

Jobsheet 8

Pemrograman Web



Arsya Fikri Sabilillah

244107060121

SIB-2C

PROGRAM STUDI SISTEM INFORMASI BISNIS

JURUSAN TEKNOLOGI INFORMASI

POLITEKNIK NEGERI MALANG

2025

PRAKTIKUM 1

1. Simpan file, kemudian buka browser dan jalankan **localhost/dasarWeb/form_upload.php**. Pilih sebuah file dan klik tombol Submit. Amati yang terjadi dan catat pemahaman Anda. (Pertanyaan No. 1)

= File tidak berhasil diunggah karena PHP tidak menemukan direktori tujuan untuk menyimpan file. Fungsi `move_uploaded_file()` gagal dijalankan.
2. Selanjutnya buat folder bernama **uploads** pada direktori **dasarWeb**. Jalankan kembali **localhost/dasarWeb/form_upload.php**. Pilih sebuah file dan klik tombol Submit. Amati yang terjadi dan catat pemahaman Anda. (Pertanyaan No. 2)

= File berhasil diunggah ke folder uploads karena direktori tujuan sudah ada dan memiliki izin tulis.
3. Simpan file, buka browser dan jalankan **localhost/dasarWeb/form_upload.php** Pilih sebuah file dengan ekstensi .pdf atau .docx. Klik tombol Submit. Amati yang terjadi dan catat pemahaman Anda. (Pertanyaan No. 3)

= Hanya file dengan ekstensi yang diizinkan yang dapat diunggah. Jika file sesuai kriteria, maka muncul pesan "Upload berhasil"; jika tidak, ditolak dengan pesan error.
4. Selanjutnya jalankan kembali **localhost/dasarWeb/form_upload.php**. Pilih sebuah file dengan ekstensi .jpg, .jpeg, .png, atau .gif. Klik tombol Submit. Amati yang terjadi dan catat pemahaman Anda. (Pertanyaan No. 4)

= Sama seperti sebelumnya, namun file gambar diterima karena masuk dalam daftar ekstensi gambar yang diizinkan.
5. 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)

= Program menampilkan pratinjau (thumbnail) gambar hasil upload dengan ukuran lebar 200px, tinggi menyesuaikan proporsi. Ini membantu pengguna melihat hasil unggahan tanpa membuka file manual.

```
echo "<p>Preview Gambar:</p>";  
echo "<img src='$targetfile' width='200'>";
```

6. Simpan file, buka browser dan jalankan **localhost/dasarWeb/form_upload.php** 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. (Pertanyaan No. 6)

= Upload gagal karena melebihi batas ukuran maksimum yang ditentukan dalam kode (misalnya 5 MB). PHP menolak file besar demi keamanan dan efisiensi server.

7. Selanjutnya jalankan kembali **localhost/dasarWeb/form_upload.php**. 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. (Pertanyaan No. 7)
- = Upload berhasil karena ukuran file masih di bawah batas maksimum. File disimpan ke folder uploads sesuai ketentuan.

PRAKTIKUM 2

8. Simpan file, buka browser dan jalankan **localhost/dasarWeb/form_multiupload.php**. Pilih beberapa file sekaligus untuk diunggah. Apa yang anda pahami dari script pada file tersebut? Catat pemahaman anda. (Pertanyaan No. 8)
- = Script menggunakan loop (foreach) untuk memproses setiap file satu per satu. Semua file valid diunggah ke folder uploads. Jadi sistem bisa mengunggah banyak file sekaligus.
9. Buat seperti langkah 3 dengan multi upload khusus gambar. Screen shoot perubahan kode dan berikan penjelasan kode tersebut. (Pertanyaan No. 9)
- = Program menambahkan validasi agar hanya file dengan ekstensi gambar (jpg, jpeg, png, gif) yang diterima. File lain akan ditolak. Tujuannya agar folder hanya berisi file gambar dan aman dari file berbahaya.

```
$allowedExtensions = array("jpg", "jpeg", "png", "gif"); // Hanya gambar

for ($i = 0; $i < $totalFiles; $i++) {
    $fileName = $_FILES['files']['name'][$i];
    $targetFile = $targetDirectory . basename(path: $fileName);
    $fileType = strtolower(string: pathinfo(path: $targetFile, flags: PATHINFO_EXTENSION));

    if (in_array(needle: $fileType, haystack: $allowedExtensions)) {
        if (move_uploaded_file(from: $_FILES['files']['tmp_name'][$i], to: $targetFile)) {
            echo "File <b>$fileName</b> berhasil diunggah.<br>";
            echo "<img src='$targetFile' width='150' style='height:auto; margin:5px; border:1px";
        } else {
            echo "Gagal mengunggah file <b>$fileName</b>.<br>";
        }
    } else {
        echo "File <b>$fileName</b> tidak valid (bukan gambar).<br>";
    }
}
} else {
    echo "Tidak ada file yang diunggah.";
```

PRAKTIKUM 3

10. Amati apa yang terjadi dan jelaskan pemahaman anda. (Pertanyaan No. 10)
- = Proses upload berlangsung **tanpa reload halaman** berkat jQuery AJAX.
- Jika file valid pesan sukses muncul secara dinamis.
 - Jika file terlalu besar atau format tidak sesuai → muncul pesan error langsung.
- Proses lebih interaktif dan efisien.
11. Ubah kode menjadi agar dapat dilakukan multi upload file khusus gambar. Screenshoot perubahan kode dan jelaskan kode tersebut. (Pertanyaan No. 11)
- = Program memvalidasi dan menampilkan hasil upload beberapa gambar sekaligus, bisa juga menampilkan preview semua gambar yang diunggah.

```

...
$allowedExtensions = array("jpg", "jpeg", "png", "gif");
$totalFiles = count(value: $_FILES['file']['name']);

for ($i = 0; $i < $totalFiles; $i++) {
    $file_name = $_FILES['file']['name'][$i];
    $file_size = $_FILES['file']['size'][$i];
    $file_tmp = $_FILES['file']['tmp_name'][$i];
    $file_type = $_FILES['file']['type'][$i];
    $file_ext = strtolower(string: pathinfo(path: $file_name, flags: PATHINFO_EXTENSION));

    if (!in_array(needle: $file_ext, haystack: $allowedExtensions)) {
        $errors[] = "File $file_name tidak valid. Hanya JPG, JPEG, PNG, dan GIF yang diperbolehkan";
        continue;
    }

    if ($file_size > 2 * 1024 * 1024) {
        $errors[] = "Ukuran file $file_name melebihi 2 MB.<br>";
        continue;
    }

    if (move_uploaded_file(from: $file_tmp, to: "uploads/" . $file_name)) {
        echo "File <b>$file_name</b> berhasil diunggah.<br>";
        echo "<img src='uploads/$file_name' width='150' style='height:auto; margin:5px; border:1px solid black;' />";
    } else {
        $errors[] = "Gagal mengunggah file $file_name.<br>";
    }
}

if (!empty($errors)) {
    echo "<b>Daftar Error:</b><br>" . implode(separator: "", array: $errors);
}

```

PRAKTIKUM 4

12. Simpan file tersebut. Buka browser dan jalankan

localhost/dasarWeb/form_upload_ajax.php. Apa yang anda pahami dari kode program di atas? Catat pemahaman anda. (Pertanyaan No. 12)

= Tampilan form upload menjadi lebih menarik (modern & responsif).

Ada efek visual pada tombol, area upload lebih rapi, dan notifikasi hasil upload terlihat jelas. Tujuannya agar user experience (UX) lebih baik tanpa mengubah fungsi utama upload.

PRAKTIKUM 5

13. Amati dan jelaskan hasil pengamatanmu (Pertanyaan No. 13)

= Tidak ada nilai cookie yang muncul karena belum dibuat. Program hanya menampilkan pesan bahwa cookie belum diset.

14. Amati dan jelaskan hasil yang ditampilkan

= Cookie berhasil dibuat dengan nama dan nilai tertentu. Saat cookiesCall.php dibuka kembali, nilai cookie sudah bisa ditampilkan.

15. Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 15)

= Cookie masih terbaca karena belum kadaluarsa (persistent cookies). Artinya cookie tetap tersimpan di sisi client meskipun browser ditutup atau komputer direstart.

PRAKTIKUM 6

16. Amati dan jelaskan hasil yang dihasilkan dari langkah 2 dan 3, kemudian beri kesimpulan. (Pertanyaan No. 16)

= Cookie sudah dihapus (dengan setcookie berwaktu lampau). Hasilnya: browser tidak lagi menampilkan data cookie.

PRAKTIKUM 7

17. Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 17)

= Tampilan kosong karena belum ada cookie yang menyimpan jumlah barang. Program menunggu input dari formBeli.html.

18. Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 18)

= Program menyimpan data jumlah buku & novel ke cookie. Lalu saat membuka keranjangBelanja.php, tampil isi keranjang (data yang tersimpan di cookie).

19. Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 19)

= Data belanja masih muncul karena cookie belum kedaluwarsa. Cookie bersifat persisten dan tetap menyimpan data meski browser ditutup.

PRAKTIKUM 8

20. Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 20)

= Tidak ada data ditampilkan karena session belum dibuat.

Setelah sessionCreate.php dijalankan, lalu buka sessionCall.php lagi → data muncul dari variabel session.

PRAKTIKUM 9

21. Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 21)

= Session telah dihapus menggunakan session_destroy(), sehingga variabel session tidak lagi bisa diakses.

PRAKTIKUM 10

22. Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 22)

= Program memeriksa input dengan data di sessionLoginProcess.php. Karena password salah atau user tidak terdaftar, maka login gagal dan menampilkan pesan error.

23. Amati dan jelaskan hasil yang ditampilkan. (Pertanyaan No. 23)

= Login berhasil. Session baru dibuat menyimpan data user. Halaman diarahkan ke homeSession.php yang hanya bisa diakses jika session aktif.

24. Jelaskan urutan proses dari login hingga logout (sebutkan juga urutan file yang diproses) (Pertanyaan No. 24)

= Urutan proses login hingga logout

- sessionLoginForm.html user isi username & password.

- sessionLoginProcess.php memverifikasi data; jika benar, buat session.
- homeSession.php menampilkan halaman utama user selama session masih aktif.
- sessionLogout.php menghapus session dengan session_destroy() lalu kembali ke halaman login.