



**Link Repository :** [https://github.com/athfizh/AthaullaHafizh\\_DESPROG\\_TI2A\\_2025](https://github.com/athfizh/AthaullaHafizh_DESPROG_TI2A_2025)

### Topik

- Konsep Form Upload dengan PHP dan JQuery.
- Pengenalan konsep cookies.
- Pengenalan konsep session.

### Tujuan

Mahasiswa diharapkan dapat:

1. Mahasiswa mampu membuat form upload menggunakan PHP dan jQuery.
2. Memahami konsep dasar cookies dan penggunaannya.
3. Memahami konsep dasar session dan penggunaannya.

### Pendahuluan

#### **File Upload**

File upload dalam PHP adalah proses mengunggah file dari komputer pengguna ke server web. File upload berguna dalam banyak kasus seperti saat pengguna ingin mengunggah gambar profil, dokumen, atau media lainnya ke situs web. Untuk melakukan ini, dapat menggunakan elemen HTML `<input type="file">` bersama dengan PHP untuk mengelola proses pengunggahan.

Berikut adalah langkah-langkah umum untuk mengimplementasikan file upload dalam PHP:

- Buat formulir HTML untuk mengunggah file:

```
<!DOCTYPE html>
<html>
<head>
  <title>File Upload</title>
</head>
<body>
  <form action="upload.php" method="post" enctype="multipart/form-data">
    <input type="file" name="fileToUpload" id="fileToUpload">
    <input type="submit" value="Upload File" name="submit">
  </form>
</body>
</html>
```

- Buat skrip PHP (**upload.php**) untuk mengelola pengunggahan file:

```
<?php
if (isset($_POST["submit"])) {
    $targetDirectory = "uploads/"; // Direktori tujuan untuk menyimpan file
    $targetFile = $targetDirectory . basename($_FILES["fileToUpload"]["name"]);

    if (move_uploaded_file($_FILES["fileToUpload"]["tmp_name"], $targetFile)) {
        echo "File berhasil diunggah.";
    } else {
        echo "Gagal mengunggah file.";
    }
}
```

Di atas, kita melakukan beberapa langkah penting:

- Membuat formulir HTML dengan elemen `<input type="file">` yang memungkinkan pengguna memilih file yang ingin diunggah.
- Mengatur atribut `enctype` formulir menjadi `"multipart/form-data"` agar file dapat diunggah.
- Ketika pengguna mengklik tombol "Upload File," data formulir akan dikirim ke `upload.php`.
- Di `upload.php`, kita memeriksa apakah pengguna telah mengklik tombol submit (`$_POST["submit"]`) dan kemudian menentukan direktori tujuan untuk menyimpan file.
- `move_uploaded_file()` digunakan untuk memindahkan file dari direktori sementara (temp) ke direktori tujuan yang telah ditentukan.
- Pesan yang sesuai (berhasil atau gagal) akan ditampilkan ke pengguna.

Pastikan telah memiliki direktori `uploads` pada server Anda, dan berikan izin yang sesuai agar PHP dapat menyimpan file di sana.

## Praktikum Bagian 1. Upload File

Langkah	Keterangan
1	Buatlah file baru di dalam direktori <code>dasarWeb</code> , beri nama <code>form_upload.php</code> .
2	<p>Ketikkan ke dalam file <code>form_upload.php</code> kode di bawah ini.</p> <pre>&lt;html&gt;   &lt;head&gt;     &lt;title&gt;File Upload&lt;/title&gt;   &lt;/head&gt;   &lt;body&gt;     &lt;form action="upload.php" method="POST" enctype="multipart/form-data"&gt;       &lt;input type="file" name="myfile"&gt;       &lt;input type="submit" name="submit"&gt;     &lt;/form&gt;   &lt;/body&gt; &lt;/html&gt;</pre>
3	Buat file baru bernama <code>upload.php</code> yang akan digunakan untuk pemrosesan <code>form_upload.php</code> .

	<pre> &lt;?php if(isset(\$_POST["submit"])){     \$targetdir = "uploads/"; //Direktori tujuan untuk menyimpan file     \$targetfile = \$targetdir . basename(\$_FILES["myfile"]["name"]);      if(move_uploaded_file(\$_FILES["myfile"]["tmp_name"], \$targetfile)){         echo "File berhasil diunggah.";     }     else{         echo "Gagal mengunggah file.";     } } ?&gt; </pre>
4	<p>Simpan file, kemudian buka browser dan jalankan <b>localhost/dasarWeb/form_upload.php</b>. Pilih sebuah file dan klik tombol Submit. Amati yang terjadi dan catat pemahaman Anda. (Pertanyaan No. 1)</p> <p><small>Warning: move_uploaded_file(uploads/—Pngtree—github logo icon_8630451.png): Failed to open stream: No such file or directory in C:\laragon\www\pertemuan08\dasarWeb\upload.php on line 6</small></p> <p><small>Warning: move_uploaded_file(): Unable to move "C:\Users\HAFIZH\AppData\Local\Temp\phpBAEC.tmp" to "uploads/—Pngtree—github logo icon_8630451.png" in C:\laragon\www\pertemuan08\dasarWeb\upload.php on line 6</small> Gagal mengunggah file.</p> <p>Gagal upload file / error, karena belum ada destinasi folder upload/ tempat untuk mengupload gambarnya.</p>
5	<p>Selanjutnya buat folder bernama <b>uploads</b> pada direktori <b>dasarWeb</b>. Jalankan kembali <b>localhost/dasarWeb/form_upload.php</b>. Pilih sebuah file dan klik tombol Submit. Amati yang terjadi dan catat pemahaman Anda. (Pertanyaan No. 2)</p> <p><b>File berhasil diunggah.</b></p> <p>File berhasil diunggah karena sudah ada destinasi folder upload/ tempat untuk mengupload gambarnya.</p>
6	<p>Ubah isi dari file <b>upload.php</b> dengan kode berikut</p> <pre> &lt;?php if(isset(\$_POST["submit"])){     \$targetdir = "uploads/"; //Direktori tujuan untuk menyimpan file     \$targetfile = \$targetdir . basename(\$_FILES["myfile"]["name"]);     \$fileType = strtolower(pathinfo(\$targetfile, PATHINFO_EXTENSION));      \$allowedExtensions = array("jpg", "jpeg", "png", "gif");     \$maxsize = 5*1024*1024;      if (in_array(\$fileType, \$allowedExtensions) &amp;&amp; \$_FILES["myfile"]["size"]&lt;=\$maxsize)     {         if(move_uploaded_file(\$_FILES["myfile"]["tmp_name"], \$targetfile)){             echo "File berhasil diunggah.";         }         else{             echo "Gagal mengunggah file.";         }     }     else{         echo "File tidak valid atau melebihi ukuran maksimum yang diizinkan";     } } ?&gt; </pre>

7	<p>Simpan file, buka browser dan jalankan <code>localhost/dasarWeb/form_upload.php</code>. Pilih sebuah file dengan ekstensi .pdf atau .docx. Klik tombol Submit. Amati yang terjadi dan catat pemahaman Anda. (Pertanyaan No. 3)</p> <p><b>File tidak valid atau melebihi ukuran maksimum yang diizinkan</b></p> <p>Karena tidak ada validasi format file .pdf pada source kode dan melebihi batas ukuran file juga.</p>
8	<p>Selanjutnya jalankan kembali <code>localhost/dasarWeb/form_upload.php</code>. Pilih sebuah file dengan ekstensi .jpg, .jpeg, .png, atau .gif. Klik tombol Submit. Amati yang terjadi dan catat pemahaman Anda. (Pertanyaan No. 4)</p> <p><b>File berhasil diunggah.</b></p> <p>File berhasil diunggah jika validasi berhasil.</p>
9	<p>Tambahkan script langkah 6 untuk menampilkan file gambar thumbnail dengan ukuran lebar 200 dan tinggi mengikuti perubahan secara otomatis setelah file gambar berhasil diunggah. Screenshoot tambahan kode. Jelaskan pemahaman Anda setelah penambahan kode program tersebut. (Pertanyaan No. 5)</p> <pre> if (in_array(\$fileType, \$allowedExtensions) &amp;&amp; \$_FILES["myfile"]["size"] &lt;= \$maxsize)     if(move_uploaded_file(\$_FILES["myfile"]["tmp_name"], \$targetfile)) {         echo "File berhasil diunggah.";         // -- BARIS TAMBAHAN DI SINI --         echo "&lt;br&gt;&lt;img src='\$targetfile' width='200'&gt;";     } else{     echo "Gagal mengunggah file."; } else{     echo "File tidak valid atau melebihi ukuran maksimum yang diizinkan"; } </pre> <p>Penambahan baris <code>echo "&lt;br&gt;&lt;img src='\$targetfile' width='200'&gt;";</code> memiliki fungsi:</p> <ul style="list-style-type: none"> <li>• <b>echo:</b> Perintah PHP untuk menampilkan output HTML ke browser.</li> <li>• <b>&lt;br&gt;:</b> Tag HTML untuk membuat baris baru, agar gambar tampil di bawah teks "File berhasil diunggah."</li> <li>• <b>&lt;img ...&gt;:</b> Tag HTML untuk menampilkan gambar.</li> <li>• <b>src='\$targetfile':</b> Atribut src (sumber) gambar diatur ke nilai variabel \$targetfile. Variabel ini berisi path lengkap ke file yang baru saja diunggah (contoh: uploads/namafile.jpg).</li> <li>• <b>width='200':</b> Atribut width mengatur lebar gambar menjadi 200 piksel, sesuai permintaan. Dengan hanya mengatur lebar, tinggi gambar akan disesuaikan secara otomatis oleh browser untuk menjaga rasio aspek (proporsi) asli gambar.</li> </ul>

10	<p>Selanjutnya, ubah isi dari file <b>upload.php</b> dengan kode berikut.</p> <pre> &lt;?php if(isset(\$_POST["submit"])){     \$targetdir = "uploads/"; //Direktori tujuan untuk menyimpan file     \$targetfile = \$targetdir . basename(\$_FILES["myfile"]["name"]);     \$fileType = strtolower(pathinfo(\$targetfile, PATHINFO_EXTENSION));      \$allowedExtensions = array("txt", "pdf", "doc", "docx");     \$maxsize = 3*1024*1024;      if (in_array(\$fileType, \$allowedExtensions) &amp;&amp; \$_FILES["myfile"]["size"]&lt;=\$maxsize)     {         if(move_uploaded_file(\$_FILES["myfile"]["tmp_name"], \$targetfile)){             echo "File berhasil diunggah";         }         else{             echo "Gagal mengunggah file.";         }     }     else{         echo "File tidak valid atau melebihi ukuran maksimum yang diizinkan";     } } ?&gt; </pre>
11	<p>Simpan file, buka browser dan jalankan <b>localhost/dasarWeb/form_upload.php</b>. Pilih sebuah file dengan ekstensi .txt, .pdf, .doc, atau .docx yang berukuran lebih dari 5 MB. Klik tombol Submit. Amati yang terjadi dan catat pemahaman Anda.</p> <p>(Pertanyaan No. 6)</p> <p><b>File tidak valid atau melebihi ukuran maksimum yang diizinkan</b></p> <p>Tidak bisa karena melebihi ketentuan.</p>
12	<p>Selanjutnya jalankan kembali <b>localhost/dasarWeb/form_upload.php</b>. Pilih sebuah file dengan ekstensi .txt, .pdf, .doc, atau .docx yang berukuran kurang dari 3 MB. Klik tombol Submit. Amati yang terjadi dan catat pemahaman Anda.</p> <p>(Pertanyaan No. 7)</p> <p><b>File berhasil diunggah</b></p> <p>Berhasil karena sesuai ketentuan.</p>

## Praktikum Bagian 2. Multi Upload File

Langkah	Keterangan
1	Buatlah file baru bernama <b>form_multiupload.php</b> .

2	<p>Ketikkan ke dalam <b>form_multiupload.php</b> kode berikut.</p> <pre> &lt;!DOCTYPE html&gt; &lt;html&gt; &lt;head&gt;     &lt;title&gt;Multiupload Dokumen&lt;/title&gt; &lt;/head&gt; &lt;body&gt;     &lt;h2&gt;Unggah Dokumen&lt;/h2&gt;     &lt;form action="proses_upload.php" method="post" enctype="multipart/form-data"&gt;         &lt;input type="file" name="files[]" multiple="multiple" accept=".pdf, .doc, .docx"&gt;         &lt;input type="submit" value="Unggah" /&gt;     &lt;/form&gt; &lt;/body&gt; &lt;/html&gt; </pre>
3	<p>Buatlah file baru bernama <b>proses_upload.php</b>. Ketikkan kode berikut.</p> <pre> &lt;?php // Lokasi penyimpanan file yang diunggah \$targetDirectory = "documents/";  // Periksa apakah direktori penyimpanan ada, jika tidak maka buat if (!file_exists(\$targetDirectory)) {     mkdir(\$targetDirectory, 0777, true); }  if (\$_FILES['files']['name'][0]) {     \$totalFiles = count(\$_FILES['files']['name']);      // Loop melalui semua file yang diunggah     for (\$i = 0; \$i &lt; \$totalFiles; \$i++) {         \$fileName = \$_FILES['files']['name'][\$i];         \$targetFile = \$targetDirectory . \$fileName;          // Pindahkan file yang diunggah ke direktori penyimpanan         if (move_uploaded_file(\$_FILES['files']['tmp_name'][\$i], \$targetFile)) {             echo "File \$fileName berhasil diunggah.&lt;br&gt;";         } else {             echo "Gagal mengunggah file \$fileName.&lt;br&gt;";         }     } } else {     echo "Tidak ada file yang diunggah."; } </pre>
4	<p>Simpan file, buka browser dan jalankan <b>localhost/dasarWeb/form_multiupload.php</b>. Pilih beberapa file sekaligus untuk diunggah. Apa yang anda pahami dari script pada file tersebut? Catat pemahaman anda. <b>(Pertanyaan No. 8)</b></p> <p>File 04_TI 2A_Athaulla Hafizh_Manajemen Waktu.docx berhasil diunggah. File 04_TI 2A_Athaulla Hafizh_Manajemen Waktu.pdf berhasil diunggah.</p> <p>Terdapat pengecekan pada folder document exist atau tidak, jika belum ada maka file document otomatis terbuat dan dokumen yang diunggah akan masuk ke dalam folder.</p>
5	<p>Buat seperti langkah 3 dengan multi upload khusus gambar. Screen shoot perubahan kode dan berikan penjelasan kode tersebut. <b>(Pertanyaan No. 9)</b></p>



```

<?php
// Lokasi penyimpanan file diubah ke 'images/'
$targetDirectory = "images/";

// Periksa apakah direktori penyimpanan ada, jika tidak maka buat
if (!file_exists($targetDirectory)) {
    mkdir($targetDirectory, 0777, true);
}

// === TAMBAHAN: Array untuk ekstensi yang diizinkan ===
$allowedExtensions = array("jpg", "jpeg", "png", "gif");

if (isset($_FILES['files']) && !empty(array_filter($_FILES['files']['name']))) {
    $totalFiles = count($_FILES['files']['name']);

    // Loop melalui semua file yang diunggah
    for ($i = 0; $i < $totalFiles; $i++) {
        $fileName = $_FILES['files']['name'][$i];
        $targetFile = $targetDirectory . $fileName;

        // === TAMBAHAN: Cek ekstensi file ===
        $fileType = strtolower(pathinfo($targetFile, PATHINFO_EXTENSION));

        // Cek apakah ekstensi ada di dalam array $allowedExtensions
        if (in_array($fileType, $allowedExtensions)) {
            // Pindahkan file yang diunggah ke direktori penyimpanan
            if (move_uploaded_file($_FILES['files']['tmp_name'][$i], $targetFile)) {
                echo "File $fileName berhasil diunggah.<br>";
            } else {
                echo "Gagal mengunggah file $fileName.<br>";
            }
        } else {
            // Tampilkan pesan error jika bukan gambar
            echo "File $fileName tidak valid. Hanya ekstensi .jpg, .jpeg, .png, .gif yang diizinkan.<br>";
        }
    }
} else {
    echo "Tidak ada file yang diunggah.";
}
?>

```

Perubahan utama dari kode proses\_upload.php sebelumnya adalah penambahan **validasi tipe file**:

1. **\$targetDirectory = "images/";**: Direktori tujuan diubah menjadi images/ agar lebih terorganisir (opsional, tapi disarankan).
2. **\$allowedExtensions = ...**: Sebuah *array* baru dibuat untuk menyimpan daftar ekstensi file gambar apa saja yang boleh diunggah.
3. **\$fileType = strtolower(...)**: Di dalam *loop*, kita mengambil ekstensi dari setiap file (misalnya .jpg) menggunakan pathinfo.
4. **if (in\_array(\$fileType, \$allowedExtensions))**: Ini adalah logika utamanya. Kode ini memeriksa apakah ekstensi file (\$fileType) ada di dalam daftar yang diizinkan (\$allowedExtensions).
5. **move\_uploaded\_file(...)**: File hanya akan dipindahkan jika lolos pengecekan in\_array.
6. **else { ... }**: Jika file tidak lolos pengecekan (bukan gambar), maka akan ditampilkan pesan error bahwa file tersebut tidak valid.

### Praktikum Bagian 3. Upload File dengan PHP dan JQuery

Langkah	Keterangan
1	Buatlah file baru bernama <b>form_upload_ajax.php</b>

2	<p>Ketikkan kode berikut pada <b>form_upload_ajax.php</b></p> <pre data-bbox="323 219 1433 913">&lt;!DOCTYPE html&gt; &lt;html&gt;  &lt;head&gt;   &lt;title&gt;Unggah File Dokumen&lt;/title&gt; &lt;/head&gt;  &lt;body&gt;   &lt;form id="upload-form" action="upload_ajax.php" method="post" enctype= "multipart/form-data"&gt;     &lt;input type="file" name="file" id="file"&gt;     &lt;input type="submit" name="submit" value="Unggah"&gt;   &lt;/form&gt;   &lt;div id="status"&gt;&lt;/div&gt;    &lt;script src="https://code.jquery.com/jquery-3.6.0.min.js"&gt;&lt;/script&gt;   &lt;script src="upload.js"&gt;&lt;/script&gt; &lt;/body&gt;  &lt;/html&gt;</pre>
3	<p>Buatlah file baru bernama <b>upload.js</b>. Tuliskan kode berikut.</p>



	<pre> \$(document).ready(function(){     \$('#upload-form').submit(function(e){         e.preventDefault();          var formData = new FormData(this);          \$.ajax({             type: 'POST',             url: 'upload_ajax.php',             data: formData,             cache: false,             contentType: false,             processData: false,             success: function(response){                 \$('#status').html(response);             },             error: function(){                 \$('#status').html('Terjadi kesalahan saat mengunggah file.');<!--            }         });     }); }); </pre--> </pre>
4	<p>Buatlah file baru bernama <b>upload_ajax.php</b>. Tuliskan kode berikut.</p> <pre> &lt;?php if (isset(\$_FILES['file'])) {     \$errors = array();     \$file_name = \$_FILES['file']['name'];     \$file_size = \$_FILES['file']['size'];     \$file_tmp = \$_FILES['file']['tmp_name'];     \$file_type = \$_FILES['file']['type'];     @ \$file_ext = strtolower(" . end(explode('.', \$_FILES['file']['name'])) . "");     \$extensions = array("pdf", "doc", "docx", "txt");      if (in_array(\$file_ext, \$extensions) === false) {         \$errors[] = "Ekstensi file yang diizinkan adalah PDF, DOC, DOCX, atau TXT.";     }      if (\$file_size &gt; 2097152) {         \$errors[] = 'Ukuran file tidak boleh lebih dari 2 MB';     }      if (empty(\$errors) == true) {         move_uploaded_file(\$file_tmp, "documents/" . \$file_name);         echo "File berhasil diunggah.";     } else {         echo implode(" ", \$errors);     } } </pre>
5	<p>Simpan file tersebut, kemudian buka browser dan jalankan <b>localhost/dasarWeb/form_upload_ajax.php</b>.</p> <ul style="list-style-type: none"> <li>- Upload file berupa gambar.</li> <li>- Upload file PDF yang berukuran &gt; 4 MB.</li> <li>- Upload file .docx dengan ukuran &lt; 2 MB.</li> </ul>

	<p>Amati apa yang terjadi dan jelaskan pemahaman anda. (Pertanyaan No. 10)</p> <ul style="list-style-type: none"> <li>- file gambar tidak diizinkan karena format gambar tidak dicantumkan pada source kode</li> <li>- file pdf sebenarnya tervalidasi benar tetapi karena ukuran &lt; 2 mb maka gagal upload</li> <li>- .docx paling memenuhi validasi sesuai source kode</li> </ul>
6	<p>Ubah kode menjadi agar dapat dilakukan multi upload file khusus gambar. Screenshoot perubahan kode dan jelaskan kode tersebut. (Pertanyaan No. 11)</p> <pre> &lt;?php // Cek apakah array 'files' dikirim if (isset(\$_FILES['files'])) {     // Tentukan ekstensi gambar yang diizinkan     \$allowedExtensions = array("jpeg", "jpg", "png", "gif");     // Tentukan ukuran file maksimum (misal: 5MB)     \$max_size = 5 * 1024 * 1024;     \$errors = array();      // Loop melalui setiap file yang diunggah menggunakan foreach     foreach (\$_FILES['files']['tmp_name'] as \$key =&gt; \$tmp_name) {         \$file_name = \$_FILES['files'][\$key]['name'];         \$file_size = \$_FILES['files'][\$key]['size'];         // Ambil ekstensi file         \$file_ext = strtolower(pathinfo(\$file_name, PATHINFO_EXTENSION));          // Validasi ekstensi         if (in_array(\$file_ext, \$allowedExtensions) === false) {             \$errors[] = "Ekstensi file \$file_name tidak diizinkan (hanya .jpg, .jpeg, .png, .gif).";         }          // Validasi ukuran file         if (\$file_size &gt; \$max_size) {             \$errors[] = "Ukuran file \$file_name tidak boleh lebih dari 5 MB.";         }          // Jika tidak ada error untuk file ini, pindahkan file         if (empty(\$errors)) {             // Pastikan folder 'images' ada             if (!file_exists('images')) {                 mkdir('images', 0777, true);             }             move_uploaded_file(\$tmp_name, "images/" . \$file_name);             echo "File \$file_name berhasil diunggah.&lt;br&gt;";         }     }      // Tampilkan semua error yang terkumpul     if (!empty(\$errors)) {         echo implode("&lt;br&gt;", \$errors);     } } ?&gt; </pre> <ul style="list-style-type: none"> <li>- <b>foreach (\$_FILES['files']['tmp_name'] as \$key =&gt; \$tmp_name):</b> Kode ini tidak lagi memproses satu file. Sebaliknya, ia menggunakan <i>looping foreach</i> untuk memproses setiap file yang ada di dalam array \$_FILES['files'].</li> <li>- <b>Validasi di dalam Loop:</b> Pengecekan ekstensi (menggunakan \$allowedExtensions yang kini berisi tipe gambar) dan ukuran file dilakukan <b>untuk setiap file secara individual</b> di dalam <i>loop</i>.</li> <li>- <b>move_uploaded_file(...):</b> Fungsi ini juga dijalankan di dalam <i>loop</i>, memindahkan setiap file yang valid satu per satu.</li> <li>- <b>Pesan Status per File:</b> echo dijalankan di dalam <i>loop</i> agar pengguna mendapatkan umpan balik untuk setiap file yang berhasil diunggah atau gagal.</li> </ul>

## Praktikum Bagian 4. Menghias Upload File

Langkah	Keterangan
1	<p>Modifikasi kode pada Praktikum Bagian 3 yaitu file <code>form_upload_ajax.php</code></p> <pre>&lt;!DOCTYPE html&gt; &lt;html&gt;  &lt;head&gt;   &lt;link rel="stylesheet" type="text/css" href="upload.css"&gt;   &lt;title&gt;Unggah File Dokumen&lt;/title&gt; &lt;/head&gt;  &lt;body&gt;   &lt;div class="upload-form-container"&gt;     &lt;h2&gt;Unggah File Dokumen&lt;/h2&gt;     &lt;form id="upload-form" action="upload.php" method="post" enctype= "multipart/form-data"&gt;       &lt;div class="file-input-container"&gt;         &lt;input type="file" name="file" id="file" class="file-input"&gt;         &lt;label for="file" class="file-label"&gt;Pilih File&lt;/label&gt;       &lt;/div&gt;       &lt;button type="submit" name="submit" class="upload-button" id="upload-button" disabled&gt;Unggah&lt;/button&gt;     &lt;/form&gt;     &lt;div id="status" class="upload-status"&gt;&lt;/div&gt;   &lt;/div&gt;    &lt;script src="https://code.jquery.com/jquery-3.6.0.min.js"&gt;&lt;/script&gt;   &lt;script src="upload.js"&gt;&lt;/script&gt; &lt;/body&gt;  &lt;/html&gt;</pre>
2	Modifikasi juga file <code>upload.js</code> seperti kode berikut.

```

$(document).ready(function(){
    $('#file').change(function(){
        if (this.files.length > 0) {
            $('#upload-button').prop('disabled', false).css('opacity', 1);
        } else {
            $('#upload-button').prop('disabled', true).css('opacity', 0.5);
        }
    });

    $('#upload-form').submit(function(e){
        e.preventDefault();

        var formData = new FormData(this);

        $.ajax({
            type: 'POST',
            url: 'upload_ajax.php',
            data: formData,
            cache: false,
            contentType: false,
            processData: false,
            success: function(response){
                $('#status').html(response);
            },
            error: function(){
                $('#status').html('Terjadi kesalahan saat mengunggah file.');
```

3

Buat file baru bernama upload.css. Ketikkan kode berikut..

```

.upload-form-container {
    max-width: 400px;
    margin: 0 auto;
    padding: 20px;
    border: 1px solid #ccc;
    border-radius: 5px;
    text-align: center;
}

h2 {
    margin: 0;
    font-size: 24px;
    color: #333;
}

.file-input-container {
    display: flex;
    justify-content: center;
    align-items: center;
    margin: 20px 0;
}

.file-input {
    display: none;
}
```

```
.file-label {
    background: #3498db;
    color: #fff;
    padding: 10px 20px;
    border-radius: 5px;
    cursor: pointer;
}

.upload-button {
    background: #2ecc71;
    color: #fff;
    padding: 10px 20px;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    opacity: 0.5; /* Opacity to make it appear faded */
}

.upload-button:disabled {
    background: #ccc; /* Change color when disabled */
    cursor: not-allowed; /* Change cursor style when disabled */
}

.upload-status {
    margin-top: 20px;
    font-weight: bold;
}
```

4

Simpan file tersebut. Buka browser dan jalankan [localhost/dasarWeb/form\\_upload\\_ajax.php](localhost/dasarWeb/form_upload_ajax.php).  
Apa yang anda pahami dari kode program di atas? Catat pemahaman anda.  
(Pertanyaan No. 12)

## Unggah File Dokumen

Pilih File

Unggah

**File berhasil diunggah.**

Penambahan style

### Pengenalan Cookies

*Cookies* adalah sebuah nilai yang dikirimkan dan ditanamkan *server* pada komputer *client*. *Cookies* berupa berkas berukuran kecil umumnya kurang dari 100 bytes, sehingga tidak akan mempengaruhi

kecepatan *browsing*. *Cookies* menampung informasi berkaitan dengan pengguna. *Cookies* berguna untuk memudahkan pengguna, seperti "mengingat" pengguna setiap kali mengunjungi website yang sama.

Informasi-informasi yang disimpan dalam *cookies* adalah data-data sederhana tentang kebiasaan pengguna. Contoh sederhananya saat pengguna mengunjungi sebuah website yang menawarkan tampilan dengan versi bahasa Inggris dan bahasa Indonesia. Apabila pengguna memilih bahasa Indonesia, *cookies* akan secara otomatis menyimpan konfigurasi tersebut. Konfigurasi tersebut akan diingat, sehingga apabila pengguna mengunjungi kembali website tersebut, pengguna tidak perlu memilih jenis bahasa lagi.



*Cookies* mempunyai umur, maksudnya setiap data yang disimpan dalam komputer pengguna suatu saat bisa hilang atau musnah. Berdasarkan umurnya *cookies* dibagi menjadi dua jenis, yaitu *session cookies* dan *persistent cookies*. *Session cookies* adalah jenis *cookies* yang tersimpan sementara dan akan hilang saat pengguna menutup *browser*. *Session cookies* ini biasanya digunakan pada fitur “keranjang belanja” di situs belanja online. *Persistent cookies* adalah jenis *cookies* yang tersimpan secara permanen di dalam komputer sampai pengguna memutuskan untuk menghapusnya. *Persistent cookies* umumnya digunakan pada fitur “remember me” saat login.

PHP memiliki fungsi yang dapat digunakan untuk membuat dan mengambil nilai *cookies*. *Cookies* harus dideklarasikan sebelum halaman ditampilkan, yang artinya dituliskan sebelum tag `<html>`. *Cookies* dibuat menggunakan fungsi `setcookie()`. Berikut adalah sintaks dasar fungsi `setcookie()`:

```
<?php
    setcookie(name, value, expire);
?>
```

Keterangan:

- **name**, adalah nama *cookies* atau variabel yang menjadi pengenal *cookies*
- **value**, berisi nilai yang tersimpan dalam *cookies*
- **expire**, adalah jangka waktu *cookies* akan disimpan di dalam komputer

Setelah mengetahui cara membuat *cookies*, selanjutnya nilai *cookies* yang telah di buat dapat diambil dengan menggunakan variabel `$_COOKIE`. Berikut ini adalah sintaks untuk mengambil nilai *cookies*:

```
<?php
    $_COOKIE['user'];
?>
```

## Praktikum Bagian 5. Membuat *Cookies*

Langkah	Keterangan
1	Buat file baru dengan nama <b>cookiesCreate.php</b> , kemudian ketikkan kode berikut. <pre>1 &lt;?php 2     setcookie("user", "Polinema", time()+3600); 3 ?&gt;</pre>
2	Buat file baru dengan nama <b>cookiesCall.php</b> , kemudian ketikkan kode berikut. <pre>1 &lt;?php 2     echo \$_COOKIE['user']; 3 ?&gt;</pre>
3	Buka <i>browser</i> dan jalankan kode program pada langkah ke-2 dengan mengetikkan <b>localhost/dasarWeb/cookiesCall.php</b>
4	Amati dan jelaskan hasil pengamatanmu (Pertanyaan No. 13)  <b>Warning:</b> Undefined array key "user" in C:\laragon\www\pertemuan08\dasarWeb\cookiesCall.php on line 2

	Karena cookies belum pernah dibuat sebelumnya.
5	Buka <i>browser</i> dan jalankan kode program langkah ke-1 dengan mengetikkan <a href="localhost/dasarWeb/cookiesCreate.php">localhost/dasarWeb/cookiesCreate.php</a>
6	Ulangi langkah ke-3.
7	Amati dan jelaskan hasil yang ditampilkan (Pertanyaan No. 14)  <b>Polinema</b>  Output value keluar karena sudah di create.
8	<i>Restart</i> komputer Anda.
9	Setelah komputer menyala, nyalakan kembali Apache pada laragon.
10	Buka <i>browser</i> yang sama dengan sebelumnya kemudian ulangi langkah ke-3.
11	Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 15)  <b>Polinema</b>  Tidak terjadi apa apa karena time kadaluwarsa 1 jam.

## Praktikum Bagian 6. Menghapus Nilai *Cookies*

Pada praktikum ini akan dibahas mengenai cara menghapus nilai *cookies*. Jika pada praktikum bagian sebelumnya *cookies* di set dengan waktu *expire* `time()+3600`, maka untuk menghapus nilai *cookies* adalah dengan sebagai berikut:

Langkah	Keterangan
1	Buat file baru dengan nama <a href="#">cookiesDel.php</a> , kemudian ketikkan kode berikut. <pre> 1 &lt;?php 2     setcookie("user", "Polinema", time()-3600); 3 ?&gt; </pre>
2	Buka <i>browser</i> dan jalankan kode program dengan mengetikkan <a href="localhost/dasarWeb/cookiesDel.php">localhost/dasarWeb/cookiesDel.php</a>
3	Buka <i>browser</i> dan jalankan kode program dari praktikum bagian 5 dengan mengetikkan <a href="localhost/dasarWeb/cookiesCall.php">localhost/dasarWeb/cookiesCall.php</a>
4	Amati dan jelaskan hasil yang dihasilkan dari langkah 2 dan 3, kemudian beri kesimpulan. (Pertanyaan No. 16)  <b>Warning:</b> Undefined array key "user" in C:\laragon\www\pertemuan08\dasarWeb\cookiesCall.php on line 2  Cookies hilang karena di hapus.

## **Praktikum Bagian 7. Penerapan *Cookies* pada Fitur Keranjang Belanja**

Salah satu contoh pemanfaatan *cookies* adalah pada fitur “keranjang belanja” pada aplikasi web toko online. Keranjang belanja berisi barang-barang yang akan dibeli oleh user. *Cookies* digunakan untuk

mengingat jumlah barang yang dipilih user. Berikut adalah contoh penggunaan *cookies* pada fitur keranjang belanja:

Langkah	Keterangan
1	<p>Buat file baru dengan nama <b>formBeli.html</b>, kemudian ketikkan kode berikut.</p> <pre> 1 &lt;html&gt; 2 &lt;head&gt; 3 &lt;/head&gt; 4 &lt;body&gt; 5     &lt;form action="prosesBeli.php" method="POST"&gt; 6         &lt;p&gt; Jumlah Novel yang dibeli : 7         &lt;input type="text" name="beliNovel" value= "0" size="2"&gt; &lt;/p&gt; 8         &lt;p&gt; Jumlah Buku Teks yang dibeli : 9         &lt;input type="text" name="beliBuku" value= "0" size="2"&gt; &lt;/p&gt; 10        &lt;input type="submit"&gt; 11    &lt;/form&gt; 12 &lt;/body&gt; 13 &lt;/html&gt;</pre>
2	<p>Buat file baru dengan nama <b>prosesBeli.php</b>, kemudian ketikkan kode berikut.</p> <pre> 1 &lt;?php 2     if(isset(\$_POST["beliNovel"]) &amp;&amp; isset(\$_POST["beliBuku"])){ 3         setcookie("beliNovel", \$_POST["beliNovel"]); 4         setcookie("beliBuku", \$_POST["beliBuku"]); 5         header("location:keranjangBelanja.php"); 6     } 7     ?&gt;</pre>
3	<p>Buat file baru dengan nama <b>keranjangBelanja.php</b>, kemudian ketikkan kode berikut.</p> <pre> 1 &lt;html&gt; 2 &lt;head&gt; 3 &lt;/head&gt; 4 &lt;body&gt; 5     &lt;h2&gt; Keranjang Belanja &lt;/h2&gt; 6 7     &lt;?php 8         \$beliNovel=\$_COOKIE['beliNovel']; 9         \$beliBuku = \$_COOKIE['beliBuku']; 10 11        echo "Jumlah Novel:" . \$beliNovel . "&lt;br&gt;"; 12        echo "Jumlah Buku :" . \$beliBuku ; 13    ?&gt; 14 &lt;/body&gt; 15 &lt;/html&gt;</pre>
4	<p>Buka <i>browser</i> dan jalankan kode program langkah ke-3 dengan mengetikkan <b>localhost/dasarWeb/keranjangBelanja.php</b></p>
5	<p>Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 17)</p> <p><b>Keranjang Belanja</b></p> <p><b>Warning:</b> Undefined array key "beliNovel" in C:\laragon\www\pertemuan08\dasarWeb\keranjangBelanja.php on line 7</p> <p><b>Warning:</b> Undefined array key "beliBuku" in C:\laragon\www\pertemuan08\dasarWeb\keranjangBelanja.php on line 8</p> <p>Jumlah Novel: Jumlah Buku :</p>

	Belum ada data karena yang diinput sehingga tidak ada data yang ditampilkan.
6	Jalankan kode program langkah ke-1 dengan mengetikkan <a href="localhost/dasarWeb/formBeli.html">localhost/dasarWeb/formBeli.html</a>
7	Isikan jumlah novel dan buku teks yang akan dibeli kemudian klik tombol “submit”.
8	<p>Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 18)</p> <hr/> <p>Jumlah Novel yang dibeli: <input type="text" value="3"/></p> <p>Jumlah Buku Teks yang dibeli : <input type="text" value="4"/></p> <p><input type="button" value="Submit"/></p> <p>Hasil :</p> <p><b>Keranjang Belanja</b></p> <p>Jumlah Novel: 3 Jumlah Buku: 4</p> <p>User input berapa jumlah yang ingin di beli lalu diarahkan ke page prosesBeli.php.</p>
9	Tutup <i>browser</i> kemudian buka kembali <i>browser</i> kemudian jalankan kembali kode program langkah ke-3 dengan mengetikkan <a href="localhost/dasarWeb/keranjangBelanja.php">localhost/dasarWeb/keranjangBelanja.php</a>
10	<p>Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 19)</p> <p><b>Keranjang Belanja</b></p> <p><b>Warning:</b> Undefined array key "beliNovel" in C:\laragon\www\pertemuan08\dasarWeb\keranjangBelanja.php on line 7</p> <p><b>Warning:</b> Undefined array key "beliBuku" in C:\laragon\www\pertemuan08\dasarWeb\keranjangBelanja.php on line 8</p> <p>Jumlah Novel: Jumlah Buku:</p> <p>Data hilang karena kode diatas adalah kode untuk cookies sesi (tanpa parameter waktu).</p>

## Pengenalan Session

*Session* adalah salah satu fasilitas yang ada pada PHP yang digunakan untuk menyimpan data sementara ke dalam variabel, sehingga data tadi dapat diakses oleh pengguna selama variabel *session* tersebut tidak dikosongkan atau dihilangkan. *Session* pada prinsipnya hampir sama dengan *cookies*, yang membedakan adalah posisi penyimpanan data. Jika *cookies* menyimpan data pada *client*, data *session* akan disimpan pada sisi *server*. Jadi *session* relatif lebih aman digunakan untuk menyimpan variabel nilai yang bersifat rahasia seperti *username* dan *password* pada saat login. Contoh penggunaan *session* diantaranya untuk menyimpan informasi login yang berlaku hanya dalam satu sesi dan menyimpan catatan pemesanan barang dalam sistem *e-commerce*/transaksi online. Tabel berikut menjelaskan perbandingan *cookies* dan *session*:

### Perbandingan *Cookies* dan *Session*

Jenis	Menyimpan Informasi	Lokasi Penyimpanan	Dapat dipanggil	Bersifat Global	Dapat dipanggil Lain Waktu	Dibatasi Browser	Dapat diedit User
<i>Cookies</i>	Ya	<i>Client</i>	Ya	Ya	Ya	Ya	Ya
<i>Session</i>	Ya	<i>Server</i>	Ya	Ya	Tidak	Tidak	Tidak

Untuk membuat *session* dalam PHP digunakan fungsi `session_start()`. Berikut adalah sintaks dasar untuk memulai *session*:

```
<?php
    session_start();
?>
```

Setelah *session* dimulai, maka variabel *session* sudah dapat mulai digunakan. Variabel yang dimaksud adalah PHP global variable `$_SESSION`. Berikut adalah sintaks menggunakan variable global `$_SESSION`:



```
<?php
    $_SESSION['namaVariabel'];
?>
```

## Praktikum Bagian 8. Membuat Session

Langkah	Keterangan
1	<p>Buat file baru dengan nama <b>sessionCreate.php</b>, kemudian ketikkan kode berikut.</p> <pre> 1  &lt;?php 2      session_start(); 3  ?&gt; 4 5  &lt;!DOCTYPE html&gt; 6  &lt;html&gt; 7      &lt;body&gt; 8          &lt;?php 9              \$_SESSION["favcolor"] = "green"; 10             \$_SESSION["favanimal"] = "cat"; 11             echo "Session variables are set."; 12         ?&gt; 13     &lt;/body&gt; 14 &lt;/html&gt;</pre>
2	<p>Buat file baru dengan nama <b>sessionCall.php</b>, kemudian ketikkan kode berikut.</p> <pre> 1  &lt;?php 2      session_start(); 3  ?&gt; 4  &lt;!DOCTYPE html&gt; 5  &lt;html&gt; 6      &lt;body&gt; 7          &lt;?php 8              echo "Favorite color is " . \$_SESSION["favcolor"] . "&lt;br&gt;"; 9              echo "Favorite animal is " . \$_SESSION["favanimal"] . "."; 10         ?&gt; 11     &lt;/body&gt; 12 &lt;/html&gt;</pre>
3	Buka <i>browser</i> dan jalankan kode program pada langkah ke-2 dengan mengetikkan <b>localhost/dasarWeb/sessionCall.php</b>
4	Buka <i>browser</i> dan jalankan kode program langkah ke-1 dengan mengetikkan <b>localhost/dasarWeb/sessionCreate.php</b>
5	Ulangi langkah ke-3
6	<p>Amati dan jelaskan hasil yang ditampilkan (Pertanyaan No. 20)</p> <p><b>Favorite color is green.</b> <b>Favorite animal is cat.</b></p> <ul style="list-style-type: none"> <li>- Saat sessionCall.php dijalankan pertama kali (Langkah 3), terjadi error "Undefined index" karena session favcolor dan favanimal belum ada.</li> <li>- sessionCreate.php (Langkah 4) dijalankan, yang</li> </ul>

	<p>memulai session dan menyimpan dua variabel di server.</p> <ul style="list-style-type: none"><li>- Saat sessionCall.php dijalankan lagi (Langkah 5), halaman akan menampilkan: Favorite color is green. Favorite animal is cat.</li></ul>
--	---

## Praktikum Bagian 9. Menghapus Nilai *Session*

PHP menyediakan fungsi `session_destroy()` yang dapat digunakan untuk menghapus *session*.

Langkah	Keterangan
1	<p>Buat file baru dengan nama <code>sessionDel.php</code>, kemudian ketikkan kode berikut.</p> <pre>1 &lt;?php 2     session_start(); 3 &gt;? 4 5 &lt;!DOCTYPE html&gt; 6 &lt;html&gt; 7     &lt;body&gt; 8         &lt;?php 9             session_unset(); 10            session_destroy(); 11 12            echo "All session variables are now removed, and the session is destroyed." 13        &gt;? 14    &lt;/body&gt; 15 &lt;/html&gt;</pre>
2	<p>Buka <i>browser</i> dan jalankan kode program dengan mengetikkan <code>localhost/dasarWeb/sessionDel.php</code></p>
3	<p>Buka <i>browser</i> dan jalankan kode program dari praktikum bagian 8 dengan mengetikkan <code>localhost/dasarWeb/sessionCall.php</code></p>
4	<p>Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 21)</p> <p><b>Warning:</b> Undefined array key "favcolor" in C:\laragon\www\pertemuan08\dasarWeb\sessionCall.php on line 8 Favorite color is .</p> <p><b>Warning:</b> Undefined array key "favanimal" in C:\laragon\www\pertemuan08\dasarWeb\sessionCall.php on line 9 Favorite animal is .</p> <ul style="list-style-type: none"><li>- <code>sessionDel.php</code> (Langkah 2) akan menampilkan pesan "All session variables are now removed, and the session is destroyed."</li><li>- Saat <code>sessionCall.php</code> dijalankan setelahnya (Langkah 3), akan muncul error "Undefined index" lagi. Ini karena <code>session_unset()</code> dan <code>session_destroy()</code> telah menghapus semua data <i>session</i> dari server.</li></ul>

## Praktikum Bagian 10. Penerapan *Session* pada Fitur Login

Langkah	Keterangan
1	<p>Buat file baru dengan nama <code>sessionLoginForm.html</code>, kemudian ketikkan kode berikut</p>

```

<html>
  <head>
    <title>File Upload</title>
  </head>
  <body>
    <form action="sessionLoginProcess.php" method="POST">
      <table>
        <tr>
          <td>Username</td>
          <td><input type="text" name="username" size="20"></td>
        </tr>
        <tr>
          <td>Password</td>
          <td><input type="password" name="password" size="20"></td>
        </tr>
        <tr>
          <td>&nbsp;</td>
          <td><input type="submit" name="login" value="Login"></td>
        </tr>
      </table>
    </form>
  </body>
</html>

```

2. Buat file dengan nama **sessionLoginProcess.php**, kemudian ketikkan kode berikut.

```

<?php
$username = $_POST['username'];
$password = $_POST['password'];

if($username=="admin" && $password=="1234"){
    session_start();
    $_SESSION["username"] = $username;
    $_SESSION["status"] = 'login';
    echo "Anda berhasil login. Silahkan menuju <a href='homeSession.php'>Halaman Home</a>";
}
else{
    echo "Gagal login. Silahkan login lagi <a href='sessionLoginForm.html'>Halaman Login</a>";
}
?>

```

3. Buat file dengan nama **homeSession.php**, kemudian ketikkan kode berikut.

```

<html>
  <head>
    <title>Halaman Home</title>
  </head>
  <body>
    <?php
      session_start();
      if($_SESSION['status']=='login'){
        echo "Selamat datang " . $_SESSION['username'];
      }
      <br><a href="sessionLogout.php">Logout</a>
    <?php
      }
      else{
        echo "Anda belum login, silahkan";
      }
      <a href="sessionLoginForm.html">Login</a>
    <?php
      }
    </body>
</html>

```

4	<p>Buat file dengan nama <code>sessionLogout.php</code>, kemudian ketikkan kode berikut.</p> <pre>&lt;?php     session_start();     session_destroy();      echo "Anda berhasil logout"; ?&gt;</pre>
5	<p>Buka <i>browser</i> dan jalankan kode program dengan mengetikkan <code>localhost/dasarWeb/sessionLoginForm.html</code></p>
6	<p>Login menggunakan username email Anda dan password “0000”.</p>
7	<p>Amati dan jelaskan hasil yang ditampilkan (Pertanyaan No. 22)</p> <hr/> <p>Gagal login. Silahkan login lagi <a href="#">Halaman Login</a></p> <p>Muncul pesan "Gagal login. Silahkan login lagi" dengan link untuk kembali ke halaman login. Ini karena username dan password yang dimasukkan tidak cocok dengan kondisi <code>if (\$username=="admin" &amp;&amp; \$password=="1234")</code>.</p>
8	<p>Jalankan kembali kode program dengan mengetikkan <code>localhost/dasarWeb/sessionLoginForm.html</code> Login menggunakan username “admin” dan password “1234”.</p>
9	<p>Amati dan jelaskan hasil yang ditampilkan (Pertanyaan No. 23)</p> <hr/> <p>Anda berhasil login. Silahkan menuju <a href="#">Halaman Home</a></p> <hr/> <p>Selamat datang admin <a href="#">Logout</a></p> <p>Muncul pesan "Anda berhasil login. Silahkan menuju Halaman Home". Saat link Halaman Home diklik, akan tampil pesan "Selamat datang admin" beserta link "Logout". Ini karena kredensial benar dan skrip membuat session status dan username.</p>
10	<p>Jelaskan urutan proses dari login hingga logout (sebutkan juga urutan file yang diproses) (Pertanyaan No. 24)</p> <ul style="list-style-type: none"> <li>- <b>Login:</b> Pengguna membuka <code>sessionLoginForm.html</code> dan mengirim data.</li> <li>- <b>Proses:</b> <code>sessionLoginProcess.php</code> menerima data. Jika benar (admin/1234), ia memulai session (<code>session_start()</code>) dan menyimpan <code>\$_SESSION["username"]</code> dan <code>\$_SESSION["status"] = 'login'</code>.</li> <li>- <b>Halaman Home:</b> Pengguna diarahkan ke <code>homeSession.php</code>. File ini memeriksa <code>if(\$_SESSION['status']=='login')</code>. Karena statusnya 'login', halaman menampilkan pesan selamat datang.</li> <li>- <b>Logout:</b> Pengguna mengklik link logout ke <code>sessionLogout.php</code>. File ini menghancurkan session (<code>session_destroy()</code>).</li> </ul>