

**LAPORAN PROYEK AKHIR PRAKTIKUM
MATA KULIAH ALGORITMA DAN PEMROGRAMAN DASAR**



SISTEM PEMESANAN TIKET DAN MAKANAN DI BIOSKOP

Oleh:

Kelompok 2 B2 2024

NAMA KETUA KELOMPOK

Sifwah Fatin Sofwani

Razib Ramadhan

M. Fahrianoor

NIM KETUA KELOMPOK

2409106072

2409106076

2409106089

**PROGRAM STUDI INFORMATIKA
FAKULTAS TEKNIK
UNIVERSITAS MULAWARMAN
SAMARINDA 2024**

KATA PENGANTAR

Pertama saya ucapkan puji syukur kepada Allah SWT yang telah membantu dan mengizinkan kami untuk menyelesaikan tugas proyek akhir praktikum algoritma pemrograman dasar yang berjudul “SISTEM PEMESANAN TIKET DAN MAKANAN DI BIOSKOP”.

Program ini berisi sistem yang berfungsi untuk memesan tiket dan minuman di bioskop yang ditambah dengan fitur tambahan seperti bisa memberi komentar, user hanya bisa membeli tiket dan makanan lalu melihat dan topup saldo serta memberi komentar dan admin dapat melakukan CRUD pada tiket ataupun makanan. Database yang digunakan untuk program ini adalah database yang bertipe json.

Kami memiliki beberapa kendala diantaranya susah dalam mengakses database yang menggunakan json seperti untuk melakukan print data ataupun melakukan CRUD serta sedikit bingung dalam pembagian tugas saat awalan tugas ini diberikan. Kami juga mengucapkan terima kasih kepada bang Muhammad Ghazali dan Pak Muhammad Bambang Firdaus yang telah membantu kami dalam penyelesaian proyek akhir ini serta kepada orang tua dan teman teman kami yang telah menyemangati kami agar penyelesaian tugas ini berakhir dengan baik.

Penulis menyadari bahwa dalam penyusunan laporan proyek akhir praktikum ini masih jauh dari kata sempurna. Oleh karena itu, penulis mengharapkan kritik dan saran yang membangun dari para pembaca demi perbaikan laporan ini di masa mendatang. Akhir kata, penulis berharap laporan proyek akhir praktikum ini dapat memberikan manfaat bagi pembaca dan pihak-pihak yang membutuhkan.

Samarinda, 11 November 2024

Kelompok 2

TAKARIR

Daftar padanan kata bahasa asing dalam bahasa Indonesia yang digunakan adalah sebagai berikut:

<i>Database</i>	Basis Data
<i>Managemen</i>	Mengatur
<i>Input</i>	Memasukkan
<i>Output</i>	Keluaran
<i>Return</i>	Kembalian
<i>True</i>	Benar
<i>Value</i>	Nilai
<i>Key</i>	Kunci
<i>Dictionary</i>	Kamus
<i>List</i>	Daftar
<i>Import</i>	Impor
<i>If</i>	Jika
<i>Print</i>	Cetak
<i>Role</i>	Peran
<i>Login</i>	Masuk
<i>Logout</i>	Keluar
<i>User</i>	Pengguna
<i>Password</i>	Kata Sandi
<i>Register</i>	Daftar
<i>Flowchart</i>	Bagan Alur
<i>Real-Time</i>	Waktu Nyata
<i>Top Up</i>	Isi Ulang

DAFTAR ISI

KATA PENGANTAR	ii
TAKARIR.....	iii
DAFTAR ISI.....	iv
DAFTAR GAMBAR	v
BAB I PENDAHULUAN.....	1
1.1 Latar Belakang	1
1.2 Kebutuhan Fungsional.....	1
1.3 Rumusan Masalah	2
1.4 Batasan Masalah.....	2
1.5 Tujuan.....	3
BAB II PERANCANGAN.....	4
2.1 Analisis Program	4
2.2 Flowchart.....	5
2.3 Konsep/Materi Praktikum yang dipakai.....	8
BAB III HASIL DAN PEMBAHASAN	11
3.1 Tampilan Program.....	11
3.2 Source Code	22
BAB IV PENUTUP	87
4.1 Kesimpulan.....	87
4.2 Saran.....	87
DAFTAR PUSTAKA	88
LAMPIRAN.....	89

DAFTAR GAMBAR

Gambar 2.1 Flowchart Utama.....	5
Gambar 2.2 Flowchart Admin	6
Gambar 2.3 Flowchart Admin	7
Gambar 3.1 Menu Awal.....	11
Gambar 3.2 Menu Login.....	11
Gambar 3.3 Menu Register	12
Gambar 3.4 Menu Utama User	12
Gambar 3.5 Menu Pembelian Tiket	13
Gambar 3.6 Menu Pembelian Makanan.....	14
Gambar 3.7 Menu Topup Saldo.....	14
Gambar 3.8 Menu Cek Saldo.....	14
Gambar 3.9 Menu Riwayat Pembelian Makanan	15
Gambar 3.10 Menu Riwayat Pembelian Tiket.....	16
Gambar 3.11 Menu Utama Forum Komentar	16
Gambar 3.12 Menu Melihat Forum Komentar	17
Gambar 3.13 Menu Menambah Komentar	17
Gambar 3.14 Menu Utama Admin.....	18
Gambar 3.15 Menu Utama Melihat Data.....	18
Gambar 3.16 Menu Utama Menambah Data	19
Gambar 3.17 Menu Utama Menambah Data (Teater)	19
Gambar 3.18 Menu Utama Mengedit Data.....	20
Gambar 3.19 Menu Utama Mengedit Salah Satu Data (Teater)	20

3.20 Menu Utama Menghapus Data.....	21
3.21 Menu Menghapus Data (Makanan).....	21
Gambar Kertas Konsul.....	89

BAB I

PENDAHULUAN

1.1 Latar Belakang

Program ini dibuat ditujukan untuk memudahkan pengguna memesan tiket makanan dan bioskop lebih mudah dengan program ini serta membuat pembeli dapat melihat jam penayangan yang ada pada teater tertentu dan kursi yang tersedia yang dimana dapat membantu pengguna untuk memesan tiket, disisi lain pengguna dapat memesan makanan dan melakukan pengecekan apakah makanan yang diinginkan masih tersedia apa tidak serta yang terakhir pengguna memberi komentar.

1.2 Kebutuhan Fungsional

Pada awal program, terdapat 3 menu yang tersedia, yaitu:

1. Login: Pengguna bisa mengakses sistem dengan memasukkan username dan password yang sesuai.
2. Register: Pengguna dapat membuat akun baru untuk bisa mengakses sistem.
3. Keluar Program: Pengguna diberi pilihan untuk keluar dari aplikasi dan menutup sesi penggunaan sistem.

Jika login sebagai user, maka akan ditampilkan menu berikut:

1. Beli Tiket: Membantu user untuk melihat film yang sedang tayang, dan memungkinkan user untuk membeli tiket bioskop jika user memiliki saldo yang cukup.
2. Beli Makanan: Membantu user untuk melihat menu makanan dan minuman yang tersedia, dan memungkinkan user untuk membeli makanan dan/atau minuman jika user memiliki saldo yang cukup.
3. Top Up Saldo: Memungkinkan user untuk menambah saldo pada akun user.
4. Cek Saldo: Memudahkan user untuk melihat jumlah saldo yang user miliki.
5. Histori Pembelian Makanan: Membantu user untuk melihat riwayat pembelian makanan yang dilakukan user sebelumnya.
6. Histori Pembelian Tiket: Membantu user untuk melihat riwayat pembelian tiket bioskop yang dilakukan user sebelumnya.

7. Beri Komentar: Memungkinkan user untuk memberikan komentar mengenai apapun yang berhubungan dengan program atau bioskop.
8. Logout: Memungkinkan user untuk keluar dari akun saat ini.
9. Keluar Program: User diberi pilihan untuk keluar dari aplikasi dan menutup sesi penggunaan sistem.

Jika login sebagai admin, maka akan ditampilkan menu berikut:

1. Lihat daftar: pada bagian lihat daftar admin dapat melihat seluruh data seperti data keseluruhan teater, film, fnb, pengguna, komentar, riwayat pembelian tiket, dan riwayat pembelian fnb.
2. Tambah data: Bagian ini digunakan admin untuk menambah data seperti data teater baru, film baru, fnb baru, user baru, dan admin juga bisa menambah komentar pada bagian ini.
3. Edit data: pada bagian ini admin memiliki fungsi untuk mengedit, terdapat 3 hal yang bisa di edit seperti teater, film, dan fnb.
4. Hapus data: admin bertugas untuk menghapus data seperti teater, film dan fnb.

1.3 Rumusan Masalah

Dalam pembuatan program ini, rumusan masalah yang dikemukakan adalah seperti berikut:

1. Bagaimana program ini dapat membantu user untuk membeli tiket dan makanan?
2. Bagaimana program agar admin mampu melakukan CRUD pada data tiket dan makanan?

1.4 Batasan Masalah

Program ini memiliki beberapa batasan diantaranya:

1. Tidak menampilkan bagian kursi apa saja yang tersedia dan tidak

1.5 Tujuan

Tujuan program sistem pemesanan tiket dan makanan di bioskop ini adalah untuk membantu dan memudahkan pelanggan membeli tiket untuk film dan makanan tanpa harus mengantri atau datang langsung ke bioskop. Program ini juga akan memberikan informasi real-time tentang ketersediaan tiket, jadwal film yang sedang tayang, serta berbagai jenis makanan dan minuman yang dapat dipesan, sehingga pelanggan dapat merencanakan perjalanan mereka ke bioskop. Selain itu, fitur top-up saldo dan sistem pembayaran terintegrasi memungkinkan pengguna melakukan transaksi dengan lebih mudah dan aman. Fitur komentar juga bertujuan untuk membantu pengguna memberikan umpan balik tentang layanan, yang dapat digunakan sebagai dasar untuk meningkatkan kualitas layanan bioskop. Selain itu, program ini memiliki sistem manajemen yang dapat mengawasi dan mengupdate informasi film dan makanan secara berkala. Ini memastikan bahwa pengguna selalu memiliki informasi yang akurat dan terkini tentang layanan yang tersedia di bioskop.

BAB II

PERANCANGAN

2.1 Analisis Program

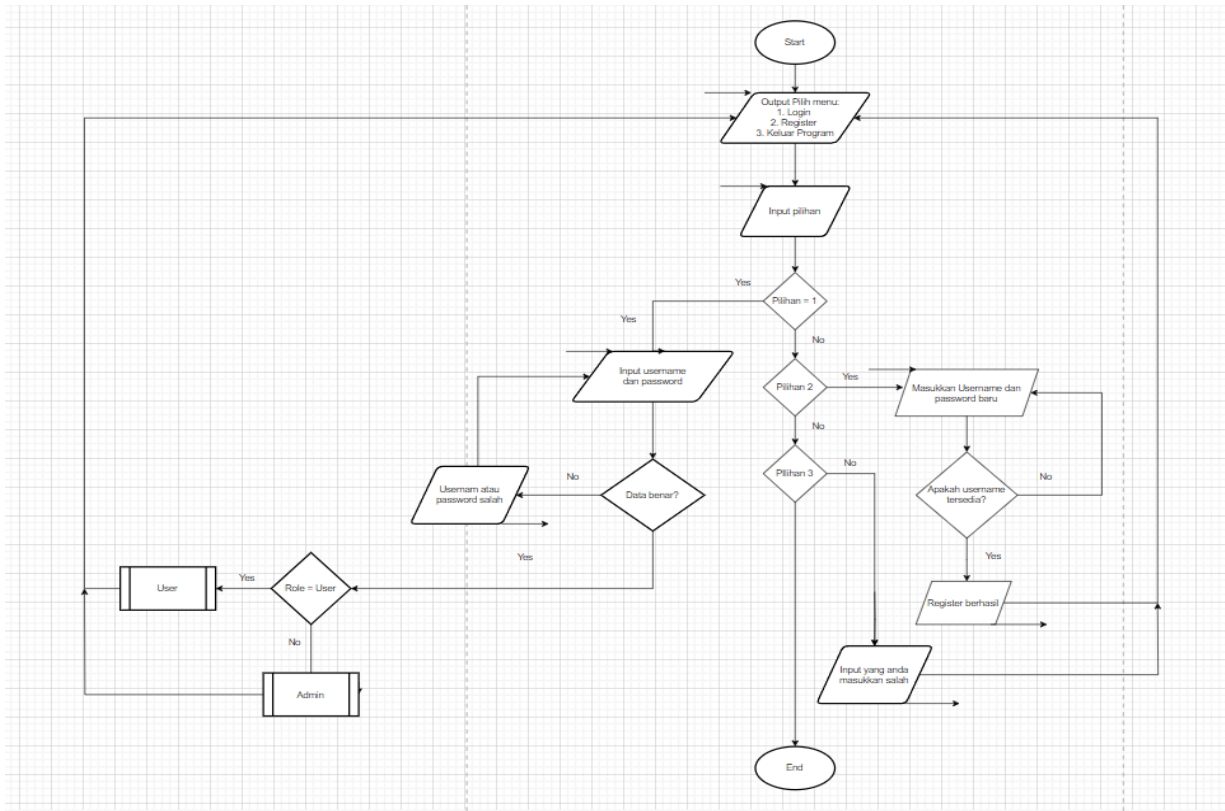
Program ini berjalan dengan menu awal yaitu dengan pilihan login, register dan keluar dari program. Jika anda memilih untuk register maka anda akan diminta untuk memasukkan username akun baru dan password yang baru setelah itu akun baru anda akan berhasil dibuat.

Setelah itu ada login yang dimana pengguna diminta untuk memasukan username dan password akun yang telah dibuat dan setelah itu mereka akan memasuki menu user atau admin tergantung jenis akun apa yang digunakan untuk login.

Akun dengan jenis user dapat membeli tiket bioskop dan makanannya, ketika membeli sesuatu pengguna juga tentunya memerlukan saldo untuk pembelian yang dimana saldo tersebut dapat dilihat dan diisi apabila user menginginkannya. Setelah melakukan pembelian pengguna dapat melihat riwayat transaksi mereka dan bisa mendiskusikan tentang makanan atau filmnya di forum diskusi.

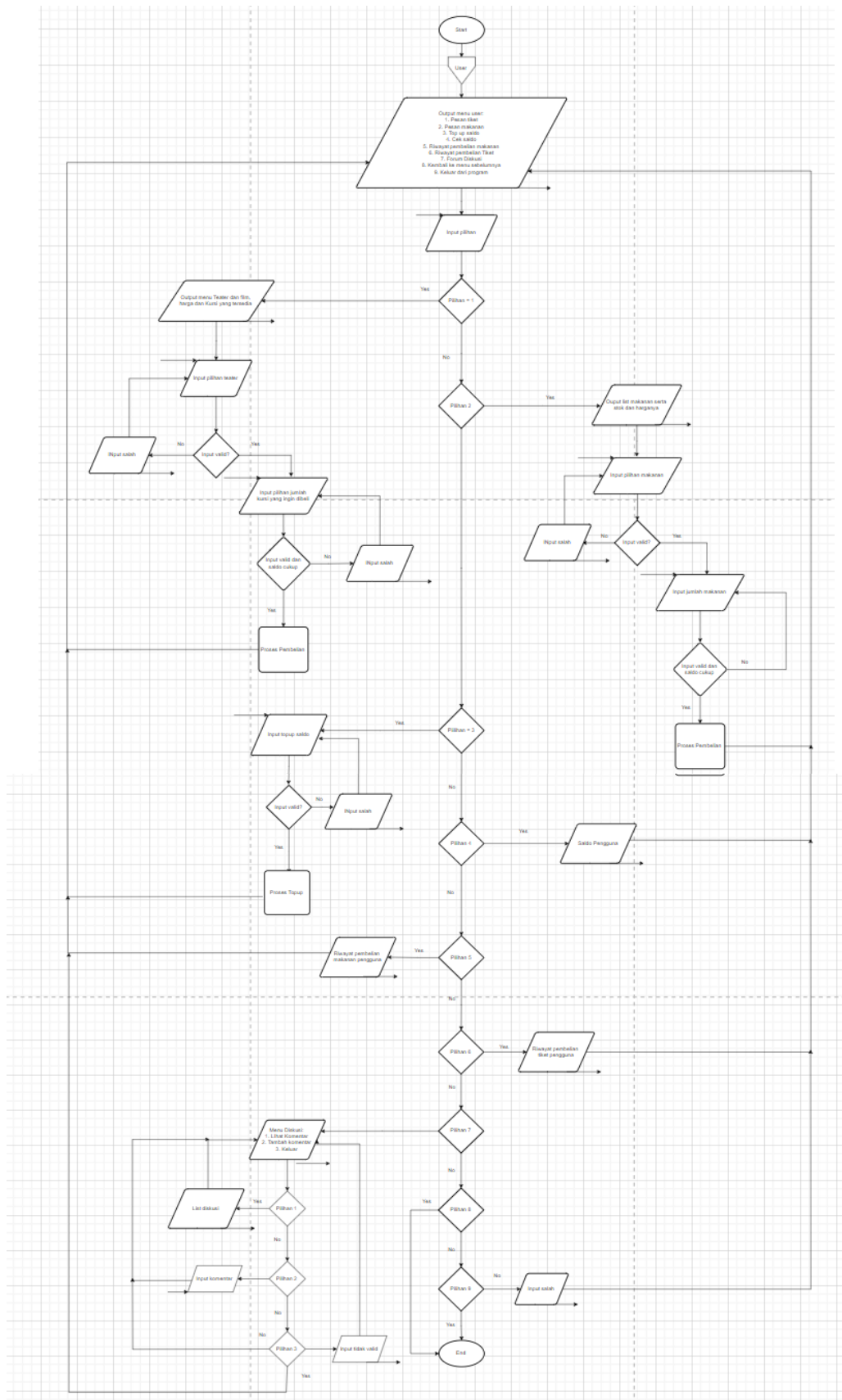
Admin yang merupakan salah satu jenis akun yang ada dalam program ini dapat melakukan CRUD untuk data teater, film, dan FnB. Misalnya admin dapat menghapus data film yang telah ada ataupun menambah jenis makanan atau minuman baru serta melihat teater apa saja yang tersedia dan mengubah data yang ada pada teater tersebut seperti mengubah nama teater 1 menjadi teater 2.

2.2 Flowchart



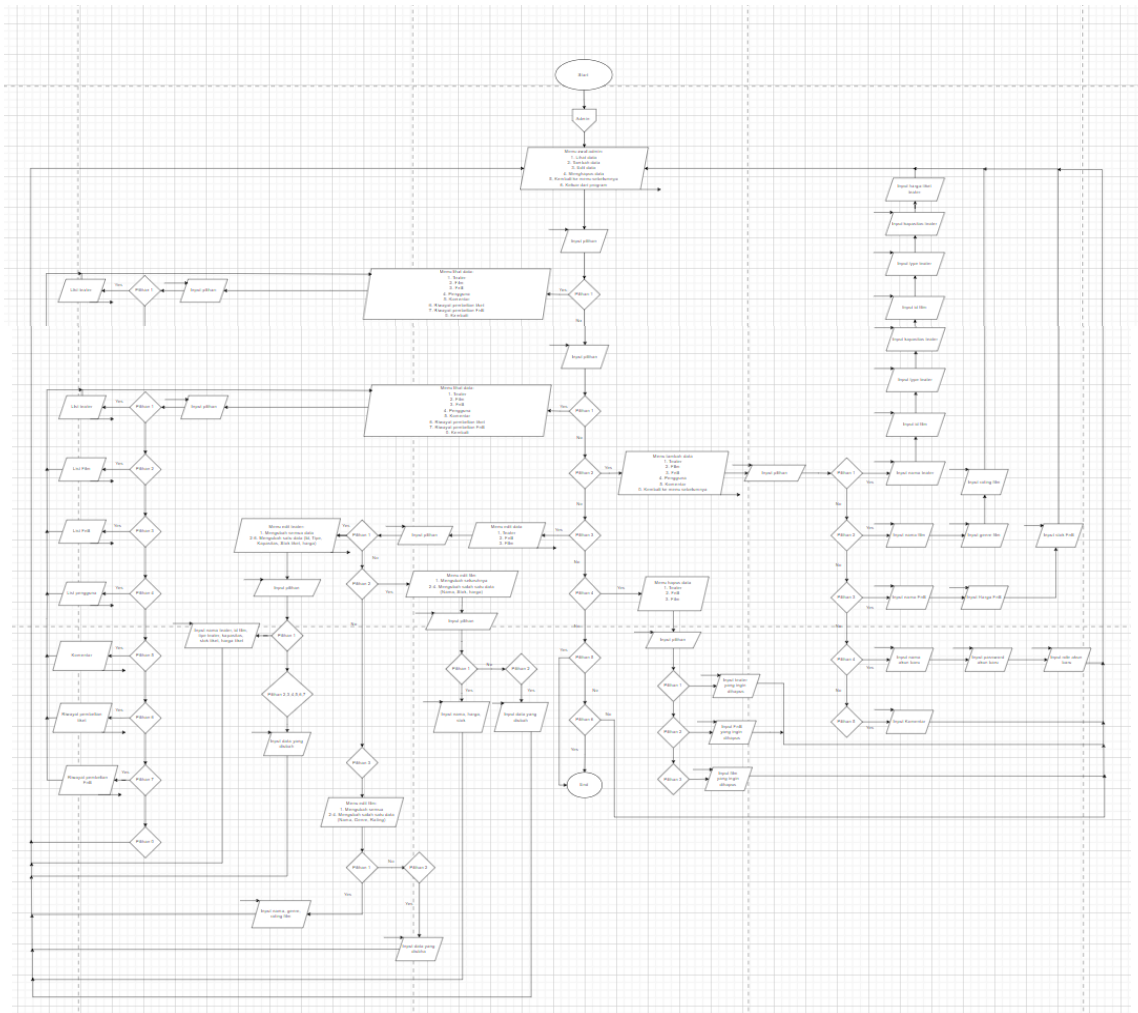
Gambar 2.1 Flowchart Utama

Program ini berawal dari menu yang menampilkan login, register atau keluar, saat pengguna memilih login pengguna akan diminta password dan username, jika passwordnya salah maka akan kembali ke menu sebelumnya namun bila benar maka akan masuk ke menu admin atau user tergantung role akun yang dimasukkan. Lalu ada register dimana pengguna akan diminta untuk memasukkan username dan password serta role untuk akun baru, namun apabila username yang dipilih sudah dipakai maka akan kembali ke menu sebelumnya. Dan ketika pengguna memilih untuk keluar dari program maka program akan berhenti.



Gambar 2.2 Flowchart User

Jika pengguna memiliki akun yang rolenya user maka pengguna akan mendapatkan menu awal yaitu menu untuk membeli tiket, makanan, cek saldo, topup saldo, memberi komentar dan dapat kembali ke menu sebelumnya serta keluar dari program.



Gambar 2.3 Flowchart Admin

Pada bagian ini pengguna dapat melakukan CRUD pada database yang tersedia, pertama pengguna dapat melihat data data yang tersedia pada database seperti dapat melihat data teater, makanan, film, riwayat transaksi makanan dan tiket serta komentar

dan pengguna. Pada menu menambahkan pengguna dapat menambahkan teater, film, makanan, pengguna, dan komentar. Lalu untuk yang terakhir data pengguna dapat mengubah dan menghapus data teater, FnB, film.

2.3 Konsep/Materi Praktikum yang dipakai

Konsep yang kami gunakan dalam pembuatan laporan ini sesuai dengan persyaratan yang terdapat pada modul diantaranya:

1. Fungsi dasar: Fungsi print.

Fungsi print pada program merupakan fungsi yang umum dipakai untuk menampilkan suatu keluaran pada layar peraga.

2. Fungsi dasar: Percabangan

Fungsi percabangan seperti if, elif, dan else digunakan untuk membuat keputusan berdasarkan kondisi tertentu. Dengan if, kita memeriksa apakah suatu kondisi benar, dan jika benar, kode yang terkandung di dalamnya akan dijalankan. Jika kondisi pertama tidak terpenuhi, kita dapat menggunakan elif (else if) untuk memeriksa kondisi lain, dan terakhir, else digunakan untuk mengeksekusi kode jika semua kondisi sebelumnya salah. Ini memungkinkan program untuk memilih jalur eksekusi yang sesuai dengan situasi, seperti memilih apa yang harus dilakukan berdasarkan input atau nilai yang diberikan.

3. Fungsi dasar: Fungsi case

Fungsi case digunakan untuk memilih dan menjalankan blok kode berdasarkan nilai tertentu yang telah ditentukan.

4. Fungsi dasar: Fungsi Import

Fungsi import digunakan dalam pemrograman untuk memasukkan modul atau pustaka ke dalam program, yang memungkinkan Anda mengakses dan menggunakan fungsi, kelas, atau variabel yang disediakan oleh modul tersebut, memungkinkan Anda untuk meningkatkan fungsionalitas program tanpa menulis ulang kode.

5. Tipe data kolektif

Tuple adalah struktur data yang dapat menyimpan data namun bersifat tidak dapat diubah. List adalah struktur data yang dapat ditambahkan, dihapus bahkan diubah dan list bersifat terurut dan pengguna dapat mengakses data sesuai indeksinya. Dan ada dictionary adalah tempat menyimpan data dalam python yang terdiri dari “key” dan “value” dimana “key” berfungsi sebagai variabel yang menyimpan sebuah data yaitu “value”. Value dalam dictionary dapat diubah, ditambah dan dihapus dengan mengakses “key” dari “value” yang ingin diubah.

6. Fungsi perulangan: While dan For

While adalah fungsi perulangan yang menjalankan blok kode berulang kali selama kondisi yang diberikan bernilai True, for adalah Pernyataan yang digunakan untuk mengulangi eksekusi blok kode dengan jumlah iterasi tertentu.

7. Fungsi

Define adalah sebuah fungsi yang memungkinkan kita membuat fungsi baru yang tidak ada dalam program bawaan python.

8. Fungsi dasar: Return

Fungsi yang berguna untuk mengembalikan nilai yang ditetapkan atau dibuat di sebuah variabel dalam proses “define”.

9. Fungsi error handling: Try and Except

Try dan except adalah blok kode dalam Python yang digunakan untuk menangani kesalahan atau exception saat program berjalan. Mereka membantu mencegah program terhenti akibat error dengan menyediakan cara untuk menangkap dan mengatasi kesalahan tersebut.

BAB III

HASIL DAN PEMBAHASAN

3.1 Tampilan Program

1. Tampilan Menu Awal

```
=====
|| Selamat datang di Program Bioskop Online ||
=====
                        MENU UTAMA
1. Login
2. Register
3. Keluar
>> Masukkan pilihan anda: |
```

Gambar 3.1 Menu Awal

Menu tersebut merupakan menu awal untuk mengakses menu login, register dan keluar dari program dan pengguna dapat melakukan input.

```
=====
|| Silahkan login terlebih dahulu ||
=====
>> Masukkan username anda: sdfs
>> Masukkan password anda: fds|
```

Gambar 3.2 Menu Login

Menu untuk pengguna menginput username dan password akun mereka.

```
=====
||          Silahkan register terlebih dahulu          ||
=====
>> Masukkan username : kecoak
>> Masukkan password : bakar
```

Gambar 3.3 Menu Register

Menu yang digunakan untuk melakukan register akun baru

2, Tampilan Menu User

```
=====
||          Behasil masuk: Selamat datang User          ||
=====
          Silahkan pilih menu di bawah ini

1. Beli tiket
2. Beli Makanan
3. Topup Saldo
4. Cek Saldo
5. Histori Pembelian Makanan
6. Histori Pembelian Tiket
7. Beri Komentar
8. Logout
9. Keluar Program
>> Masukkan pilihan yang anda inginkan: 
```

Gambar 3.4 Menu Utama User

Menu utama yang dapat diakses user.

```
=====
||          Daftar Teater dan Film:          ||
=====

-----
teater 1
Nama Film      : Agak Laen
Rating         : 13+
Harga          : Rp.10000
Jumlah Kursi   : 69/30
Tipe Teater    : Reguler
-----

Teater 2
Nama Film      : Peaky blinders
Rating         : 18+
Harga          : Rp.40000
Jumlah Kursi   : 80/70
Tipe Teater    : Reguler
-----

Sifwah
Nama Film      : Kocak gaming
Rating         : 20
Harga          : Rp.10000
Jumlah Kursi   : 72/72
Tipe Teater    : Regular
-----

>> Masukkan urutan teater yang anda inginkan (0 untuk kembali ke halaman utama): 2
>> Masukkan jumlah kursi yang ingin anda beli: 10
```

Gambar 3.5 Menu Pembelian Tiket

Menu user untuk membeli tiket.

```
=====
||                      Daftar makanan dan minuman:                      ||
=====

1. Hotdog
Stok   : 19
Harga  : Rp.25000
-----

2. es krim
Stok   : 14
Harga  : Rp.4000
-----

>> Masukkan nomor makanan yang ingin anda beli (0 untuk kembali ke halaman utama): 2
>> Masukkan jumlah yang ingin anda beli: 100
```

Gambar 3,6 Menu Pembelian Makanan

Menu yang dimana user dapat membeli makanan

```
=====
||                      Top up                      ||
=====

>> Masukkan nominal top up anda (0 untuk kembali ke halaman utama): Rp.200
```

Gambar 3.7 Menu Topup Saldo

Menu untuk pengguna menambahkan saldo

```
Saldo Anda adalah Rp.2313154123

>> Klik enter untuk kembali ke halaman utama
```

Gambar 3.8 Menu Cek Saldo

Menu untuk pengguna melakukan pengecekan saldo

Riwayat Pembelian FnB untuk user	
FnB	: es krim
Harga	: Rp.4000
Jumlah	: 2
Total Harga	: Rp.8000
Waktu Pembelian	: 2024-11-20 15:13:33
FnB	: es krim
Harga	: Rp.4000
Jumlah	: 2
Total Harga	: Rp.8000
Waktu Pembelian	: 2024-11-20 20:40:32
FnB	: es krim
Harga	: Rp.4000
Jumlah	: 2
Total Harga	: Rp.8000
Waktu Pembelian	: 2024-11-20 20:40:54
FnB	: Hotdog
Harga	: Rp.25000
Jumlah	: 1
Total Harga	: Rp.25000
Waktu Pembelian	: 2024-11-22 10:40:10
FnB	: Hotdog
Harga	: Rp.25000
Jumlah	: 0
Total Harga	: Rp.0
Waktu Pembelian	: 2024-11-22 14:50:24

Gambar 3.9 Menu Riwayat Pembelian Makanan

Menu untuk pengguna melihat riwayat pembelian makanan.

```
=====
||          Riwayat Pembelian Tiket untuk user          ||
=====
-----
Film          : Agak Laen
Teater        : teater 1
Harga Tiket   : Rp.10000
Jumlah        : 1
Total         : Rp.10000
Waktu Pembelian : 2024-11-20 14:59:33
-----
Film          : Agak Laen
Teater        : teater 1
Harga Tiket   : Rp.10000
Jumlah        : 0
Total         : Rp.0
Waktu Pembelian : 2024-11-22 14:43:25
```

3.10 Menu Riwayat Pembelian Tiket

Menu untuk pengguna melihat riwayat pembelian tiket.

```
=====
||          Forum Komentar          ||
=====
          Silahkan pilih menu di bawah ini

1. Lihat semua komentar
2. Tambah komentar baru
3. Keluar
>> Pilih opsi (1-3): █
```

3.11 Menu Utama Forum Komentar

Menu utama komentar.

```
=====
||                               Daftar Komentar                               ||
=====

-----

Username: admin
Komentar: This is a comment
-----

Username: user
Komentar: This is another comment
-----

Username: user
Komentar: filmnya asik bagerrr
-----

Username: admin
Komentar: filmnya asik kan?
-----

Username: admin
Komentar: asik banget omaga
-----
```

Gambar 3.12 Menu Melihat Forum Komentar

Tampilan forum komentar.

```
>> Masukkan komentar Anda: dsadas
```

Gambar 3.13 Menu Menambah Komentar

3. Tampilan Admin

```
=====
||          Behasil masuk: Selamat datang Admin          ||
=====
                Silahkan pilih menu di bawah ini

1. Lihat Daftar
2. Tambah Data
3. Edit Data
4. Hapus Data
5. Logout
6. Keluar Program
>> Masukkan pilihan Anda: |
```

Gambar 3.14 Menu Utama Admin

```
=====
||          Daftar Data          ||
=====

1. Teater
2. Film
3. FnB
4. Pengguna
5. Komentar
6. Riwayat Pembelian Tiket
7. Riwayat Pembelian FnB
```

Gambar 3.15 Menu Utama Melihat Data


```
=====
||                                ||
||                                ||
=====
1. Teater
2. Film
3. FnB
4. User
5. Komentar
```

Gambar 3.16 Menu Utama Tambah Data

```
=====
||                                ||
||                                ||
=====

-----
ID      : 1
Nama    : Agak Laen
-----

ID      : 2
Nama    : Peaky blinders
-----

ID      : 3
Nama    : kcoakk
-----

ID      : 4
Nama    : 2
-----

ID      : 5
Nama    : Kocak gaming
-----

=====
||                                ||
||                                ||
=====

>> Nama Teater: anjaysda
>> ID Film (0 untuk None): 2
>> Tipe Teater (R untuk Regular, P untuk Premiere): r
```

Gambar 3.17 Menu Utama Menambah Data (Teater)

```
=====
||                      Edit Teater, Fnb, dan Film                      ||
=====

1. Teater
2. Fnb
3. Film

>> Masukkan pilihan (0 untuk kembali):
```

Gambar 3.18 Menu Utama Mengedit Data

```
=====
||                      Edit teater                      ||
=====

1.Seluruhnya
2>Nama Teater
3.ID Film
4.Tipe
5.Kapasitas
6.Stok Tiket
7.Harga

>> Masukkan pilihan(0 untuk kembali):
```

Gambar 3.19 Menu Utama Mengedit Salah Satu Data (Teater)

```
=====
||                      Hapus Teater, Fnb, dan Film                      ||
=====

1. Teater
2. Fnb
3. Film

>> Masukkan Pilihan (pilih 0 untuk kembali):
```

3.20 Menu Utama Menghapus Data

```
=====
||                      Hapus Fnb                      ||
=====
||                      Daftar Fnb                      ||
=====

-----
ID      : 1
Nama    : Hotdog
Stok    : 19
Harga   : Rp.25000
-----
ID      : 2
Nama    : es krim
Stok    : 14
Harga   : Rp.4000
-----

>> Masukkan ID fnb:
```

Gambar 3.21 Menu Menghapus Data (Makanan)

3.2 Source Code

```
import os
import json
import datetime
import colorama
from colorama import Fore, Back, Style
```

Fungsi dari kode tersebut adalah agar program dapat memasukkan modul eksternal ataupun bawaan python, dapat digunakan untuk membaca file json, waktu dan hari, untuk melakukan clear terminal serta untuk mengakses pustaka colorama yang membantu mencetak teks berwarna di terminal atau Command Prompt.

```
colorama.init(autoreset=True)
```

Berfungsi sebagai mereset warna setiap ada fungsi print dan tidak perlu menambahkan code yang berguna untuk mereset warna setiap menggunakan fungsi print

```
# default data
default_data = {
    "users": [
        {
            "id": 1,
            "username": "admin",
            "password": "admin",
            "role": "admin",
            "balance": None
        },
        {
            "id": 2,
            "username": "user",
            "password": "user",
            "role": "user",
            "balance": 0
        }
    ]
}
```

```

    }
],
"teaters": [],
"films": [],
"ticketTransactions": [],
"fnb": [],
"fnbTransactions": [],
"comments": []
}

# mengakses database untuk pertama kali
try:
    with open('database.json') as f:
        database = json.load(f)
        # Membuat dictionary untuk menyimpan data film
        film_dict = {film["id"]: film["name"] for film in
database["films"]}
        film_rating = {film["id"]: film["filmRating"] for film in
database["films"]}
        user_dict = {user["id"]: user["username"] for user in
database["users"]}
        teater_dict = {teater["id"]: teater["name"] for teater in
database["teaters"]}
        fnb_dict = {fnb["id"]: fnb["name"] for fnb in
database["fnb"]}
        fnb_price = {fnb["id"]: fnb["price"] for fnb in
database["fnb"]}
except FileNotFoundError:
    print("File database.json tidak ditemukan. Membuat file baru
dengan data default.")
    input("\nTekan enter untuk lanjut")
    database = default_data
    with open('database.json', 'w') as f:
        json.dump(database, f, indent=2)

    film_dict = {}
    film_rating = {}
    user_dict = {}
    teater_dict = {}
    fnb_dict = {}
    fnb_price = {}

```



Fungsi ini berguna untuk Mengakses file database saat program pertama kali dijalankan. Jika file database tidak terdeteksi, program akan membuat file database baru dengan data default.

```
def cls():  
    os.system('cls')
```

Code tersebut berfungsi untuk mendefinisikan fungsi clear terminal

```
def buka():  
    try:  
        with open('database.json') as f:  
            return json.load(f)  
    except FileNotFoundError:  
        print("File database.json tidak ditemukan")  
        return
```

Fungsi tersebut digunakan untuk mengakses file database sistem dan bila file tidak ditemukan maka akan muncul pemberitahuan file database tidak ditemukan.

```
def kamus():  
    global film_dict, film_rating, user_dict, teater_dict, fnb_dict,  
    fnb_price  
    film_dict = {film["id"]: film["name"] for film in  
database["films"]}  
    film_rating = {film["id"]: film["filmRating"] for film in  
database["films"]}  
    user_dict = {user["id"]: user["username"] for user in  
database["users"]}  
    teater_dict = {teater["id"]: teater["name"] for teater in  
database["teaters"]}  
    fnb_dict = {fnb["id"]: fnb["name"] for fnb in database["fnb"]}  
    fnb_price = {fnb["id"]: fnb["price"] for fnb in database["fnb"]}
```

Ini adalah sebuah fungsi yang bertujuan untuk membuat beberapa dictionary global berdasarkan data yang ada dalam variabel database. Fungsi ini melakukan pemrosesan data menjadi bentuk yang lebih mudah diakses dengan memetakan ID suatu item tertentu dengan satu value lain dari item yang sama.

```
def save(data):  
    with open('database.json', 'w') as f:  
        json.dump(data, f, indent=2)
```

Fungsi tersebut berguna untuk menyimpan data file database

```
def get_next_id(data_list):  
    id_terbesar = 0  
    for item in data_list:  
        if item["id"] > id_terbesar:  
            id_terbesar = item["id"]  
    return id_terbesar + 1
```

Fungsi tersebut berguna sebagai mencari id terbesar di dalam id database yang digunakan sebagai id untuk transaksi baru.

```
def menuawal():  
    while True:  
        print(Fore.CYAN + Style.BRIGHT + "=" * 60)  
        print(Fore.CYAN + Style.BRIGHT + "||" + " Selamat datang di  
Program Bioskop Online ".center(56) + "||")  
        print(Fore.CYAN + Style.BRIGHT + "=" * 60)  
        print(Fore.MAGENTA + Style.BRIGHT + "MENU UTAMA".center(60))  
        print(Fore.CYAN + "1. " + Fore.GREEN + Style.BRIGHT + "Login  
)")  
        print(Fore.CYAN + "2. " + Fore.LIGHTYELLOW_EX + Style.BRIGHT  
+ "Register")  
        print(Fore.CYAN + "3. " + Fore.RED + Style.BRIGHT + "Keluar")  
        pilihan = input(Fore.BLUE + Style.BRIGHT + ">> Masukkan  
pilihan anda: ").strip()  
        cls()  
  
        match pilihan:
```

```

        case "1":
            try:
                level, masuk, username = menulogin()
                return level, masuk, username
            except:
                continue
        case "2":
            menuregis()
        case "3":
            print("Terima kasih! Program selesai.")
            quit()
        case "":
            continue
        case _:
            print(Fore.LIGHTYELLOW_EX + Style.BRIGHT + "Pilihan
tidak tersedia.\n")

```

Fungsi tersebut berguna sebagai menu awal yang dimana pengguna akan diminta untuk melakukan input menu apa saja yang diinginkan yang dimana jika pengguna menginput 1 maka pengguna akan masuk ke menu login, 2 untuk registrasi akun baru dan ke 3 untuk keluar dari program. Dalam fungsi print terdapat syntax Fore dan Style yang dimana fore berguna untuk mewarnai output teks terminal dan style untuk menentukan jenis teksnya terang atau gelap.

```

def menulogin():
    while True:
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.CYAN + Style.BRIGHT + "||" + "Silahkan login
terlebih dahulu".center(56)+"||")
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        username = input(Fore.BLUE + Style.BRIGHT + ">> Masukkan
username anda: ")
        password = input(Fore.BLUE + Style.BRIGHT + ">> Masukkan
password anda: ")
        cls()
        # mengecek username dan password
        for user in database['users']:

```



```

        if user['username'] == username and user['password'] ==
password:
            if user['role'] == "user":
                masuk = 2
                level = "user"
                username = user['username']
                return level, masuk, username
            elif user['role'] == "admin":
                masuk = 2
                level = "admin"
                username = user['username']
                return level, masuk, username
        print(Fore.LIGHTYELLOW_EX + Style.BRIGHT + "Username atau
password yang anda masukkan salah.\n")
        break

```

Disini pengguna akan diminta untuk memasukkan username dan password akun, apabila salah maka mereka akan mendapatkan pemberitahuan bahwa password atau username yang mereka masukkan salah dan akan dikembalikan ke menu sebelumnya, sebaliknya apabila benar maka mereka akan masuk ke menu selanjutnya sesuai dengan role akun yang mereka masukkan.

```

def menuregis():
    while True:
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.CYAN + Style.BRIGHT + "||" + "Silahkan register
terlebih dahulu".center(56)+"||" )
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        username = input(Fore.BLUE + Style.BRIGHT + ">> Masukkan
username : ").strip()

        usnterpakai = False
        # mengecek apakah username sudah terpakai
        for user in database['users']:
            if user['username'] == username:
                usnterpakai = True
                break

```

```

        if usnterpakai:
            cls()
            print(Fore.LIGHTYELLOW_EX + Style.BRIGHT + "Username
telah dipakai.\n")
            break

        password = input(Fore.BLUE + Style.BRIGHT + ">> Masukkan
password : ").strip()

        # mengecek apakah username dan password kosong
        if username == "" or password == "":
            cls()
            print(Fore.LIGHTYELLOW_EX + Style.BRIGHT + "Username dan
password tidak boleh kosong!\n")
            break

        # menentukan role
        role = "user"

        # membuat user baru
        regis = {
            "id": get_next_id(database['users']),
            "username": username,
            "password": password,
            "role": role,
            "balance": 0
        }

        # menambahkan user baru ke database
        database['users'].append(regis)

        # menyimpan database baru ke file
        save(database)

        cls()
        print(Fore.LIGHTGREEN_EX + Style.BRIGHT + "Register
berhasil!\n")
        break

```

Fungsi ini adalah dimana pengguna dapat membuat akun baru untuk digunakan, awalnya pengguna akan diminta untuk menginput username yang mereka inginkan, jika username mereka sudah terpakai maka mereka tidak dapat menggunakannya dan akan kembali ke menu sebelumnya, jika belum maka mereka akan diminta untuk memasukkan password dan setelah itu akun berhasil ditambahkan, register pada menu awal hanya akan mendapatkan role user.

```
def menuuser():
    while True:
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.CYAN + Style.BRIGHT + "||" + "Behasil masuk:
Selamat datang User".center(56)+"||")
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.BLUE + Style.BRIGHT + "Silahkan pilih menu di
bawah ini\n".center(60))
        print(Fore.BLUE + "1. " + Fore.GREEN + Style.BRIGHT + "Beli
tiket")
        print(Fore.BLUE + "2. " + Fore.GREEN + Style.BRIGHT + "Beli
Makanan")
        print(Fore.BLUE + "3. " + Fore.GREEN + Style.BRIGHT + "Topup
Saldo")
        print(Fore.BLUE + "4. " + Fore.GREEN + Style.BRIGHT + "Cek
Saldo")
        print(Fore.BLUE + "5. " + Fore.GREEN + Style.BRIGHT +
"Histori Pembelian Makanan")
        print(Fore.BLUE + "6. " + Fore.GREEN + Style.BRIGHT +
"Histori Pembelian Tiket")
        print(Fore.BLUE + "7. " + Fore.GREEN + Style.BRIGHT + "Berikomentar")
        print(Fore.BLUE + "8. " + Fore.LIGHTYELLOW_EX + Style.BRIGHT
+ "Logout")
        print(Fore.BLUE + "9. " + Fore.LIGHTRED_EX + Style.BRIGHT +
"Keluar Program")
        inputmenu = input( Fore.LIGHTBLUE_EX + Style.BRIGHT + ">>
Masukkan pilihan yang anda inginkan: ")
        cls()
        return inputmenu
```

Pada fungsi ini kami hanya menampilkan apa saja menu user serta user dapat memasukkan input yang mereka inginkan untuk menuju menu selanjutnya yang mereka inginkan

```
def menutiket(username, masuk):
    try:
        while masuk == 3:
            # menampilkan daftar teater dan film
            print(Fore.CYAN + Style.BRIGHT + "="*60)
            print(Fore.CYAN + Style.BRIGHT + "||"+"Daftar Teater dan
Film:".center(50)+"||")
            print(Fore.CYAN + Style.BRIGHT + "="*60)

            print(Fore.WHITE + "-" * 50)
            for data in database['teaters']:
                film_name = film_dict.get(data["film_id"])
                rating = film_rating.get(data["film_id"])
                if data['film_id'] is not None:
                    print(Fore.GREEN + Style.BRIGHT +
f"  {data['name']})")
                    print(Fore.LIGHTWHITE_EX + f"  Nama Film      :
{film_name}")
                    print(Fore.LIGHTWHITE_EX + f"  Rating        :
{rating}")
                    print(Fore.LIGHTWHITE_EX + f"  Harga          :
Rp.{data['price']})")
                    print(Fore.LIGHTWHITE_EX + f"  Jumlah Kursi :
{data['available']}/{data['capacity']})")
                    print(Fore.LIGHTWHITE_EX + f"  Tipe Teater   :
{data['type'].capitalize()})")
                    print(Fore.WHITE + "-" * 50)

                beli = int(input(Fore.BLUE + Style.BRIGHT + ">> Masukkan
urutan teater yang anda inginkan (0 untuk kembali ke halaman utama):
"))
                if beli == 0:
                    cls()
                    return masuk == 2
                teaterada= False
                for cekteater in database['teaters']:
```

```

        if cekteater['film_id'] is None:
            continue
        elif cekteater['id'] == beli:
            teaterada = True
            if cekteater['available'] <= 0:
                cls()
                print(Fore.LIGHTRED_EX + Style.BRIGHT +
"Kursi telah habis")
                return menutiket(masuk)
                break

            if teaterada == False:
                cls()
                print(Fore.LIGHTRED_EX + Style.BRIGHT + "Nomor teater
tidak valid")
                menutiket(masuk)

            belikursi = int(input(Fore.BLUE + Style.BRIGHT + ">>
Masukkan jumlah kursi yang ingin anda beli: "))

            for cekkursi in database['teaters']:
                if beli == cekkursi['id']:
                    cekkursi = cekkursi['available']

            if belikursi > cekkursi:
                cls()
                print(Fore.LIGHTRED_EX + Style.BRIGHT + "Anda memesan
kursi lebih dari yang tersedia\n")
                continue

            elif cekkursi > 0 and belikursi <= cekkursi:
                for ceksaldo in database['users']:
                    if ceksaldo['username'] == username:
                        iduser = ceksaldo['id']
                        saldopengguna = ceksaldo['balance']
                        for belitiket in database['teaters'] :
                            if saldopengguna >=
belikursi*belitiket['price']:

                                if belitiket['id'] == beli:
                                    namateater = belitiket['name']

```

```

        idteater = belitiket['id']
        belitiket['available'] -=

belikursi

        ceksaldo['balance'] -=

belikursi*belitiket['price']

        for idfilm in database['films']:
            if beli == idfilm['id']:
                idnamafilam = idfilm['id']
                namafilam = idfilm['name']

                riwayat = {
                    "id":
get_next_id(database['ticketTransactions']),
                    "film_id":
idnamafilam,
                    "teater_id":
idteater,
                    "user_id": iduser,
                    "price":
belitiket['price'],
                    "totalTicket":
belikursi,
                    "totalPrice":
belikursi*belitiket['price'],
                    "purchaseTime":
datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")
                }
                database['ticketTransacti
ons'].append(riwayat)

                save(database)

            cls()
            print(Fore.LIGHTGREEN_EX +
Style.BRIGHT + f"Anda telah membeli sebanyak {belikursi} kursi untuk
{namateater} dengan film {namafilam}\nHarga satuan tiket adalah Rp
{belitiket['price']} dan total harganya adalah Rp
{belikursi*belitiket['price']} \n")
            return masuk == 2
        else:
            cls()

```

```

print(Fore.LIGHTRED_EX + Style.BRIGHT
+ "Saldo anda tidak cukup\n")
menutiket(username, masuk)

except ValueError:
    cls()
    print(Fore.LIGHTRED_EX + Style.BRIGHT + "Input yang anda
masukkan harus berupa angka\n")
    menutiket(username, masuk)
except TypeError:
    cls()
    print(Fore.LIGHTRED_EX + Style.BRIGHT + "Anda memasukkan input
yang salah\n")
    menutiket(username, masuk)

```

Pada fungsi ini fungsi ada parameter username dan masuk, parameter masuk berfungsi sebagai agar perulangan dapat berjalan dan username untuk mengetahui username pengguna, pada fungsi ini pengguna dapat membeli tiket dengan awalan pengguna diminta untuk membeli teater apa yang diinginkan setelah itu akan ada pengecekan, apabila teater tersedia maka akan lanjut ke menu pemesanan kursi, jika tidak maka pengguna akan kembali ke menu sebelumnya, begitu pula saat membeli kursi namun inputan kursi 0 atau melebihi kursi yang tersedia maka akan kembali ke menu sebelumnya, jika pengguna membeli kursi dan kursinya masih tersedia maka sistem akan membaca saldo pengguna, apabila cukup maka sistem akan mengurangi saldo pengguna dan membuat riwayat pemesanan.

```

def menutopup():
    try:
        while True:
            print(Fore.BLUE + Style.BRIGHT + "="*60)
            print(Fore.BLUE + Style.BRIGHT + "||" + "Top
up".center(56) + "||")
            print(Fore.BLUE + Style.BRIGHT + "="*60)

```

```

        topup = int(input(Fore.LIGHTGREEN_EX + Style.BRIGHT + ">>
Masukkan nominal top up anda (0 untuk kembali ke halaman utama):
Rp."))

        if topup > 0:
            for topups in database['users']:
                if topups['username'] == username:
                    topups['balance'] += topup
            save(database)
            cls()
            print(Fore.LIGHTGREEN_EX + Style.BRIGHT + f"Anda
telah top up sebanyak Rp.{topup}\n")
            return
        elif topup == 0:
            cls()
            return
        else:
            cls()
            print(Fore.LIGHTYELLOW_EX + Style.BRIGHT + "Nominal
top up harus lebih besar dari 0.\n")
    except ValueError:
        cls()
        print(Fore.LIGHTRED_EX + Style.BRIGHT + "Input yang anda
masukkan harus berupa angka.\n")
        menutopup()
    except TypeError:
        cls()
        print(Fore.LIGHTRED_EX + Style.BRIGHT + "Anda memasukkan input
yang salah.\n")
        menutopup()

```

Pada fungsi ini pengguna dapat mengisi saldo mereka yang dimana saat mereka memilih menu tersebut, pengguna akan diminta untuk menginput jumlah saldo yang diinginkan, jika inputnya valid maka jumlah saldo pengguna akan bertambah sesuai input pengguna.

```

def menumakanan():
    try:
        while True:

```



```

        # Menampilkan daftar makanan
        print(Fore.BLUE + Style.BRIGHT + "="*60)
        print(Fore.BLUE + Style.BRIGHT + "||" + "Daftar makanan dan
minuman:".center(56) + "||")
        print(Fore.BLUE + Style.BRIGHT + "="*60)

        print(Fore.WHITE + "-" * 50)
        for data in database['fnb']:
            print(Fore.GREEN + Style.BRIGHT + f" {data['id']}.
{data['name']}"")
            print(Fore.LIGHTWHITE_EX + f" Stok :
{data['stock']}"")
            print(Fore.LIGHTWHITE_EX + f" Harga :
Rp.{data['price']}"")
            print(Fore.WHITE + "-" * 50)

        beli = int(input(Fore.BLUE + Style.BRIGHT + ">> Masukkan
nomor makanan yang ingin anda beli (0 untuk kembali ke halaman
utama): "))

        if beli == 0:
            cls()
            return

        for data in database['fnb']:
            if data['id'] == beli:
                idmakanan = data['id']
                if data['stock'] <= 0:
                    cls()
                    print(Fore.LIGHTRED_EX + Style.BRIGHT + "Stok
habis\n")
                    continue

                totalfnb = int(input(Fore.BLUE + Style.BRIGHT
+ ">> Masukkan jumlah yang ingin anda beli: "))
                if totalfnb > data['stock']:
                    cls()
                    print(Fore.LIGHTRED_EX + Style.BRIGHT + "Anda
membeli melebihi stok yang tersedia\n")
                    continue

```

```

        if totalfnb == 0:
            cls()
            print("Anda tidak membeli makanan apapun")
            break
        elif totalfnb <= data['stock']:
            for ceksaldo in database['users']:
                if ceksaldo['username'] == username:
                    iduser = ceksaldo['id']
                    if ceksaldo['balance'] >=
totalfnb*data['price']:
                    ceksaldo['balance'] -=
totalfnb*data['price']
                    data['stock'] -= totalfnb

                    transaksi_baru = {
                        "id":
get_next_id(database['fnbTransactions']),
                        "user_id": iduser,
                        "fnb_id": idmakanan,
                        "quantity": totalfnb,
                        "total":
totalfnb*data['price'],
                        "purchaseTime":
datetime.datetime.now().strftime("%Y-%m-%d %H:%M:%S")
                    }

                    database['fnbTransactions'].appen
d(transaksi_baru)

                    save(database)

                    cls()
                    print(Fore.LIGHTGREEN_EX +
Style.BRIGHT + f"Anda telah membeli {data['name']} sebanyak
{totalfnb} buah\nHarga satuannya adalah Rp {data['price']} dan
totalnya adalah Rp {totalfnb*data['price']}")
                    return masuk == 2
            else:
                cls()
                print(Fore.LIGHTRED_EX +
Style.BRIGHT + "Saldo anda tidak cukup")

```

```

except ValueError:
    cls()
    print(Fore.LIGHTRED_EX + Style.BRIGHT + "Input yang anda
masukkan harus berupa angka\n")
    menumakanan()
except TypeError:
    cls()
    print(Fore.LIGHTRED_EX + Style.BRIGHT + "Anda memasukkan input
yang salah\n")
    menumakanan()

```

Fungsi ini berfungsi sebagai tempat memesan makanan bagi user, awalnya akan muncul jenis makanan apa saja yang ada lalu pengguna akan diminta untuk memasukkan makanan apa yang ingin dibeli, setelah itu pengguna akan diminta untuk memasukkan jumlah pembelian yang diinginkan, jika stok masih tersedia maka sistem akan mengurangi saldo pengguna dan membuat riwayat pembelian makanan.

```

def menuhistorymakanan():
    user_found = 0
    for user in database['users']:
        if user['username'] == username:
            user_id = user['id']
            user_found = 1
            break

    if user_found == 0:
        print(Fore.LIGHTRED_EX + Style.BRIGHT + "Pengguna tidak
ditemukan\n")
        return

    print(Fore.CYAN + Style.BRIGHT + "="*60)
    print(Fore.CYAN + Style.BRIGHT + "||"+f"Riwayat Pembelian FnB
untuk {username}".center(56) + "||")
    print(Fore.CYAN + Style.BRIGHT + "="*60)
    print(Fore.WHITE + "-" * 50)

    transaksi_ditemukan = 0

```

```

for transaksi in database['fnbTransactions']:
    if transaksi['user_id'] == user_id:
        transaksi_ditemukan = 1
        for fnb_item in database['fnb']:
            if fnb_item['id'] == transaksi['fnb_id']:
                nama_makanan = fnb_item['name']
                total_pembelian = transaksi['total']
                harga_per_item = fnb_item['price']
                jumlah_dibeli = transaksi['quantity']
                waktu_pembelian = transaksi['purchaseTime']

                print(Fore.LIGHTWHITE_EX + f"FnB          :
{nama_makanan}")
                print(Fore.LIGHTWHITE_EX + f"Harga          :
Rp.{harga_per_item}")
                print(Fore.LIGHTWHITE_EX + f"Jumlah          :
{jumlah_dibeli}")
                print(Fore.LIGHTWHITE_EX + f"Total Harga      :
Rp.{total_pembelian}")
                print(Fore.LIGHTWHITE_EX + f"Waktu Pembelian  :
{waktu_pembelian}")
                print(Fore.WHITE + "-" * 50)

        if transaksi_ditemukan == 0:
            print(Fore.LIGHTYELLOW_EX + Style.BRIGHT + "Tidak ada riwayat
pembelian FnB untuk pengguna ini.\n")
            input(Fore.LIGHTBLUE_EX + Style.BRIGHT + ">> Klik enter untuk
kembali ke halaman utama\n")
            cls()

```

Fungsi ini bermanfaat agar pengguna dapat melihat riwayat pembelian makanan mereka yang dimana sistem akan membaca username pengguna, apabila sama dengan yang ada pada database riwayat transaksi makanan maka akan muncul riwayat pembelian makanan user tersebut, jika tidak ada maka akan muncul pemberitahuan bahwa user tidak memiliki riwayat pembelian makanan.

```

def menuhistorytiket():
    user_found = 0
    for user in database['users']:
        if user['username'] == username:
            user_id = user['id']
            user_found = 1
            break

    if user_found == 0:
        print("Pengguna tidak ditemukan\n")
        return

    print(Fore.CYAN + Style.BRIGHT + "="*60)
    print(Fore.CYAN + Style.BRIGHT + "||"+f"Riwayat Pembelian Tiket
untuk {username}".center(56) + "||")
    print(Fore.CYAN + Style.BRIGHT + "="*60)
    print(Fore.WHITE + "-" * 50)

    tiket_ditemukan = 0
    for transaksi in database['ticketTransactions']:
        if transaksi['user_id'] == user_id:
            tiket_ditemukan = 1
            for film in database['films']:
                if film['id'] == transaksi['film_id']:
                    nama_film = film['name']
                    for teater in database['teaters']:
                        if teater['id'] == transaksi['teater_id']:
                            nama_teater = teater['name']

                            print(Fore.LIGHTWHITE_EX +
f"Film          : {nama_film}")
                            print(Fore.LIGHTWHITE_EX +
f"Teater       : {nama_teater}")
                            print(Fore.LIGHTWHITE_EX + f"Harga
Tiket        : Rp.{transaksi['price']}")
                            print(Fore.LIGHTWHITE_EX +
f"Jumlah       : {transaksi['totalTicket']}")
                            print(Fore.LIGHTWHITE_EX +
f"Total        : Rp.{transaksi['totalPrice']}")
                            print(Fore.LIGHTWHITE_EX + f"Waktu
Pembelian : {transaksi['purchaseTime']}")

```

```

print(Fore.WHITE + "-" * 50)

if tiket_ditemukan == 0:
    print(Fore.LIGHTYELLOW_EX + Style.BRIGHT + "Tidak ada riwayat pembelian\n")
    input(Fore.LIGHTBLUE_EX + Style.BRIGHT + ">> Klik enter untuk kembali ke halaman utama\n")
    cls()

```

Pada fungsi ini pengguna dapat melihat riwayat transaksi pembelian tiket mereka yang dimana, awalnya sistem akan melakukan pengecekan pada id user, jika id user sama dengan id yang ada pada riwayat transaksi user maka sistem akan memunculkan riwayat pembelian tiket pengguna, jika tidak akan muncul pemberitahuan bahwa user tidak memiliki riwayat pembelian tiket.

```

def menukomentar(username):
    cls()
    while True:

        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.CYAN + Style.BRIGHT + "||" + "Forum
Komentar".center(56)+"||")
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.BLUE + Style.BRIGHT + "Silahkan pilih menu di
bawah ini\n".center(60))
        print(Fore.BLUE + "1. " + Fore.GREEN + Style.BRIGHT + "Lihat
semua komentar")
        print(Fore.BLUE + "2. " + Fore.GREEN + Style.BRIGHT + "Tambah
komentar baru")
        print(Fore.BLUE + "3. " + Fore.LIGHTYELLOW_EX + Style.BRIGHT +
"Keluar")

        pilihan = input(Fore.LIGHTBLUE_EX + Style.BRIGHT + ">> Pilih
opsi (1-3): ")
        cls()

        if pilihan == '1':

```

```

        # Tampilkan semua komentar
        if not database['comments']:
            print(Fore.LIGHTYELLOW_EX + "\nBelum ada komentar.")
        else:
            print(Fore.CYAN + Style.BRIGHT + "="*60)
            print(Fore.CYAN + Style.BRIGHT + "||" + "Daftar
Komentar".center(56)+"||")
            print(Fore.CYAN + Style.BRIGHT + "="*60)
            print(Fore.WHITE + "-" * 50)
            for comment in database['comments']:
                # Cari nama pengguna berdasarkan user_id
                username = next(
                    (user['username'] for user in
database['users'] if user['id'] == comment['user_id']), "Unknown
User"
                )
                print(Fore.LIGHTWHITE_EX + f"\nUsername:
{username}")
                print(Fore.LIGHTWHITE_EX + f"Komentar:
{comment['comment']}")
                print(Fore.WHITE + "-" * 50)

            elif pilihan == '2':
                # Tambah komentar baru
                for user in database['users']:
                    if user['username'] == username:
                        user_id = user['id']
                        komentar_baru = input(Fore.BLUE + Style.BRIGHT
+ ">> Masukkan komentar Anda: ").strip()
                        if komentar_baru == "":
                            cls()
                            print(Fore.LIGHTYELLOW_EX + "Komentar tidak
boleh kosong!\n")
                            break
                        komentar = {
                            "user_id": user_id,
                            "comment": komentar_baru
                        }
                        database['comments'].append(komentar)
                        save(database)
                        cls()

```

```

        print(Fore.LIGHTGREEN_EX + "Komentar berhasil
ditambahkan!\n")

        elif pilihan == '3':
            cls()
            print(Fore.LIGHTYELLOW_EX + "Keluar dari forum
diskusi.\n")
            break

        else:
            cls()
            print(Fore.LIGHTYELLOW_EX + "Pilihan tidak valid. Silakan
coba lagi.")

```

Pada menu ini pengguna dapat melakukan diskusi, pertama pengguna dapat melihat diskusi apa saja yang ada dan dapat menambahkan komentar untuk mengikuti diskusi yang ada.

```

def menuadmin():
    while True:
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.CYAN + Style.BRIGHT + "||" + "Behasil masuk:
Selamat datang Admin".center(56)+"||")
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.BLUE + Style.BRIGHT + "Silahkan pilih menu di
bawah ini\n".center(60))
        print(Fore.BLUE + "1. " + Fore.GREEN + Style.BRIGHT + "Lihat
Daftar")
        print(Fore.BLUE + "2. " + Fore.GREEN + Style.BRIGHT + "Tambah
Data ")
        print(Fore.BLUE + "3. " + Fore.GREEN + Style.BRIGHT + "Edit
Data")
        print(Fore.BLUE + "4. " + Fore.GREEN + Style.BRIGHT + "Hapus
Data")
        print(Fore.BLUE + "5. " + Fore.LIGHTYELLOW_EX + Style.BRIGHT
+ "Logout")
        print(Fore.BLUE + "6. " + Fore.LIGHTRED_EX + Style.BRIGHT +
"Keluar Program")
        inputmenu = input(Fore.LIGHTBLUE_EX + Style.BRIGHT + ">>
Masukkan pilihan Anda: ")

```



```

cls()
return inputmenu

```

Fungsi ini berguna sebagai menu utama admin dan pengguna dapat memasukan menu apa saja yang ingin di akses.

```

def lihatadmin():
    while True:
        # mengecek apakah database ada
        if not database:
            print(Fore.LIGHTYELLOW_EX + "Data tidak tersedia")
            input(Fore.LIGHTYELLOW_EX + "\n>> Tekan enter untuk
kembali")
            return

        # menampilkan daftar database
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.CYAN + Style.BRIGHT + "||"+ "Daftar
Data".center(56)+"||")
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.BLUE + "1. " + Fore.GREEN + Style.BRIGHT + "Teater")
        print(Fore.BLUE + "2. " + Fore.GREEN + Style.BRIGHT + "Film")
        print(Fore.BLUE + "3. " + Fore.GREEN + Style.BRIGHT + "FnB")
        print(Fore.BLUE + "4. " + Fore.GREEN + Style.BRIGHT
+"Pengguna")
        print(Fore.BLUE + "5. " + Fore.GREEN + Style.BRIGHT
+"Komentar")
        print(Fore.BLUE + "6. " + Fore.GREEN + Style.BRIGHT + "Riwayat
Pembelian Tiket")
        print(Fore.BLUE + "7. " + Fore.GREEN + Style.BRIGHT + "Riwayat
Pembelian FnB")
        pilihlihat = input(Fore.LIGHTBLUE_EX + Style.BRIGHT + "\n>>
Masukkan angka yang sesuai (0 untuk kembali): ").strip()
        cls()

        match pilihlihat:
            case "0":
                return
            case "1":

```

```

        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.CYAN + Style.BRIGHT + "||"+"Daftar
Teater".center(56)+"||")
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.WHITE + "-" * 50)
        if not database['teaters']:
            print(Fore.LIGHTYELLOW_EX + "Tidak ada database
teater\n")
        else:
            for data in database['teaters']:
                film_name = film_dict.get(data["film_id"])
                print(Fore.LIGHTWHITE_EX + f" ID          :
{data['id']}")
                print(Fore.LIGHTWHITE_EX + f" Nama Teater  :
{data['name']}")
                print(Fore.LIGHTWHITE_EX + f" Nama Film    :
{film_name}")
                print(Fore.LIGHTWHITE_EX + f" Harga        :
{'None' if data['price'] is None else 'Rp.' + str(data['price'])}")
                print(Fore.LIGHTWHITE_EX + f" Jumlah Kursi :
{data['available']}/{data['capacity']}")
                print(Fore.LIGHTWHITE_EX + f" Tipe Teater  :
{data['type'].capitalize()}")
                print(Fore.WHITE + "-" * 50)
            input(Fore.LIGHTBLUE_EX + "\n>> Tekan enter untuk
kembali\n")
            cls()
        case "2":
            print(Fore.CYAN + Style.BRIGHT + "="*60)
            print(Fore.CYAN + Style.BRIGHT + "||"+"Daftar
Film".center(56)+"||")
            print(Fore.CYAN + Style.BRIGHT + "="*60)
            print(Fore.WHITE + "-" * 50)
            if not database['films']:
                print(Fore.LIGHTYELLOW_EX + "Tidak ada database
film\n")
            else:
                for data in database['films']:
                    print(Fore.LIGHTWHITE_EX + f" ID          :
{data['id']}")

```

```

        print(Fore.LIGHTWHITE_EX + f" Nama :
{data['name']}")
        print(Fore.LIGHTWHITE_EX + f" Genre : {'',
'.join(data['genre'])}")
        print(Fore.LIGHTWHITE_EX + f" Rating :
{data['filmRating']}")
        print(Fore.WHITE + "-" * 50)
        input(Fore.LIGHTBLUE_EX + "\n>> Tekan enter untuk
kembali\n")
        cls()
        case "3":
            print(Fore.CYAN + Style.BRIGHT + "="*60)
            print(Fore.CYAN + Style.BRIGHT + "||"+ "Daftar
FnB".center(56)+"||")
            print(Fore.CYAN + Style.BRIGHT + "="*60)
            print(Fore.WHITE + "-" * 50)
            if not database['fnb']:
                print(Fore.LIGHTYELLOW_EX + "Tidak ada database
FnB\n")
            else:
                for data in database['fnb']:
                    print(Fore.LIGHTWHITE_EX + f" ID :
{data['id']}")
                    print(Fore.LIGHTWHITE_EX + f" Nama :
{data['name']}")
                    print(Fore.LIGHTWHITE_EX + f" Stok :
{data['stock']}")
                    print(Fore.LIGHTWHITE_EX + f" Harga :
Rp.{data['price']}")
                    print(Fore.WHITE + "-" * 50)
                input(Fore.LIGHTBLUE_EX + "\n>> Tekan enter untuk
kembali\n")
                cls()
                case "4":
                    print(Fore.CYAN + Style.BRIGHT + "="*60)
                    print(Fore.CYAN + Style.BRIGHT + "||"+ "Daftar
Pengguna".center(56)+"||")
                    print(Fore.CYAN + Style.BRIGHT + "="*60)
                    print(Fore.WHITE + "-" * 50)
                    if not database['users']:

```

```

        print(Fore.LIGHTYELLOW_EX + "Tidak ada database
pengguna\n")
    else:
        for data in database['users']:
            print(Fore.LIGHTWHITE_EX + f" ID      :
{data['id']}")
            print(Fore.LIGHTWHITE_EX + f" Username :
{data['username']}")
            print(Fore.LIGHTWHITE_EX + f" Role      :
{data['role'].capitalize()}")
            print(Fore.WHITE + "-" * 50)
            input(Fore.LIGHTBLUE_EX + "\n>> Tekan enter untuk
kembali\n")
            cls()
        case "5":
            print(Fore.CYAN + Style.BRIGHT + "="*60)
            print(Fore.CYAN + Style.BRIGHT + "||"+ "Daftar
Komentar".center(56)+"||")
            print(Fore.CYAN + Style.BRIGHT + "="*60)
            print(Fore.WHITE + "-" * 50)
            if not database['comments']:
                print(Fore.LIGHTYELLOW_EX + "Tidak ada database
komentar\n")
            else:
                for data in database['comments']:
                    user_name = user_dict.get(data["user_id"])
                    print(Fore.LIGHTWHITE_EX + f" ID      :
{data['user_id']}")
                    print(Fore.LIGHTWHITE_EX + f" Nama      :
{user_name}")
                    print(Fore.LIGHTWHITE_EX + f" Komentar   :
{data['comment']}")
                    print(Fore.WHITE + "-" * 50)
                    input(Fore.LIGHTBLUE_EX + "\n>> Tekan enter untuk
kembali\n")
                    cls()
                case "6":
                    print(Fore.CYAN + Style.BRIGHT + "="*60)
                    print(Fore.CYAN + Style.BRIGHT + "||"+ "Daftar Riwayat
Pembelian Tiket".center(56)+"||")
                    print(Fore.CYAN + Style.BRIGHT + "="*60)

```

```

        print(Fore.WHITE + "-" * 50)
        if not database['ticketTransactions']:
            print(Fore.LIGHTYELLOW_EX + "Tidak ada database
riwayat pembelian tiket\n")
        else:
            for data in database['ticketTransactions']:
                user_name = user_dict.get(data["user_id"])
                teater_name =
teater_dict.get(data["teater_id"])
                film_name = film_dict.get(data["film_id"])
                print(Fore.LIGHTWHITE_EX +
f"ID          : {data['id']}"")
                print(Fore.LIGHTWHITE_EX +
f>Nama        : {user_name}"")
                print(Fore.LIGHTWHITE_EX +
f"Teater      : {teater_name}"")
                print(Fore.LIGHTWHITE_EX +
f"Film        : {film_name}"")
                print(Fore.LIGHTWHITE_EX +
f"Harga       : Rp.{data['price']}"")
                print(Fore.LIGHTWHITE_EX + f"Jumlah
Tiket      : {data['totalTicket']}"")
                print(Fore.LIGHTWHITE_EX + f"Waktu Pembelian
: {data['purchaseTime']}"")
                print(Fore.WHITE + "-" * 50)
            input("\nTekan enter untuk kembali\n")
            cls()
        case "7":
            print(Fore.CYAN + Style.BRIGHT + "="*60)
            print(Fore.CYAN + Style.BRIGHT + "||"+ "Daftar Riwayat
Pembelian Tiket".center(56)+"||")
            print(Fore.CYAN + Style.BRIGHT + "="*60)
            print(Fore.WHITE + "-" * 50)
            if not database['fnbTransactions']:
                print(Fore.LIGHTYELLOW_EX + "Tidak ada database
riwayat pembelian FnB\n")
            else:
                for data in database['fnbTransactions']:
                    user_name = user_dict.get(data["user_id"])
                    fnb_name = fnb_dict.get(data["fnb_id"])
                    harga = fnb_price.get(data["fnb_id"])

```

```

        waktu_pembelian = data['purchaseTime']
        print(Fore.LIGHTWHITE_EX +
f"ID          : {data['id']}")
        print(Fore.LIGHTWHITE_EX + f>Nama
User          : {user_name}")
        print(Fore.LIGHTWHITE_EX +
f"FnB         : {fnb_name}")
        print(Fore.LIGHTWHITE_EX +
f"Harga       : Rp.{harga}")
        print(Fore.LIGHTWHITE_EX +
f"Jumlah      : {data['quantity']}")
        print(Fore.LIGHTWHITE_EX + f"Total
Harga        : Rp.{data['total']}")
        print(Fore.LIGHTWHITE_EX + f"Waktu
Pembelian   : {waktu_pembelian}")
        print(Fore.WHITE + "-" * 50)
        input(Fore.LIGHTBLUE_EX + "\n>> Tekan enter untuk
kembali\n")

        cls()
        case "":
            continue
        case _:
            print("Pilihan tidak tersedia.\n")

```

Fungsi diatas merupakan fungsi dimana admin dapat melihat data yang ada pada database diantaranya teater, film, FnB, pengguna, komentar, riwayat pembelian makanan dan minuman.

```

def tambahadmin(username):
    while True:
        # mengecek apakah database ada
        if not database:
            print(Fore.LIGHTYELLOW_EX + "Data tidak tersedia")
            input(Fore.LIGHTYELLOW_EX + "\n>> Tekan enter untuk
kembali")
            return

        # menampilkan menu tambah database

```

```

        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.CYAN + Style.BRIGHT + "||" + "Tambah
Data".center(56) + "||")
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.BLUE + "1. " + Fore.GREEN + Style.BRIGHT +
"Teater")
        print(Fore.BLUE + "2. " + Fore.GREEN + Style.BRIGHT + "Film")
        print(Fore.BLUE + "3. " + Fore.GREEN + Style.BRIGHT + "FnB")
        print(Fore.BLUE + "4. " + Fore.GREEN + Style.BRIGHT + "User")
        print(Fore.BLUE + "5. " + Fore.GREEN + Style.BRIGHT +
"Komentar")
        pilih_tambah = input(Fore.LIGHTBLUE_EX + "\n>> Masukkan angka
yang sesuai (0 untuk kembali): ").strip()
        cls()

        match pilih_tambah:
            case "0":
                return
            case "1":
                try:
                    # menampilkan daftar film
                    print(Fore.CYAN + Style.BRIGHT + "="*60)
                    print(Fore.CYAN + Style.BRIGHT + "||" + "Daftar
Film".center(56) + "||")
                    print(Fore.CYAN + Style.BRIGHT + "="*60)
                    print(Fore.WHITE + "-" * 50)
                    for data in database['films']:
                        print(Fore.LIGHTWHITE_EX + f" ID      :
{data['id']}")
                        print(Fore.LIGHTWHITE_EX + f" Nama    :
{data['name']}")
                        print("-" * 50)

                    print(Fore.CYAN + Style.BRIGHT + "="*60)
                    print(Fore.CYAN + Style.BRIGHT + "||" + "Tambah
Teater".center(56) + "||")
                    print(Fore.CYAN + Style.BRIGHT + "="*60)
                    name = input(Fore.LIGHTBLUE_EX + ">> Nama Teater:
").capitalize().strip()
                    # cek apakah nama teater kosong
                    if name == "":

```

```

        cls()
        print(Fore.LIGHTYELLOW_EX + "Nama teater
tidak boleh kosong\n")
        continue
    # cek apakah nama sudah ada
    for teater in database["teaters"]:
        if teater["name"].lower() == name.lower():
            cls()
            print(Fore.LIGHTYELLOW_EX + "Teater
dengan nama tersebut sudah ada\n")
            break

    film_id = int(input(Fore.LIGHTBLUE_EX + ">>> ID
Film (0 untuk None): ").strip())
    if film_id == 0:
        film_id = None
    else:
        # Cek apakah id film ada
        film_found = False # Inisialisasi flag untuk
ID film

        for film in database["films"]:
            if film["id"] == film_id:
                film_found = True
                break # Keluar dari loop jika film
ditemukan

        if not film_found:
            print(Fore.LIGHTYELLOW_EX + "Film dengan
ID tersebut tidak ditemukan\n")
            break

    type = input(Fore.LIGHTBLUE_EX + ">>> Tipe Teater
(R untuk Regular, P untuk Premiere): ").capitalize().strip()
    match type:
        case "R":
            type = "Regular"
        case "P":
            type = "Premiere"
        case "Regular":
            type = "Regular"
        case "Premiere":

```



```

        type = "Premiere"
    case _:
        cls()
        print(Fore.LIGHTYELLOW_EX + "Input yang
anda masukkan salah\n")
        continue

    capacity = int(input(Fore.LIGHTBLUE_EX + ">>
Kapasitas Teater: ").strip())
    if capacity <= 0:
        cls()
        print(Fore.LIGHTYELLOW_EX + "Kapasitas harus
lebih dari 0\n")
        continue

    if film_id is not None:
        price = int(input(Fore.LIGHTBLUE_EX + ">>
Harga Tiket: ").strip())
        if price <= 0:
            cls()
            print(Fore.LIGHTYELLOW_EX + "Harga harus
lebih dari 0\n")
            continue
        else:
            price = None

    teater_baru = {
        "id": get_next_id(database["teaters"]),
        "name": name,
        "film_id": film_id,
        "type": type,
        "capacity": capacity,
        "available": capacity,
        "price": price
    }
    database["teaters"].append(teater_baru)
    save(database)
    cls()
    print(Fore.LIGHTGREEN_EX + "Teater baru berhasil
ditambahkan!\n")
except ValueError:

```

```

        cls()
        print(Fore.LIGHTYELLOW_EX + "Input yang anda
masukkan harus berupa angka\n")
    except TypeError:
        cls()
        print(Fore.LIGHTYELLOW_EX + "Anda memasukkan
input yang salah\n")
    case "2":
        try:
            while True:
                print(Fore.CYAN + Style.BRIGHT + "="*60)
                print(Fore.CYAN + Style.BRIGHT + "||"
+"Tambah Film".center(56)+"||")
                print(Fore.CYAN + Style.BRIGHT + "="*60)

                nama = input(Fore.LIGHTBLUE_EX + ">> Nama
Film: ").strip().capitalize()
                # cek apakah nama kosong
                if nama == "":
                    cls()
                    print(Fore.LIGHTYELLOW_EX + "Nama tidak
boleh kosong\n")

                    continue
                # cek apakah nama sudah ada
                if nama in [data['name'] for data in
database['films']]:
                    cls()
                    print(Fore.LIGHTYELLOW_EX + "Film dengan
nama tersebut sudah ada di daftar\n")
                    break

                genre_input = input(Fore.LIGHTBLUE_EX + ">>
Genre film (pisahkan dengan koma jika lebih dari satu): ").strip()
                # Memisahkan input berdasarkan koma,
menghilangkan spasi di sekitar, dan mengabaikan entri kosong
                genre = [g.strip().capitalize() for g in
genre_input.split(",") if g.strip()]

                # cek apakah genre kosong
                if not genre:
                    cls()

```

```

        print(Fore.LIGHTYELLOW_EX + "Genre tidak
boleh kosong\n")

        tambahadmin()

        filmRating = input(Fore.LIGHTBLUE_EX + ">>
Rating Film: ").strip()
        # cek apakah rating kosong
        if filmRating == "":
            cls()
            print(Fore.LIGHTYELLOW_EX + "Rating tidak
boleh kosong\n")

            continue

        film_baru = {
            "id": get_next_id(database["films"]),
            "name": nama,
            "genre": genre,
            "filmRating": filmRating
        }
        database["films"].append(film_baru)
        save(database)
        cls()
        print(Fore.LIGHTGREEN_EX + "Film baru
berhasil ditambahkan!\n")
        break
    except ValueError:
        cls()
        print(Fore.LIGHTYELLOW_EX + "Input yang anda
masukkan harus berupa angka\n")
    except TypeError:
        cls()
        print(Fore.LIGHTYELLOW_EX + "Anda memasukkan
input yang salah\n")
    case "3":
        try:
            while True:
                print(Fore.CYAN + Style.BRIGHT + "="*60)
                print(Fore.CYAN + Style.BRIGHT + "||"
+"Tambah FnB".center(56)+"||")
                print(Fore.CYAN + Style.BRIGHT + "="*60)

```

```

        nama = input(Fore.LIGHTBLUE_EX + ">> Nama: ")
        nama = nama.strip().capitalize()
        # cek apakah nama kosong
        if nama == "":
            cls()
            print(Fore.LIGHTYELLOW_EX + "Nama tidak boleh kosong\n")
            continue
        # cek apakah nama sudah ada
        if nama in [data['name'] for data in database['fnb']]:
            cls()
            print(Fore.LIGHTYELLOW_EX + "Fnb dengan nama tersebut sudah ada di daftar\n")
            break

        harga = int(input(Fore.LIGHTBLUE_EX + ">> Harga: ").strip())
        # cek apakah harga kurang dari atau sama dengan 0
        if harga <= 0:
            cls()
            print(Fore.LIGHTYELLOW_EX + "Harga tidak boleh kurang dari atau sama dengan 0\n")
            break

        stock = int(input(Fore.LIGHTBLUE_EX + ">> Stok awal: ").strip())
        # cek apakah stok kurang dari 0
        if stock < 0:
            cls()
            print(Fore.LIGHTYELLOW_EX + "Stok tidak boleh kurang dari 0\n")
            break

        fnb_baru = {
            "id": get_next_id(database["fnb"]),
            "name": nama,
            "price": harga,
            "stock": stock
        }

```

```

        database["fnb"].append(fnb_baru)
        save(database)
        cls()
        print(Fore.LIGHTGREEN_EX + "Fnb baru berhasil
ditambahkan!\n")
        break
    except ValueError:
        cls()
        print(Fore.LIGHTYELLOW_EX + "Input yang anda
masukkan harus berupa angka\n")
    except TypeError:
        cls()
        print(Fore.LIGHTYELLOW_EX + "Anda memasukkan
input yang salah\n")
    case "4":
        try:
            while True:
                print(Fore.CYAN + Style.BRIGHT + "="*60)
                print(Fore.CYAN + Style.BRIGHT + "||"
+"Tambah Pengguna".center(56)+"||")
                print(Fore.CYAN + Style.BRIGHT + "="*60)

                username = input(Fore.LIGHTBLUE_EX + ">>
Username: ").strip()

                # cek apakah username kosong
                if username == "":
                    cls()
                    print(Fore.LIGHTYELLOW_EX + "Username
tidak boleh kosong\n")
                    break
                # cek apakah username sudah ada
                if username in [data['username'] for data in
database['users']]:
                    cls()
                    print(Fore.LIGHTYELLOW_EX + "Pengguna
dengan username tersebut sudah ada\n")
                    continue

                password = input(Fore.LIGHTBLUE_EX + ">>
Password: ").strip()

                # cek apakah password kosong

```

```

        if password == "":
            cls()
            print(Fore.LIGHTYELLOW_EX + "Password
tidak boleh kosong\n")

            break

        role = input(Fore.LIGHTBLUE_EX + ">> Role (U
untuk User, A untuk Admin): ").strip().capitalize()
        # cek apakah role kosong
        match role:
            case "U":
                role = "user"
                balance = 0
            case "A":
                role = "admin"
                balance = None
            case "User":
                role = "user"
                balance = 0
            case "Admin":
                role = "admin"
                balance = None
            case _:
                cls()
                print(Fore.LIGHTYELLOW_EX + "Role
tidak valid\n")

                continue

        # membuat user baru
        regis = {
            "id": get_next_id(database["users"]),
            "username": username,
            "password": password,
            "role": role,
            "balance": balance
        }

        # menambahkan user baru ke database
        database['users'].append(regis)

        # menyimpan database baru ke file

```

```

        save(database)

        cls()
        print(Fore.LIGHTGREEN_EX + "Register
berhasil!\n")

        break
    except TypeError:
        cls()
        print(Fore.LIGHTYELLOW_EX + "Anda memasukkan
input yang salah\n")

    case "5":
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.CYAN + Style.BRIGHT + "||" + "Tambah
Komentar".center(56)+"||")
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        for user in database['users']:
            if user['username'] == username:
                user_id = user['id']
                komentar_baru = input(Fore.BLUE +
Style.BRIGHT + ">> Masukkan komentar Anda: ").strip()
                if komentar_baru == "":
                    cls()
                    print(Fore.LIGHTYELLOW_EX + "Komentar
tidak boleh kosong!\n")

                    break
                komentar = {
                    "user_id": user_id,
                    "comment": komentar_baru
                }
                database['comments'].append(komentar)
                save(database)
                cls()
                print(Fore.LIGHTGREEN_EX + "Komentar berhasil
ditambahkan!\n")

    case "":
        continue
    case _:
        print(Fore.LIGHTYELLOW_EX + "Pilihan tidak
tersedia.\n")

```

Fungsi diatas merupakan fungsi untuk menambahkan data baru seperti teater, film, FnB, Pengguna, dan komentar. Pada penambahan teater pengguna akan diminta nama teater, film yang ingin ditayangkan, harga tiket, kursi yang ingin disediakan dan tipe teaternya, untuk FnB pengguna diminta untuk menambahkan harga, stok dan nama makanan. Lalu untuk Film pengguna diminta untuk menginput nama film, genre dan ratingnya, yang terakhir menambahkan akun baru, admin diminta untuk menginput username, password dan rolenya.

```
def edit():
    while True:
        print(Fore.CYAN + Style.BRIGHT+"="*60)
        print(Fore.CYAN + Style.BRIGHT+"||"+"Edit Teater, Fnb, dan Film".center(56)+"||")
        print(Fore.CYAN + Style.BRIGHT+"="*60)
        print()
        print(Fore.BLUE + "1. " + Fore.GREEN + Style.BRIGHT + "Teater")
        print(Fore.BLUE + "2. " + Fore.GREEN + Style.BRIGHT + "Fnb")
        print(Fore.BLUE + "3. " + Fore.GREEN + Style.BRIGHT + "Film")
        print()
        edit_pilih = input(Fore.LIGHTBLUE_EX+">> Masukkan pilihan (0 untuk kembali): ").strip()
        cls()

        match edit_pilih:
            case "0":
                return

            case "1":
                try:
                    while True:
                        print(Fore.CYAN + Style.BRIGHT+"="*60)
                        print(Fore.CYAN + Style.BRIGHT+"||"+"Edit Teater, Fnb, dan Film".center(56)+"||")
                        print(Fore.CYAN + Style.BRIGHT+"="*60)
                        print(Fore.CYAN + Style.BRIGHT + "||"+"Daftar Teater dan Film:".center(56)+"||")
                        print(Fore.CYAN + Style.BRIGHT + "="*60)
```



```

        print(Fore.WHITE + "-" * 50)
        for data in database['teaters']:
            film_name =
film_dict.get(data["film_id"])
            rating = film_rating.get(data["film_id"])
            if data['film_id'] is not None:
                print(Fore.GREEN + Style.BRIGHT +
f" {data['name']})")
                print(Fore.LIGHTWHITE_EX +
f" ID          : {data['id']})")
                print(Fore.LIGHTWHITE_EX + f" Nama
Film      : {film_name}")
                print(Fore.LIGHTWHITE_EX +
f" Rating      : {rating}")
                print(Fore.LIGHTWHITE_EX +
f" Harga       : Rp.{data['price']})")
                print(Fore.LIGHTWHITE_EX + f" Jumlah
Kursi : {data['available']}/{data['capacity']})")
                print(Fore.LIGHTWHITE_EX + f" Tipe
Teater  : {data['type'].capitalize()})")
                print(Fore.WHITE + "-" * 50)
            id_teater = int(input(Fore.LIGHTBLUE_EX+">>
Masukkan ID teater yang ingin diedit (0 untuk kembali): "))
            cls()

            if id_teater == 0:
                break

            teater = next((t for t in database["teaters"]
if t["id"] == id_teater), None)

            if teater:
                print(Fore.CYAN + Style.BRIGHT+"="*60)
                print(Fore.CYAN + Style.BRIGHT+"||"+"Edit
teater".center(56)+"||")
                print(Fore.CYAN + Style.BRIGHT+"="*60)
                print()
                print(Fore.BLUE + "1."+ Fore.GREEN +
Style.BRIGHT + "Seluruhnya")

```

```

        print(Fore.BLUE + "2." + Fore.GREEN +
Style.BRIGHT + "Nama Teater")
        print(Fore.BLUE + "3." + Fore.GREEN +
Style.BRIGHT + "ID Film")
        print(Fore.BLUE + "4." + Fore.GREEN +
Style.BRIGHT + "Tipe")
        print(Fore.BLUE + "5." + Fore.GREEN +
Style.BRIGHT + "Kapasitas")
        print(Fore.BLUE + "6." + Fore.GREEN +
Style.BRIGHT + "Stok Tiket")
        print(Fore.BLUE + "7." + Fore.GREEN +
Style.BRIGHT + "Harga")

        print()
        pilih = input(Fore.LIGHTBLUE_EX + ">>
Masukkan pilihan(0 untuk kembali): ").strip()
        cls()

        match pilih:
            case "0":
                continue

            case "1":
                while True:
                    print(Fore.CYAN +
Style.BRIGHT + "=" * 60)

                    print(Fore.CYAN +
Style.BRIGHT + "||" + "Edit teater".center(56) + "||")
                    print(Fore.CYAN +
Style.BRIGHT + "=" * 60)

                    name =
input(Fore.LIGHTBLUE_EX + ">> Nama Teater: ").capitalize().strip()
                    # cek apakah nama teater
kosong

                    if name == "":
                        cls()
                        print(Fore.LIGHTYELLOW_EX
+ "Nama teater tidak boleh kosong\n")

                        continue
                    # cek apakah nama sudah ada
                    for teater in
database["teaters"]:

```

```

        if teater["name"].lower()
== name.lower():
            cls()
            print(Fore.LIGHTYELLOW
W_EX + "Teater dengan nama tersebut sudah ada\n")
            break

            film_id =
int(input(Fore.LIGHTBLUE_EX + ">> ID Film (0 untuk None): ").strip())
            if film_id == 0:
                film_id = None
            else:
                # Cek apakah id film ada
                film_found = False #
Inisialisasi flag untuk ID film

                for film in
database["films"]:

                    if film["id"] ==
film_id:

                        film_found = True
                        break # Keluar

dari loop jika film ditemukan

                if not film_found:
                    print(Fore.LIGHTYELLOW
W_EX + "Film dengan ID tersebut tidak ditemukan\n")
                    break

            type =
input(Fore.LIGHTBLUE_EX + ">> Tipe Teater (R untuk Regular, P untuk
Premiere): ").capitalize().strip()

            match type:
                case "R":
                    type = "Regular"
                case "P":
                    type = "Premiere"
                case "Regular":
                    type = "Regular"
                case "Premiere":
                    type = "Premiere"
                case _:

```

```

        cls()
        print(Fore.LIGHTYELLOW
W_EX + "Input yang anda masukkan salah\n")
        continue

        capacity =
int(input(Fore.LIGHTBLUE_EX + ">> Kapasitas Teater: ").strip())
        if capacity <= 0:
            cls()
            print(Fore.LIGHTYELLOW_EX
+ "Kapasitas harus lebih dari 0\n")
            continue

        if film_id is not None:
            price =
int(input(Fore.LIGHTBLUE_EX + ">> Harga Tiket: ").strip())
            if price <= 0:
                cls()
                print(Fore.LIGHTYELLOW
W_EX + "Harga harus lebih dari 0\n")
                continue
            else:
                price = None

        teater.update({
            "id":
get_next_id(database["teaters"]),

            "name": name,
            "film_id": film_id,
            "type": type,
            "capacity": capacity,
            "available": capacity,
            "price": price
        })
        save(database)
        cls()
        print(Fore.LIGHTBLUE_EX+"Data
teater berhasil diubah.\n")

        break

    case "2":

```

```

print("="*60)
print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit Nama".center(56)+"||")
print(Fore.CYAN +
Style.BRIGHT+"="*60)

print()
name = input(Fore.LIGHTBLUE_EX +
">> Nama Teater: ").capitalize().strip()
# cek apakah nama teater kosong
if name == "":
    cls()
    print(Fore.LIGHTYELLOW_EX +
"Nama teater tidak boleh kosong\n")
    break
# cek apakah nama sudah ada
for teater in
database["teaters"]:
    if teater["name"].lower() ==
name.lower():
        cls()
        print(Fore.LIGHTYELLOW_EX
+ "Teater dengan nama tersebut sudah ada\n")
        break
    teater["name"] = name
    save(database)
    print(Fore.LIGHTBLUE_EX+"Nama
teater berhasil diubah.")
    input(Fore.LIGHTBLUE_EX+"\nSilahk
an tekan enter untuk kembali")
    cls()

case "3":
    print(Fore.CYAN +
Style.BRIGHT+"="*60)
    print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit teater".center(56)+"||")
    print(Fore.CYAN +
Style.BRIGHT+"="*60)

    print()
    film_id =
int(input(Fore.LIGHTBLUE_EX + ">> ID Film (0 untuk None): ").strip()))

```

```

        if film_id == 0:
            film_id = None
        else:
            # Cek apakah id film ada
            film_found = False #

            for film in
                database["films"]:
                    if film["id"] == film_id:
                        film_found = True
                        break # Keluar dari

            loop jika film ditemukan

            if not film_found:
                print(Fore.LIGHTYELLOW_EX
+ "Film dengan ID tersebut tidak ditemukan\n")
                break
            teater["film_id"] = film_id
            save(database)
            print(Fore.LIGHTBLUE_EX+"ID film
berhasil diubah.")

            input(Fore.LIGHTBLUE_EX+"\nSilahk
an tekan enter untuk kembali")

            cls()

        case "4":
            os.system('cls')
            print(Fore.CYAN +
Style.BRIGHT+"="*60)

            print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit teater".center(56)+"||")
            print(Fore.CYAN +
Style.BRIGHT+"="*60)

            print()
            type = input(Fore.LIGHTBLUE_EX +
">> Tipe Teater (R untuk Regular, P untuk Premiere):
").capitalize().strip()

            match type:
                case "R":
                    type = "Regular"
                case "P":

```

```

        type = "Premiere"
    case "Regular":
        type = "Regular"
    case "Premiere":
        type = "Premiere"
    case _:
        cls()
        print(Fore.LIGHTYELLOW_EX
+ "Input yang anda masukkan salah\n")
        continue
    teater["type"] = type
    save(database)
    print(Fore.LIGHTBLUE_EX+"Tipe
teater berhasil diubah.")
    input(Fore.LIGHTBLUE_EX+"\nSilahk
an tekan enter untuk kembali")
    cls()

    case "5":
        print(Fore.CYAN +
Style.BRIGHT+"="*60)
        print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit teater".center(56)+"||")
        print(Fore.CYAN +
Style.BRIGHT+"="*60)
        print()
        capacity =
int(input(Fore.LIGHTBLUE_EX + ">> Kapasitas Teater: ").strip())
        if capacity <= 0:
            cls()
            print(Fore.LIGHTYELLOW_EX +
"Kapasitas harus lebih dari 0\n")
            continue
        teater["capacity"] = capacity
        save(database)
        print(Fore.LIGHTBLUE_EX+"Kapasita
s teater berhasil diubah.")
        input(Fore.LIGHTBLUE_EX+"\nSilahk
an tekan enter untuk kembali")

    case "6":

```

```

print(Fore.CYAN +
Style.BRIGHT+"="*60)

print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit teater".center(56)+"||")
print(Fore.CYAN +
Style.BRIGHT+"="*60)

print()
available =
int(input(Fore.LIGHTBLUE_EX + ">> Stok Tiket: ").strip())
if available < 0:
    cls()
    print(Fore.LIGHTYELLOW_EX +
"Stok tiket harus lebih dari 0\n")
    continue
elif available >
teater["capacity"]:
    cls()
    print(Fore.LIGHTYELLOW_EX +
"Stok tiket tidak boleh melebihi kapasitas teater\n")
    continue

teater["available"] = available
save(database)
print(Fore.LIGHTBLUE_EX+"Stok
tiket teater berhasil diubah.")
input(Fore.LIGHTBLUE_EX+"\nSilahk
an tekan enter untuk kembali")

cls()

case "7":
    print("="*60)
    print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit teater".center(56)+"||")
    print(Fore.CYAN +
Style.BRIGHT+"="*60)

    print()
    if film_id is not None:
        price =
int(input(Fore.LIGHTBLUE_EX + ">> Harga Tiket: ").strip())
        if price <= 0:

```



```

        cls()
        print(Fore.LIGHTYELLOW_EX
+ "Harga harus lebih dari 0\n")
        continue
    else:
        print(Fore.LIGHTYELLOW_EX+"Harap memilih film terlebih dahulu\n")
        break
    teater["price"] =
int(input(Fore.LIGHTBLUE_EX+">> Masukkan harga tiket yang baru: "))
    save(database)
    print(Fore.LIGHTBLUE_EX+"Harga
tiket teater berhasil diubah.")
    input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan enter untuk kembali")

    case "":
        continue

    case _:
        print(Fore.LIGHTBLUE_EX+"Pilihan
tidak ada.")
        input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan enter untuk kembali")
    else:
        print(Fore.LIGHTYELLOW_EX+"Teater dengan
ID tersebut tidak ditemukan.")
        input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan enter untuk kembali")

    except ValueError:
        cls()
        print(Fore.LIGHTYELLOW_EX + "Input yang anda
masukkan harus berupa angka\n")
    except TypeError:
        cls()
        print(Fore.LIGHTYELLOW_EX + "Anda memasukkan
input yang salah\n")

    case "2":
        try:

```

```

        while True:
            print(Fore.CYAN + Style.BRIGHT+"="*60)
            print(Fore.CYAN + Style.BRIGHT+"||"+"Edit
Teater, Fnb, dan Film".center(56)+"||")
            print(Fore.CYAN + Style.BRIGHT+"="*60)
            print(Fore.CYAN + Style.BRIGHT + "||"+"Daftar
FnB".center(56)+"||")

            print(Fore.CYAN + Style.BRIGHT + "="*60)
            print(Fore.WHITE + "-" * 50)
            if not database['fnb']:
                print(Fore.LIGHTYELLOW_EX + "Tidak ada
data FnB\n")

            else:
                for data in database['fnb']:
                    print(Fore.LIGHTWHITE_EX +
f" ID      : {data['id']}")
                    print(Fore.LIGHTWHITE_EX +
f" Nama    : {data['name']}")
                    print(Fore.LIGHTWHITE_EX +
f" Stok    : {data['stock']}")
                    print(Fore.LIGHTWHITE_EX +
f" Harga   : Rp.{data['price']}")
                    print(Fore.WHITE + "-" * 50)
                id_fnb = int(input(Fore.LIGHTBLUE_EX+">>
Masukkan ID fnb yang ingin diedit (0 untuk kembali): "))
                cls()

                if id_fnb == 0:
                    break

                fnb = next((t for t in database["fnb"] if
t["id"] == id_fnb), None)

                if fnb:
                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print(Fore.CYAN + Style.BRIGHT+"||"+"Edit
fnb".center(56)+"||")

                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print()
                    print(Fore.BLUE + "1."+ Fore.GREEN +
Style.BRIGHT + "Seluruhnya")

```

```

Style.BRIGHT + "Nama")
print(Fore.BLUE + "2." + Fore.GREEN +
Style.BRIGHT + "Harga")
print(Fore.BLUE + "3." + Fore.GREEN +
Style.BRIGHT + "Stok")
print()
pilih = input(Fore.LIGHTBLUE_EX + ">>
Masukkan pilihan (0 untuk kembali): ").strip()
cls()

match pilih:
    case "0":
        continue

    case "1":
        while True:
            print(Fore.CYAN +
Style.BRIGHT + "=" * 60)
            print(Fore.CYAN +
Style.BRIGHT + "||" + "Edit fnb".center(56) + "||")
            print(Fore.CYAN +
Style.BRIGHT + "=" * 60)

            print()
            nama =
input(Fore.LIGHTBLUE_EX + ">> Nama: ").strip().capitalize()
            # cek apakah nama kosong
            if nama == "":
                cls()
                print(Fore.LIGHTYELLOW_EX
+ "Nama tidak boleh kosong\n")
                continue
            # cek apakah nama sudah ada
            if nama in [data['name'] for
data in database['fnb']]:
                cls()
                print(Fore.LIGHTYELLOW_EX
+ "FnB dengan nama tersebut sudah ada di daftar\n")
                break

```

```

        harga =
int(input(Fore.LIGHTBLUE_EX + ">> Harga: ").strip())
        # cek apakah harga kurang
dari atau sama dengan 0

        if harga <= 0:
            cls()
            print(Fore.LIGHTYELLOW_EX
+ "Harga tidak boleh kurang dari atau sama dengan 0\n")
            break

        stock =
int(input(Fore.LIGHTBLUE_EX + ">> Stok awal: ").strip())
        # cek apakah stok kurang dari
0

        if stock < 0:
            cls()
            print(Fore.LIGHTYELLOW_EX
+ "Stok tidak boleh kurang dari 0\n")
            break

        fnb.update({
            "id":
get_next_id(database["fnb"]),

            "name": nama,
            "price": harga,
            "stock": stock
        })
        save(database)
        cls()
        print(Fore.LIGHTBLUE_EX+"Selu
ruh data fnb berhasil diubah.\n")

        case "2":
            print(Fore.CYAN +
Style.BRIGHT+"="*60)

            print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit fnb".center(56)+"||")
            print(Fore.CYAN +
Style.BRIGHT+"="*60)

            print()

```

```

nama = input(Fore.LIGHTBLUE_EX +
">> Nama: ").strip().capitalize()
# cek apakah nama kosong
if nama == "":
    cls()
    print(Fore.LIGHTYELLOW_EX +
"Nama tidak boleh kosong\n")
    continue
# cek apakah nama sudah ada
if nama in [data['name'] for data
in database['fnb']]:
    cls()
    print(Fore.LIGHTYELLOW_EX +
"FnB dengan nama tersebut sudah ada di daftar\n")
    break
fnb["food"] = nama
save(database)
cls()
print(Fore.LIGHTBLUE_EX+"Data
makanan fnb berhasil diubah.\n")

case "3":
    print(Fore.CYAN +
Style.BRIGHT+"="*60)
    print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit fnb".center(56)+"||")
    print(Fore.CYAN +
Style.BRIGHT+"="*60)
    print()
    harga =
int(input(Fore.LIGHTBLUE_EX + ">> Harga: ").strip())
# cek apakah harga kurang dari
atau sama dengan 0
    if harga <= 0:
        cls()
        print(Fore.LIGHTYELLOW_EX +
"Harga tidak boleh kurang dari atau sama dengan 0\n")
        break
    fnb["price"] = harga
    save(database)
    cls()

```

```

print(Fore.LIGHTBLUE_EX+"Data
harga fnb berhasil diubah.\n")

    case "4":
        print(Fore.CYAN +
Style.BRIGHT+"="*60)

        print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit fnb".center(56)+"||")
        print(Fore.CYAN +
Style.BRIGHT+"="*60)

        print()
        stock =
int(input(Fore.LIGHTBLUE_EX + ">> Stok awal: ").strip())
        # cek apakah stok kurang dari 0
        if stock < 0:
            cls()
            print(Fore.LIGHTYELLOW_EX +
"Stok tidak boleh kurang dari 0\n")

            break
        fnb["stock"] = stock
        save(database)
        cls()
        print(Fore.LIGHTBLUE_EX+"Data
stok fnb berhasil diubah.\n")

    case "":
        continue

    case _:
        print(Fore.LIGHTBLUE_EX+"Tidak
ada dalam pilihan")

        input(Fore.LIGHTBLUE_EX+"\nSilahk
an tekan enter untuk kembali")

    else:
        print(Fore.LIGHTBLUE_EX+"Fnb dengan ID
tersebut tidak ditemukan.")
        input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan
enter untuk kembali")

        cls()

```

```

        except ValueError:
            cls()
            print(Fore.LIGHTYELLOW_EX + "Input yang anda
masukkan harus berupa angka\n")
        except TypeError:
            cls()
            print(Fore.LIGHTYELLOW_EX + "Anda memasukkan
input yang salah\n")

    case "3":
        try:
            while True:
                print(Fore.CYAN + Style.BRIGHT + "="*60)
                print(Fore.CYAN + Style.BRIGHT+"||"+"Edit
Teater, Fnb, dan Film".center(56)+"||")
                print(Fore.CYAN + Style.BRIGHT + "="*60)
                print(Fore.CYAN + Style.BRIGHT + "||"+"Daftar
Film".center(56)+"||")
                print(Fore.CYAN + Style.BRIGHT + "="*60)
                print(Fore.WHITE + "-" * 50)
                if not database['films']:
                    print(Fore.LIGHTYELLOW_EX + "Tidak ada
data film\n")
                else:
                    for data in database['films']:
                        print(Fore.LIGHTWHITE_EX +
f" ID      : {data['id']}"")
                        print(Fore.LIGHTWHITE_EX +
f" Nama    : {data['name']}"")
                        print(Fore.LIGHTWHITE_EX +
f" Genre   : {', '.join(data['genre'])}"")
                        print(Fore.LIGHTWHITE_EX + f" Rating
: {data['filmRating']}"")
                        print(Fore.WHITE + "-" * 50)
                    id_film =
int(input(Fore.LIGHTBLUE_EX+"Masukkan ID Film (0 untuk kembali): "))
                    cls()

                    if id_film == 0:
                        break

```

```

        films = next((t for t in database["films"] if
t["id"] == id_film), None)
        if films:
            print(Fore.CYAN + Style.BRIGHT+"="*60)
            print(Fore.CYAN + Style.BRIGHT+"||"+"Edit
Film".center(56)+"||")

            print(Fore.CYAN + Style.BRIGHT+"="*60)
            print()
            print(Fore.BLUE + "1."+ Fore.GREEN +
Style.BRIGHT + " Seluruhnya")
            print(Fore.BLUE + "2."+ Fore.GREEN +
Style.BRIGHT + " Nama")
            print(Fore.BLUE + "3."+ Fore.GREEN +
Style.BRIGHT + " Genre")
            print(Fore.BLUE + "4."+ Fore.GREEN +
Style.BRIGHT + " Rating")
            print()
            pilih = input(Fore.LIGHTBLUE_EX+">>
Masukkan pilihan (0 untuk kembali): ").strip()
            cls()

            match pilih:
                case "0":
                    continue
                case "1":
                    print(Fore.CYAN +
Style.BRIGHT+"="*60)

                    print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit Film".center(56)+"||")
                    print(Fore.CYAN +
Style.BRIGHT+"="*60)

                    print()
                    nama = input(Fore.LIGHTBLUE_EX +
">> Nama Film: ").strip().capitalize()
                    # cek apakah nama kosong
                    if nama == "":
                        cls()
                        print(Fore.LIGHTYELLOW_EX +
>Nama tidak boleh kosong\n")

                        continue
                    # cek apakah nama sudah ada

```



```

        if nama in [data['name'] for data
in database['films']]:
            cls()
            print(Fore.LIGHTYELLOW_EX +
"Film dengan nama tersebut sudah ada di daftar\n")
            break

            genre_input =
input(Fore.LIGHTBLUE_EX + ">> Genre film (pisahkan dengan koma jika
lebih dari satu): ").strip()
            # Memisahkan input berdasarkan
koma, menghilangkan spasi di sekitar, dan mengabaikan entri kosong
            genre = [g.strip().capitalize()
for g in genre_input.split(",") if g.strip()]

            # cek apakah genre kosong
            if not genre:
                cls()
                print(Fore.LIGHTYELLOW_EX +
"Genre tidak boleh kosong\n")
                continue

            filmRating =
input(Fore.LIGHTBLUE_EX + ">> Rating Film: ").strip()
            # cek apakah rating kosong
            if filmRating == "":
                cls()
                print(Fore.LIGHTYELLOW_EX +
"Rating tidak boleh kosong\n")
                continue

            films.update({
                "id":
get_next_id(database["films"]),
                "name": nama,
                "genre": genre,
                "filmRating": filmRating
            })
            save(database)
            print(Fore.LIGHTBLUE_EX+"\nSeluruh
data film berhasil diubah.")

```

```

an tekan enter untuk kembali")
        input(Fore.LIGHTBLUE_EX+"\nSilahk

        cls()

        case "2":
            print(Fore.CYAN +
Style.BRIGHT+"="*60)

            print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit Film".center(56)+"||")
            print(Fore.CYAN +
Style.BRIGHT+"="*60)

            print()
            nama = input(Fore.LIGHTBLUE_EX +
">> Nama Film: ").strip().capitalize()
            # cek apakah nama kosong
            if nama == "":
                cls()
                print(Fore.LIGHTYELLOW_EX +
"Nama tidak boleh kosong\n")

                continue
            # cek apakah nama sudah ada
            if nama in [data['name'] for data
in database['films']]:
                cls()
                print(Fore.LIGHTYELLOW_EX +
"Film dengan nama tersebut sudah ada di daftar\n")
                break
            films["name"] = nama
            save(database)
            print(Fore.LIGHTBLUE_EX+"\nData
nama film berhasil diubah.")

            input(Fore.LIGHTBLUE_EX+"\nSilahk
an tekan enter untuk kembali")

            cls()

            case "3":
                print(Fore.CYAN +
Style.BRIGHT+"="*60)

                print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit Film".center(56)+"||")

```

```

print(Fore.CYAN +
Style.BRIGHT+"="*60)

print()
genre_input =
input(Fore.LIGHTBLUE_EX + ">> Genre film (pisahkan dengan koma jika
lebih dari satu): ").strip()

# Memisahkan input berdasarkan
koma, menghilangkan spasi di sekitar, dan mengabaikan entri kosong
genre = [g.strip().capitalize()
for g in genre_input.split(",") if g.strip()]

# cek apakah genre kosong
if not genre:
    cls()
    print(Fore.LIGHTYELLOW_EX +
"Genre tidak boleh kosong\n")
    continue
films["genre"] = genre
save(database)
print(Fore.LIGHTBLUE_EX+"\nData
genre berhasil diubah.")

input(Fore.LIGHTBLUE_EX+"\nSilahk
an tekan enter untuk kembali")

cls()

case "4":
    print(Fore.CYAN +
Style.BRIGHT+"="*60)

    print(Fore.CYAN +
Style.BRIGHT+"||"+"Edit Film".center(56)+"||")
    print(Fore.CYAN +
Style.BRIGHT+"="*60)

    print()
    filmRating =
input(Fore.LIGHTBLUE_EX + ">> Rating Film: ").strip()
    # cek apakah rating kosong
    if filmRating == "":
        cls()
        print(Fore.LIGHTYELLOW_EX +
"Rating tidak boleh kosong\n")
        continue

```

```

        films["filmRating"] = filmRating
        save(database)
        print(Fore.LIGHTBLUE_EX+"\nData
rating film berhasil diubah.")
        input(Fore.LIGHTBLUE_EX+"\nSilahk
an tekan enter untuk kembali")
        cls()

    case "":
        continue

    case _:
        print(Fore.LIGHTBLUE_EX+"\nTidak
ada dalam pilihan")
        input(Fore.LIGHTBLUE_EX+"\nSilahk
an tekan enter untuk kembali")
        cls()

    else:
        print(Fore.LIGHTBLUE_EX+"Film dengan ID
tersebut tidak ditemukan.")
        input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan
enter untuk kembali")

    except ValueError:
        cls()
        print(Fore.LIGHTYELLOW_EX + "Input yang anda
masukkan harus berupa angka\n")
    except TypeError:
        cls()
        print(Fore.LIGHTYELLOW_EX + "Anda memasukkan
input yang salah\n")

    case "":
        continue

    case _:
        cls()
        print(Fore.LIGHTBLUE_EX+"Tidak ada dalam pilihan")
        input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan enter untuk
kembali")

```

```
cls()
```

Pada fungsi ini pengguna dapat mengubah data teater yaitu film yang ingin ditayangkan pada teater tersebut, tipe teater, kapasitas teater, stok tiket, dan harga. Lalu untuk FnB admin dapat mengubah harga, stok dan nama makanannya, yang terakhir ada film dimana admin dapat merubah nama, rating dan genre film tersebut.

```
def hapus():
    print(Fore.CYAN + Style.BRIGHT+"="*60)
    print(Fore.CYAN + Style.BRIGHT+"||"+"Hapus Teater, Fnb, dan
Film".center(56)+"||")
    print(Fore.CYAN + Style.BRIGHT+"="*60)
    print()
    print(Fore.BLUE +"1."+ Fore.GREEN + Style.BRIGHT + " Teater")
    print(Fore.BLUE +"2."+ Fore.GREEN + Style.BRIGHT + " FnB")
    print(Fore.BLUE +"3."+ Fore.GREEN + Style.BRIGHT + " Film")
    print()
    pilih_hapus = input(Fore.LIGHTBLUE_EX+">> Masukkan Pilihan (pilih
0 untuk kembali): ")
    cls()

    if pilih_hapus == "1":
        print(Fore.CYAN + Style.BRIGHT+"="*60)
        print(Fore.CYAN + Style.BRIGHT+"||"+"Hapus
teater".center(56)+"||")
        print(Fore.CYAN + Style.BRIGHT+"="*60)
        print(Fore.CYAN + Style.BRIGHT + "||"+"Daftar
Teater".center(56)+"||")
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.WHITE + "-" * 50)
        if not database['teaters']:
            print(Fore.LIGHTYELLOW_EX + "Tidak ada database
teater\n")
        else:
            for data in database['teaters']:
                film_name = film_dict.get(data["film_id"])
                print(Fore.LIGHTWHITE_EX + f" ID          :
{data['id']}")
```

```

        print(Fore.LIGHTWHITE_EX + f" Nama Teater :
{data['name']}")
        print(Fore.LIGHTWHITE_EX + f" Nama Film :
{film_name}")
        print(Fore.LIGHTWHITE_EX + f" Harga : {'None'
if data['price'] is None else 'Rp.' + str(data['price'])}")
        print(Fore.LIGHTWHITE_EX + f" Jumlah Kursi :
{data['available']}/{data['capacity']}")
        print(Fore.LIGHTWHITE_EX + f" Tipe Teater :
{data['type'].capitalize()}")
        print(Fore.WHITE + "-" * 50)
    try:
        id_teater = int(input(Fore.LIGHTBLUE_EX + ">> Masukkan ID
teater: "))
        cls()
    except ValueError:
        print(Fore.LIGHTRED_EX + "Input harus berupa angka")
        input(Fore.LIGHTBLUE_EX + "\nTekan Enter untuk kembali")
        cls()
        return
    teater = next((t for t in database["teaters"] if t["id"] ==
id_teater), None)

    if not teater:
        print(Fore.LIGHTRED_EX + "Teater dengan ID tersebut
tidak ditemukan.")
        input(Fore.LIGHTBLUE_EX + "\nTekan Enter untuk
kembali.")
        cls()
        return
    while True:
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.CYAN + Style.BRIGHT + "||" + "Hapus
teater".center(56) + "||")
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print()
        print(Fore.BLUE + "1." + Fore.GREEN + Style.BRIGHT + "
Yes")
        print(Fore.BLUE + "2." + Fore.GREEN + Style.BRIGHT + " No")
        menghapus = input(Fore.LIGHTBLUE_EX + ">> Masukkan pilihan:
")

```

```

        try:
            match menghapus:
                case "0":
                    os.system('cls')
                    return
                case "1":
                    database["teaters"].remove(teater)
                    save(database)
                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print(Fore.CYAN + Style.BRIGHT+f"Teater
dengan ID {id_teater} berhasil dihapus".center(56))
                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print()
                    input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan
enter untuk kembali")
                    cls()
                    return

                case "2":
                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print(Fore.CYAN + Style.BRIGHT+"Anda batal
menghapus".center(56))
                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print()
                    input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan
enter untuk kembali")
                    cls()
                    return

            except ValueError:
                print(Fore.LIGHTRED_EX + "Input salah")
                input(Fore.LIGHTBLUE_EX + "\nTekan Enter untuk
mencoba lagi")
                cls()

        elif pilih_hapus == "2":
            print(Fore.CYAN + Style.BRIGHT+"="*60)
            print(Fore.CYAN + Style.BRIGHT+"||"+"Hapus
Fnb".center(56)+"||")

```

```

        print(Fore.CYAN + Style.BRIGHT+"="*60)
        print(Fore.CYAN + Style.BRIGHT + "||"+"Daftar
FnB".center(56)+"||")
        print(Fore.CYAN + Style.BRIGHT + "="*60)
        print(Fore.WHITE + "-" * 50)
        if not database['fnb']:
            print(Fore.LIGHTYELLOW_EX + "Tidak ada database FnB\n")
        else:
            for data in database['fnb']:
                print(Fore.LIGHTWHITE_EX + f" ID      :
{data['id']}")
                print(Fore.LIGHTWHITE_EX + f" Nama    :
{data['name']}")
                print(Fore.LIGHTWHITE_EX + f" Stok    :
{data['stock']}")
                print(Fore.LIGHTWHITE_EX + f" Harga   :
Rp.{data['price']}")
                print(Fore.WHITE + "-" * 50)
            try:
                id_fnb = int(input(Fore.LIGHTBLUE_EX+">> Masukkan ID fnb:
"))
                cls()
            except ValueError:
                print(Fore.LIGHTRED_EX + "Input harus berupa angka.")
                input(Fore.LIGHTBLUE_EX + "\nTekan Enter untuk kembali.")
                cls()
                return
            fnb = next((t for t in database["fnb"] if t["id"] == id_fnb),
None)

            if not fnb:
                print(Fore.LIGHTRED_EX + "Fnb dengan ID tersebut
tidak ditemukan.")
                input(Fore.LIGHTBLUE_EX + "\nTekan Enter untuk
kembali.")
                cls()
                return
            while True:
                print(Fore.CYAN + Style.BRIGHT+"="*60)
                print(Fore.CYAN + Style.BRIGHT+"||"+"Hapus
fnb".center(56)+"||")

```



```

        print(Fore.CYAN + Style.BRIGHT+"="*60)
        print()
        print(Fore.BLUE + "1." + Fore.GREEN + Style.BRIGHT + "
Yes")

        print(Fore.BLUE + "2." + Fore.GREEN + Style.BRIGHT + " No")
        menghapus = input(Fore.LIGHTBLUE_EX+">> Masukkan pilihan
(pilih 0 untuk kembali): ")
        try:
            match menghapus:
                case "0":
                    os.system('cls')
                    return
                case "1":
                    database["fnb"].remove(fnb)
                    save(database)
                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print(Fore.CYAN + Style.BRIGHT+f"Fnb dengan
ID {id_fnb} berhasil dihapus".center(56))
                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print()
                    input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan
enter untuk kembali")
                    cls()
                    return

                case "2":
                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print(Fore.CYAN + Style.BRIGHT+"Anda batal
menghapus")

                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print()
                    input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan
enter untuk kembali")
                    cls()
                    return

            except ValueError:
                print(Fore.LIGHTRED_EX + "Input salah")
                input(Fore.LIGHTBLUE_EX + "\nTekan Enter untuk
mencoba lagi")
                cls()

```

```

elif pilih_hapus == "3":
    print(Fore.CYAN + Style.BRIGHT+"="*60)
    print(Fore.CYAN + Style.BRIGHT+"||"+"Hapus
Film".center(56)+"||")
    print(Fore.CYAN + Style.BRIGHT+"="*60)
    print(Fore.CYAN + Style.BRIGHT + "||"+"Daftar
Film".center(56)+"||")
    print(Fore.CYAN + Style.BRIGHT + "="*60)
    print(Fore.WHITE + "-" * 50)
    if not database['films']:
        print(Fore.LIGHTYELLOW_EX + "Tidak ada database film\n")
    else:
        for data in database['films']:
            print(Fore.LIGHTWHITE_EX + f" ID      :
{data['id']}")
            print(Fore.LIGHTWHITE_EX + f" Nama      :
{data['name']}")
            print(Fore.LIGHTWHITE_EX + f" Genre : {'',
'.join(data['genre'])}")
            print(Fore.LIGHTWHITE_EX + f" Rating :
{data['filmRating']}")
            print(Fore.WHITE + "-" * 50)
        try:
            id_film = int(input(Fore.LIGHTBLUE_EX+">> Masukkan ID
film: "))
            cls()
        except ValueError:
            print(Fore.LIGHTRED_EX + "Input harus berupa angka.")
            input(Fore.LIGHTBLUE_EX + "\nTekan Enter untuk kembali.")
            cls()
            return
        film = next((t for t in database["films"] if t["id"] ==
id_film), None)

        if not film:
            print(Fore.LIGHTRED_EX + "Film dengan ID tersebut
tidak ditemukan.")
            input(Fore.LIGHTBLUE_EX + "\nTekan Enter untuk
kembali.")
            cls()

```

```

        return
    while True:
        print(Fore.CYAN + Style.BRIGHT+"="*60)
        print(Fore.CYAN + Style.BRIGHT+"||"+"Hapus
Film".center(56)+"||")
        print(Fore.CYAN + Style.BRIGHT+"="*60)
        print()
        print(Fore.BLUE + "1." + Fore.GREEN + Style.BRIGHT + "
Yes")
        print(Fore.BLUE + "2." + Fore.GREEN + Style.BRIGHT + " No")
        menghapus = input(Fore.LIGHTBLUE_EX+"> Masukkan pilihan
(pilih 0 untuk kembali): ")

        try:
            match menghapus:
                case "0":
                    os.system('cls')
                    return
                case "1":
                    database["films"].remove(film)
                    save(database)
                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print(Fore.CYAN + Style.BRIGHT+f"Film dengan
ID {id_film} berhasil dihapus".center(56))
                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print()
                    input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan
enter untuk kembali")
                    cls()
                    return

                case "2":
                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print(Fore.CYAN + Style.BRIGHT+"Anda batal
menghapus".center(56))
                    print(Fore.CYAN + Style.BRIGHT+"="*60)
                    print()
                    input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan
enter untuk kembali")
                    cls()
                    return

```

```
        except ValueError:
            print(Fore.LIGHTRED_EX + "Input salah")
            input(Fore.LIGHTBLUE_EX + "\nTekan Enter untuk
mencoba lagi")
            cls()

        else:
            print(Fore.LIGHTBLUE_EX+"Tidak ada dalam pilihan")
            input(Fore.LIGHTBLUE_EX+"\nSilahkan tekan enter untuk
kembali")
            cls()
```

Fungsi ini memiliki kode yang panjang namun sebenarnya simple yaitu untuk menghapus data yang ada pada database seperti menghapus teater, film, dan FnB yang diinginkan.

BAB IV

PENUTUP

4.1 Kesimpulan

Program yang kami membuat diharapkan dapat membantu user dalam melakukan pembelian ataupun keperluan lainnya seperti melihat riwayat transaksi bahkan berdiskusi tentang makanan ataupun film yang ada di bioskop. Program ini juga diharapkan dapat membantu admin dalam mengelola bioskop dan melakukan perubahan-perubahan yang diinginkan seperti menghapus, menambahkan, ataupun mengubah data yang ada di bioskop. Admin juga dapat berdiskusi kepada user tentang apa saja di bioskop seperti saran film untuk kedepannya ataupun melihat keluhan konsumen apabila maknannya kurang enak atau mahal harga tiket bioskop.

4.2 Saran

Untuk kedepannya kami harap program kami dapat berkembang dengan poin-poin berikut:

1. Pengembangan Sistem:

Perlu ditambahkan fitur penentuan kursi yang lebih rinci, sehingga pengguna dapat memilih kursi yang diinginkan secara langsung berdasarkan visualisasi tata letak teater.

2. Keamanan Sistem:

Mengimplementasikan enkripsi untuk data pengguna, terutama kata sandi, agar privasi dan keamanan pengguna tetap terjaga.

3. Fitur Tambahan:

Menambahkan fitur rekomendasi film atau makanan berdasarkan histori pembelian pengguna untuk meningkatkan personalisasi layanan.

DAFTAR PUSTAKA

“Python Modules.” GeeksforGeeks, 9 Aug. 2024

www.geeksforgeeks.org/python-modules/. Accessed 22 Nov. 2024

Json - JSON Encoder and Decoder.” Python Documentation,
docs.python.org/3/library/json.html. Accessed 22 Nov. 2024.

“W3Schools.Com.” W3Schools Online Web Tutorials,
www.w3schools.com/python/. Accessed 22 Nov. 2024.

LAMPIRAN

<p>Tanggal Konsultasi : 21 Oktober 2024</p> <p>Uraian / Pembahasan : 1. Mendefinisikan flourescent 2. pembuatan bakteri program 3. teknik kultur dengan baki 4. Pengelompokan masing-masing fungsi</p>	
<p>Asisten Lab</p> <p><i>[Signature]</i> Nama: Muhammad Ghazali</p>	<p>Ketua Kelompok</p> <p><i>[Signature]</i> Nama: Muhammad Ghazali</p>

<p>Tanggal Konsultasi : 14 November 2024</p> <p>Uraian / Pembahasan : 1. Fitur kromatografi dengan pengelompokan 2. Bagaimana prosedur kultur 3. Pembuatan kultur dengan 4. Fitur kromatografi dengan pengelompokan</p>	
<p>Asisten Lab</p> <p><i>[Signature]</i> Nama: Md. Ghazali</p>	<p>Ketua Kelompok</p> <p><i>[Signature]</i> Nama: Fikri Fauzan Kurniawan</p>

Gambar Kertas Konsul