

PRAKTIKUM 6

FILE UPLOAD, COOKIE, SESSION DAN AJAX

1. Tujuan Praktikum

Setelah mengikuti praktikum ini mahasiswa diharapkan dapat

- 1) Membuat dan memproses form untuk upload file
- 2) Menggunakan cookies di PHP
- 3) Menggunakan session di PHP, contohnya untuk login dan shopping cart
- 4) Menggunakan Ajax untuk melakukan request ke server dengan sintaks javascript secara langsung maupun dengan sintaks jQuery.

2. Tools

Tools yang dibutuhkan untuk melakukan praktikum ini adalah Apache web server dan PHP, text editor.

3. Langkah-Langkah Praktikum

3.1. Praktikum 1 : Form Upload

Form untuk upload file

```
<!--File      : form_upload.html
Deskripsi    : menampilkan form upload
-->
<html>
<head>
<title>Administration - upload new files</title>
</head>
<body>
<h1>Upload new news files</h1>
<form action="upload.php" method="post" enctype="multipart/form-data">
  <div>
    File allowed : jpg, jpeg, png or gif<br /><br />
    <input type="hidden" name="MAX_FILE_SIZE" value="1000000" />
    <label for="userfile">Upload a file:</label>
    <input type="file" name="userfile" id="userfile"/>
    <input type="submit" value="Send File"/>
  </div>
</form>
</body>
</html>
```

Pemrosesan upload file

```
<!--File      : upload.php
Deskripsi    : memproses upload file
-->
<html>
<head>
<title>Uploading...</title>
```

```

</head>
<body>
<h1>Uploading file...</h1>
<?php
if ($_FILES['userfile']['error'] > 0)
{
    echo 'Problem: ';
    switch ($_FILES['userfile']['error'])
    {
        case 1: echo 'File exceeded upload_max_filesize';
                break;
        case 2: echo 'File exceeded max_file_size';
                break;
        case 3: echo 'File only partially uploaded';
                break;
        case 4: echo 'No file uploaded';
                break;
        case 6: echo 'Cannot upload file: No temp directory
specified';
                break;
        case 7: echo 'Upload failed: Cannot write to disk';
                break;
    }
    exit;
}

$target_dir = "uploads/";
$target_file = $target_dir . basename($_FILES['userfile']['name']);
$upload_ok = 1;
$file_type = pathinfo($target_file,PATHINFO_EXTENSION);

// Check if file already exists
if (file_exists($target_file)) {
    echo "Sorry, file already exists.<br />";
    $upload_ok = 0;
}
// Check file size if you not use hidden input 'MAX_FILE_SIZE'
/*if ($_FILES['userfile']['size'] > 1000000) {
    echo "Sorry, your file is too large.<br />";

    $upload_ok = 0;
}*/
// Allow certain file formats
$allowed_type = array("jpg", "png", "jpeg", "gif");
if(!in_array($file_type, $allowed_type)) {
    echo "Sorry, only JPG, JPEG, PNG & GIF files are allowed.";
    $upload_ok = 0;
}
// Does the file have the right MIME type?
/*if ($_FILES['userfile']['type'] != 'text/plain'){
    echo 'Problem: file is not plain text';
    $uploadOk = 0;
}*/
// put the file where we'd like it
if ($upload_ok != 0){
    if (is_uploaded_file($_FILES['userfile']['tmp_name'])){
        if (!move_uploaded_file($_FILES['userfile']['tmp_name'],
$target_file)){
            echo 'Problem: Could not move file to destination
directory';
        }else{
            echo 'File uploaded successfully<br /><br />';
            echo 'Filename =
'.basename($_FILES['userfile']['name']).'<br />';
            echo 'Size = '.$_FILES['userfile']['size'].' byte';
        }
    }else{
        echo 'Problem: Possible file upload attack. Filename: ';
    }
}

```

```
        echo $_FILES['userfile']['name'];
    }
}
?>
</body>
</html>
```

3.2. Praktikum 2 : Cookies

Mengeset cookie

```
<?php
//File           : set_cookie.php
//Deskripsi : untuk mengeset sebuah cookie

//Remember that setcookie must come before any other line that generates
output
setcookie("username","michele");
echo 'Cookie created.';
?>
```

Mengakses cookie

```
<?php
//File           : get_cookie.php
//Deskripsi : untuk mengakses nilai cookie

if (!isset($_COOKIE['username'])) {
    echo ("Oops, the cookie isn't set!");
}
else {
    echo ("The stored username is ". $_COOKIE['username'] . ".");
}
?>
```

Menghapus cookie

```
<?php
//File           : destroy_cookie.php
//Deskripsi : untuk menghapus cookie

//Remember that setcookie must come before any other line that generates
output
setcookie("username","", time() - 10 );
echo "Cookie has been destroyed.";
?>
```

Jalankan file set_cookie.php → get_cookie.php → destroy_cookie.php → get_cookie.php secara berurutan. Apa kesimpulan Anda?

3.3. Praktikum 3 : Session

```
<?php
//File           : set_session.php
//Deskripsi : untuk mengeset session

//Remember that session_start must come before any other line that
generates output
session_start();
$_SESSION['username'] = 'adam';
```

```
$_SESSION['level'] = 'admin';
echo 'Session has been set<br />';
?>
```

Mengakses session

```
<?php
//File      : get_session.php
//Deskripsi : untuk mengakses nilai session

session_start();
if (isset($_SESSION['username']) && isset($_SESSION['level'])) {
    echo 'Username : ' . $_SESSION['username'] . '<br />';
    echo 'Level : ' . $_SESSION['level'] . '<br />';
} else {
    echo 'Session has not been set.<br />';
}
?>
```

Menghapus session

```
<?php
//File      : destroy_session.php
//Deskripsi : untuk menghapus session

session_start();
unset($_SESSION['username']);
unset($_SESSION['level']);
session_destroy();
echo "sessions has been destroyed.";
?>
```

Jalankan file set_session.php → get_session.php → destroy_session.php → get_session.php.

Apa kesimpulan Anda?

Contoh session untuk shopping cart

Edit file view_books.php pada praktikum kedua, tambahkan space untuk ‘Add to Cart’

```
echo '<td><a href="show_cart.php?isbn='.$row['isbn'].'">Add to
Cart</a></td>';
```

sehingga tampilannya seperti berikut:

ISBN	Author	Title	Price	Option
0-672-31697-8	Michael Morgan	Java 2 for Professional Developers	34.99	Add to Cart
0-672-31745-1	Thomas Down	Installing Debian GNU/Linux	24.99	Add to Cart
0-672-31509-2	Pruitt, et al.	Teach Yourself GIMP in 24 Hours	24.99	Add to Cart
0-672-31769-9	Thomas Schenk	Caldera OpenLinux System Administration Unleashed	49.99	Add to Cart

```
<?php
//File      : show_cart.php
//Deskripsi : untuk memproses dan menampilkan shopping cart
session_start();
$isbn = $_GET['isbn'];
```

```

if($isbn){
    if (!isset($_SESSION['cart'])){
        $_SESSION['cart'] = array();
        $_SESSION['total_item'] = 0;
        $_SESSION['total_price'] = 0;
    }

    if (isset($_SESSION['cart'][$isbn])){
        $_SESSION['cart'][$isbn]++;
    }else{
        $_SESSION['cart'][$isbn]++;
    }
    $_SESSION['total_items'] = calculate_items($_SESSION['cart']);
    $_SESSION['total_price'] = calculate_price($_SESSION['cart']);

    display_cart($_SESSION['cart']);
    echo '<br /><br /><a href="view_books.php">Continue Shopping</a>';
}

```

Variabel `$_SESSION['cart']` digunakan untuk menyimpan data shopping cart, isi dari `$_SESSION['cart']` adalah sebuah array asosiatif dengan indeks array menyatakan isbn buku yang dibeli dan nilainya menyatakan jumlah buku yang dibeli, sebagai contoh:

```
$_SESSION['cart']['0-672-31697-8'] = 2
```

```
$_SESSION['cart']['0-672-31745-1'] = 1
```

Lengkapi kode program untuk fungsi **calculate_items(\$cart)** untuk menghitung jumlah item yang ada di shopping cart, **calculate_price(\$cart)** untuk menghitung total harga item yang ada di shopping cart, dan **display_cart(\$cart)** untuk menampilkan isi shopping cart.

Latihan Session

Buatlah sebuah form login dengan masukan username dan password. Sebelumnya edit tabel customer, tambahkan field username dan password. Username berupa email.

Buatlah file untuk login dan logout, kemudian tambahkan kode program pada file `view_books.php` sehingga hanya dapat dibuka oleh customer yang sudah login.

3.4. Praktikum 4 : Ajax dan PHP - Membuat objek XMLHttpRequest

Fungsi berikut digunakan untuk membuat objek `XMLHttpRequest` sebagai core Ajax. Buatlah fungsi tersebut dalam sebuah file dan simpan di folder `ajax/js` dengan nama `ajax.js`.

```

function getXMLHttpRequest() {
    var req = false;
    try {
        /* for Firefox */
        req = new XMLHttpRequest();
    } catch (err) {
        try {
            /* for some versions of IE */
            req = new ActiveXObject("Msxml2.XMLHTTP");
        } catch (err) {

```

```

        try {
            /* for some other versions of IE */
            req = new ActiveXObject("Microsoft.XMLHTTP");
        } catch (err) {
            req = false;
        }
    }
    return req;
}

```

Perhatikan bahwa objek `XMLHttpRequest` didefinisikan dengan cara yang berbeda pada browser IE dan browser non IE.

3.5. Praktikum 5 : Ajax dan PHP – Method GET

Bagian ini bertujuan untuk melakukan request melalui Ajax dengan method GET. Sebagai contoh, file berikut ini digunakan untuk menerima input dari user, selanjutnya input tersebut akan ditampilkan pada halaman yang sama menggunakan Ajax. Event yang men-trigger pemanggilan fungsi javascript adalah `onClick` pada button `Submit`. Selanjutnya, elemen di halaman web yang akan diisi respon dari server adalah `<div id="user_input"></div>`.

Buatlah file berikut dan simpan di dalam folder `ajax`. File tersebut akan menggunakan file `ajax.js` yang telah dibuat sebelumnya.

```

<!--
File : user_form.html
Deskripsi : menampilkan isi input form dengan ajax
-->
<!DOCTYPE HTML>
<html>
<head>
    <script src="js/ajax.js" type="text/javascript"></script>
</head>
<body>
<h2>User Input</h2>
<form>
<table>
    <tr>
        <td valign="top">Name</td>
        <td valign="top">:</td>
        <td valign="top"><input type="text" name="name" id="name"
size="30" maxlength="50"></td>
    </tr>
    <tr>
        <td valign="top">Address</td>
        <td valign="top">:</td>
        <td valign="top"><textarea name="address" id="address"
rows="5" cols="30"></textarea></td>
    </tr>
    <tr>
        <td valign="top">City</td>
        <td valign="top">:</td>
        <td valign="top"><select name="city" id="city">
            <option value="none">--Select a city--</option>
            <option value="Airport West">Airport West</option>
            <option value="Box Hill">Box Hill</option>
            <option value="Yarraville">Yarraville</option>
        </select>
    </td>

```

```
        </tr>
        <tr>
            <td valign="top" colspan="3"><br />
                <input type="button" id="submit" value="Submit"
onClick=" javascript:view_input_get(); ">
            </td>
        </tr>
    </table>
</form>
<div id="user_input"></div>
</body>
</html>
```

Berikut ini adalah fungsi javascript `view_input_get` yang digunakan untuk melakukan request di server dengan Ajax menggunakan method `GET`. Buatlah fungsi tersebut di dalam file `js/ajax.js`. Fungsi tersebut melakukan request ke file `process_form_get.php`.

```

function view_input_get(){
    var xmlhttp = getXMLHttpRequest();
    //get input value
    var name = encodeURIComponent(document.getElementById('name').value);
    var address = encodeURIComponent(document.getElementById('address').value);
    var city = encodeURIComponent(document.getElementById('city').value);
    //set url and inner
    var url = "process_form_get.php?name=" + name + "&address=" +
address + "&city=" + city;
    var inner = "user_input";
    //open request
    xmlhttp.open('GET', url, true);
    xmlhttp.onreadystatechange = function() {
        document.getElementById(inner).innerHTML = '';
        if ((xmlhttp.readyState == 4) && (xmlhttp.status == 200)){
            document.getElementById(inner).innerHTML =
xmlhttp.responseText;
        }
        return false;
    }
    xmlhttp.send(null);
}

```

Berikut ini adalah isi dari file `process_form_get.php`.

```

<?php
echo "<h2>Your Input:</h2>";
echo 'Name = ' . $_GET['name'] . '<br />';
echo 'Address = ' . $_GET['address'] . '<br />';
echo 'City = ' . $_GET['city'] . '<br />';
?>

```

3.6. Praktikum 6 : Ajax dan PHP – Method POST

Bagian ini bertujuan untuk melakukan request melalui Ajax dengan method POST. Ubahlah baris program pada file `user_form.html`

```

<input type="button" id="submit" value="Submit" onClick="
javascript:view_input_get(); ">

```

menjadi

```

<input type="button" id="submit" value="Submit" onClick="
javascript:view_input_post(); ">

```

Tambahkan fungsi berikut pada file `js/ajax.js`. Fungsi tersebut digunakan untuk melakukan request ke server melalui Ajax dengan method POST. File yang di-request adalah `process_form_post.php`.

```

function view_input_post(){
    var xmlhttp = getXMLHttpRequest();
    //get input value
    var name = encodeURIComponent(document.getElementById('name').value);
    var address = encodeURIComponent(document.getElementById('address').value);
    var city = encodeURIComponent(document.getElementById('city').value);
    //set url and inner

```



```

var inner = "user_input";
var url = "process_form_post.php";
var params = "name=" + name + "&address=" + address + "&city=" +
city;
xmlhttp.open("POST", url, true);
xmlhttp.setRequestHeader("Content-type", "application/x-www-form-
urlencoded");

xmlhttp.onreadystatechange = function() {
    document.getElementById(inner).innerHTML = '';
    if ((xmlhttp.readyState == 4) && (xmlhttp.status == 200)){
        document.getElementById(inner).innerHTML =
xmlhttp.responseText;
    }
    return false;
}
xmlhttp.send(params);
}

```

Berikut ini adalah kode program untuk file `process_form_post.php`.

```

<?php
echo "<h2>Your Input:</h2>";
echo 'Name = '.$_POST['name'].'<br />';
echo 'Address = '.$_POST['address'].'<br />';
echo 'City = '.$_POST['city'].'<br />';
?>

```

Amati perbedaan kode program pada fungsi `view_input_get` dan `view_input_post`. Pada file `js/ajax.js`. Apa perbedaannya saat dilakukan request dengan metode `GET` dan `POST`?

3.7. Praktikum 7 : Ajax – XML

Bagian ini bertujuan untuk menerima respon dari server dalam format XML.

Berikut ini adalah contoh halaman XML yang akan di-request. Tulislah kode XML berikut dan simpan dengan nama `note.xml`.

```

<?xml version="1.0" encoding="UTF-8"?>
<!-- Edited by XMLSpy -->
<note>
    <to>Tove</to>
    <from>Jani</from>
    <heading>Reminder</heading>
    <body>Don't forget me this weekend!</body>
</note>

```

Berikut ini adalah file yang digunakan untuk melakukan request dokumen `note.xml` dan membaca responnya.

Tulislah kode program berikut dan simpan dengan nama `ajax_xml.html`.

```

<!DOCTYPE html>
<html>
<head>
<script src="js/ajax.js" type="text/javascript"></script>
<script>
function loadXMLDoc(url){
    var xmlhttp = getXMLHttpRequest();

```

```

        var txt,x,xx,i;
        xmlhttp.open("GET",url,true);
        xmlhttp.onreadystatechange=function() {
            if (xmlhttp.readyState==4 && xmlhttp.status==200){
                x = xmlhttp.responseXML.getElementsByTagName("note");
                to =
x[0].getElementsByTagName("to")[0].childNodes[0].nodeValue;
                from =
x[0].getElementsByTagName("from")[0].childNodes[0].nodeValue;
                heading =
x[0].getElementsByTagName("heading")[0].childNodes[0].nodeValue;
                body =
x[0].getElementsByTagName("body")[0].childNodes[0].nodeValue;

                txt = 'Message: <br />To : ' + to + '<br />From : ' +
from + '<br />Heading : ' + heading + '<br />Body : ' + body + '<br />';
                document.getElementById('info').innerHTML = txt;

            }
        }
        xmlhttp.send();
    }
</script>
</head>
<body>

<button onclick="loadXMLDoc('note.xml')">View Text</button>
<div id="info"></div>

</body>
</html>

```

3.8. Praktikum 8 : Ajax – jQuery

Bagian ini bertujuan untuk melakukan hal yang sama dengan bagian 3.2 dan 3.3, yaitu menampilkan isi dari input form, hanya saja kode untuk Ajax dituliskan dengan sintaks jQuery.

Ada beberapa pilihan fungsi jQuery yang dapat digunakan untuk Ajax , yaitu:

jQuery load method

```
$(selector).load(URL,data,callback);
```

jQuery get method

```
$.get( url, [data], [callback], [type] )
```

jQuery post method

```
$.post( url, [data], [callback], [type] )
```

jQuery ajax method

```
$.ajax( options )
```

Lakukan langkah-langkah berikut:

1. Buatlah folder baru dengan nama `jquery`, di dalamnya buat folder baru lagi bernama `js`.
Download lib jQuery di <http://jquery.com> dan simpan di folder `jquery/js`.
2. Copy file PHP `process_form_get.php` dan `process_form_post.php` ke folder `jquery`.

3. Buatlah file `user_form.html` di dalam folder `jquery` seperti berikut. Pada contoh ini digunakan jQuery load method untuk melakukan Ajax. Saat menggunakan method load, method yang digunakan untuk request data ke server adalah GET, sehingga alamat url yang di-request ke server adalah `process_form_get.php`.

```
<!DOCTYPE HTML>
<html>
<head>
  <script src="js/jquery-1.11.1.js" type="text/javascript"></script>
  <script>
    $(document).ready(function(){
      $("#btn01").click(function(){
        var vname = $("#name").val();
        var vaddress = $("#address").val();
        var vcity = $("#city").val();

        $("#user_input").load('process_form_get.php',
        {
          name: vname,
          address: vaddress,
          city: vcity
        },function(){
          alert('Page has been loaded');
        });
      });
    });
  </script>
</head>
<body>
<h2>User Input</h2>

<table>
  <tr>
    <td valign="top">Name</td>
    <td valign="top">:</td>
    <td valign="top"><input type="text" name="name" id="name"
size="30" maxlength="50" placeholder="Name (max 50 characters)"></td>
  </tr>
  <tr>
    <td valign="top">Address</td>
    <td valign="top">:</td>
    <td valign="top"><textarea name="address" id="address"
rows="5" cols="30" placeholder="Address (max 100
characters)"></textarea></td>
  </tr>
  <tr>
    <td valign="top">City</td>
    <td valign="top">:</td>
    <td valign="top"><select name="city" id="city">
      <option value="none">--Select a city--</option>
      <option value="Airport West">Airport West</option>
      <option value="Box Hill">Box Hill</option>
      <option value="Yarraville">Yarraville</option>
    </select>
  </td>
  </tr>
  <tr>
    <td colspan="3" valign="top"><br />
      <button id="btn01">Submit</button>
    </td>
  </tr>
</table>

<div id="user_input"></div>
```

```
</body>
</html>
```

Selain method load dapat juga digunakan method yang lain. Untuk mencoba gantilah kode di atas yang menggunakan method load dengan beberapa method berikut.

jQuery get method (method untuk request ke server adalah GET)

```
$.get("process_form_get.php",
{
    name: vname,
    address: vaddress,
    city: vcity
}, function(data,status){
    alert(status);
    if (status === "error"){
        $("#user_input").html("The page can't be loaded");
    }else{
        $("#user_input").html(data);
    }
});
```

atau

```
var url = "process_form_get.php?name=" + vname + "&address=" +
vaddress + "&city=" + vcity;

$.get(url,function(data,status){
    $("#user_input").html(data);
});
```

jQuery post method (method untuk request ke server adalah POST, sehingga file yang direquest adalah process_form_post.php)

```
var url = "process_form_post.php";

$.post(url,
{
    name: vname,
    address: vaddress,
    city: vcity
},
function(data,status){
    $("#user_input").html(data);
});
```

jQuery ajax method (method untuk request ke server pada contoh ini adalah GET)

```
$.ajax({
    url: "process_form_get.php",
    type: "GET",
    dataType: 'html',
    beforeSend: function(){
        $("#user_input").html('');
    },
    success: function(data){
        $("#user_input").html(data);
    },
    error: function(){
        $("#user_input").html("The page can't be loaded");
    }
});
```

4. Latihan

Buatlah program untuk menampilkan detail data customer. Sebelumnya nama-nama customer ditampilkan dalam select item. Saat user memilih nama customer, maka program akan menampilkan detail data customer di bawahnya.

Customers Data

Julie Smith ▼

Name = Julie Smith
Address = 25 Oak Street
City = Airport West

5. Daftar Pustaka

Welling, L. and Thomson, L., 2009, *PHP and MySQL Web Development*, Pearson Education.

Davis, M.E. and Phillips, J.A., 2007, *Learning PHP and MySQL Second Edition*, O'Reilly.

W3Schools, 2017, *AJAX Introduction*, <https://www.w3schools.com/js>, diakses pada 21 Agustus 2017, pukul 15:40.

W3Schools, 2017, *JQuery Tutorial*, <https://www.w3schools.com/js>, diakses pada 22 Agustus 2017, pukul 20:12.