

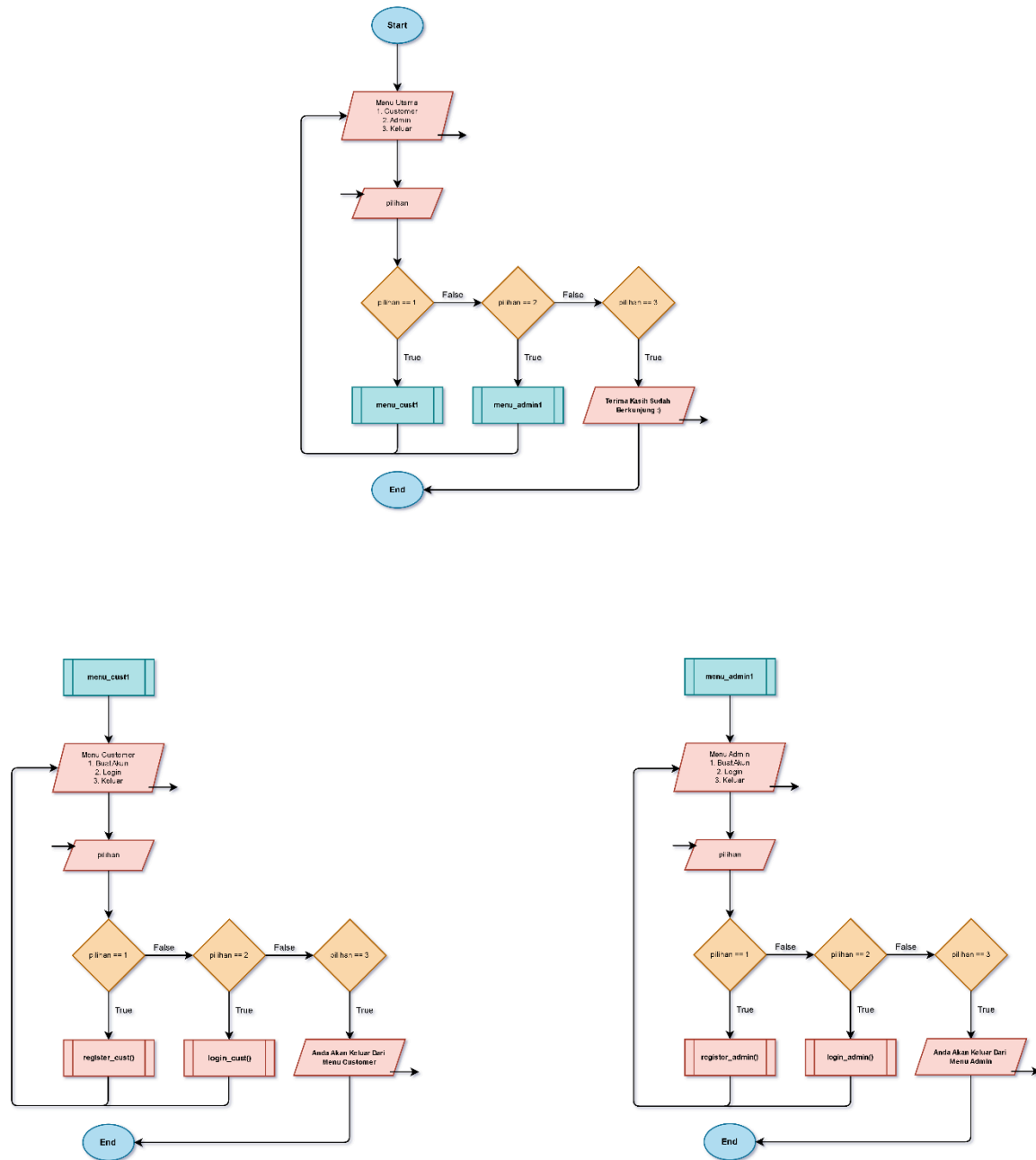
**LAPORAN PRAKTIKUM**  
**POSTTEST 8**  
**ALGORITMA PEMROGRAMAN DASAR**



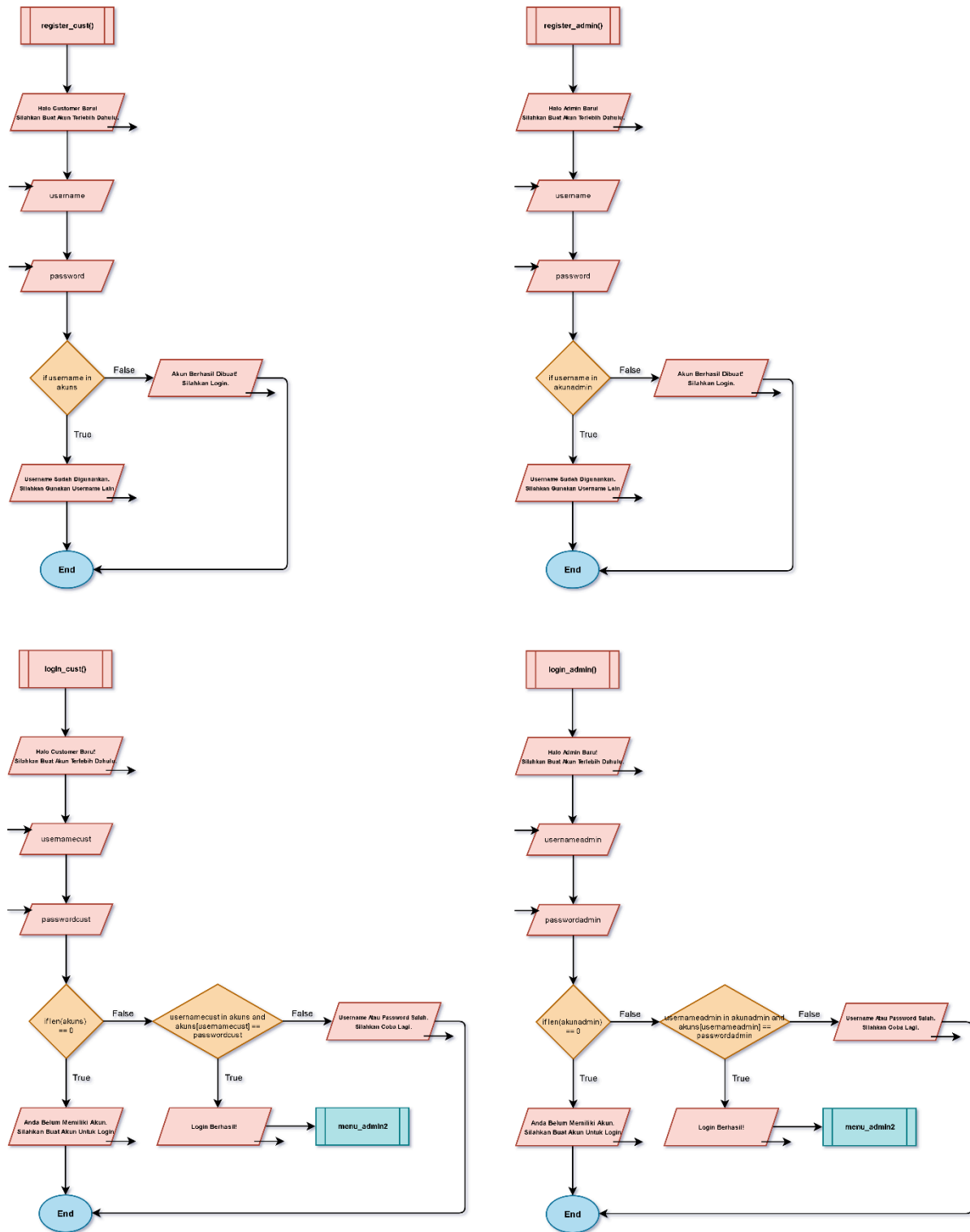
**Disusun oleh:**  
**Muhamad Farid Al Mubarak (2509106087)**  
**Kelas (B2 '25)**

**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA**  
**2025**

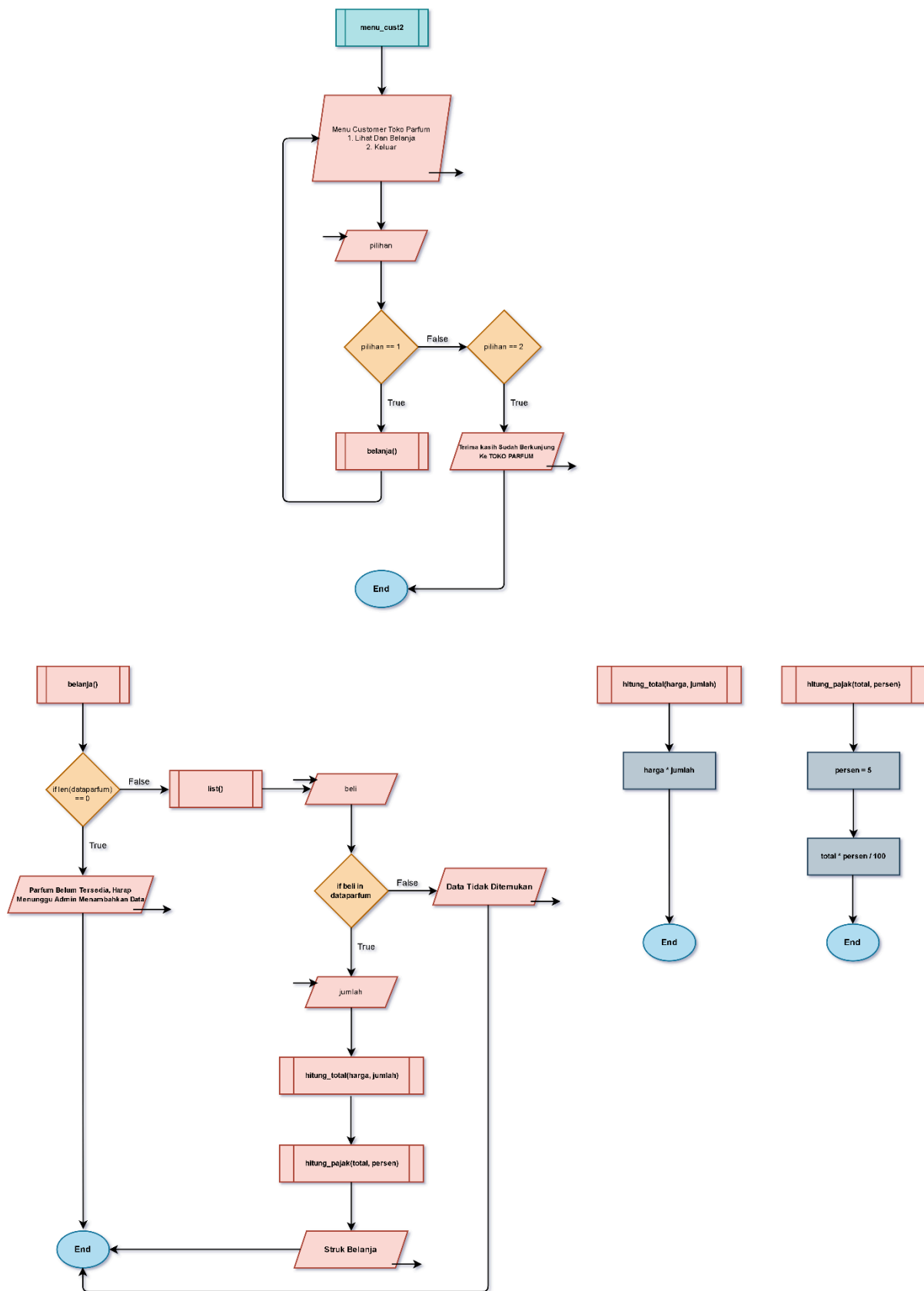
# 1. Flowchart



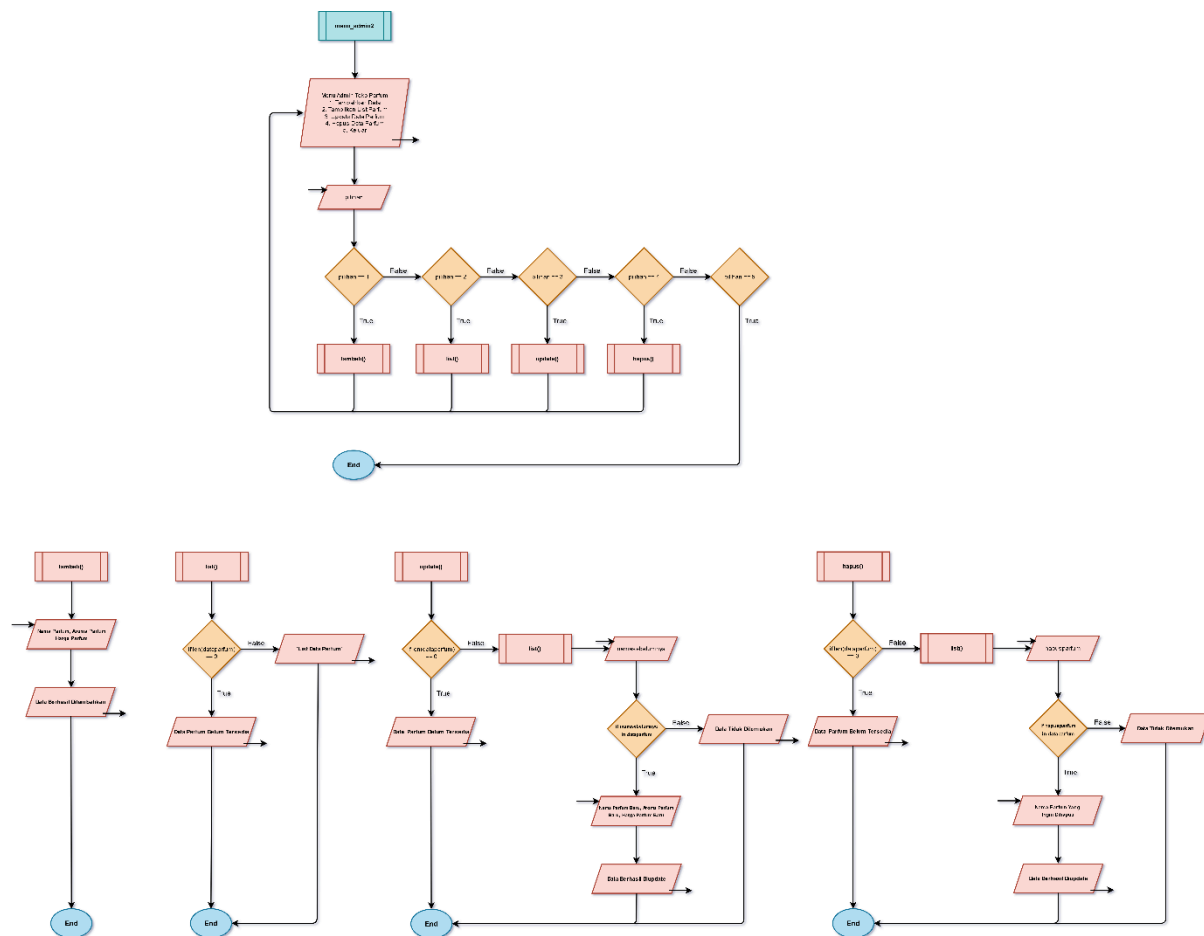
Gambar 1.1 Flowchart Bagian Pertama



Gambar 1.2 Flowchart Bagian Kedua



Gambar 1.3 Flowchart Bagian Ketiga



Gambar 1.4 Flowchart Bagian Keempat

## Penjelasan Tentang Flowchart

### 1. Gambar 1.1

Flowchart akan dimulai dengan **START** dan menampilkan menu utama, di mana pengguna bisa memilih opsi login sebagai Customer atau Admin, dan Keluar dari program. Perbedaan Customer dan Admin yaitu Customer hanya bisa melihat dan berbelanja barang di toko, sedangkan Admin Memiliki Akses Manajemen Toko untuk menambah, menampilkan, memperbarui, dan menghapus data (Sistem CRUD).

Jika pengguna memilih opsi 1 untuk login sebagai Customer, lalu akan diarahkan ke menu Customer. Setelah memilih opsi 1, Program akan menampilkan menu Customer yang berisi tiga pilihan, yaitu Registrasi atau buat akun, Login, dan Keluar dari menu Customer.

Jika pengguna memilih opsi 2 untuk login sebagai Admin lalu akan diarahkan ke menu Admin. Sama halnya dengan menu Customer, menu Admin juga bisa Registrasi atau buat akun, Login dan Keluar dari menu Admin.

Jika pengguna memilih opsi 3, maka program akan berhenti dan Flowchart diakhiri **END**.

## **2. Gambar 1.2**

Pada bagian ini terdapat empat fungsi yang digunakan untuk registrasi dan login, masing-masing untuk Customer dan Admin. Dalam proses registrasi, pengguna diminta untuk memasukkan username dan password. Program kemudian akan memeriksa apakah username tersebut sudah pernah digunakan atau belum. Jika belum, akun akan berhasil dibuat, dan pengguna bisa melanjutkan ke proses login.

Pada bagian login, pengguna harus memasukkan username dan password yang telah dibuat sebelumnya. Program akan memeriksa apakah keduanya benar. Jika sesuai, login dinyatakan berhasil. Namun, jika salah, maka login akan gagal.

## **3. Gambar 1.3**

Bagian ini menampilkan menu Customer Toko Parfum yang memiliki dua pilihan, yaitu Belanja dan Keluar.

Jika Customer memilih opsi 1 (Belanja), maka pengguna dapat melihat dan membeli parfum. Setelah itu, Customer diminta untuk memasukkan jumlah parfum yang diinginkan. Program kemudian akan menghitung pajak dan total harga, lalu menampilkan struk belanja, sebelum akhirnya kembali ke menu Customer Toko Parfum. Jika Customer memilih opsi 2 (Keluar), maka Customer akan keluar dari menu Customer Toko Parfum.

## **4. Gambar 1.4**

Pada bagian ini akan ditampilkan menu admin Toko Parfum, dimana admin bisa menambah, menampilkan, memperbarui dan menghapus data.

- Jika Admin memilih opsi 1 (Tambah Data), maka Admin dapat menambahkan data baru seperti nama parfum, aroma parfum dan harga parfum.
- Jika Admin memilih opsi 2 (Tampilkan Data), maka program akan menampilkan data yang sebelumnya ditambahkan oleh Admin.

- Jika Admin memilih opsi 3 (Update Data), maka Admin bisa memilih data yang ingin diupdate, dengan menginput nama parfum yang ingin diubah, Admin bisa memperbarui data.
- Jika Admin memilih opsi 4 (Hapus Data), maka Admin bisa memilih data yang ingin dihapus, dengan menginput nama parfum yang ingin dihapus, lalu data otomatis akan dihapus dari list.
- Jika Admin memilih opsi 5 (Keluar), maka Admin keluar dari menu Admin Toko Parfum

## 2. Deskripsi Singkat Program

Program ini adalah simulasi sederhana sistem manajemen dan penjualan parfum yang memungkinkan pengguna untuk mendaftar dan masuk sebagai Admin atau Customer. Sebagai Admin, pengguna dapat menambah, melihat, mengubah, dan menghapus daftar parfum. Sebagai Customer, pengguna dapat melihat daftar parfum yang telah ditambahkan oleh Admin dan melakukan simulasi pembelian. Semua data akun dan parfum disimpan sementara di memori selama program berjalan.

Ada juga beberapa Library yang saya gunakan di Program ini:

- Library Inquirer :
- Library PrettyTable :
- Library Pwinput :
- Library DateTime :

## 3. Source Codes

### 3.1 Penyimpanan Data

Bagian ini digunakan untuk menyimpan data selama program berjalan. Terdiri dari 5 variabel global yaitu akuns, akunadmin, dataparfum, transaksi, dan nama terakhir pembeli.

#### Source Code:

```
akuns = {}
akunadmin = {}
dataparfum = {}
transaksi = 0
```

```
nama_terakhir = ''
```

### 3.2 Menu Utama

Bagian ini berfungsi untuk menampilkan pilihan login sebagai Customer atau Admin serta opsi untuk keluar dari program.

#### Source Code:

```
import inquirer
import os
os.system('cls')
from prettytable import PrettyTable
from customer1 import menu_cust1
from admin1 import menu_admin1

def menu_utama():
    while True:
        menu_utama = PrettyTable()
        menu_utama.field_names = ["No", "Pilihan Menu"]
        menu_utama.align = "l"
        menu_utama.hrules = 1
        menu_utama.add_row(['1', 'CUSTOMER'])
        menu_utama.add_row(['2', 'ADMIN'])
        menu_utama.add_row(['3', 'KELUAR'])
        os.system('cls')
        print('=== SELAMAT DATANG DI TOKO PARFUM X ===')
        print('Anda Login Sebagai Siapa?')
        print(menu_utama)

        pertanyaan = [
            inquirer.List(
                'pilihan',
                message='Silahkan Pilih Opsi Login',
                choices=[('1. Customer', 1),
                        ('2. Admin', 2),
                        ('3. Keluar', 3)]
```



```
)
]

answers = inquirer.prompt(pertanyaan)

if answers['pilihan'] == 1:
    menu_cust1()
elif answers['pilihan'] == 2:
    menu_admin1()
else:
    answers["pilihan"] == 3
    print('\nTerima Kasih Sudah Berkunjung :)')
    exit()
menu_utama()
```

### 3.3 Menu Customer Pertama

Bagian ini Menampilkan Menu khusus untuk memberikan pilihan kepada Customer untuk membuat akun, login, atau keluar dari menu ini.

**Source Code:**

```

import inquirer
import os
os.system('cls')
from prettytable import PrettyTable
from registrasi import register_cust
from login import login_cust

def menu_cust1():
    while True:
        os.system('cls')
        menu_customer_1 = PrettyTable()
        menu_customer_1.field_names = ["No", "Pilihan Menu"]
        menu_customer_1.align = "l"
        menu_customer_1.hrules = 1
        menu_customer_1.add_row(['1', 'BUAT AKUN'])
        menu_customer_1.add_row(['2', 'LOGIN'])
        menu_customer_1.add_row(['3', 'KELUAR'])
        print('Selamat Datang Di MENU CUSTOMER!')
        print(menu_customer_1)

        pertanyaan = [
            inquirer.List(
                'pilihan',
                message='Silahkan Pilih Opsi Yang Ada Di Menu',
                choices=[('1. Buat Akun', 1),
                        ('2. Login', 2),
                        ('3. Keluar', 3)]
            )
        ]

        answers = inquirer.prompt(pertanyaan)

        if answers['pilihan'] == 1:
            register_cust()
        elif answers['pilihan'] == 2:
            login_cust()
        else:
            answers['pilihan'] == 3
            print('\nAnda Akan Keluar Dari Menu Customer')
            input('\nTekan Enter Untuk Kembali Ke Menu...')
            break

```

### 3.4 Menu Customer Kedua

Bagian ini akan ditampilkan setelah Customer berhasil login. Berisi fitur untuk melihat dan belanja parfum atau opsi keluar dari menu ini.

**Source Code:**

```

import inquirer
import os
os.system('cls')
from prettytable import PrettyTable
from belanja import belanja

def menu_cust2():
    while True:
        os.system('cls')
        menu_customer_2 = PrettyTable()
        menu_customer_2.field_names = ["No", "Pilihan Menu"]
        menu_customer_2.align = "l"
        menu_customer_2.hrules = 1
        menu_customer_2.add_row(['1', 'LIHAT & BELANJA'])
        menu_customer_2.add_row(['2', 'KELUAR'])
        print('Selamat Datang Di MENU CUSTOMER TOKO PARFUM X!')
        print(menu_customer_2)

        pertanyaan = [
            inquirer.List(
                'pilihan',
                message='Silahkan Pilih Opsi Yang Ada Di Menu',
                choices=[('1. Lihat & Belanja', 1),
                        ('2. Keluar', 2) ]
            )
        ]

        answers = inquirer.prompt(pertanyaan)

        if answers['pilihan'] == 1:
            belanja()
        else:
            answers['pilihan'] == 2
            print('\nAnda Akan Keluar Dari Menu Customer')
            input('\nTekan Enter Untuk Kembali Ke Menu...')
            break

```

### 3.5 Menu Admin Pertama

Bagian ini Menampilkan Menu khusus untuk memberikan pilihan kepada Admin untuk membuat akun, login, atau keluar dari menu ini.

**Source Code:**

```

import inquirer
import os
os.system('cls')
from prettytable import PrettyTable
from registrasi import register_admin
from login import login_admin

def menu_admin1():
    while True:
        os.system('cls')
        menu_admin_1 = PrettyTable()
        menu_admin_1.field_names = ["No", "Pilihan Menu"]
        menu_admin_1.align = "l"
        menu_admin_1.hrules = 1
        menu_admin_1.add_row(['1', 'BUAT AKUN'])
        menu_admin_1.add_row(['2', 'LOGIN'])
        menu_admin_1.add_row(['3', 'KELUAR'])
        print('Selamat Datang Di MENU ADMIN!')
        print(menu_admin_1)

        pertanyaan = [
            inquirer.List(
                'pilihan',
                message='Silahkan Pilih Opsi Yang Ada Di Menu',
                choices=[('1. Buat Akun', 1),
                        ('2. Login', 2),
                        ('3. Keluar', 3)]
            )
        ]

        answers = inquirer.prompt(pertanyaan)

        if answers['pilihan'] == 1:
            register_admin()
        elif answers['pilihan'] == 2:
            login_admin()
        else:
            answers['pilihan'] == 3
            print('\nAnda Akan Keluar Dari Menu Login')
            input('\nTekan Enter Untuk Kembali Ke Menu Utama...')
            break

```

### 3.6 Menu Admin Kedua

Bagian ini akan ditampilkan setelah Admin berhasil login. Berisi fitur CRUD untuk menambah, menampilkan, meng-update, dan menghapus data parfum serta keluar dari menu ini.

**Source Code:**

```

import inquirer
import os
os.system('cls')
from prettytable import PrettyTable
from crud import tambah, list, update, hapus

def menu_admin2():
    while True:
        os.system('cls')
        menu_admin_2 = PrettyTable()
        menu_admin_2.field_names = ["No", "Pilihan Menu"]
        menu_admin_2.align = "l"
        menu_admin_2.hrules = 1
        menu_admin_2.add_row(['1', 'TAMBAHKAN PARFUM'])
        menu_admin_2.add_row(['2', 'TAMPILKAN LIST PARFUM'])
        menu_admin_2.add_row(['3', 'UPDATE DATA PARFUM'])
        menu_admin_2.add_row(['4', 'HAPUS DATA PARFUM'])
        menu_admin_2.add_row(['5', 'KELUAR'])
        print('Selamat Datang Di MENU ADMIN TOKO PARFUM X!')
        print(menu_admin_2)

        pertanyaan = [
            inquirer.List(
                'pilihan',
                message='Silahkan Pilih Opsi Yang Ada Di Menu',
                choices=[('1. TAMBAH', 1),
                        ('2. LIST', 2),
                        ('3. UPDATE', 3),
                        ('4. HAPUS', 4),
                        ('5. KELUAR', 5), ]
            )
        ]

        answers = inquirer.prompt(pertanyaan)

        if answers['pilihan'] == 1:
            tambah()
        elif answers['pilihan'] == 2:
            list()
        elif answers['pilihan'] == 3:
            update()
        elif answers['pilihan'] == 4:
            hapus()
        else:
            answers['pilihan'] == 5
            print('\nAnda Akan Keluar Dari Menu Admin')
            input('\nTekan Enter Untuk Kembali Ke Menu...')
            break

```

### 3.7 Registrasi dan Login

Bagian ini berfungsi untuk pembuatan akun dan proses login Customer maupun Admin. Setiap Customer atau Admin mempunyai fungsi masing-masing untuk registrasi dan login.

#### Source Code:

```
def register_cust():
    print('\nHalo Customer baru!, Silahkan Buat Akun Terlebih Dahulu.')

    username = input('Masukkan Username: ')
    password = input('Masukkan Password: ')
    if username in akuns:
        print('\nUsername Sudah Digunakan. Silahkan Gunakan Username Lain!')
        input('\nTekan Enter Untuk Kembali...')
    else:
        akuns[username] = password
        print('\nAkun Berhasil Dibuat! Silakan Login.')
        input('\nTekan Enter Untuk Kembali Ke Menu...')

def register_admin():
    print('\nHalo Admin baru!, Silahkan Buat Akun Terlebih Dahulu.')

    username = input('Masukkan Username: ')
    password = input('Masukkan Password: ')
    if username in akunadmin:
        print('\nUsername Sudah Digunakan. Silahkan Gunakan Username Lain!')
        input('\nTekan Enter Untuk Kembali...')
    else:
        akunadmin[username] = password
        print('\nAkun Berhasil Dibuat! Silakan Login.')
        input('\nTekan Enter Untuk Kembali Ke Menu...')

def login_cust():
    print('\nHalo Customer!, Silahkan Login Terlebih Dahulu.')

    usernamecust = input('Masukkan Username: ')
    passwordcust = input('Masukkan Password: ')

    if len(akuns) == 0:
        print('\nAnda Belum Memiliki Akun. Silahkan Buat Akun Untuk Login!')
        input('\nTekan Enter Untuk Kembali...')
    elif usernamecust in akuns and akuns[usernamecust] == passwordcust:
        print('\nLogin Berhasil!')
        input('\nTekan Enter Untuk Melanjutkan...')
        menu_cust2()
    else:
        print('\nUsername atau Password Salah. Silahkan Coba Lagi')
```

```

        input('\nTekan Enter Untuk Kembali...')

def login_admin():
    print('\nHalo Admin!, Silahkan Login Terlebih Dahulu.')

    usernameadmin = input('Masukkan Username: ')
    passwordadmin = input('Masukkan Password: ')

    if len(akunadmin) == 0:
        print('\nAnda Belum Memiliki Akun. Silahkan Buat Akun Untuk Login!')
        input('\nTekan Enter Untuk Kembali...')
    elif usernameadmin in akunadmin and akunadmin[usernameadmin] ==
passwordadmin:
        print('\nLogin Berhasil!')
        input('\nTekan Enter Untuk Melanjutkan...')
        menu_admin2()
    else:
        print('\nUsername atau Password Salah. Silahkan Coba Lagi')
        input('\nTekan Enter Untuk Kembali...')

```

### 3.8 Fungsi CRUD

Bagian ini akan ditampilkan setelah Admin berhasil login. Berisi fitur CRUD untuk menambah, menampilkan, meng-update, dan menghapus data parfum serta keluar dari menu ini.

#### Source Code:

```

def tambah():
    print('\n=== TAMBAH DATA PARFUM ===')
    nama = input('Nama Parfum: ')
    aroma = input('Aroma Parfum: ')
    try:
        harga = int(input('Harga Parfum: Rp. '))
    except ValueError:
        print('\nHarga Harus Berupa Angka!')
        input('\nTekan Enter Untuk Kembali...')
    return
    dataparfum[nama] = {'Aroma': aroma, 'Harga': harga}
    print('\nData berhasil ditambahkan!')
    input('\nTekan Enter Untuk Kembali...')

def list():
    print('\n=== LIST DATA PARFUM ===')
    if len(dataparfum) == 0:
        print('\nData Parfum Belum Tersedia.')
        input('\nTekan Enter Untuk Kembali...')
    else:

```

```

        urutan = 1
        for nama, info in dataparfum.items():
            print(f'\nParfum Ke-{urutan}')
            print(f>Nama : {nama}')
            print(f'Aroma : {info['Aroma']}')
            print(f>Harga : Rp. {info['Harga']}')
            urutan += 1
        input('\nTekan Enter Untuk Lanjut...')

def update():
    if len(dataparfum) == 0:
        print("\nParfum Belum Tersedia, Silahkan Pilih 1 Untuk Menambahkan Data.")
        input('\nTekan Enter Untuk Kembali...')
        return
    list()

    print('\n=== UPDATE DATA PARFUM ===')
    namasebelumnya = input('\nMasukkan Nama Parfum yang Ingin Diupdate: ')
    if namasebelumnya in dataparfum:
        namabarur = input('Nama baru: ')
        aromabarur = input('Aroma baru: ')
        try:
            hargabarur = int(input('Harga baru: Rp. '))
        except ValueError:
            print('\nHarga Harus Berupa Angka!')
            input('\nTekan Enter Untuk Kembali...')
            return
        dataparfum.pop(namasebelumnya)
        dataparfum[namabarur] = {'Aroma': aromabarur, 'Harga': hargabarur}
        print('\nData berhasil diupdate!')
        input('\nTekan Enter Untuk Kembali...')
    else:
        print('\nData Tidak Ditemukan')
        input('\nTekan Enter Untuk Kembali...')

def hapus():
    if len(dataparfum) == 0:
        print("\nParfum Belum Tersedia, Silahkan Pilih 1 Untuk Menambahkan Data.")
        input('\nTekan Enter Untuk Kembali...')
        return
    list()
    print('\n=== HAPUS DATA PARFUM ===')
    hapusparfum = input('\nMasukkan Nama Parfum yang Ingin Dihapus: ')
    if hapusparfum in dataparfum:
        dataparfum.pop(hapusparfum)
        print('\nData berhasil dihapus!')
        input('\nTekan Enter Untuk Kembali...')
    else:
        print('\nData Tidak Ditemukan')

```



```
input('\nTekan Enter Untuk Kembali...')
```

### 3.9 Fungsi Perhitungan

Bagian ini berfungsi untuk menghitung total harga dan pajak pembelian parfum. Fungsi ini digunakan di Fungsi belanja.

#### Source Code:

```
def hitung_total(harga, jumlah):  
    return harga * jumlah  
  
def hitung_pajak(total, persen):  
    persen = 5  
    return total * persen / 100
```

### 3.10 Fungsi Belanja

Fungsi ini akan ditampilkan setelah Customer memilih opsi melihat dan belanja. Setelah itu Customer akan memilih parfum yang ingin dibeli, memasukkan jumlah, lalu program akan menghitung total harga dan pajak.

#### Source Code:

```
def belanja():  
    global transaksi, nama_terakhir  
    if len(dataparfum) == 0:  
        print("\nParfum Belum Tersedia, Silahkan Menunggu Admin Menambahkan  
Data.")  
        input('\nTekan Enter Untuk Kembali...')  
        return  
    list()  
    beli = input('\nMasukkan Nama Parfum Yang Ingin Dibeli: ')  
    if beli in dataparfum:  
        try:  
            jumlah = int(input('Masukkan Jumlah yang Ingin Dibeli: '))  
        except ValueError:  
            print("\nJumlah Harus Berupa Angka!")  
            input('\nTekan Enter Untuk Kembali...')  
            return  
        total = hitung_total(dataparfum[beli]['Harga'], jumlah)  
        pajak = hitung_pajak(total, 5)  
        total += pajak  
        transaksi += total  
        nama_terakhir = beli  
        print('\n=== STRUK BELANJA ===')
```

```

    print(f>Nama Parfum                : {beli}')
    print(f'Aroma                      : {dataparfum[beli]["Aroma"]}')
    print(f'Harga Satuan               : Rp. {dataparfum[beli]["Harga"]}')
    print(f'Jumlah                     : {jumlah}')
    print(f'Total Harga + Pajak (5%) : Rp. {total}')
    print('\nTerima Kasih Sudah Berbelanja!')
    input('\nTekan Enter Untuk Kembali...')
else:
    print("\nNama Parfum Tidak Ditemukan")
    input('\nTekan Enter Untuk Kembali...')

```

### 3.11 Program Utama

Menjalankan keseluruhan program dengan memanggil fungsi *menu\_utama()* secara berulang hingga pengguna memilih opsi keluar dan program akan berhenti.

#### Source Code:

```

while True:
    menu_utama()

```

## 4. Hasil Output

### Output Menu Utama

- Output yang akan ditampilkan Program pertama kali.

```
=== SELAMAT DATANG DI TOKO PARFUM X ===
Anda Login Sebagai Siapa?
+---+-----+
| No | Pilihan Menu |
+---+-----+
| 1  | CUSTOMER    |
+---+-----+
| 2  | ADMIN       |
+---+-----+
| 3  | KELUAR      |
+---+-----+
[?] Silahkan Pilih Opsi Login:
> 1. Customer
   2. Admin
   3. Keluar
```

Gambar 4.1 Output Menu Utama

### Output Admin

- Output jika pengguna memilih opsi 2 yaitu login sebagai Admin.

```
Selamat Datang Di MENU ADMIN!
+---+-----+
| No | Pilihan Menu |
+---+-----+
| 1  | BUAT AKUN    |
+---+-----+
| 2  | LOGIN        |
+---+-----+
| 3  | KELUAR       |
+---+-----+
[?] Silahkan Pilih Opsi Yang Ada Di Menu:
> 1. Buat Akun
   2. Login
   3. Keluar
```

Gambar 4.2 Output Admin

- Output jika pengguna memilih opsi 1 untuk membuat Akun baru dan berhasil Registrasi.

```
Halo Admin baru!, Silahkan Buat Akun Terlebih Dahulu.  
Masukkan Username: farid  
Masukkan Password: 087  
  
Akun Berhasil Dibuat! Silakan Login.  
  
Tekan Enter Untuk Kembali Ke Menu...
```

Gambar 4.3 Output Admin

- Output jika pengguna memilih opsi 2 untuk melakukan Login setelah Berhasil Registrasi.

```
Halo Admin!, Silahkan Login Terlebih Dahulu.  
Masukkan Username: farid  
Masukkan Password: 087  
  
Login Berhasil!  
  
Tekan Enter Untuk Melanjutkan...
```

Gambar 4.4 Output Admin

- Output jika pengguna memilih opsi 2 untuk melakukan Login tetapi belum membuat akun.

```
Halo Admin!, Silahkan Login Terlebih Dahulu.  
Masukkan Username: farid  
Masukkan Password: 087  
  
Tekan Enter Untuk Melanjutkan...  
  
Anda Belum Memiliki Akun. Silahkan Buat Akun Untuk Login!  
  
Tekan Enter Untuk Kembali...
```

Gambar 4.5 Output Admin

- Output jika Login gagal.

```
Halo Admin!, Silahkan Login Terlebih Dahulu.  
Masukkan Username: varzed  
Masukkan Password: 000  
  
Username atau Password Salah. Silahkan Coba Lagi  
  
Tekan Enter Untuk Kembali...
```

Gambar 4.6 Output Admin

- Output jika pengguna memilih opsi 3 untuk keluar dari Menu Admin

```
[?] Silahkan Pilih Opsi Yang Ada Di Menu:
  1. Buat Akun
  2. Login
> 3. Keluar

Anda Akan Keluar Dari Menu Login

Tekan Enter Untuk Kembali Ke Menu Utama...
```

Gambar 4.7 Output Admin

- Output jika pengguna berhasil Login dan lanjut ke Menu Admin Toko Parfum.

```
Selamat Datang Di MENU ADMIN TOKO PARFUM X!
+-----+
| No | Pilihan Menu |
+-----+
| 1 | TAMBAHKAN PARFUM |
+-----+
| 2 | TAMPILKAN LIST PARFUM |
+-----+
| 3 | UPDATE DATA PARFUM |
+-----+
| 4 | HAPUS DATA PARFUM |
+-----+
| 5 | KELUAR |
+-----+
[?] Silahkan Pilih Opsi Yang Ada Di Menu:
> 1. TAMBAH
  2. LIST
  3. UPDATE
  4. HAPUS
  5. KELUAR
█
```

Gambar 4.8 Output Admin

- Output jika pengguna memilih opsi 1 untuk menambahkan data.

```
Tekan Enter Untuk Melanjutkan...

=== TAMBAH DATA PARFUM ===
Nama Parfum: █
```

Gambar 4.9 Output Admin

```
=== TAMBAH DATA PARFUM ===
Nama Parfum: MyKonos Sea
Aroma Parfum: Aquatic
Harga Parfum: Rp. 250000

Data berhasil ditambahkan!

Tekan Enter Untuk Kembali...█
```

Gambar 4.10 Output Admin

```
=== TAMBAH DATA PARFUM ===
Nama Parfum: HMAN Natural
Aroma Parfum: Fruity
Harga Parfum: Rp. 300000

Data berhasil ditambahkan!

Tekan Enter Untuk Kembali...█
```

Gambar 4.11 Output Admin

```
=== TAMBAH DATA PARFUM ===
Nama Parfum: Bior Amber
Aroma Parfum: Soft
Harga Parfum: Rp. 420000

Data berhasil ditambahkan!

Tekan Enter Untuk Kembali...█
```

Gambar 4.12 Output Admin

```
=== TAMBAH DATA PARFUM ===
Nama Parfum: Dawn Angel
Aroma Parfum: Heaven
Harga Parfum: Rp. 120000

Data berhasil ditambahkan!

Tekan Enter Untuk Kembali...█
```

Gambar 4.13 Output Admin

- Output jika pengguna memilih opsi 2 untuk menampilkan data.

```
=== LIST DATA PARFUM ===

Parfum Ke-1
Nama : MyKonos Sea
Aroma : Aquatic
Harga : Rp. 250000

Parfum Ke-2
Nama : HMAN Natural
Aroma : Fruity
Harga : Rp. 300000

Parfum Ke-3
Nama : Bior Amber
Aroma : Soft
Harga : Rp. 420000

Parfum Ke-4
Nama : Dawn Angel
Aroma : Heaven
Harga : Rp. 120000

Tekan Enter Untuk Lanjut...█
```

Gambar 4.14 Output Admin

- Output jika pengguna memilih opsi 3 untuk meng-update data yang diinginkan.

```

=== UPDATE DATA PARFUM ===
Masukkan Nama Parfum yang Ingin Diupdate: MyKonos Sea
Nama baru: MyKonos Star
Aroma baru: Space
Harga baru: Rp. 220000

Data berhasil diupdate!

```

Gambar 4.15 Output Admin

- Output jika pengguna memilih opsi 3 untuk meng-update data yang diinginkan namun nama parfum tidak ada yang sesuai di List.

```

=== UPDATE DATA PARFUM ===

Masukkan Nama Parfum yang Ingin Diupdate: Fragment

Data Tidak Ditemukan

Tekan Enter Untuk Kembali...

```

Gambar 4.16 Output Admin

- Output jika pengguna memilih opsi 2 lagi untuk menampilkan data yang telah diupdate sebelumnya.

```

=== LIST DATA PARFUM ===

Parfum Ke-1
Nama : HMAN Natural
Aroma : Fruity
Harga : Rp. 300000

Parfum Ke-2
Nama : Bior Amber
Aroma : Soft
Harga : Rp. 420000

Parfum Ke-3
Nama : Dawn Angel
Aroma : Heaven
Harga : Rp. 120000

Parfum Ke-4
Nama : MyKonos Star
Aroma : Space
Harga : Rp. 220000

Tekan Enter Untuk Lanjut...

```

Gambar 4.17 Output Admin

- Output jika pengguna memilih opsi 4 untuk menghapus data yang diinginkan.

```
=== HAPUS DATA PARFUM ===
Masukkan Nama Parfum yang Ingin Dihapus: MyKonos Star

Data berhasil dihapus!
```

Gambar 4.18 Output Admin

- Output jika pengguna memilih opsi 4 untuk menghapus data yang diinginkan namun nama parfum tidak ada yang sesuai di List.

```
=== HAPUS DATA PARFUM ===

Masukkan Nama Parfum yang Ingin Dihapus: Fragment

Data Tidak Ditemukan

Tekan Enter Untuk Kembali...█
```

Gambar 4.19 Output Admin

- Output jika pengguna memilih opsi 2 lagi untuk menampilkan data yang telah diubah sebelumnya.

```
=== LIST DATA PARFUM ===

Parfum Ke-1
Nama : HMAN Natural
Aroma : Fruity
Harga : Rp. 300000

Parfum Ke-2
Nama : Bior Amber
Aroma : Soft
Harga : Rp. 420000

Parfum Ke-3
Nama : Dawn Angel
Aroma : Heaven
Harga : Rp. 120000

Tekan Enter Untuk Lanjut...█
```

Gambar 4.20 Output Admin

- Output jika pengguna memilih opsi 5 untuk keluar dari menu.

```
[?] Silahkan Pilih Opsi Yang Ada Di Menu:
  1. TAMBAH
  2. LIST
  3. UPDATE
  4. HAPUS
> 5. KELUAR

Anda Akan Keluar Dari Menu Admin

Tekan Enter Untuk Kembali Ke Menu...█
```

Gambar 4.21 Output Admin



## Output Customer

- Output jika pengguna memilih opsi 1 yaitu login sebagai Customer.

```
Selamat Datang Di MENU CUSTOMER!
+-----+
| No | Pilihan Menu |
+-----+
| 1 | BUAT AKUN   |
+-----+
| 2 | LOGIN       |
+-----+
| 3 | KELUAR      |
+-----+
[?] Silahkan Pilih Opsi Yang Ada Di Menu:
> 1. Buat Akun
  2. Login
  3. Keluar
█
```

Gambar 4.22 Output Customer

- Output jika pengguna memilih opsi 1 untuk membuat Akun baru dan berhasil Registrasi.

```
Halo Customer baru!, Silahkan Buat Akun Terlebih Dahulu.
Masukkan Username: varzed
Masukkan Password: 087

Akun Berhasil Dibuat! Silakan Login.

Tekan Enter Untuk Kembali Ke Menu...█
```

Gambar 4.23 Output Customer

- Output jika pengguna memilih opsi 2 untuk Login namun belum membuat akun.

```
Halo Customer!, Silahkan Login Terlebih Dahulu.
Masukkan Username: varzed
Masukkan Password: 087

Anda Belum Memiliki Akun. Silahkan Buat Akun Untuk Login!

Tekan Enter Untuk Kembali...█
```

Gambar 4.17 Output Customer

- Output jika pengguna memilih opsi 2 dan Login berhasil.

```
Halo Customer!, Silahkan Login Terlebih Dahulu.  
Masukkan Username: varzed  
Masukkan Password: 087  
  
Login Berhasil!  
  
Tekan Enter Untuk Melanjutkan...
```

Gambar 4.24 Output Customer

- Output jika Login gagal.

```
Halo Customer!, Silahkan Login Terlebih Dahulu.  
Masukkan Username: varzed  
Masukkan Password: 000  
  
Username atau Password Salah. Silahkan Coba Lagi  
  
Tekan Enter Untuk Kembali...
```

Gambar 4.25 Output Customer

- Output jika pengguna memilih opsi 3 untuk keluar dari menu Customer.

```
[?] Silahkan Pilih Opsi Yang Ada Di Menu:  
  1. Buat Akun  
  2. Login  
> 3. Keluar  
  
Anda Akan Keluar Dari Menu Login  
  
Tekan Enter Untuk Kembali Ke Menu Utama...
```

Gambar 4.26 Output Customer

- Output jika pengguna Memilih Opsi 1 untuk melihat dan belanja parfum.

```

=== LIST DATA PARFUM ===

Parfum Ke-1
Nama : HMAN Natural
Aroma : Fruity
Harga : Rp. 300000

Parfum Ke-2
Nama : Bior Amber
Aroma : Soft
Harga : Rp. 420000

Parfum Ke-3
Nama : Dawn Angel
Aroma : Heaven
Harga : Rp. 120000

Tekan Enter Untuk Lanjut...

Masukkan Nama Parfum Yang Ingin Dibeli: █

```

Gambar 4.27 Output Customer

- Output setelah pengguna memasukkan nama parfum yang ingin dibeli dan menampilkan struk belanja.

```

Masukkan Nama Parfum Yang Ingin Dibeli: Dawn Angel
Masukkan Jumlah yang Ingin Dibeli: 1

=== STRUK BELANJA ===
Nama Parfum      : Dawn Angel
Aroma           : Heaven
Harga Satuan    : Rp. 120000
Jumlah          : 1
Total Harga + Pajak (5%) : Rp. 126000.0

Terima Kasih Sudah Berbelanja!

```

Gambar 4.28 Output Customer

- Output jika pengguna memilih opsi 2 untuk keluar dari menu Toko Parfum.

```

[?] Silahkan Pilih Opsi Yang Ada Di Menu:
    1. Lihat & Belanja
    > 2. Keluar

Anda Akan Keluar Dari Menu Customer

Tekan Enter Untuk Kembali Ke Menu... █

```

Gambar 4.29 Output Customer

## Output Keluar Dari Program

- Output jika pengguna memilih opsi 3 untuk keluar dari program.

```
=== SELAMAT DATANG DI TOKO PARFUM X ===
Anda Login Sebagai Siapa?
+---+-----+
| No | Pilihan Menu |
+---+-----+
| 1  | CUSTOMER    |
+---+-----+
| 2  | ADMIN       |
+---+-----+
| 3  | KELUAR      |
+---+-----+
[?] Silahkan Pilih Opsi Login:
    1. Customer
    2. Admin
    > 3. Keluar

Terima Kasih Sudah Berkunjung :)
```

Gambar 4.30 Output Keluar Program

## OutPut Error / Tidak Valid

- Output jika pengguna menginput selain angka saat mengupdate data

```
=== TAMBAH DATA PARFUM ===
Nama Parfum: Fragment Pink
Aroma Parfum: Floral
Harga Parfum: Rp. tiga ratus ribu

Harga Harus Berupa Angka!

Tekan Enter Untuk Kembali...
```

Gambar 4.31 Output ValueError (Berlaku juga untuk Update Data)

- Output jika data belum tersedia.

```
=== LIST DATA PARFUM ===

Data Parfum Belum Tersedia.

Tekan Enter Untuk Kembali...
```

Gambar 4.32 Output Data Parfum Belum Tersedia (List)

```
Tekan Enter Untuk Melanjutkan...

Parfum Belum Tersedia, Silahkan Pilih 1 Untuk Menambahkan Data.

Tekan Enter Untuk Kembali...
```

Gambar 4.33 Data Parfum Belum Tersedia (Update)

```
Tekan Enter Untuk Melanjutkan...  
Parfum Belum Tersedia, Silahkan Pilih 1 Untuk Menambahkan Data.  
Tekan Enter Untuk Kembali...█
```

Gambar 4.34 Output Data Parfum Belum Tersedia (Hapus)

- Output jika ada akun duplikat.

```
Halo Admin baru!, Silahkan Buat Akun Terlebih Dahulu.  
Masukkan Username: farid  
Masukkan Password: 087  
  
Username Sudah Digunakan. Silahkan Gunakan Username Lain!  
Tekan Enter Untuk Kembali...█
```

Gambar 4.45 Output Akun Admin Duplikat

```
Halo Customer baru!, Silahkan Buat Akun Terlebih Dahulu.  
Masukkan Username: varzed  
Masukkan Password: 087  
  
Username Sudah Digunakan. Silahkan Gunakan Username Lain!  
Tekan Enter Untuk Kembali...█
```

Gambar 4.46 Output Akun Customer Duplikat

## 5. Langkah-langkah GIT

### 5.1 GIT Add

Memilih file yang berubah untuk disiapkan masuk ke commit.

```
git add .
```

Gambar 5.1 Git Add

### 5.2 GIT Commit

Menyimpan perubahan yang sudah dipilih dengan pesan tertentu.

```
git commit -m "The End Of PostTest APD"
```

Gambar 5.2 Git Commit

### 5.3 GIT Push

Mengirim commit dari komputer ke repo online.

```
git push
```

Gambar 5.3 Git Push