lokasi lain menggunakan piranti komunikasi

Pemrosesan File

- *File* merupakan tempat untuk menyimpan data sekunder secara permanen di dalam suatu media penyimpan (misal *hardisk*, *flashdisk*)
- Secara umum format file berupa: teks atau *binary*
- Tiga tahap pengaksesan dalam skema dasar pengaksesan file:
 1. Membuka file
 2. Memproses file/manipulasi file
 3. Menutup file

Pemrosesan File [2]

Beberapa fungsi dalam pemrosesan file:

- Fungsi **fopen()**
 - untuk membuka/mengakses file
 - fungsi ini dapat mengakses file dari sistem file, atau melalui HTTP atau FTP di internet.
- Fungsi **fclose()**
 - untuk menutup file/berkas digunakan
- Fungsi **fgets()**
 - untuk melihat isi dari file digunakan
- Fungsi **fputs()**
 - untuk menginputkan data ke dalam file
- Fungsi **feof()**
 - untuk menentukan akhir dari sebuah file. Jika sudah pada akhir file, fungsi ini akan bernilai *True*

Pemrosesan File [3]

- Fungsi **copy()**
 - untuk mengkopi suatu file
 - file yang akan dikopi dapat diambil melalui masukan data *form*
- Fungsi **unlink()**
 - untuk menghapus file
 - Keluarannya adalah TRUE jika file berhasil dihapus dan FALSE jika terjadi kegagalan dalam proses penghapusan
- Fungsi **rename()**
 - untuk mengganti nama file
 - Keluarannya adalah TRUE jika file berhasil diubah/diganti dan FALSE jika terjadi kegagalan dalam proses pengubahan

Pemrosesan File [4]

- Fungsi **file_exists()**
 - untuk memeriksa kondisi suatu file, apakah file tersebut ada atau tidak
 - Keluarannya adalah TRUE jika file ada dan FALSE jika file tidak ada
- Fungsi **filesize()**
 - untuk mengetahui besar ukuran suatu file digunakan
 - Fungsi ini mempunyai hasil bertipe *integer*, yang menyatakan ukuran file dalam satuan byte

Cookie dan Session

- Untuk mencatat jejak (*track*) tertentu yang dilakukan oleh pengunjung web site (jumlah kunjungan, nama dan kunjungan terakhir). Mesin *client* atau *web browser* menyimpan informasi tersebut dan mengirimnya ke *web server* pada saat terjadi *request*
 - *Cookie*:
 - disimpan di dalam *harddisk client*
 - mungkin tidak dapat berjalan jika di-blok oleh pengguna dengan mengubah *setting security web browser*
 - secara umum, *cookie* disimpan bersama dengan *username* dan *password* (*password* dapat di-enkripsi sebelum disimpan demi keamanan data) sehingga di dalam *cookie* akan tersimpan *username* dan *password* yang telah terenkripsi
 - *Session*:
 - disimpan di *server*
 - menyerupai *token*, di-*generate* saat otentikasi dilakukan
 - aktif selama *browser* yang mengakses halaman tercatat masih terbuka

Contoh Aplikasi PHP

- Membuat *Counter*
- Membuat *Guest Book* (Buku Tamu)

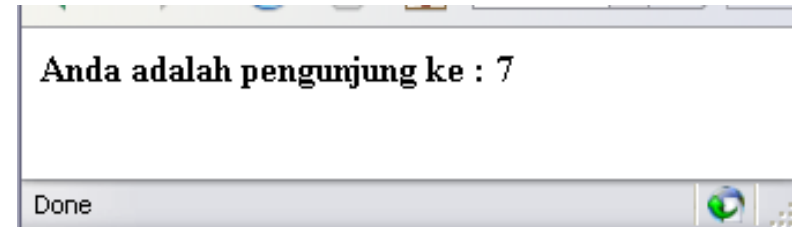
Membuat Counter

- Fungsi *counter* di sini adalah untuk menghitung berapa kali suatu halaman situs web telah dibuka oleh pengunjung.
- Langkah-langkah:
 1. Membuat file teks (**counter.txt**) agar setiap kali suatu halaman *web* ditampilkan maka terlebih dulu isi file tersebut akan dibaca nilainya dan diperbaharui (ditambahkan nilainya dengan 1). Nilai *counter* diinisialisasi bernilai 0 (nol)
 2. Membuat *file* php (**counter.php**) untuk:
 - membaca nilai di dalam file teks
 - memperbaharui nilai *counter*
 - menyimpan nilai yang baru di file
 - menampilkan nilainya di layar *browser*

Membuat Counter [2]

counter.txt

0



counter.php

```
<?php
    $filecounter="counter.txt";
    $open=fopen($filecounter,'r+');
    $counter=fread($open,filesize($filecounter));
    fclose($open);
    $counter++;
    $write=fopen($filecounter,'w');
    fputs($write,$counter);
    fclose($write);
    echo "<b> Anda adalah pengunjung ke :
    $counter</b>";
?>
```

Membuat *Guest Book* (Buku Tamu)

- Salah satu bagian penting dari sebuah website adalah fasilitas pengisian *guest book* (buku tamu) untuk mengetahui siapa saja pengunjung website dan juga mendapatkan komentar dari pengunjung tersebut. Di dalam contoh berikut ini, data isian *guest book* akan dikirimkan ke suatu alamat *email* tertentu.
- Langkah-langkah:
 1. Membuat *file* php (**guestbook_1.php**) untuk membuat *form* pengisian *guest book*. Data yang dimasukkan adalah nama, alamat email, alamat, dan pesan
 2. Membuat *file* php (**guestbook_2.php**) untuk menerima masukan data dari *form* di atas yang selanjutnya akan dikirimkan ke alamat *email* tertentu

Membuat *Guest Book* (Buku Tamu) [2]

guestbook_1.php

```
<table>
  <caption align="top"> <b>GUEST BOOK</b>
</caption>
  <tr> <td colspan="2"> <hr size=1 width=100% >
</hr> </td> </tr>
  <form action="guestbook_2.php" method=POST>
    <tr> <td> Name : </td> <td>
      <input type=text name=nama size=30> </td>
    </tr>
    <tr> <td> E-Mail : </td> <td>
      <input type=text name=email size=30> </td>
    </tr>
    <tr> <td> Address : </td> <td>
      <input type=text name=alamat size=30> </td>
    </tr>
    <tr> <td valign ="top"> Message : </td> <td>
      <textarea name=pesan rows=9 cols=40>
</textarea> </td> </tr>
    <tr> <td align="center" colspan="2">
      <input type=submit value="Submit"> <input
type=reset value="Reset"> </td> </tr>
  </form>
</table>
```

GUEST BOOK

Name : Arianti Wega

E-Mail : arianti.wega@yahoo.com

Address : Jl. Dharmawangsa I/10 Surabaya

Message : Hello...this is my first visit to your website.

Submit Reset

Done

Membuat *Guest Book* (Buku Tamu) [3]

guestbook_2.php

```
<?php
    $to="admin-myweb@yahoo.com";
    $subject="Guest Book";
    $from="From: $nama <$email>";
    $content="";
    $content.="Guest Book\n";
    $content.="Name : $nama\n";
    $content.="E-Mail : $email\n";
    $content.="Address : $alamat\n";
    $content.="Message: $pesan\n";
    mail($to,$subject,$content,$from);
    echo("Thank you for your visit:\n");
?>
```

