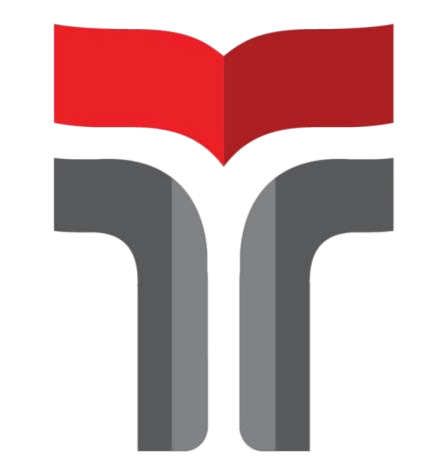
**LAPORAN TUGAS BESAR**

**PRAKTIKUM PEMROGRAMAN 1**



**Disusun oleh :**

Adam Darmawan (2211104034)

Satria Ariq Adelard Dompas (2211104033)

Salman Alfa Rizzi (2211104056)

Arwin Nabiel Arrofif (2211104057)

Yogi Hafidh Maulana (2211104061)

**Dosen :**

Arif Amrulloh, S.Kom., M.Kom.

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK**

**FAKULTAS INFORMATIKA INSTITUT TEKNOLOGI TELKOM**

**PURWOKERTO**

**SAVEMONEY**

**Program pencatatan keuangan pribadi**

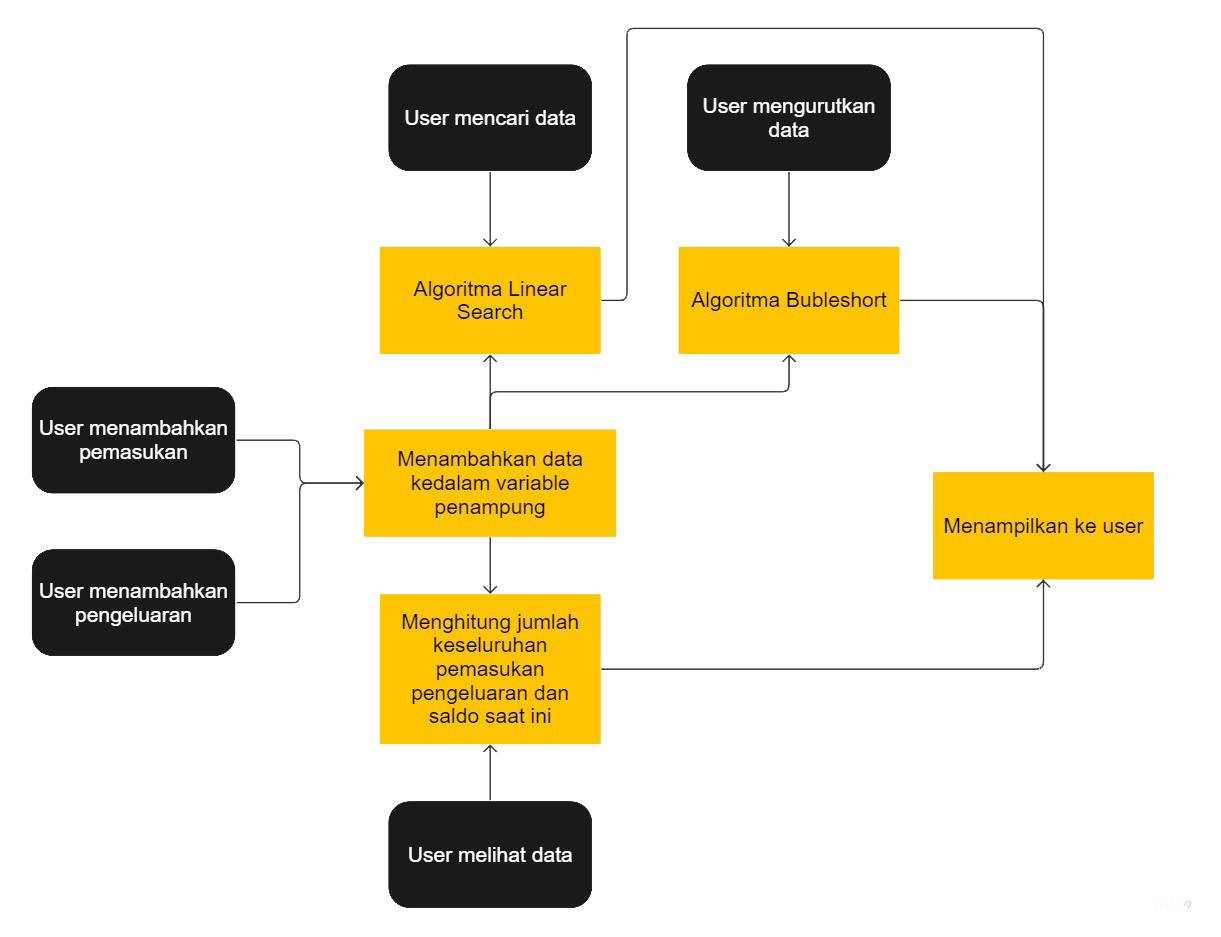
1. **Latar Belakang Aplikasi**

Dalam kehidupan sehari-hari, manajemen keuangan menjadi aspek penting yang harus diperhatikan oleh setiap individu. Pencatatan keuangan yang baik dapat membantu seseorang mengelola pendapatan, mengontrol pengeluaran, dan merencanakan anggaran dengan lebih efektif. Namun, seringkali orang kesulitan melakukan pencatatan keuangan secara manual dan terorganisir. Oleh karena itu, diperlukan adanya aplikasi yang dapat membantu pengguna dalam mencatat dan mengelola keuangan mereka.

Berikut ini beberapa alasan penting mengapa aplikasi pencatatan keuangan menjadi relevan dan diperlukan. Pertama, pengelolaan keuangan yang Lebih Efisien, dengan menggunakan aplikasi pencatatan keuangan, pengguna dapat memantau secara akurat arus kas mereka. Aplikasi tersebut dapat menghitung pemasukan dan pengeluaran, memanatu arus kas keuangan, serta memberikan visualisasi yang jelas tentang bagaimana uang mereka digunakan. Hal ini memungkinkan pengguna untuk melihat pola pengeluaran, mengidentifikasi area pengeluaran yang berlebihan, dan membuat perubahan yang diperlukan untuk mencapai tujuan keuangan mereka. Kedua, pencatatan yang Terorganisir, Aplikasi pencatatan keuangan menyediakan fitur-fitur yang mempermudah pengguna dalam mencatat setiap transaksi keuangan. Pengguna dapat menambahkan keterangan dari pemasukan atau pengeluaran tersebut. Dengan adanya pencatatan yang terorganisir, pengguna dapat dengan mudah melacak dan mencari informasi keuangan yang diperlukan, serta menghindari kesalahan atau kelalaian dalam mencatat transaksi.

Dalam rangka memenuhi kebutuhan pengguna, pengembangan aplikasi pencatatan keuangan perlu mempertimbangkan fitur fitur utama seperti menambahkan pemasukan, menambahkan pengeluaran, mengurutkan data pemasukan dan pengeluaran, mencari data pemasukan dan pengeluaran, dan melihat data. Dengan adanya fiitur fitur ini, diharapkan pengguna dapat mendapatakan kemudahan dalam mengelola keuangan mereka dengan lebih baik.

1. **Analisis Kerja Aplikasi**
2. **Desain Aplikasi**

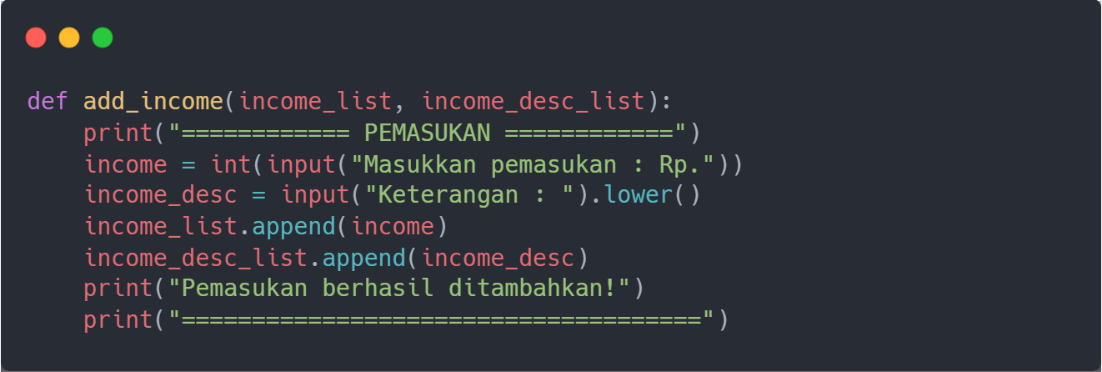


Aplikasi pencatat keuangan ini diberi nama Savemoney dan menggunakan bahasa pemrograman Python untuk membangunnya. Program ini dapat digunakan oleh siapa saja yang ingin melacak pemasukan dan pengeluaran pribadi. Pada Aplikasi ini, kami menawarkan fitur fitur penting untuk mengelola keuangan peribadi, antara lain:

* Fitur Menambah pemasukan, digunakan untuk menambahkan pemasukan.
* Fitur Menambah pengeluaran, digunakan untuk menambahkan pengeluaran.
* Fitur Mengurutkan data, digunakan untuk mengurutkan data pemasukan dan pengeluaran dari terkecil sampai terbesar.
* Fitur Mencari data, digunakan untuk mencari data pemasukan atau pengeluaran berdasarkan keterangan.
* Fitur Melihat data, digunakan untuk melihat data pemasukan dan pengeluaran secara keseluruhan.

Link repository GitHub : <https://bit.ly/savemoney-Apps>

1. **Fitur Menambah Pemasukan**



Fungsi **add\_income** digunakan untuk menambahkan pemasukan. Fungsi ini memiliki dua parameter, yaitu income\_list dan income\_desc\_list. Fungsi ini bertujuan untuk menambahkan pemasukan ke dalam dua list tersebut.

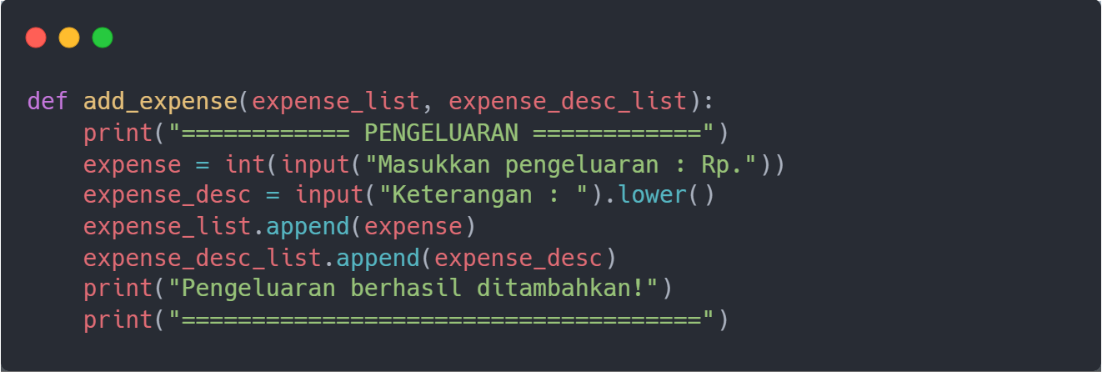
Kemudian, pengguna diminta untuk memasukkan jumlah pemasukan melalui fungsi input dengan prompt "Masukkan pemasukan : Rp.". Input tersebut kemudian diubah menjadi tipe data integer menggunakan fungsi int() dan disimpan dalam variabel income.

Selanjutnya, pengguna diminta untuk memasukkan keterangan mengenai pemasukan tersebut melalui fungsi input dengan prompt "Keterangan : ". Input tersebut kemudian diubah menjadi huruf kecil menggunakan metode lower() dan disimpan dalam variabel income\_desc.

Setelah itu, nilai income dan income\_desc ditambahkan ke akhir income\_list dan income\_desc\_list menggunakan metode append(), sehingga pemasukan tersebut ditambahkan ke dalam daftar.

Kemudian, pesan "Pemasukan berhasil ditambahkan!" dicetak untuk memberi tahu pengguna bahwa pemasukan telah berhasil ditambahkan.

1. **Fitur Menambahkan Pengeluaran**



Fungsi **add\_expense** digunakan untuk menambahkan pengeluaran. Fungsi ini memiliki dua parameter, yaitu expense\_list dan expense\_desc\_list. Fungsi ini digunakan untuk menambahkan pengeluaran ke dalam dua list tersebut. Fungsi add\_expense ini memiliki logika yang serupa dengan add\_income, tetapi fungsinya adalah untuk menambahkan pengeluaran ke dalam daftar pengeluaran dan keterangan pengeluaran ke dalam daftar keterangan pengeluaran.

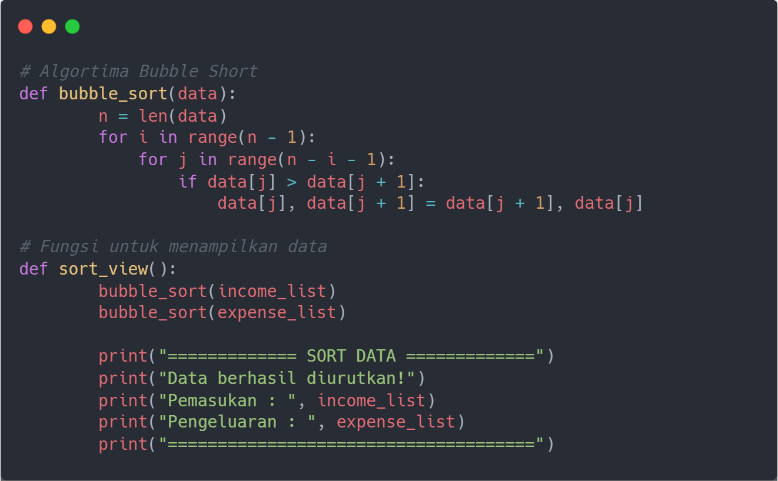
Pertama, kode meminta pengguna untuk memasukkan jumlah pengeluaran melalui fungsi input dengan prompt "Masukkan pengeluaran : Rp.". Input tersebut kemudian diubah menjadi tipe data integer menggunakan fungsi int() dan disimpan dalam variabel expense.

Selanjutnya, pengguna diminta untuk memasukkan keterangan mengenai pengeluaran tersebut melalui fungsi input dengan prompt "Keterangan : ". Input tersebut kemudian diubah menjadi huruf kecil menggunakan metode lower() dan disimpan dalam variabel expense\_desc.

Setelah itu, nilai expense dan expense\_desc ditambahkan ke akhir expense\_list dan expense\_desc\_list menggunakan metode append(), sehingga pengeluaran tersebut ditambahkan ke dalam daftar.

Kemudian, pesan "Pengeluaran berhasil ditambahkan!" dicetak untuk memberi tahu pengguna bahwa pengeluaran telah berhasil ditambahkan.

1. **Fitur Mengurutkan data**



Kode di atas terdiri dari dua bagian, yaitu fungsi bubble\_sort(data) dan fungsi sort\_view(). Fungsi-fungsi ini digunakan untuk mengurutkan data dalam income\_list dan expense\_list menggunakan algoritma pengurutan gelembung (bubble sort).

Fungsi bubble\_sort digunakan untuk mengurutkan data pemasukan dan pengeluaran secara terurut.

1. Fungsi ini menerima satu parameter data, yang merupakan sebuah list yang akan diurutkan.
2. Variabel n diinisialisasi dengan panjang (len()) dari data.
3. Dilakukan perulangan for dengan variabel i untuk mengatur jumlah iterasi yang sama dengan n - 1.
4. Di dalam perulangan for pertama, terdapat perulangan for kedua dengan variabel j. Perulangan ini akan mengiterasi dari 0 hingga n - i - 1.
5. Di dalam perulangan for kedua, dilakukan pengecekan apakah data[j] lebih besar dari data[j + 1]. Jika iya, maka terjadi pertukaran nilai menggunakan teknik unpacking data[j], data[j + 1] = data[j + 1], data[j], sehingga nilai yang lebih kecil bergerak ke posisi sebelumnya.
6. Setelah kedua perulangan selesai, fungsi bubble\_sort selesai menjalankan proses pengurutan dan kembali ke pemanggilannya.
7. Fungsi sort\_view():

Fungsi **sort\_view** digunakan untuk menampilkan data pemasukan atau pengeluaran uang sudah diurutkan kepada user sehingga user mudah melihat data pemasukan dan pengeluaran. Kedua fungsi ini saling berhubungan. Fungsi ini tidak menerima parameter.

1. Pertama, fungsi ini memanggil bubble\_sort untuk mengurutkan income\_list menggunakan algoritma pengurutan gelembung.
2. Selanjutnya, fungsi ini memanggil bubble\_sort untuk mengurutkan expense\_list menggunakan algoritma pengurutan gelembung.
3. Setelah itu, fungsi ini mencetak pesan "Data berhasil diurutkan!" untuk memberi tahu pengguna bahwa data telah berhasil diurutkan.
4. Kemudian, fungsi ini mencetak income\_list dan expense\_list yang sudah diurutkan.
5. Fungsi sort\_view selesai menjalankan semua perintahnya dan kembali ke pemanggilannya.
6. **Fitur Pencarian Data**



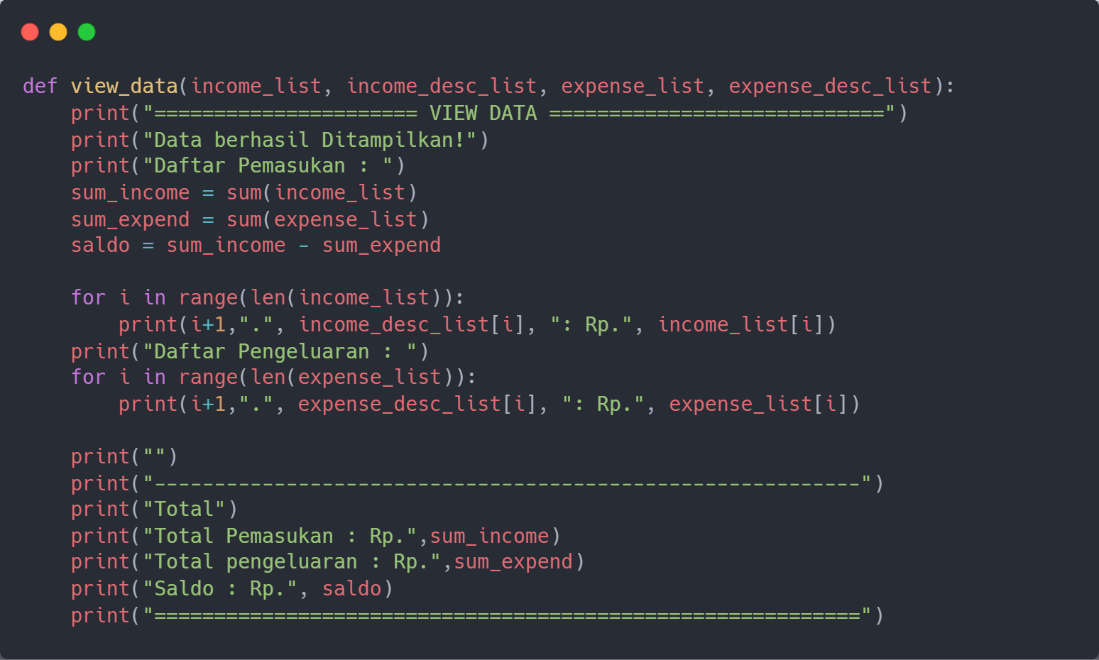
Fungsi **linear\_search** digunakan untuk mencari data pemasukan atau pengeluaran berdasarkan kategori. Fungsi **search** digunakan untuk mencari data pemasukan atau pengeluaran berdasarkan kategori pemasukan atau pengeluaran. Hal ini akan memudahkan user mencari pengeluaran atau pemasukan pada suatu waktu. Kedua fungsi ini saling berhubungan.

Fungsi linear\_search melakukan pencarian linear (secara berurutan) dalam list arr untuk mencari kemunculan pertama dari keyword. Jika keyword ditemukan dalam arr, fungsi ini akan mengembalikan indeks pertama di mana keyword ditemukan. Jika tidak ditemukan, fungsi ini mengembalikan nilai -1. Cara kerjanya yaitu

1. Fungsi linear\_search menerima dua parameter, yaitu arr yang merupakan list yang akan dicari, dan keyword yang merupakan kata kunci yang akan dicari dalam list.
2. Di dalam fungsi ini, terdapat perulangan for yang akan mengiterasi melalui setiap elemen dalam arr.
3. Pada setiap iterasi, dilakukan pengecekan apakah elemen saat ini yang diubah menjadi string (str(arr[i])) sama dengan keyword. Jika sama, maka fungsi akan mengembalikan indeks elemen tersebut menggunakan pernyataan return i.
4. Jika tidak ada elemen yang sesuai dengan keyword, maka fungsi akan mengembalikan -1 menggunakan pernyataan return -1.

Fungsi search memberikan pilihan kepada pengguna untuk mencari data pemasukan atau pengeluaran. Berdasarkan pilihan pengguna, fungsi ini melakukan langkah-langkah berikut dari fungsi ini

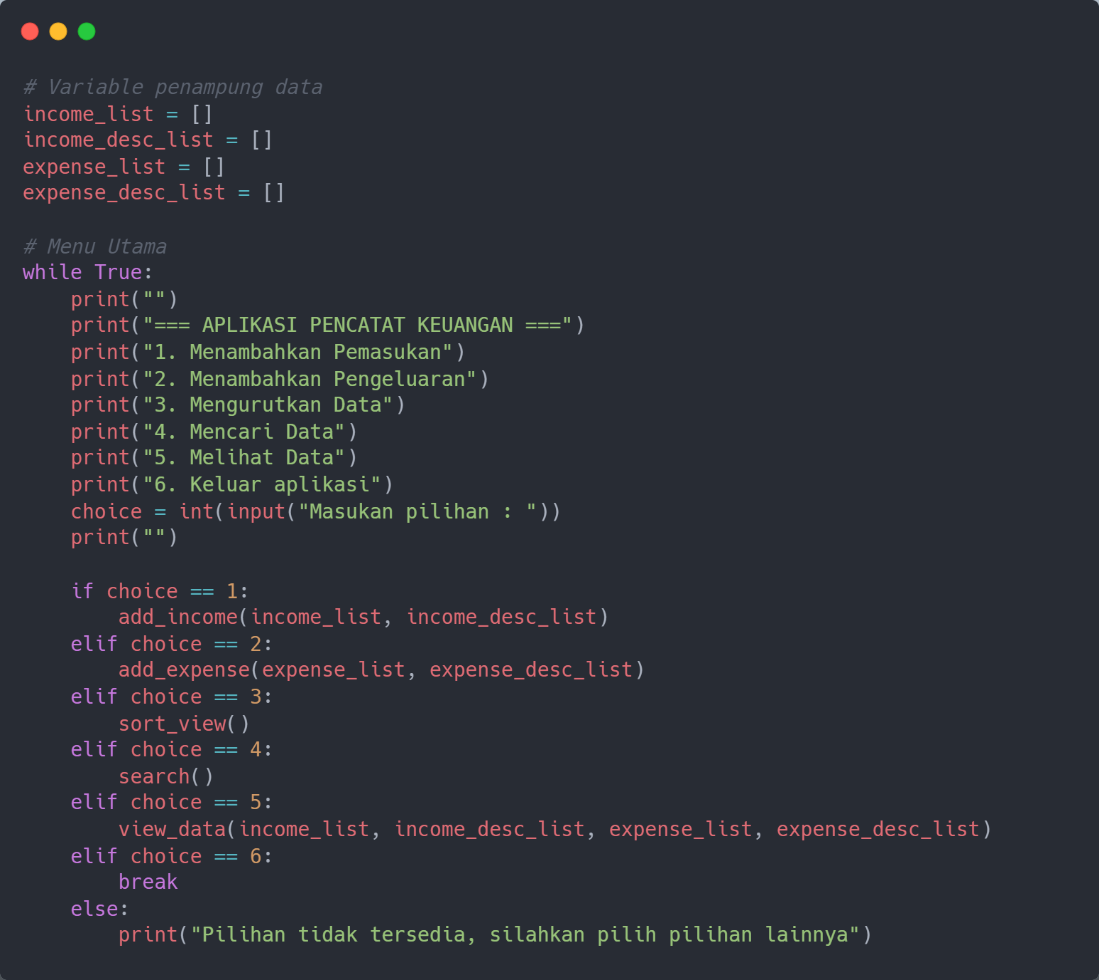
1. Fungsi ini tidak menerima parameter.
2. Pertama, fungsi ini mencetak pilihan menu yang tersedia.
3. Kemudian, fungsi ini meminta pengguna untuk memasukkan pilihan (choice) melalui fungsi input.
4. Jika pilihan (choice) adalah 1, maka fungsi akan mencetak daftar keterangan pemasukan (income\_desc\_list).
5. Selanjutnya, fungsi meminta pengguna untuk memasukkan kata kunci (keyword) yang sesuai dengan data di atas menggunakan fungsi input.
6. Setelah itu, fungsi akan memanggil fungsi linear\_search(income\_desc\_list, keyword) untuk mencari indeks pertama di mana keyword ditemukan dalam income\_desc\_list.
7. Jika hasil pencarian (result) adalah -1, maka fungsi mencetak pesan "Keyword tidak ditemukan". Jika tidak, fungsi mencetak pesan yang menyatakan pemasukan dengan keyword dan jumlahnya menggunakan indeks yang ditemukan dalam income\_list.
8. Jika pilihan (choice) adalah 2, fungsi akan melakukan langkah-langkah yang serupa untuk mencari pengeluaran (expense\_list) berdasarkan expense\_desc\_list.
9. Fungsi search selesai menjalankan semua perintahnya dan kembali ke pemanggilannya.
10. **Fitur Melihat data**



Fungsi **view\_data** digunakan untuk melihat data pemasukan dan pengeluaran secara keseluruhan. Berikut adalah implementasi fungsi tersebut. Fungsi ini memiliki empat parameter, yaitu income\_list, income\_desc\_list, expense\_list, dan expense\_desc\_list. Fungsi ini bertujuan untuk menampilkan data pemasukan dan pengeluaran beserta total pemasukan, total pengeluaran, dan saldo.

Langkah-langkah yang dilakukan oleh fungsi view\_data adalah sebagai berikut:

1. Pertama, fungsi mencetak pesan "Data berhasil Ditampilkan!" untuk memberi tahu pengguna bahwa data akan ditampilkan.
2. Selanjutnya, fungsi menghitung total pemasukan dengan menggunakan fungsi sum() pada income\_list dan menyimpannya dalam variabel sum\_income. Begitu juga dengan total pengeluaran, dihitung dengan fungsi sum() pada expense\_list dan disimpan dalam variabel sum\_expend.
3. Fungsi juga menghitung saldo dengan mengurangi sum\_expend dari sum\_income dan menyimpannya dalam variabel saldo.
4. Setelah itu, fungsi melakukan perulangan for untuk menampilkan daftar pemasukan. Setiap elemen dalam income\_list dan income\_desc\_list dicetak dengan format yang sesuai, yaitu nomor urut, keterangan pemasukan, dan jumlah pemasukan.
5. Setelah mencetak daftar pemasukan, fungsi melakukan perulangan for untuk menampilkan daftar pengeluaran dengan format yang serupa seperti langkah sebelumnya.
6. Setelah itu, fungsi mencetak pesan "Total" untuk menandakan bahwa data total akan ditampilkan.
7. Fungsi mencetak total pemasukan dengan format yang sesuai, yaitu "Total Pemasukan : Rp." diikuti oleh nilai sum\_income.
8. Selanjutnya, fungsi mencetak total pengeluaran dengan format yang sesuai, yaitu "Total pengeluaran : Rp." diikuti oleh nilai sum\_expend.
9. Terakhir, fungsi mencetak saldo dengan format yang sesuai, yaitu "Saldo : Rp." diikuti oleh nilai saldo.
10. Dengan demikian, fungsi view\_data akan menampilkan secara terstruktur daftar pemasukan, daftar pengeluaran, total pemasukan, total pengeluaran, dan saldo ke layar.
11. **Fitur Menu**



Kode di atas merupakan sebuah loop while yang akan menjalankan sebuah menu aplikasi pengelolaan keuangan. Pengguna akan terus diberikan pilihan menu hingga memilih untuk keluar dari aplikasi.

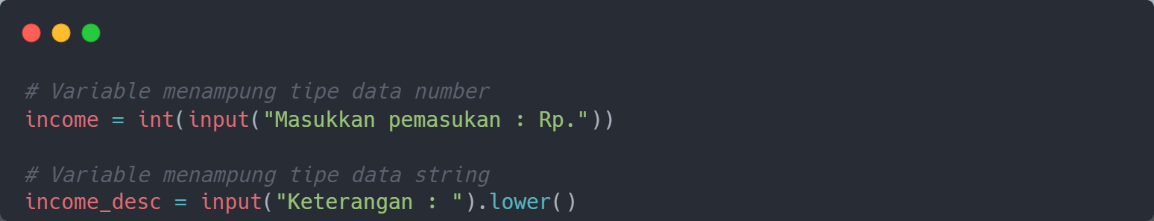
Variable income\_list, income\_desc\_list, expense\_list, expense\_desc\_list adalah variable utama yang menampung data pemasukan dan pengeluaran.

Cara kerja fitur menu adalah sebagai berikut :

1. Loop while akan terus berjalan selama kondisi True, sehingga menu akan terus ditampilkan sampai pengguna memilih untuk keluar.
2. Pada setiap iterasi loop, kode akan mencetak pilihan menu yang tersedia, yaitu menambahkan pemasukan, menambahkan pengeluaran, mengurutkan data, mencari data, melihat data, atau keluar dari aplikasi.
3. Pengguna diminta untuk memasukkan pilihan menu melalui fungsi input yang kemudian dikonversi menjadi bilangan bulat dengan menggunakan int(input()).
4. Setelah memasukkan pilihan, akan ada pengujian kondisi if untuk menentukan langkah selanjutnya berdasarkan pilihan yang dimasukkan.
5. Jika pilihan adalah 1, fungsi add\_income akan dipanggil dengan argumen income\_list dan income\_desc\_list untuk menambahkan pemasukan.
6. Jika pilihan adalah 2, fungsi add\_expense akan dipanggil dengan argumen expense\_list dan expense\_desc\_list untuk menambahkan pengeluaran.
7. Jika pilihan adalah 3, fungsi sort\_view akan dipanggil untuk mengurutkan data pemasukan dan pengeluaran.
8. Jika pilihan adalah 4, fungsi search akan dipanggil untuk mencari data pemasukan atau pengeluaran berdasarkan kata kunci.
9. Jika pilihan adalah 5, fungsi view\_data akan dipanggil untuk melihat semua data yang telah diinputkan.
10. Jika pilihan adalah 6, loop akan dihentikan dengan pernyataan break dan pengguna akan keluar dari aplikasi.
11. Jika pilihan tidak cocok dengan menu yang tersedia, pesan "Pilihan tidak tersedia, silahkan pilih pilihan lainnya" akan dicetak.
12. Dengan demikian, pengguna dapat menggunakan menu yang disediakan untuk mengelola dan melihat data pemasukan, pengeluaran, dan saldo mereka.
13. **Implementasi Materi Kuliah**

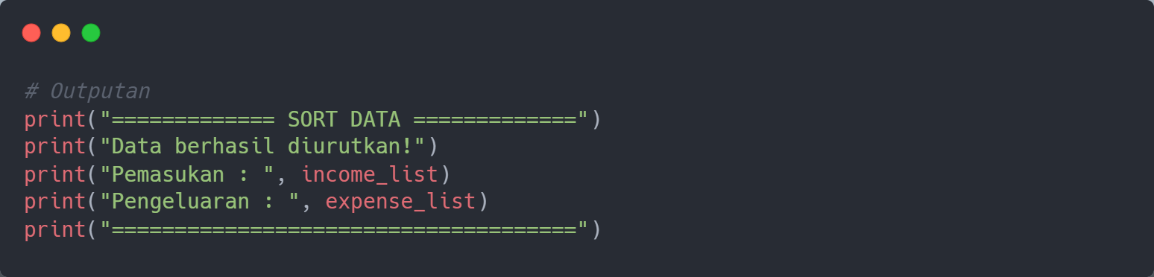
Berikut implementasi materi kuliah yang ada di program kita.

* 1. Variable dan Tipe Data



* 1. Inputan dan Outputan





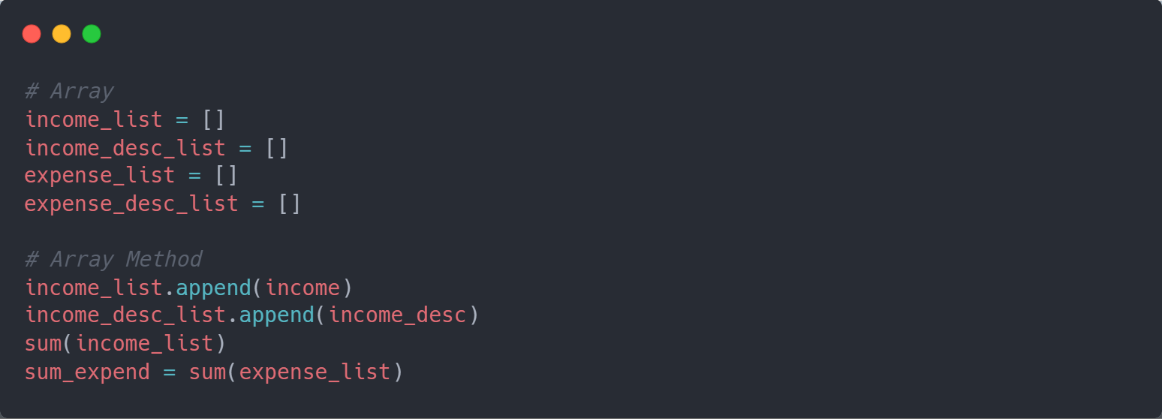
* 1. Percabangan



* 1. Perulangan



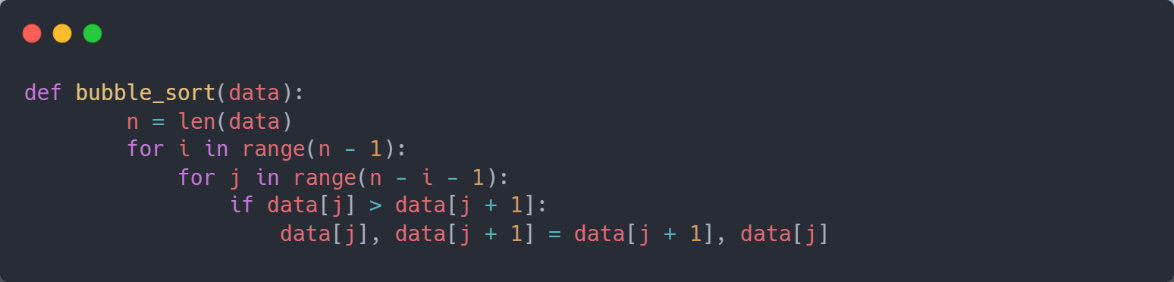
* 1. Array dan Array method



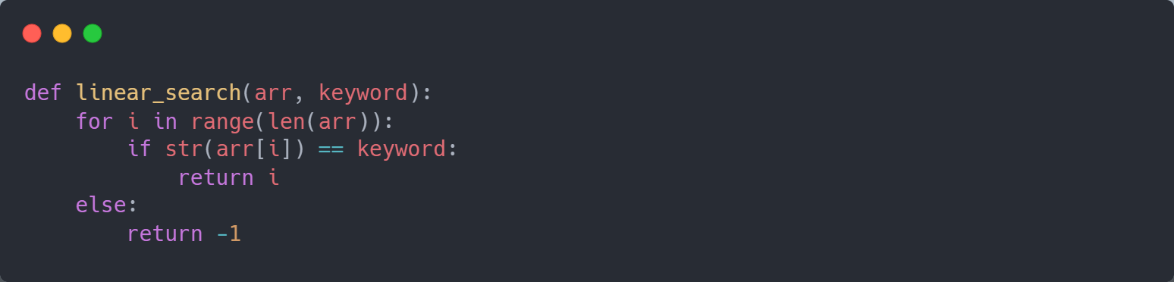
* 1. Fungsi dan Prosedur



* 1. Algoritma Sorting



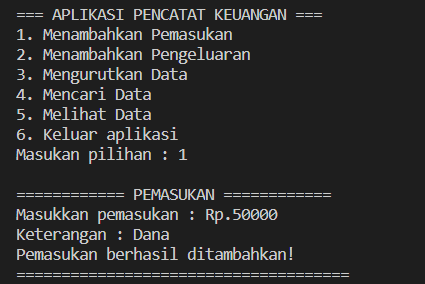
* 1. Algoritma Searching

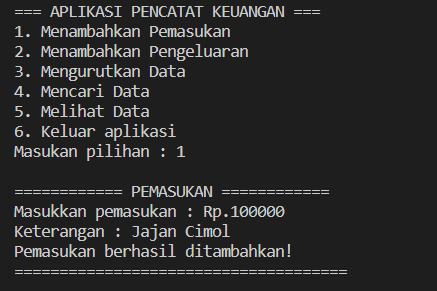


1. **Pengujian Aplikasi**

Aplikasi ini telah melalui tahap pengujian yang meliputi:

1. Menambahkan pemasukan



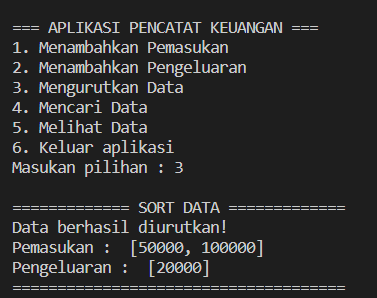


1. Menambahkan pengeluaran

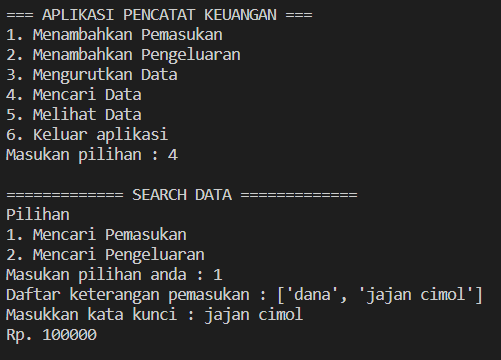
A screen shot of a computer

Description automatically generated with medium confidence

1. Mengurutkan data



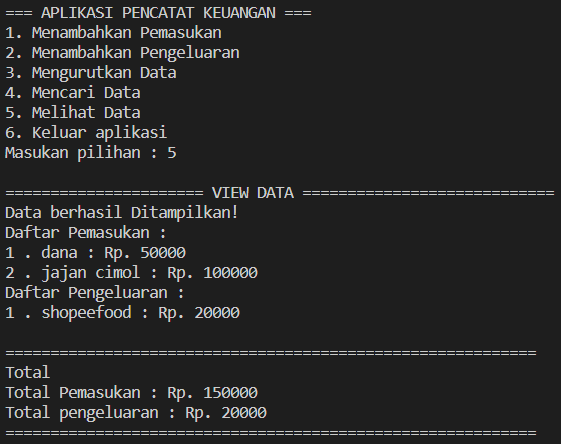
1. Mencari data



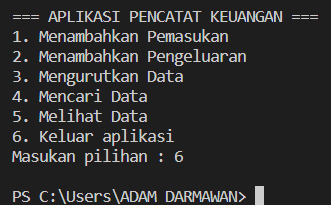
A screenshot of a computer

Description automatically generated with medium confidence

1. Melihat data



1. Keluar Aplikasi



1. **Timeline Pengerjaan**

Berikut adalah timeline pengerjaan aplikasi Savemoney beserta program pendukung yang terkait dengan pengerjaan

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Jenis Kegiatan** | | **Juni** | | | | | | | | | |
| **15** | **16** | **17** | **18** | **19** | **20** | **21** | **22** | **23** | **24** |
| 1 | Perencanaan Program   * Ide * Skema Dasar * Sistem dan Fitur * Repository Github * Pembagian Tugas | |  |  |  |  |  |  |  |  |  |  |
| 2 | Fitur Pemasukan | |  |  |  |  |  |  |  |  |  |  |
| 3 | Fitur Pengeluaran |  |  |  |  |  |  |  |  |  |  |
| 4 | Fitur Mengurutkan Data | |  |  |  |  |  |  |  |  |  |  |
| 5 | Fitur Mencari Data |  |  |  |  |  |  |  |  |  |  |
| 6 | Fitur Melihat Data | |  |  |  |  |  |  |  |  |  |  |
| 7 | Pembuatan Laporan | |  |  |  |  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **No** | **Jenis Kegiatan** | **Juni** | | | | | | **Juli** | | | |
| **25** | **26** | **27** | **28** | **29** | **30** | **1** | **2** | **3** | **4** |
| 1 | Perencanaan Program   * Ide * Skema Dasar * Sistem dan Fitur * Repository Github * Pembagian Tugas |  |  |  |  |  |  |  |  |  |  |
| 2 | Fitur Pemasukan |  |  |  |  |  |  |  |  |  |  |
| 3 | Fitur Pengeluaran |  |  |  |  |  |  |  |  |  |  |
| 4 | Fitur Mengurutkan Data |  |  |  |  |  |  |  |  |  |  |
| 5 | Fitur Mencari Data |  |  |  |  |  |  |  |  |  |  |
| 6 | Fitur Melihat Data |  |  |  |  |  |  |  |  |  |  |
| 7 | Pengecekan Ulang, Pengujiam Akhir, dan Debugging |  |  |  |  |  |  |  |  |  |  |
| 8 | Pembuatan Laporan |  |  |  |  |  |  |  |  |  |  |

1. **Pembagian Tugas**

|  |  |  |
| --- | --- | --- |
| 1 | Nama Lengkap | Adam Darmawan |
| 2 | NIM | 2211104034 |
| 3 | Kelas | S1-SE06B |
| 4 | Tugas | * Programmer 3   + Fitur view data   + Sistem Transaksi   + Pemeriksaan Program * Program tester 2   + Pengujian akhir * Pembuat dan Penanggung jawab Laporan 1 |

|  |  |  |
| --- | --- | --- |
| 1 | Nama Lengkap | Satria Ariq Adelard Dompas |
| 2 | NIM | 2211104033 |
| 3 | Kelas | S1-SE06B |
| 4 | Tugas | * Programmer 2   + Fitur pengurutan data   + Debugger * Program tester 3   + Pengujian akhir * Pengecekan ulang laporan |

|  |  |  |
| --- | --- | --- |
| 1 | Nama Lengkap | Salman Alfa Rizzi |
| 2 | NIM | 2211104056 |
| 3 | Kelas | S1-SE06B |
| 4 | Tugas | * Programmer 5   + Fitur pemasukan   + Debugger * Program tester 5   + Pengecekan ulang * Pengecekan ulang laporan |

|  |  |  |
| --- | --- | --- |
| 1 | Nama Lengkap | Arwin Nabiel Arrofif |
| 2 | NIM | 2211104057 |
| 3 | Kelas | S1-SE06B |
| 4 | Tugas | * Programmer 4   + Fitur pengeluaran   + Debugger * Program tester 4   + Pengecekan ulang * Pengecekan ulang laporan |

|  |  |  |
| --- | --- | --- |
| 1 | Nama Lengkap | Yogi Hafidh Maulana |
| 2 | NIM | 2211104061 |
| 3 | Kelas | S1-SE06B |
| 4 | Tugas | * Developer   + Pengembangan Program Utama   + Fitur Menu   + Fitur saldo   + Fitur Pencarian data   + Debugger * Program Tester 1   + Pengujian akhir   + Pengecekan ulang * Pembuatan dan Penanggung jawab laporan 2 |