

JOBSHEET 8

PHP – FORM UPLOAD



Dibuat oleh: Silmy Maulia Dewi

Kelas: SIB 2E

NIM: 2241760090

D4 Sistem Informasi Bisnis

Teknologi Informasi

Politeknik Negeri Malang

2024



Topik

- Konsep Form Upload dengan PHP dan JQuery

Tujuan

Mahasiswa diharapkan dapat:

1. Mahasiswa mampu membuat form upload menggunakan PHP
2. Mahasiswa mampu membuat form upload menggunakan jQuery

Perhatian

Jobsheet ini harus dikerjakan step-by-step sesuai langkah-langkah praktikum yang sudah diberikan. Soal dapat dijawab langsung di dalam kolom yang disediakan dengan menggunakan PDF Editor.

Pendahuluan

File Upload

File upload dalam PHP adalah proses mengunggah file dari komputer pengguna ke server web. Ini berguna dalam banyak kasus, seperti saat pengguna ingin mengunggah gambar profil, dokumen, atau media lainnya ke situs web Anda. Untuk melakukan ini, Anda 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:

```
</html>
```

Buat skrip PHP (**upload.php**) untuk mengelola pengunggahan 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 Anda memiliki direktori **uploads** yang ada di server Anda, dan berikan izin yang sesuai agar PHP dapat menyimpan file di sana.

Praktikum 1. Upload File

Langkah	Keterangan								
1	Buatlah satu file baru di dalam direktori dasarWeb, beri nama <code>form_upload.php</code> .								
2	Ketikkan ke dalam file <code>form_upload.php</code> tersebut kode di bawah ini.								
3	<pre><!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></pre> <pre><?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."; } }</pre>								
4	<p>Simpan file tersebut, kemudian buka browser dan jalankan <code>localhost/dasarWeb/form_upload.php</code>. Apa yang anda pahami dari script pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 1.1)</p> <p>Hasil:</p> <div><div><div>Choose File</div><div>Additional E...st Tense.pdf</div><div>Upload File</div></div><table><thead><tr><th>Name</th><th>Date modified</th><th>Type</th><th>Size</th></tr></thead><tbody><tr><td> Additional Exercise on Simple Past Tense</td><td>02/04/2024 22:30</td><td>Microsoft Edge PD...</td><td>396 KB</td></tr></tbody></table></div> <p>Jawab: Kode tersebut adalah form HTML yang memungkinkan pengguna mengunggah file. Formulir tersebut memiliki atribut <code>enctype="multipart/form-data"</code>, yang diperlukan ketika mengunggah file. Ketika formulir disubmit, skrip PHP (<code>upload.php</code>) akan dijalankan. Di dalam skrip PHP, kita memeriksa apakah tombol submit telah ditekan (<code>isset(\$_POST["submit"])</code>). Jika ya, kita menentukan direktori tujuan (<code>uploads/</code>) di mana file akan disimpan. Kemudian, kita memindahkan file yang diunggah dari direktori sementara (<code>temporarily uploaded file</code>) ke direktori tujuan menggunakan fungsi <code>move_uploaded_file()</code>.</p>	Name	Date modified	Type	Size	 Additional Exercise on Simple Past Tense	02/04/2024 22:30	Microsoft Edge PD...	396 KB
Name	Date modified	Type	Size						
 Additional Exercise on Simple Past Tense	02/04/2024 22:30	Microsoft Edge PD...	396 KB						

	Jika proses pemindahan berhasil, pesan "File berhasil diunggah." akan ditampilkan; jika tidak, pesan "Gagal mengunggah file." akan ditampilkan
5	<p>Ubah isi dari file <code>upload.php</code> tersebut dengan kode di bawah ini.</p> <pre> <?php if (isset(\$_POST["submit"])) { \$targetDirectory = "uploads/"; // Direktori tujuan untuk menyimpan file \$targetFile = \$targetDirectory . basename(\$_FILES["fileToUpload"]["name"]); \$fileType = strtolower(pathinfo(\$targetFile, PATHINFO_EXTENSION)); \$allowedExtensions = array("jpg", "jpeg", "png", "gif"); \$maxFileSize = 5 * 1024 * 1024; if (in_array(\$fileType, \$allowedExtensions) && \$_FILES["fileToUpload"]["size"] <= \$maxFileSize) { if (move_uploaded_file(\$_FILES["fileToUpload"]["tmp_name"], \$targetFile)) { echo "File berhasil diunggah."; } else { echo "Gagal mengunggah file."; } } else { echo "File tidak valid atau melebihi ukuran maksimum yang diizinkan."; } } </pre>
6	<p>Simpan file tersebut, kemudian buka browser dan jalankan <code>localhost/dasarWeb/form_upload.php</code>. Apa yang anda pahami dari penggunaan script tersebut. Catat di bawah ini pemahaman anda. (soal no 1.2)</p> <p>Hasil: File tidak valid atau melebihi ukuran maksimum yang diizinkan. File berhasil diunggah.</p> <p>Jawab: Kode telah ditambahkan untuk memeriksa apakah jenis file yang diunggah adalah salah satu dari jenis yang diizinkan (jpg, jpeg, png, gif). Jika jenis file tidak termasuk dalam daftar ini, maka akan muncul pesan kesalahan "File tidak valid atau melebihi ukuran maksimum yang diizinkan." Juga ditambahkan validasi untuk memeriksa apakah ukuran file tidak melebihi ukuran maksimum yang diizinkan (5 MB). Jika ukuran file melebihi batas ini, pesan kesalahan yang sama akan muncul. Ketika mencoba mengupload file berekstensi pdf maka keluar hasil file tidak valid dan ketika mencoba mengupload file berekstensi jpg maka proses upload berhasil</p>
7	<p>Soal 1.3: Tambahkan script langkah 5 untuk membuat file gambar thumbnail dengan ukuran lebar 200 dan tinggi mengikuti perubahan secara otomatis. Catat di sini apa yang anda amati dari penambahan kode program di atas. (soal no 1.3)</p> <p>Kode program:</p>

```

if (isset($_POST["submit"])){
    $targetDirectory = "uploads/"; // direktori tujuan untuk menyimpan file
    $targetFile = $targetDirectory . basename($_FILES["fileToUpload"]["name"]);
    $fileType = strtolower(pathinfo($targetFile, PATHINFO_EXTENSION));

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

    $maxFileSize = 5 * 1024 * 1024;

    if (in_array($fileType, $allowedExtensions) && $_FILES["fileToUpload"]["size"] <= $maxFileSize){
        if (move_uploaded_file($_FILES["fileToUpload"]["tmp_name"], $targetFile)){
            // Membuat thumbnail
            $thumbnailWidth = 200; // Lebar thumbnail tetap 200px

            // Mendapatkan informasi gambar asli
            list($originalWidth, $originalHeight) = getimagesize($targetFile);

            // Menghitung tinggi thumbnail sesuai dengan lebar yang ditetapkan
            $thumbnailHeight = floor($originalHeight * ($thumbnailWidth / $originalWidth));

            // Membuat gambar thumbnail baru
            $thumbnail = imagecreatetruecolor($thumbnailWidth, $thumbnailHeight);

            // Memuat gambar asli
            $source = null;
            if ($fileType === 'jpg' || $fileType === 'jpeg') {
                $source = imagecreatefromjpeg($targetFile);
            } elseif ($fileType === 'png') {
                $source = imagecreatefrompng($targetFile);
            } elseif ($fileType === 'gif') {
                $source = imagecreatefromgif($targetFile);
            }

            // Menyalin dan menyesuaikan ukuran gambar asli ke thumbnail
            imagecopyresampled($thumbnail, $source, 0, 0, 0, 0, $thumbnailWidth, $thumbnailHeight, $originalWidth, $originalHeight);

            // Simpan thumbnail
            $thumbnailFileName = $targetDirectory . "thumbnail_" . basename($_FILES["fileToUpload"]["name"]);
            if ($fileType === 'jpg' || $fileType === 'jpeg') {
                imagejpeg($thumbnail, $thumbnailFileName);
            } elseif ($fileType === 'png') {
                imagepng($thumbnail, $thumbnailFileName);
            } elseif ($fileType === 'gif') {
                imagegif($thumbnail, $thumbnailFileName);
            }

            echo "File dan thumbnail berhasil diunggah.";
        }else{
            echo "Gagal mengunggah file.";
        }
    }else{
        echo "File tidak valid atau melebihi ukuran maksimum yang diizinkan.";
    }
}

```

File dan thumbnail berhasil diunggah.

Hasil:

Jawab: Kode tersebut melakukan penambahan fungsionalitas untuk membuat thumbnail dari file gambar yang diunggah. Saat file gambar diunggah, script akan membuat thumbnail dengan lebar tetap 200 piksel dan tinggi yang disesuaikan secara proporsional. Thumbnail tersebut kemudian disimpan di direktori yang sama dengan file gambar asli dengan nama file yang dimulai dengan "thumbnail_". Proses ini dilakukan menggunakan fungsi-fungsi dari ekstensi GD di PHP, termasuk imagecreatetruecolor() untuk membuat gambar baru, getimagesize() untuk mendapatkan informasi ukuran gambar asli, imagecreatefrom...() untuk memuat gambar asli, dan imagecopyresampled() untuk menyalin dan menyesuaikan ukuran gambar asli ke thumbnail.

8

Ubah isi dari file upload.php tersebut dengan kode di bawah ini.

	<pre> <?php if (isset(\$_POST["submit"])) { \$targetDirectory = "documents/"; // Direktori tujuan untuk menyimpan dokumen \$targetFile = \$targetDirectory . basename(\$_FILES["documentToUpload"]["name"]); \$documentFileType = strtolower(pathinfo(\$targetFile, PATHINFO_EXTENSION)); \$allowedExtensions = array("txt", "pdf", "doc", "docx"); \$maxFileSize = 10 * 1024 * 1024; if (in_array(\$documentFileType, \$allowedExtensions) && \$_FILES["documentToUpload"]["size"] <= \$maxFileSize) { if (move_uploaded_file(\$_FILES["documentToUpload"]["tmp_name"], \$targetFile)) { echo "Dokumen berhasil diunggah."; } else { echo "Gagal mengunggah dokumen."; } } else { echo "Jenis dokumen tidak valid atau melebihi ukuran maksimum yang diizinkan."; } } </pre>
9	<p>Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/form_upload.php . Apa yang anda pahami dari penggunaan script tersebut. Catat di bawah ini pemahaman anda. (soal no 1.4)</p> <p>Hasil: Dokumen berhasil diunggah.</p> <p>Jawab: Perubahan pada kode tersebut terletak pada pemilihan nama field untuk input file pada form HTML dan penggunaannya dalam kode PHP. Pada form HTML, nama input file diubah menjadi name="documentToUpload", sehingga sesuai dengan apa yang diharapkan oleh kode PHP. Dalam kode PHP, kunci array \$_FILES["documentToUpload"] digunakan untuk mengakses informasi tentang file yang diunggah. Kode juga memeriksa jenis file yang diunggah dan ukurannya sesuai dengan aturan yang ditetapkan sebelumnya. Jika file yang diunggah sesuai dengan kriteria yang ditetapkan, maka file akan dipindahkan ke direktori tujuan yang ditentukan. Jika tidak, akan muncul pesan kesalahan yang sesuai.</p>

Praktikum 2. Multi Upload File

Langkah	Keterangan
1	Buatlah satu file baru di dalam direktori dasarWeb, beri nama form_multiupload.php.
2	Ketikkan ke dalam file form_multiupload.php tersebut kode di bawah ini.

```
<!DOCTYPE html>
<html>
<head>
|   <title>Multiupload Dokumen</title>
</head>
<body>
|   <h2>Unggah Dokumen</h2>
|   <form action="proses_upload.php" method="post" enctype="multipart/form-data">
|       <input type="file" name="files[]" multiple="multiple" accept=".pdf, .doc, .docx" />
|       <input type="submit" value="Unggah" />
|   </form>
</body>
</html>
```

proses_upload.php

3

```
<?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 < $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.<br>";
        } else {
            echo "Gagal mengunggah file $fileName.<br>";
        }
    }
} else {
    echo "Tidak ada file yang diunggah.";
}
```

4

Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/form_multiupload.php. Apa yang anda pahami dari script pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 2.1)

Hasil: File MODUL BAHASA INGGRIS 2_UNIT 3_8th Edition.pdf berhasil diunggah.

Jawab: Kode tersebut adalah form HTML untuk mengunggah multiple dokumen dengan menggunakan input file yang diatur untuk menerima ekstensi file .pdf, .doc, dan .docx. Kemudian, diikuti oleh skrip PHP yang menangani proses pengunggahan file. Skrip PHP membuat direktori "documents/" jika belum ada, kemudian memeriksa apakah ada file yang diunggah. Jika ada, skrip akan melakukan loop melalui setiap file yang diunggah, memindahkan file ke direktori penyimpanan yang ditentukan, dan memberikan pesan berhasil atau gagal sesuai dengan hasilnya. Jika tidak ada file yang diunggah, akan diberikan pesan bahwa tidak ada file yang diunggah

Soal 2.2: Buat seperti langkah 3 dengan multi upload khusus gambar. Catat di sini apa yang anda amati dari kode program di atas. (soal no 2.2)

```
<html>
<head>
  <title>Multiupload Gambar</title>
</head>
<body>
  <h2>Unggah Gambar</h2>
  <form action="proses_upload.php" method="post" enctype="multipart/form-data">
    <input type="file" name="images[]" multiple accept="image/*" />
    <input type="submit" value="Unggah">
  </form>
</body>
</html>
```

Kode program:

```
// Lokasi penyimpanan file yang diunggah
$targetDirectory = "uploads/";

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

if (!empty($_FILES['images']['name'][0])) {
  $totalImages = count($_FILES['images']['name']);

  // Loop melalui semua gambar yang diunggah
  for ($i = 0; $i < $totalImages; $i++){
    $imageFileName = $_FILES['images']['name'][$i];
    $targetFile = $targetDirectory . basename($imageFileName);

    // Periksa apakah file adalah gambar valid
    $imageFileType = strtolower(pathinfo($targetFile, PATHINFO_EXTENSION));
    if (getimagesize($_FILES['images']['tmp_name'][$i]) !== false) {
      // Pindahkan file yang diunggah ke direktori penyimpanan
      if (move_uploaded_file($_FILES['images']['tmp_name'][$i], $targetFile)) {
        echo "Gambar $imageFileName berhasil diunggah.<br>";
      } else {
        echo "Gagal mengunggah gambar $imageFileName.<br>";
      }
    } else {
      echo "File $imageFileName bukan gambar yang valid.<br>";
    }
  }
} else {
  echo "Tidak ada gambar yang diunggah.";
}
```

Hasil: Gambar ui.png berhasil diunggah.

Jawab: Script tersebut memungkinkan pengguna untuk mengunggah beberapa gambar sekaligus melalui form HTML dengan atribut multiple, sementara batasan jenis file diterapkan dengan atribut accept untuk memastikan hanya gambar yang diunggah. Script PHP memproses setiap gambar yang diunggah, memeriksa validitasnya dengan getimagesize(), dan jika valid, memindahkannya ke direktori penyimpanan. Pesan status ditampilkan untuk setiap gambar yang diunggah, memberikan umpan balik langsung kepada pengguna tentang status pengunggahan gambar

Praktikum 3. Upload File dengan PHP dan JQuery

Langkah	Keterangan
1	Buatlah satu file baru di dalam direktori dasarWeb, beri nama form_upload_ajax.php.

2

Ketikkan ke dalam file `form_upload_ajax.php` tersebut kode di bawah ini.

```
<!DOCTYPE html>
<html>

<head>
  <title>Unggah File Dokumen</title>
</head>

<body>
  <form id="upload-form" action="upload_ajax.php" method="post" enctype=
"multipart/form-data">
    <input type="file" name="file" id="file">
    <input type="submit" name="submit" value="Unggah">
  </form>
  <div id="status"></div>

  <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script>
  <script src="upload.js"></script>
</body>

</html>
```

3

`upload.js`

```
$(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.');
      }
    });
  });
});
```

`upload_ajax.php`

	<pre> <?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 > 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>
4	<p>Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/form_upload_ajax.php. Apa yang anda pahami dari script pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 3.1)</p> <div> <input type="button" value="Choose File"/> RESUME JURNAL.docx <input type="button" value="Unggah"/> </div> <p>Hasil: File berhasil diunggah.</p> <p>Jawab: Kode HTML di atas adalah formulir yang memungkinkan pengguna untuk mengunggah file dokumen. Ketika formulir disubmit, JavaScript jQuery di bawahnya akan mencegah tindakan default, yaitu pengiriman formulir secara tradisional, dan menggunakan AJAX untuk mengirim data formulir ke upload_ajax.php. File PHP kemudian akan memeriksa file yang diunggah untuk memastikan bahwa itu memiliki ekstensi yang diizinkan (PDF, DOC, DOCX, atau TXT) dan ukurannya tidak lebih dari 2 MB. Jika tidak ada kesalahan, file akan dipindahkan ke direktori "documents". Hasilnya akan ditampilkan dalam div dengan id "status" melalui AJAX</p>
5	<p>Soal 3.2: Buat seperti langkah 3 dengan multi upload khusus gambar dan JQuery. Catat di sini apa yang anda amati dari kode program di atas. (soal no 3.2)</p> <p>Unggah Gambar</p> <div> <input type="button" value="Choose Files"/> cat.jpg <input type="button" value="Unggah"/> </div> <p>Hasil: Gambar cat.jpg berhasil diunggah.</p> <p>Jawab: Kode tersebut adalah implementasi form HTML untuk unggah gambar dengan kemampuan multi-upload menggunakan jQuery AJAX. Formulir HTML memiliki elemen <code><input type="file" multiple></code> yang memungkinkan pengguna untuk memilih beberapa file gambar sekaligus. Ketika formulir disubmit, skrip JavaScript menggunakan AJAX untuk mengirim data ke server tanpa perlu me-refresh halaman. Di sisi server (dalam file upload_ajax.php), PHP digunakan untuk memeriksa dan memproses setiap file gambar yang diunggah. Setiap file diperiksa untuk memastikan bahwa itu memiliki ekstensi file yang diizinkan (JPG, JPEG, PNG, atau GIF) dan tidak melebihi batas ukuran file (2 MB). Jika semua validasi berhasil, file gambar dipindahkan ke direktori "uploads/" dan pesan berhasil diunggah ditampilkan kembali kepada pengguna. Jika ada kesalahan dalam validasi, pesan</p>

	kesalahan ditampilkan kepada pengguna. Dengan demikian, pengguna dapat dengan mudah mengunggah beberapa gambar secara bersamaan tanpa meninggalkan atau me-refresh halaman.
--	---

Praktikum 4. Menghias Upload File

Langkah	Keterangan
1	Memodifikasi script yang sudah di buat pada Praktikum 3 yaitu form_upload_ajax.php dan upload.js
2	Ketikkan ke dalam semua file dan tambahkan 1 file berupa css dengan kode di bawah ini.
3	<div> <pre> <!DOCTYPE html> <html> <head> <link rel="stylesheet" type="text/css" href="upload.css"> <title>Unggah File Dokumen</title> </head> <body> <div class="upload-form-container"> <h2>Unggah File Dokumen</h2> <form id="upload-form" action="upload.php" method="post" enctype= "multipart/form-data"> <div class="file-input-container"> <input type="file" name="file" id="file" class="file-input"> <label for="file" class="file-label">Pilih File</label> </div> <button type="submit" name="submit" class="upload-button" id="upload-button" disabled>Unggah</button> </form> <div id="status" class="upload-status"></div> </div> <script src="https://code.jquery.com/jquery-3.6.0.min.js"></script> <script src="upload.js"></script> </body> </html> </pre> </div> <div>upload.js</div>

```

$(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.');
```

upload.css

```

.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;
}

```

Simpan file tersebut, kemudian buka browser dan jalankan localhost/dasarWeb/form_upload_ajax.php. Apa yang anda pahami dari script pada file tersebut. Catat di bawah ini pemahaman anda. (soal no 4.1)



Unggah File Dokumen

Pilih File

Unggah

File berhasil diunggah.

Hasil:

Jawab: Kode tersebut mencakup form HTML untuk unggah file dokumen, ditambah dengan elemen-elemen JavaScript untuk mengaktifkan tombol unggah dan mengirimkan permintaan AJAX saat formulir diserahkan. Sisi server PHP bertanggung jawab atas validasi dan penanganan file yang diunggah, memeriksa ekstensi dan ukuran file sebelum memindahkannya ke direktori yang sesuai