

JOBSHEET 8

PHP - Form Upload, Cookies and Session

Web Design and Programming Courses



Febrian Arka Samudra

2341720066 / 10

STUDY PROGRAM D-IV INFORMATICS ENGINEERING

INFORMATICS TECHNOLOGY MAJOR

POLITEKNIK NEGERI MALANG

Jl. Soekarno Hatta No.9, Jatimulyo, kec, Lowokwaru, Kota Malang, Jawa Timur 65141

Practical Section 1. Files

Save the file, then open a browser and run localhost/week8/form_upload.php.

Select a file and click the Submit button. Observe what happens and record your understanding. (Question No. 1)

```
form_upload.php
1  <html>
2    <head>
3      <title>File Upload</title>
4    </head>
5    <body>
6      <form action="upload.php" method="POST" enctype="multipart/form-data">
7        <input type="file" name="myfile">
8        <input type="submit" name="submit">
9      </form>
10   </body>
11 </html>
```

Choose File No file chosen

Submit

Explanation :

Form_upload.php

- Allows users to select and upload a file using `<input type="file">` and submit it using `<input type="submit">`.
- The form uses the POST method and `enctype="multipart/form-data"` to handle file uploads.

Next, create a folder named uploads in the week8. Re-run

localhost/week8/form_upload.php. Select a file and click the Submit button. Observe what happens and record your understanding. (Question No. 2)

```
upload.php
1  <?php
2  if (isset($_POST["submit"])) {
3      $targetdir = "uploads/";
4      $targetfile = $targetdir . basename($_FILES["myfile"]["name"]);
5
6      if (move_uploaded_file($_FILES["myfile"]["tmp_name"], $targetfile)) {
7          echo "file berhasil diunggah.";
8      }
9      else {
10         echo "Gagal mengunggah file.";
11     }
12 }
13 ?>
```

file berhasil diunggah.

Explanation : After creating the uploads folder and re-running the file upload script

1. File Submission The form allows you to select and submit a file
2. PHP Execution The script moves the uploaded file to the uploads folder
3. Successful Upload If the folder exists and has the right permissions, the file is uploaded, and the message "file berhasil diunggah." is displayed
4. If permissions are incorrect, the upload fails with the message "Gagal mengunggah file."

Save the file, open a browser and run localhost/week8/form_upload.php

Select a file with the .pdf extension or .docx. Click the Submit button.

Observe what happens and record your understanding.

(Question No. 3)

```
upload.php
1  <?php
2  if (isset($_POST["submit"])) {
3      $targetdir = "uploads/";
4      $targetfile = $targetdir . basename($_FILES["myfile"]["name"]);
5      $filetype = strtolower(pathinfo($targetfile, PATHINFO_EXTENSION));
6
7      $allowedExtensions = array("jpg", "jpeg", "png", "gif");
8
9      if (in_array($filetype, $allowedExtensions) && $_FILES["myfile"]["size"]<=$maxsize)
10     {
11         if (move_uploaded_file($_FILES["myfile"]["tmp_name"], $targetfile)) {
12             echo "file berhasil diunggah.";
13         }
14         else {
15             echo "Gagal mengunggah file.";
16         }
17     }else {
18         echo "File tidak valid atau melebihi ukuran maksimum yang diizinkan";
19     }
20 }
21
22 ?>
```

File tidak valid atau melebihi ukuran maksimum yang diizinkan

Explanation : The code `in_array($fileType, $allowedExtensions)` checks if the file type is among the permitted extensions (such as jpg, jpeg, png, or gif). If the file type is allowed, it outputs "File berhasil diunggah" (File successfully uploaded). Otherwise, it displays "File tidak valid atau melebihi ukuran maksimum yang diizinkan" (Invalid file or exceeds the maximum allowed size).

Next run localhost/week8/form_upload.php again.

Select a file with the extension .jpg, .jpeg, .png, or .gif. Click the Submit button.

Observe what happens and record your understanding.

(Question No. 4)

```
upload.php
1  <?php
2  if (isset($_POST["submit"])) {
3      $targetdir = "image1.jpg";
4      $targetfile = $targetdir . basename($_FILES["myfile"]["name"]);
5      $filetype = strtolower(pathinfo($targetfile, PATHINFO_EXTENSION));
6
7      $allowedExtensions = array("jpg", "jpeg", "png", "gif");
8      $maxsize = 5*1024*1024;
9
10     if (in_array($filetype, $allowedExtensions) && $_FILES["myfile"]["size"]<=$maxsize)
11     {
12         if(move_uploaded_file($_FILES["myfile"]["tmp_name"], $targetfile)) {
13             echo "file berhasil diunggah.";
14         }
15         else {
16             echo "Gagal mengunggah file.";
17         }
18     }else {
19         echo "File tidak valid atau melebihi ukuran maksimum yang diizinkan";
20     }
21 }
22 ?>
```

file berhasil diunggah.

Explanation : It says "File berhasil diunggah" because the file meets the criteria, which include being a PNG file and not exceeding the 5MB size limit.

Add script from step 6 to display thumbnail image files with a width of 200 and height

following the changes automatically after the image file is successfully uploaded.

Screen shoot the additional code. Explain your understanding after adding the program code.

(Question No. 5)

```
if(move_uploaded_file($_FILES["myfile"]["tmp_name"], $targetfile)) {
    echo "file berhasil diunggah.";
    echo "<img src='$targetfile' width='200' style='height: auto;
    '><br>";
}
```

Explanation : The line echo ""; displays the uploaded image. The src attribute points to \$targetfile, which is the file path of the uploaded image. The width='200' sets the image's width to 200 pixels, and the height adjusts automatically to

preserve the original aspect ratio. The alt='Thumbnail' attribute provides alternative text for the image, which is helpful for accessibility purposes.

Save the file, open a browser and run localhost/week8/form_upload.php

Select a file with an extension of .txt, .pdf, .doc, or .docx that is more than 5 MB in size. Click the Submit button. Observe what happens and record your understanding.

(Question No. 6)

```
upload.php
1  <?php
2  if (isset($_POST["submit"])) {
3      $targetdir = "image1.jpg";
4      $targetfile = $targetdir . basename($_FILES["myfile"]["name"]);
5      $filetype = strtolower(pathinfo($targetfile, PATHINFO_EXTENSION));
6
7      $allowedExtensions = array("txt", "pdf", "doc", "docx");
8      $maxsize = 3*1024*1024;
9
10     if (in_array($filetype, $allowedExtensions) && $_FILES["myfile"]["size"]<=$maxsize)
11     {
12         if(move_uploaded_file($_FILES["myfile"]["tmp_name"], $targetfile)) {
13             echo "file berhasil diunggah.";
14             echo "<img src='$targetfile' width='200' style='height: auto;
15             '><br>";
16         }
17         else {
18             echo "Gagal mengunggah file.";
19         }
20     }else {
21         echo "File tidak valid atau melebihi ukuran maksimum yang diizinkan";
22     }
23 }
24 ?>
```

File tidak valid atau melebihi ukuran maksimum yang diizinkan

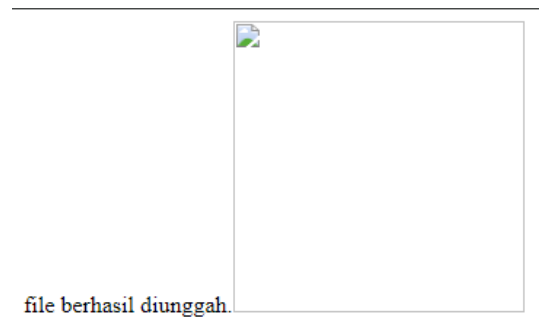
Explanation : It displays "File tidak valid atau melebihi ukuran maksimum yang diizinkan" because the file doesn't meet the size requirements. The code specifies a maximum file size of 3MB.

Next run localhost/week8/form_upload.php again.

Select a file with the extension .txt, .pdf, .doc, or .docx that is less than 3 MB in size.

Click the Submit button. Observe what happens and record your understanding.

(Question No. 7)



Explanation :

It displays "File berhasil diunggah" because the file meets the size requirement of a maximum of 3MB (My file is 600kb) and also satisfies the type requirement, which is a PDF.

Practical Section 2. Multi Upload File

Save the file, open the browser and run

localhost/week8/form_multiupload.php. Select multiple files at once to upload. What do you understand from the script in the file? Record your understanding.

(Question No. 8)

```
form_multiupload.php
1 <!DOCTYPE html>
2 <html lang="en">
3 <head>
4   <meta charset="UTF-8">
5   <meta name="viewport" content="width=device-width, initial-scale=1.0">
6   <title>Multiupload Dokumen</title>
7 </head>
8 <body>
9   <h2>Unggah Dokumen</h2>
10  <form action="proses_upload.php" method="post" enctype="multipart/form-data">
11    <input type="file" name="files[]" multiple="multiple" accept=".pdf, .doc, .docx">
12    <input type="submit" value="Unggah" />
13  </form>
14 </body>
15 </html>
```

```

proses_upload.php
1  <?php
2  // Lokasi penyimpanan file yang diunggah
3  $targetDirectory = "documents/";
4
5  // Periksa apakah direktori penyimpanan ada, jika tidak maka buat
6  if (!file_exists($targetDirectory)) {
7      mkdir($targetDirectory, 0777, true);
8  }
9
10 if (isset($_FILES['files']['name'][0])) {
11     $totalFiles = count($_FILES['files']['name']);
12
13     // Loop melalui semua file yang diunggah
14     for ($i = 0; $i < $totalFiles; $i++) {
15         $fileName = $_FILES['files']['name'][$i];
16         $targetFile = $targetDirectory . $fileName;
17
18         // Pindahkan file yang diunggah ke direktori penyimpanan
19         if (move_uploaded_file($_FILES['files']['tmp_name'][$i], $targetFile)) {
20             echo "File $fileName berhasil diunggah.<br>";
21         } else {
22             echo "Gagal mengunggah file $fileName.<br>";
23         }
24     }
25 } else {
26     echo "Tidak ada file yang diunggah.";
27 }
28
29 ?>

```

Unggah Dokumen

Choose Files JOBSHEET...066_10.pdf Unggah

File JOBSHEET 7_PHP – Form Processing_Febrian Arka Samudra_2341720066_10.pdf berhasil diunggah.

Explanation :

- Handling the Uploaded Files:**
 - `if ($_FILES['files']['name'][0]):` Verifies if at least one file has been uploaded.
 - `$totalFiles = count($_FILES['files']['name']);`: Counts the total number of uploaded files. This enables the script to handle multiple file uploads in a single form submission.
- Looping Through the Uploaded Files:**
 - `for ($i = 0; $i < $totalFiles; $i++):` Loops through each uploaded file.
 - `$fileName = $_FILES['files']['name'][$i];`: Gets the name of each uploaded file.
 - `$targetFile = $targetDirectory . $fileName;`: Defines the destination path for each file by combining the target directory (documents/) with the file name.

Change the code for multi upload of images.

Screen shoot the code changes and provide an explanation of the code.

(Question No. 9)

```
form_multiupload.php
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title>Multiupload Gambar</title>
7  </head>
8  <body>
9      <h2>Unggah Gambar</h2>
10     <form action="proses_upload.php" method="post" enctype="multipart/form-data">
11         <input type="file" name="images[]" multiple="multiple" accept=".jpg, .jpeg, .png, .gif">
12         <input type="submit" value="Unggah" />
13     </form>
14 </body>
15 </html>
```

```
proses_upload.php
1  <?php
2  // Lokasi penyimpanan file yang diunggah
3  $targetDirectory = "images/";
4
5  // Periksa apakah direktori penyimpanan ada, jika tidak maka buat
6  if (!file_exists($targetDirectory)) {
7      mkdir($targetDirectory, 0777, true);
8  }
9
10 // Check if files were uploaded
11 if (isset($_FILES['images']) && $_FILES['images']['name'][0]) {
12     $totalFiles = count($_FILES['images']['name']);
13
14     // Loop through all uploaded files
15     for ($i = 0; $i < $totalFiles; $i++) {
16         $fileName = $_FILES['images']['name'][$i];
17         $targetFile = $targetDirectory . $fileName;
18         $fileType = strtolower(pathinfo($targetFile, PATHINFO_EXTENSION));
19
20         // Only allow certain image file types
21         $allowedExtensions = array("jpg", "jpeg", "png", "gif");
22
23         if (in_array($fileType, $allowedExtensions)) {
24             // Move uploaded file to the target directory
25             if (move_uploaded_file($_FILES['images']['tmp_name'][$i], $targetFile)) {
26                 echo "File $fileName berhasil diunggah.<br>";
27             } else {
28                 echo "Gagal mengunggah file $fileName.<br>";
29             }
30         } else {
31             echo "File $fileName memiliki tipe tidak valid. Hanya file JPG, JPEG, PNG, dan GIF yang diizinkan.<br>";
32         }
33     }
34 } else {
35     echo "Tidak ada file yang diunggah.";
36 }
37 >
```


Unggah Gambar

Choose Files

WhatsApp Image 2023-10-30 at 21.48.28_3a45aafc.jpg

Unggah

File WhatsApp Image 2023-10-30 at 21.48.28_3a45aafc.jpg berhasil diunggah.

Explanation : The \$targetDirectory is set to "images/", meaning that uploaded images will be saved in the "images/" folder.

- An array \$allowedExtensions is used to define the permitted image file types (jpg, jpeg, png, gif).
- When a user selects multiple image files using the file input and submits the form, the files are sent to proses_upload.php.
- For each file, the script verifies if it is a valid image type and tries to move it to the "images/" directory.
- If the upload is successful, a success message is shown. Otherwise, an error message is displayed if the file type is invalid or the upload fails for any reason.

Practical Section 3. Upload Files with PHP and Jquery

Observe what is happening and explain your understanding.

(Question No. 10)

```
form_upload_ajax.php
1  <!DOCTYPE html>
2  <html>
3
4  <head>
5  |   <title>Unggah File Dokumen</title>
6  </head>
7
8  <body>
9  |   <form id="upload-form" action="upload_ajax.php" method="post" enctype="multipart/form-data">
10 |       <input type="file" name="file" id="file">
11 |       <input type="submit" name="submit" value="Unggah">
12 |   </form>
13 |   <div id="status"></div>
14
15 |   <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
16 |   <script src="upload.js"></script>
17 </body>
18
19 </html>
```

```

JS upload.js > ...
1  $(document).ready(function() {
2      $('#upload-form').submit(function(e) {
3          e.preventDefault();
4
5          var formData = new FormData(this);
6
7          $.ajax({
8              type: 'POST',
9              url: 'upload_ajax.php',
10             data: formData,
11             cache: false,
12             contentType: false,
13             processData: false,
14             success: function(response) {
15                 $('#status').html(response);
16             },
17             error: function() {
18                 $('#status').html('Terjadi kesalahan saat mengunggah file.');

```

```

upload_ajax.php
1  <?php
2  if (isset($_FILES['file'])) {
3      $errors = array();
4      $file_name = $_FILES['file']['name'];
5      $file_size = $_FILES['file']['size'];
6      $file_tmp = $_FILES['file']['tmp_name'];
7      $file_type = $_FILES['file']['type'];
8      $file_ext = strtolower(end(explode('.', $_FILES['file']['name'])));
9      $extensions = array("pdf", "doc", "docx", "txt");
10
11      if (in_array($file_ext, $extensions) === false) {
12          $errors[] = "Ekstensi file yang diizinkan adalah PDF, DOC, DOCX, atau TXT.";
13      }
14
15      if ($file_size > 2097152) {
16          $errors[] = 'Ukuran file tidak boleh lebih dari 2 MB';
17      }
18
19      if (empty($errors) == true) {
20          move_uploaded_file($file_tmp, "documents/" . $file_name);
21          echo "File berhasil diunggah.";
22      } else {
23          echo implode(" ", $errors);
24      }
25  }
26  ?>

```

Choose File WhatsApp I...3a45aafc.jpg Unggah

Ekstensi file yang diizinkan adalah PDF, DOC, DOCX, atau TXT.

Choose File JOBSHEET ...66_10.docx Unggah

Ukuran file tidak boleh lebih dari 2 MB

JOBSHEET ...0066_10.pdf

File berhasil diunggah.

Explanation : The first condition occurs because the file extension of the uploaded file is not in the \$extensions array. The second condition happens because the \$file_size exceeds 2,097,152 bytes or 2MB. The final condition occurs when the file extension is valid, and the file size does not exceed 2MB.

Change the code to be able to do multi-upload image files.

Screenshot the code changes and explain the code.

(Question No. 11)

```
$extensions = array("jpg", "jpeg", "png", "gif");

if (in_array($file_ext, $extensions) === false) {
    $errors[] = "Ekstensi file yang diizinkan adalah jpg, jpeg, png, atau gif.";
}
```

image1.jpg

File berhasil diunggah.

Explanation : Change the contents of the \$extensions array to an array of image extensions.

Practical Section 4. Decorate Upload Files

Save the file. Open a browser and run

localhost/week8/form_upload_ajax.php.

What do you understand from the program code above? Record your understanding.

(Question No. 12)

```

form_upload_ajax.php
1  <!DOCTYPE html>
2  <html>
3
4  <head>
5      <link rel="stylesheet" type="text/css" href="upload.css">
6      <title>Unggah File Dokumen</title>
7  </head>
8
9  <body>
10     <div class="upload-form-container">
11         <h2>Unggah File Dokumen</h2>
12         <form id="upload-form" action="upload.php" method="post" enctype="multipart/form-data">
13             <div class="file-input-container">
14                 <input type="file" name="file" id="file" class="file-input">
15                 <label for="file" class="file-label">Pilih File</label>
16             </div>
17             <button type="submit" name="submit" class="upload-button" id="upload-button" disabled>Unggah</button>
18         </form>
19         <div id="status" class="upload-status"></div>
20     </div>
21
22     <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
23     <script src="upload.js"></script>
24 </body>
25
26 </html>








```

```

JS upload.js > ready() callback > submit() callback
1  $(document).ready(function() {
2      $('#file').change(function() {
3          if (this.files.length > 0) {
4              $('#upload-button').prop('disabled', false).css('opacity', 1);
5          } else {
6              $('#upload-button').prop('disabled', true).css('opacity', 0.5);
7          }
8      });
9
10     $('#upload-form').submit(function(e) {
11         e.preventDefault();
12
13         var formData = new FormData(this);
14
15         $.ajax({
16             type: 'POST',
17             url: 'upload_ajax.php',
18             data: formData,
19             cache: false,
20             contentType: false,
21             processData: false,
22             success: function(response) {
23                 $('#status').html(response);
24             },
25             error: function() {
26                 $('#status').html('Terjadi kesalahan saat mengunggah file.');


```

```

# upload.css >  .upload-status
1  .upload-form-container {
2      max-width: 400px;
3      margin: 0 auto;
4      padding: 20px;
5      border: 1px solid  #ccc;
6      border-radius: 5px;
7      text-align: center;
8  }
9
10 h2 {
11     margin: 0;
12     font-size: 24px;
13     color:  #333;
14 }
15
16 .file-input-container {
17     display: flex;
18     justify-content: center;
19     align-items: center;
20     margin: 20px 0;
21 }
22
23 .file-input {
24     display: none;
25 }
26
27 .file-label {
28     background:  #3498db;
29     color:  #fff;
30     padding: 10px 20px;
31     border-radius: 5px;
32     cursor: pointer;
33 }
34
35 .upload-button {
36     background:  #2ecc71;
37     color:  #fff;
38     padding: 10px 20px;
39     border: none;

```

```

40     border-radius: 5px;
41     cursor: pointer;
42     opacity: 0.5;
43 }
44
45 .upload-button:disabled {
46     background:  #ccc;
47     cursor: not-allowed;
48 }
49
50 .upload-status {
51     margin-top: 20px;
52     font-weight: bold;
53 }

```



Explanation : The code adds styling to the appearance of the web page as follows.

Practical Section 5. Creating Cookies

Observe and explain your observations

(Question No. 13)

```
🐘 cookiesCreate.php
1  <?php
2  setcookie("user","Polinema",time()+3600);
3  ?>
```

```
🐘 cookiesCall.php
1  <?php
2  |      echo $_COOKIE ['user'];
3  ?>
```

Warning: Undefined array key "user" in D:\laragon\www\PemogrammanWeb\Week8\cookiesCall.php on line 2

Explanation : The error occurs because the first time you run the code, `$_COOKIE['user']` will not be set, so nothing will be displayed from the echo statement. After reloading the page or accessing it again, the user cookie will be set to "Polinema," and the next time you visit the page, it will show the value "Polinema."

Observe and explain the results displayed

(Question No. 14)

Polinema

Explanation : The code `setcookie("user", "Polinema", time() + 3600);` creates a cookie named "user" with the value "Polinema," which will expire after one hour (3600 seconds). On the first load, the cookie is set but is not available until the next request. On the second page load, `$_COOKIE['user']` becomes accessible, and it will display "Polinema."

Observe and explain the results displayed.

(Question No. 15)

Polinema

Explanation : If the cookie has expired, it will be reset, and nothing will be displayed at first until you refresh the page, which will set the cookie again.

Practical Section 6. Deleting the Value of Cookies

Observe and describe the results from steps 2 and 3, then draw conclusions.

(Question No. 16)

```
cookiesDel.php
1  <?php
2  setcookie("user", "Polinema", time()-3600);
3  ?>
```

Warning: Undefined array key "user" in `D:\laragon\www\PemogramanWeb\Week8\cookiesCall.php` on line 2

Explanation : The error occurs because it deletes the cookies by setting the expiration time of the user cookie to one hour in the past (`time() - 3600`).

Practical Section 7. Application of Cookies to the Shopping Cart Feature

Observe and explain the results displayed.

(Question No. 17)

```
<> formBeli.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title></title>
7  </head>
8  <body>
9      <form action="prosesBeli.php" method="post">
10         <p>Jumlah Novel yang dibeli :
11         | <input type="text" name="beliNovel" value="0" size="2">
12         </p>
13         <p>Jumlah Buku Teks yang dibeli :
14         | <input type="text" name="beliBuku" value="0" size="2">
15         </p>
16         <input type="submit">
17     </form>
18 </body>
19 </html>
```

```
prosesBeli.php
1  <?php
2      if(isset($_POST["beliNovel"]) && isset($_POST["beliBuku"])) {
3          setcookie('beliNovel', $_POST["beliNovel"], time() + 60);
4          setcookie('beliBuku', $_POST["beliBuku"], time() + 60);
5          header("Location: keranjangBelanja.php");
6      }
7  ?>
```

```
keranjangBelanja.php
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4      <meta charset="UTF-8">
5      <meta name="viewport" content="width=device-width, initial-scale=1.0">
6      <title></title>
7  </head>
8  <body>
9      <h2>Keranjang Belanja</h2>
10
11      <?php
12          $beliNovel = $_COOKIE['beliNovel'];
13          $beliBuku = $_COOKIE['beliBuku'];
14
15          echo "Jumlah Novel : " . $beliNovel . "<br>";
16          echo "Jumlah Buku : " . $beliBuku . "<br>";
17      ?>
18
19 </body>
20 </html>
```


Keranjang Belanja

Warning: Undefined array key "beliNovel" in D:\laragon\www\PemrogramanWeb\Week8\keranjangBelanja.php on line 12

Warning: Undefined array key "beliBuku" in D:\laragon\www\PemrogramanWeb\Week8\keranjangBelanja.php on line 13

Jumlah Novel :

Jumlah Buku :

Explanation : The web page displays an error when the program reaches step three.

Observe and explain the results displayed.

(Question No. 18)

Jumlah Novel yang dibeli :

Jumlah Buku Teks yang dibeli :

Keranjang Belanja

Jumlah Novel : 1
Jumlah Buku : 2

Explanation : It retrieves the values of two cookies: beliNovel and beliBuku. Then, it outputs the number of novels (beliNovel) and books (beliBuku) stored in those cookies.

Observe and explain the results displayed.

(Question No. 19)

Keranjang Belanja

Jumlah Novel : 1

Jumlah Buku : 2

Explanation : Since cookies are stored in the browser, they should remain even after you close and reopen it, unless:

The cookies are set to expire after 60 seconds. After setting the cookies, the user is redirected to keranjangBelanja.php, where the items in the cart will be shown.

Practical Section 8. Creating a Session

Observe and explain the results displayed

(Question No. 20)

```
sessionCreate.php
1  <?php
2      session_start();
3  ?>
4
5  <!DOCTYPE html>
6  <html>
7  <body>
8      <?php
9          $_SESSION["favcolor"] = "green";
10         $_SESSION["favanimal"] = "cat";
11         echo "Session variable are set.";
12     ?>
13 </body>
14 </html>
```

```
SessionCall.php
1  <?php
2      session_start();
3  ?>
4
5  <!DOCTYPE html>
6  <html>
7  <body>
8      <?php
9          echo "Favorite color is : " . $_SESSION["favcolor"] . "<br>";
10         echo "Favorite animal is : " . $_SESSION["favanimal"] . "<br>";
11     ?>
12 </body>
13 </html>
```

Warning: Undefined array key "favcolor" in D:\laragon\www\PemogramanWeb\Week8\SessionCall.php on line 9
Favorite color is : ,

Warning: Undefined array key "favanimal" in D:\laragon\www\PemogramanWeb\Week8\SessionCall.php on line 10
Favorite animal is :

Session variable are set.

Favorite color is : green,
Favorite animal is : cat

Explanation : After establishing the session in sessionCreate.php, you should see the session variables displayed when you access sessionCall.php. If you attempt to access sessionCall.php before

running sessionCreate.php, there will be no session data set, which may result in an error or undefined values, depending on how PHP is configured to manage missing session variables.

Practical Section 9. Removing Session Values

Observe and explain the results displayed.

(Question No. 21)

```
sessionDel.php
1  <?php
2      session_start();
3  ?>
4
5  <!DOCTYPE html>
6  <html>
7  <body>
8      <?php
9          session_unset();
10         session_destroy();
11
12         echo "All session variables are now removed, and the session is destroyed"
13     ?>
14 </body>
15 </html>
```

All session variables are now removed, and the session is destroyed

Warning: Undefined array key "favcolor" in D:\laragon\www\PemogramanWeb\Week8\SessionCall.php on line 9
Favorite color is : ,

Warning: Undefined array key "favanimal" in D:\laragon\www\PemogramanWeb\Week8\SessionCall.php on line 10
Favorite animal is :

Explanation : Session Handling: The code utilizes session_unset() to clear all session variables and session_destroy() to eliminate the session itself.

Practical Section 10. Implementation of Session on the Login Feature

Observe and explain the results displayed

(Question No. 22)

```
<> sessionLoginForm.html > html > body > form > table > tr > td > input
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <title>File Upload</title>
5  </head>
6  <body>
7  |   <form action="sessionLoginProcess.php" method="post">
8  |   |   <table>
9  |   |   |   <tr>
10 |   |   |   |   <td>Username</td>
11 |   |   |   |   <td><input type="text" name="username" size="20"></td>
12 |   |   |   |   </tr>
13 |   |   |   |   <tr>
14 |   |   |   |   <td>Password</td>
15 |   |   |   |   <td><input type="password" name="password" size="20"></td>
16 |   |   |   |   </tr>
17 |   |   |   |   <tr>
18 |   |   |   |   <td>&nbsp;</td>
19 |   |   |   |   <td><input type="submit" name="Login" value="Login"></td>
20 |   |   |   |   </tr>
21 |   |   |   </table>
22 |   </form>
23 </body>
24 </html>
```

```
sessionLoginProcess.php
1  <?php
2  $username = $_POST['username'];
3  $password = $_POST['password'];
4
5  if($username == 'admin' && $password == '1234'){
6  |   session_start();
7  |   $_SESSION['username'] = $username;
8  |   $_SESSION['status'] = 'login';
9  |   echo "Anda berhasil login. Silahkan menuju <a href='homeSession.php'>Halaman Home</a>";
10 | } else {
11 |   echo "Gagal login. Silahkan login lagi <a href='sessionLoginForm.html'>Halaman Login</a>";
12 | }
13 ?>
```

```
sessionLogout.php
1  <?php
2  session_start();
3  session_destroy();
4
5  echo "Anda berhasil logout";
6  ?>
```

```

homeSession.php
1  <!DOCTYPE html>
2  <html>
3  <head>
4  |   <title>Halaman Home</title>
5  </head>
6  <body>
7  |   <?php
8  |       session_start();
9  |
10 |       if($_SESSION['status'] == 'login') {
11 |           echo "Selamat datang " . $_SESSION['username'];
12 |           ?>
13 |           <br> <a href="sessionLogout.php"> Log Out</a>
14 |       <?php
15 |       } else {
16 |           echo "Anda belum logi, silahkan "
17 |           ?>
18 |           <a href="sessionLoginForm.html">Log In</a>
19 |       <?php
20 |       }
21 |   ?>
22 </body>
23 </html>

```

Username

Password

Gagal login. Silahkan login lagi [Halaman Login](#)

Explanation : When you enter "admin" as the username and "0000" as the password and click "Log in," the sessionLoginProcess.php script will receive the POST request. The script verifies the provided credentials against the hardcoded values:

- Username: admin
- Password: 1234

Since the provided password (0000) does not match the expected password (1234), the login attempt fails.

Observe and explain the results displayed

(Question No. 23)

Anda berhasil login. Silahkan menuju [Halaman Home](#)

Explanation : It is successful because the credentials match:

- Username: admin
- Password: 1234

Describe the sequence of the process from login to logout (also mention the order in which the files are processed)

(Question No. 24)

Explanation :

- **sessionLoginForm.html:** The initial form is presented to the user.
- **sessionLoginProcess.php:** Handles the login attempt and verifies the credentials.
- **homeSession.php (if login is successful):** Displays a welcome message and a logout link.
- **sessionLogout.php (upon logging out):** Ends the session and confirms the logout.

The process from login to logout begins with the user accessing sessionLoginForm.html, where they enter their username and password. Upon submission, the form data is sent to sessionLoginProcess.php, which checks the credentials. If the login is successful, the user is redirected to homeSession.php, where a welcome message and a logout link are displayed. If the login fails, a failure message prompts the user to try again. When the user clicks the logout link, they are taken to sessionLogout.php, which destroys the session and confirms that the user has logged out. This sequence illustrates how user actions guide the flow between these PHP files while effectively managing the session state.

Github : <https://github.com/FebrianArkaSamudra/PemrogramanWeb/tree/main/Week8>