

**Sistem Pengelolaan Keuangan Pribadi dengan File CSV**

**LAPORAN FINAL PROJECT STRUKTUR DATA**

Oleh:

Basthiar Rofil Ahdistyo(24416255201200)



PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS BUANA PERJUANGAN KARAWANG  
2025

## PENDAHULUAN

### 1.1 Latar Belakang

Manajemen keuangan pribadi adalah hal yang penting dalam kehidupan sehari-hari, terutama bagi mereka yang ingin mengelola pengeluaran dan pemasukan secara lebih efektif. Banyak individu menghadapi kesulitan dalam memantau pengeluaran mereka yang dapat menyebabkan pemborosan atau kesulitan finansial di masa depan. Oleh karena itu, pengembangan aplikasi manajemen keuangan pribadi menggunakan Python dapat menjadi solusi untuk membantu pengguna dalam mencatat dan memonitor keuangan mereka dengan mudah.

Aplikasi ini bertujuan untuk memberikan pengguna kemudahan dalam melakukan pencatatan pemasukan dan pengeluaran, serta menghasilkan laporan bulanan dan tahunan. Dengan menggunakan file CSV sebagai basis data, aplikasi ini memanfaatkan struktur data yang efisien untuk menyimpan dan mengelola transaksi.

### 1.2 Rumusan Masalah

1. Banyak individu yang kesulitan mengelola pengeluaran mereka dengan baik, sehingga mengarah pada pemborosan atau ketidakseimbangan finansial.
2. Kurangnya alat yang sederhana dan efisien untuk mencatat pemasukan dan pengeluaran secara teratur, yang membuat orang kesulitan memonitor keuangan mereka.

### 1.3 Tujuan

1. Mengembangkan aplikasi manajemen keuangan pribadi yang memungkinkan pengguna untuk mencatat pemasukan dan pengeluaran secara efektif.
2. Menyediakan laporan bulanan dan tahunan untuk membantu pengguna memonitor pengeluarannya dan mengevaluasi keuangan mereka.
3. Membantu pengguna dalam mengelompokkan pengeluaran berdasarkan kategori tertentu, yang dapat meningkatkan kontrol terhadap anggaran dan pengeluaran mereka.

## PERENCANAAN DAN DESAIN

### 2.1 Deskripsi Aplikasi

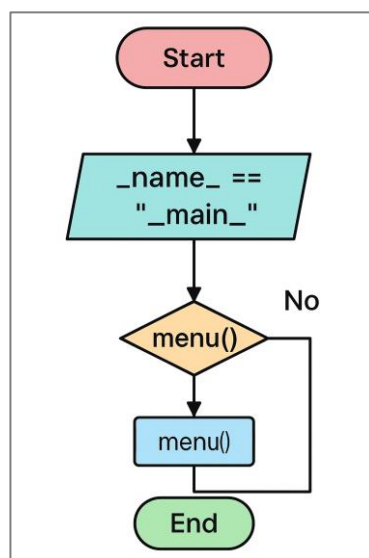
Aplikasi yang dikembangkan ini adalah aplikasi manajemen keuangan pribadi berbasis Python yang memungkinkan pengguna untuk:

1. Mencatat pemasukan dan pengeluaran.
2. Melihat laporan bulanan dan tahunan.
3. Mengelompokkan pengeluaran berdasarkan kategori.
1. Pencatatan pemasukan dan pengeluaran untuk setiap transaksi.
2. Laporan pengeluaran berdasarkan bulan dan tahun.
3. Mengelompokkan pengeluaran dalam kategori tertentu.

Alur Aplikasi (Flowchart):

1. Pengguna membuka aplikasi.
2. Pengguna dapat memilih untuk menambah transaksi atau melihat laporan.
3. Aplikasi akan menyimpan data transaksi dalam file CSV.
4. Laporan ditampilkan berdasarkan bulan atau tahun yang dipilih.

**Gambar 1.** Alur Aplikasi



Struktur File CSV:

- Tanggal: Tanggal transaksi.
- Kategori: Kategori pengeluaran (misalnya Makanan, Transportasi).
- Pemasukan: Jumlah uang yang diterima.
- Pengeluaran: Jumlah uang yang dikeluarkan.
- Sisa Pemasukan: Sisa dari pemasukan yang belum terpakai.

Tanggal	Kategori	Pemasukan	Pengeluaran	Sisa Pemasukan
2025-07-15	Gaji	4.000.000	-	4.000.000
2025-02-30	Gaji	5.800.000	-	5.800.000
2025-03-15	Body Care	-	1.500.900	8.300.000

**Tabel 1.** File CSV

## PERANCANGAN SOLUSI

### 3.1 Penjelasan Teknis

Aplikasi ini menggunakan Python dan CSV sebagai basis data untuk menyimpan transaksi keuangan. Struktur data yang digunakan dalam aplikasi ini adalah:

- List: Untuk menyimpan daftar transaksi.
- HashMap (Dictionary): Untuk mengelompokkan pengeluaran berdasarkan kategori.

```
main.py > ...
1 import csv
2 from datetime import datetime
3
4 # File database
5 FILE_NAME = 'keuangan.csv'
6
7 # Inisialisasi file CSV jika belum ada
8 def inisialisasi_file():
9     try:
10         with open(FILE_NAME, mode='r') as file:
11             pass
12     except FileNotFoundError:
13         with open(FILE_NAME, mode='w', newline='') as file:
14             writer = csv.writer(file)
15             writer.writerow(['Tanggal', 'Jenis', 'Kategori', 'Deskripsi', 'Jumlah'])
16
17 def tambah_data():
18     tanggal = input("Masukkan tanggal (YYYY-MM-DD): ")
19     jenis = input("Jenis (Pemasukan/Pengeluaran): ")
20     kategori = input("Kategori: ")
21     deskripsi = input("Deskripsi: ")
22     jumlah = input("Jumlah: ")
23
24     with open(FILE_NAME, mode='a', newline='') as file:
25         writer = csv.writer(file)
26         writer.writerow([tanggal, jenis, kategori, deskripsi, jumlah])
27     print("Data berhasil ditambahkan.")
```

**Gambar 2.** Kode Utama

### 3.2 Hasil Uji Coba

Contoh Pengujian:

- Pemasukan: 9.800.000X2 bulan (Gaji)
- Pengeluaran: 1.500.900 untuk kategori Body Care.

Saldo yang dihitung setelah transaksi adalah:

- Saldo Awal: 9.800.000.
- Pengeluaran: 1.500,900, sehingga saldo menjadi 8.300.000

*Tangkapan Layar ada pada lampiran*

Pengujian Laporan Bulanan:

Laporan bulan Juli 2025 menunjukkan transaksi yang terjadi pada bulan tersebut, dengan informasi tentang kategori, pemasukan, dan pengeluaran.

## KESIMPULAN

Aplikasi manajemen keuangan pribadi yang dikembangkan telah berhasil menyediakan fitur pencatatan pemasukan dan pengeluaran serta laporan bulanan dan tahunan. Aplikasi ini dapat membantu pengguna dalam mengelola keuangan mereka dengan cara yang mudah dan terstruktur.

Rekomendasi Pengembangan:

- Penambahan fitur pencarian transaksi berdasarkan kategori atau tanggal.
- Implementasi antarmuka grafis (GUI) untuk kemudahan penggunaan.
- Menambahkan fitur pengingat tagihan untuk mengingatkan pengguna tentang pembayaran yang akan datang.

## DAFTAR PUSTAKA

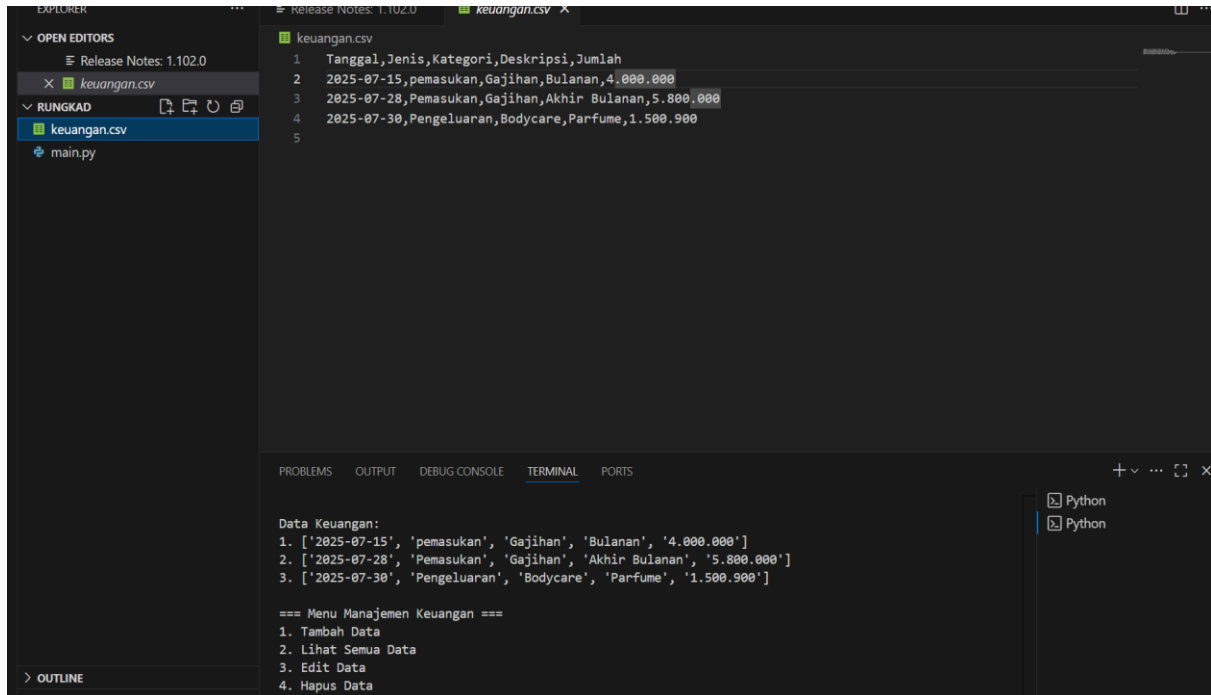
1. Dokumentasi Python: <https://docs.python.org/3/>
2. Panduan penggunaan CSV: <https://realpython.com/python-csv/>

## LAMPIRAN

### Lampiran A : Tangkapan Layar Program

```
Data Keuangan:
1. ['2025-07-15', 'pemasukan', 'Gajihan', 'Bulanan', '4.000.000']
2. ['2025-07-28', 'Pemasukan', 'Gajihan', 'Akhir Bulanan', '5.800.000']
3. ['2025-07-30', 'Pengeluaran', 'Bodycare', 'Parfume', '1.500.900']

=== Menu Manajemen Keuangan ===
1. Tambah Data
2. Lihat Semua Data
3. Edit Data
4. Hapus Data
5. Laporan Bulanan
```





## Lampiran B : Kode Program

```
main.py X
main.py > ...
Click to add a breakpoint {}:
30     with open(FILE_NAME, mode='r') as file:
31         reader = csv.reader(file)
32         next(reader)
33         print("\nData Keuangan:")
34         for i, row in enumerate(reader, start=1):
35             print(f"{i}. {row}")
36
37     def edit_data():
38         lihat_data()
39         index = int(input("Pilih nomor data yang ingin diedit: "))
40         with open(FILE_NAME, mode='r') as file:
41             reader = list(csv.reader(file))
42
43             header = reader[0]
44             data = reader[1:]
45
46             if 0 < index <= len(data):
47                 new = []
48                 for i, val in zip(['Tanggal', 'Jenis', 'Kategori', 'Deskripsi', 'Jumlah'], data[index - 1]):
49                     new_val = input(f"{i} [{val}]: ") or val
50                     new.append(new_val)
51                 data[index - 1] = new
52
53                 with open(FILE_NAME, mode='w', newline='') as file:
54                     writer = csv.writer(file)
55                     writer.writerow(header)
56                     writer.writerows(data)
57                 print("Data berhasil diperbarui.")
...
main.py X
main.py > ...
37     def edit_data():
58         else:
59             print("Index tidak valid.")
60
61     def hapus_data():
62         lihat_data()
63         index = int(input("Pilih nomor data yang ingin dihapus: "))
64         with open(FILE_NAME, mode='r') as file:
65             reader = list(csv.reader(file))
66
67             header = reader[0]
68             data = reader[1:]
69
70             if 0 < index <= len(data):
71                 data.pop(index - 1)
72                 with open(FILE_NAME, mode='w', newline='') as file:
73                     writer = csv.writer(file)
74                     writer.writerow(header)
75                     writer.writerows(data)
76                 print("Data berhasil dihapus.")
77             else:
78                 print("Index tidak valid.")
79
80     def laporan_bulanan():
81         bulan = input("Masukkan bulan (01-12): ")
82         tahun = input("Masukkan tahun (YYYY): ")
83         total_masuk, total_keluar = 0, 0
84
85         with open(FILE_NAME, mode='r') as file:
```

```

86     reader = csv.DictReader(file)
87     for row in reader:
88         if row['Tanggal'][0:7] == f"{tahun}-{bulan}":
89             if row['Jenis'].lower() == 'pemasukan':
90                 total_masuk += int(row['Jumlah'])
91             else:
92                 total_keluar += int(row['Jumlah'])
93     print(f"\nLaporan Bulanan {bulan}/{tahun}:")
94     print(f"Total Pemasukan : Rp{total_masuk}")
95     print(f"Total Pengeluaran: Rp{total_keluar}")
96
97 def laporan_tahunan():
98     tahun = input("Masukkan tahun (YYYY): ")
99     total_masuk, total_keluar = 0, 0
100
101     with open(FILE_NAME, mode='r') as file:
102         reader = csv.DictReader(file)
103         for row in reader:
104             if row['Tanggal'][0:4] == tahun:
105                 if row['Jenis'].lower() == 'pemasukan':
106                     total_masuk += int(row['Jumlah'])
107                 else:
108                     total_keluar += int(row['Jumlah'])
109     print(f"\nLaporan Tahunan {tahun}:")
110     print(f"Total Pemasukan : Rp{total_masuk}")
111     print(f"Total Pengeluaran: Rp{total_keluar}")
112
113 def menu():

```

main.py > laporan\_tahunan

```

113 def menu():
114     inisialisasi_file()
115     while True:
116         print("\n=== Menu Manajemen Keuangan ===")
117         print("1. Tambah Data")
118         print("2. Lihat Semua Data")
119         print("3. Edit Data")
120         print("4. Hapus Data")
121         print("5. Laporan Bulanan")
122         print("6. Laporan Tahunan")
123         print("7. Keluar")
124
125         pilihan = input("Pilih menu (1-7): ")
126
127         if pilihan == '1':
128             tambah_data()
129         elif pilihan == '2':
130             lihat_data()
131         elif pilihan == '3':
132             edit_data()
133         elif pilihan == '4':
134             hapus_data()
135         elif pilihan == '5':
136             laporan_bulanan()
137         elif pilihan == '6':
138             laporan_tahunan()
139         elif pilihan == '7':
140             print("Terima kasih telah menggunakan aplikasi ini.")
141             break

```

main.py ×

main.py > laporan\_tahunan

```
113 def menu():
140     print("Terima kasih telah menggunakan aplikasi ini.")
141     break
142 else:
143     print("Pilihan tidak valid.")
144
145 if __name__ == "__main__":
146     menu()
147
148
```