

****🔥 Tantangan Python: File Handling (Baca, Tulis, dan Manipulasi File)****

Berdasarkan catatanmu tentang file handling, berikut 5 tantangan untuk menguji pemahamanmu:

🎯 1. **Baca File dan Hitung Jumlah Baris**

Buat program untuk:

1. Baca file `data.txt` (berisi beberapa baris teks).
2. Hitung dan cetak jumlah baris dalam file.
3. Cetak baris ke-3 saja (jika ada).

****Contoh Output:****

```

Jumlah baris: 5

Baris ke-3: Ini adalah baris ketiga

```

🎯 2. **Copy Isi File dengan Filter**

Buat program untuk:

1. Baca file `source.txt`.
2. Buat file baru `filtered.txt` yang hanya berisi baris yang mengandung kata "Python".
3. Gunakan `with` statement untuk kedua file.

****Contoh:****

```

```
source.txt
```

```
Saya belajar Python
```

```
Java juga keren
```

```
Python itu menyenangkan
```

```
filtered.txt
```

```
Saya belajar Python
```

```
Python itu menyenangkan
```

```
```\n
```

```
---\n
```

```
### 🎯 3. **Append Data ke File Log**
```

Buat program untuk:

1. Minta input user: nama dan waktu (format bebas).
2. Append data ke file `log.txt` dalam format: `[waktu] - nama`
3. Setiap input baru ditambahkan di baris baru.

****Contoh Output di log.txt:****

```
```\n
```

```
[10:30] - Budi
```

```
[11:45] - Ani
```

```
[12:00] - Citra
```

```
```\n
```

```
---\n
```

```
### 🎯 4. **File Encryption Sederhana**
```

Buat program untuk:

1. Baca file `plain.txt`.
2. Enkripsi setiap karakter dengan Caesar cipher (geser 3 karakter).
3. Simpan hasil enkripsi di `encrypted.txt`.

****Rumus Caesar Cipher:****

- 'a' → 'd', 'b' → 'e', ..., 'z' → 'c' (geser 3)

****Contoh:****

```

# plain.txt

hello world

# encrypted.txt

khoor zruog

```

🎯 5. ****Merge Multiple Files****

Buat program untuk:

1. Gabungkan isi dari 3 file: `file1.txt`, `file2.txt`, `file3.txt`.
2. Simpan hasil gabungan di `merged.txt`.
3. Beri separator antara file: `\n=== File X ===\n`

****Contoh Output di merged.txt:****

```

=== File 1 ===

Isi file pertama

=== File 2 ===

Isi file kedua

=== File 3 ===

Isi file ketiga

` ``

---

### 💡 **\*\*Tips Penyelesaian:\*\***

- Gunakan `with open(...) as file` untuk auto-close.
- Untuk tantangan 4, gunakan `ord()` dan `chr()` untuk manipulasi karakter.
- Untuk tantangan 5, loop melalui list nama file.

**\*\*Contoh Solusi Tantangan 1:\*\***

` `` python

with open("data.txt", "r") as file:

lines = file.readlines()

print(f"Jumlah baris: {len(lines)}")

if len(lines) >= 3:

print(f"Baris ke-3: {lines[2].strip()}")

` ``

**\*\*Selamat Mengerjakan!\*\*** 🚀

**\*\*File handling adalah pintu gerbang ke dunia persistence data!\*\***