

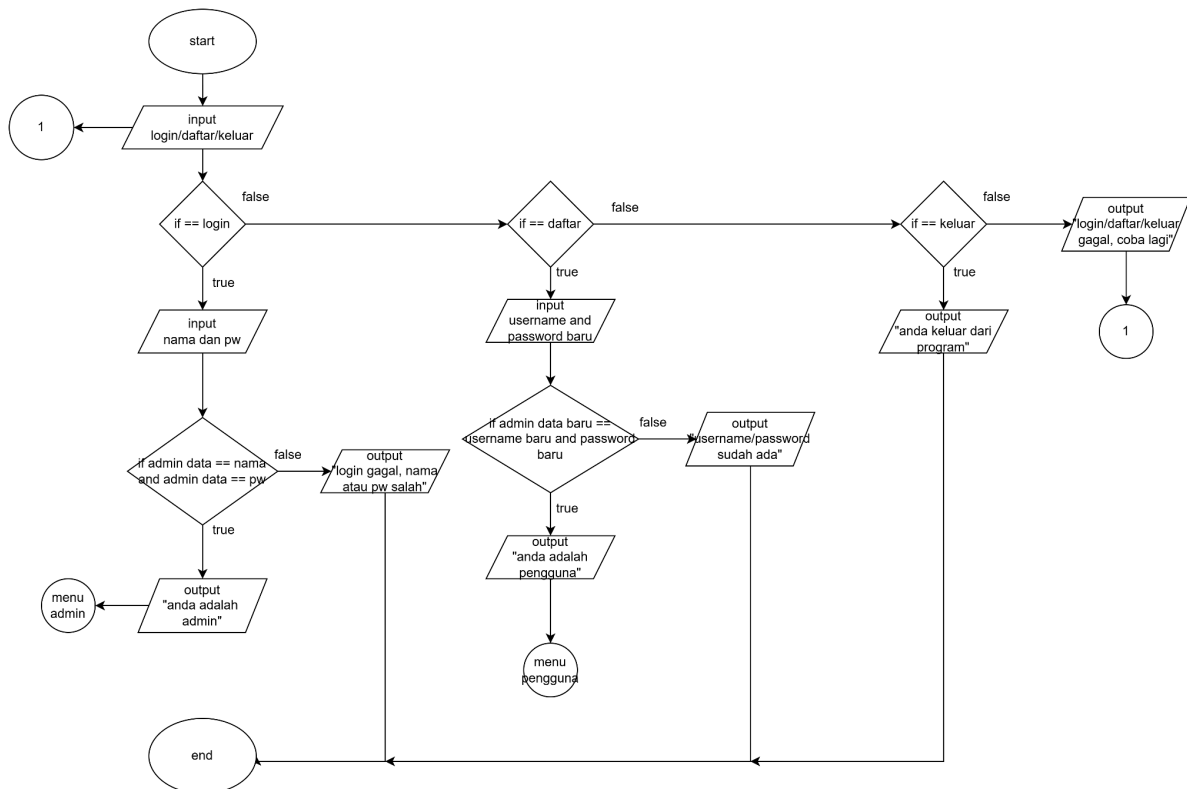
LAPORAN PRAKTIKUM
POSTTEST (8)
ALGORITMA PEMROGRAMAN DASAR



Disusun oleh:
Intan Indah Sari (2509106088)
Kelas (B2 '25)

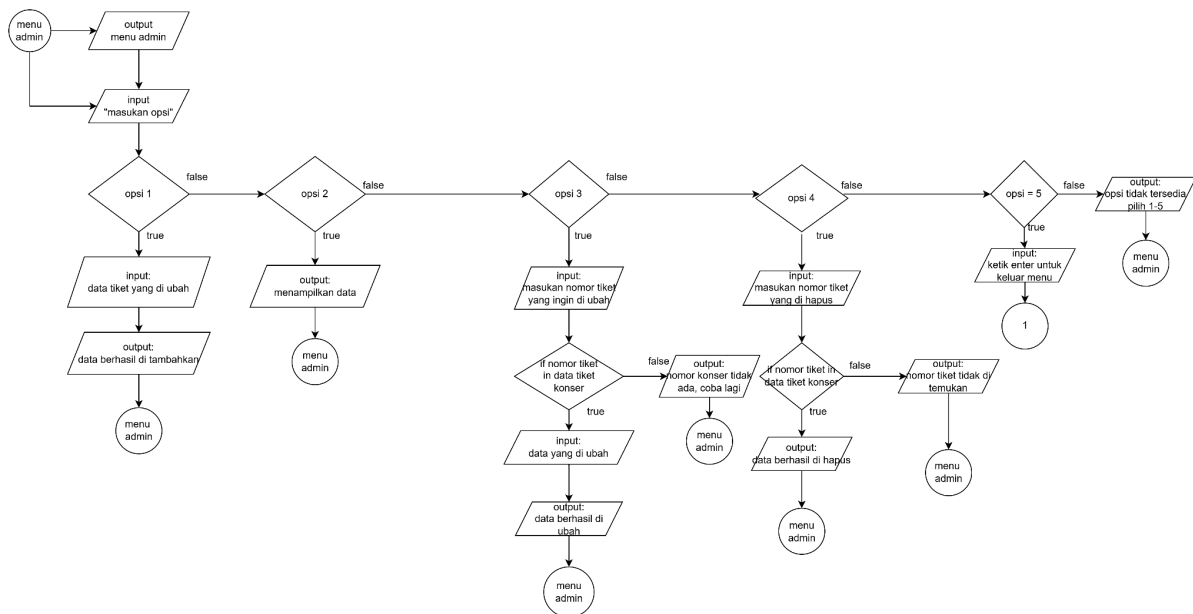
PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA
2025

1. Flowchart



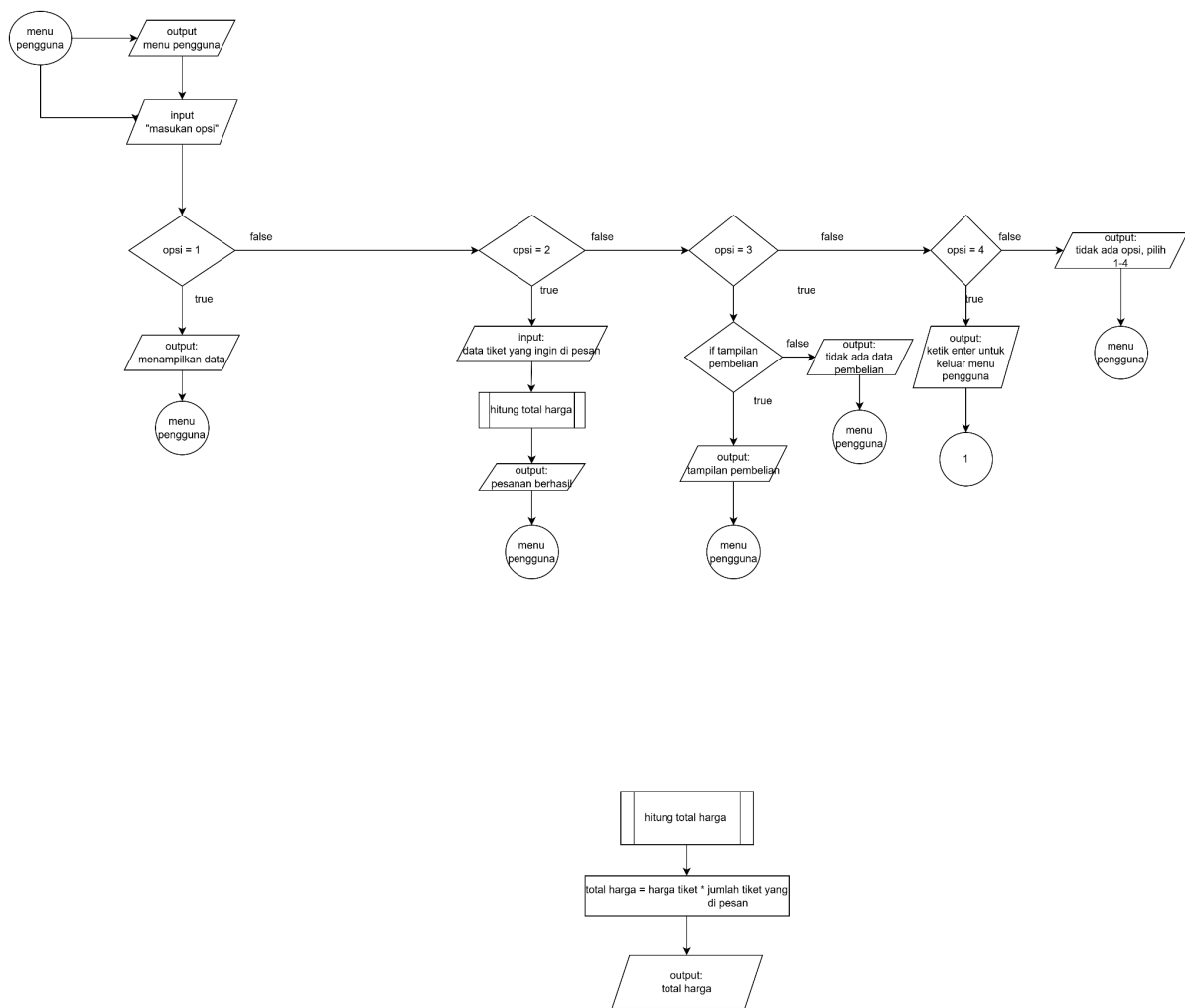
Gambar flowchart 1.1 bagian login/regs/logout

Pada bagian ini program meminta untuk memilih jika memilih login maka program akan membawa untuk masuk ke dalam menu admin jika berhasil, dan login ini dia sudah memiliki ketentuan username dan password, sementara jika memilih daftar/regs itu harus memasukan username dan password baru kemudian program akan mengarahkan ke menu pengguna, tetapi username dan password ini tidak boleh sama dengan data loginnya admin, dan untuk bagian keluar/logout ini jika pilihannya keluar maka program akan berhenti dan jika salah memasukan inputan tidak sesuai dengan inputan login/daftar/keluar ini maka program akan meminta inputan ulang.



Gambar flowchart 1.2 menu admin

Pada menu admin ini, admin memiliki akses untuk melakukan fitur CRUD pada data penjualan tiket konsernya dan juga akan setiap opsi memiliki fungsinya masing-masing sesuai dengan CRUD, opsi 1 admin bisa menambahkan data tiket, opsi 2 untuk melihat tiket apa saja yang tersedia, opsi 3 untuk mengubah data tiket, dan opsi 4 itu untuk menghapus data tiket, dan opsi 5 itu untuk keluar dari menu admin dan kembali lagi ke bagian login/daftar/keluar.



Gambar flowchart 1.3 menu pengguna

Pada menu pengguna ini pengguna tidak memiliki akses penuh seperti admin, pengguna hanya memiliki beberapa fitur seperti opsi 1 untuk melihat data tiket yang tersedia yang telah di perbarui oleh admin, opsi 2 untuk menambahkan data pembelian tiket yang ingin di pesan oleh pengguna dan opsi 3 untuk menampilkan data pembelian dari pengguna dan juga menjumlahkan total dari tiket yang di pesan dan yang terakhir opsi 4 untuk keluar dari menu pengguna.

2. Deskripsi Singkat Program

Tujuan dari program ini adalah untuk mengelola data penjualan tiket konser dengan menggunakan CRUD (khusus admin) sementara untuk pengguna itu hanya menampilkan, menambahkan pembelian tiket dan juga membantu untuk menghitung total harga tiket yang di pesan.

Fungsi :

- Membantu admin untuk mengelola data penjualan dengan menggunakan CRUD.
- Membantu pengguna untuk menambahkan data pembelian dan juga menghitung total harga tiket yang di beli.
- Membedakan menu admin dan juga menu pengguna agar memudahkan yang mengakses program.

3. Source Code

3.1 source code prosedur

```
judul_program = "PROGRAM PENJUALAN TIKET KONSER"
def sapaan_awal (judul_program):
    sapaan = " ***** SELAMAT DATANG
***** "
    print(sapaan)
    sapa = "                di dalam                "
    print(sapa)
    print(f"~~~~~{judul_program}~~~~~")
```

```
def menuadmin():
    print("~~~~~MENU ADMIN~~~~~")
    print("OPSI 1 = MENAMBAHKAN DATA PENJUALAN TIKET")
    print("OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET")
    print("OPSI 3 = MENGUBAH DATA PENJUALAN TIKET")
    print("OPSI 4 = MENGHAPUS DATA PENJUALAN TIKET")
    print("OPSI 5 = KELUAR")
```

```
def menupengguna():
    print("~~~~~MENU PENGGUNA~~~~~")
    print("OPSI 1 = UNTUK MELIHAT DATA TIKET KONSER")
    print("OPSI 2 = MENAMBAHKAN DATA PEMBELIAN TIKET")
    print("OPSI 3 = MENAMPILKAN PEMBELIAN TIKET")
    print("OPSI 4 = KELUAR")
```

3.2 source code fungsi tanpa parameter

```
def jumlah_admin_terdaftar():
    jumlah = len(admin_terdaftar)
    return jumlah
```

```
def data_tiket_tersedia():
    table_konser = PrettyTable()
    table_konser.field_names = ["no", "nama konser", "kategori tiket",
    "tempat", "harga tiket", "stock"]
    data_tiket_baru = {}
    for nomor_tiket, tiket in data_tiket_konser.items():
        table_konser.add_row([
            nomor_tiket,
            tiket["nama konser"],
            tiket["kategori tiket"],
            tiket["tempat"],
```

```

        tiket["harga tiket"],
        tiket["stock"]
    ])
    return table_konser

```

```

def tampilan_data_pembelian_tiket():
    table_tampilan = PrettyTable()
    table_tampilan.field_names = ["no", "nama konser", "jumlah tiket",
    "total harga"]
    for nomor_pembelian, beli in data_pembelian_tiket.items():
        total_harga_angka = beli['total harga']
        table_tampilan.add_row([
            nomor_pembelian,
            beli["nama konser"],
            beli["jumlah tiket"],
            total_harga_angka
        ])
    return table_tampilan

```

3.3 source code fungsi dengan parameter

```

def hitung_total_harga (harga_tiket, jumlah_tiket_yang_ingin_di_pesan):
    harga = harga_tiket
    jumlah = jumlah_tiket_yang_ingin_di_pesan
    total = harga_tiket*jumlah_tiket_yang_ingin_di_pesan
    return total

def cek_stock_tiket (sisa_stock_tiket, jumlah_tiket_yang_ingin_di_pesan):
    if sisa_stock_tiket >= jumlah_tiket_yang_ingin_di_pesan:
        return True
    else:
        return False

```

3.4 source code login

```

pilihan = input("login/daftar/keluar : ")
if pilihan == "login":
    login = admin_terdaftar
    if login == admin_terdaftar:
        login_berhasil = True
        nama = input("masukan username: ")
        pw = input ("masukan password: ")
        for admin_key, admin_data in admin_terdaftar.items():
            if admin_data ["nama"] == nama and admin_data ["pw"]
            == pw:
                print("anda adalah admin")

```

```

                                print(140* "=")
                                login_berhasil = True
                                break
                    else:
                        print("login gagal, username/password yang anda
masukan salah")
                                exit ()

```

3.5 source code daftar

```

elif pilihan == "daftar":
    nama_baru = input("masukan username baru : ")
    pw_baru = input ("masukan password baru : ")
    data_admin = False
    for admin_data_baru in admin_terdaftar.values():
        if admin_data_baru ["nama"] == nama_baru:
            data_admin = True
            print ("username/password sudah ada")
            exit ()
    else:
        keys_baru = 5
        keys_baru = len(admin_terdaftar.keys()) + 1
        admin_terdaftar [keys_baru] = {"nama" : nama_baru, "pw" :
pw_baru}
        print("anda adalah pengguna")
        print(140* "=")

```

3.6 source code keluar

```

elif pilihan == "keluar":
    print("anda keluar dari program")
    print("bye bye, jumpa lagi")
    exit()
else:
    print("login/daftar/keluar, gagal")
    print("coba lagi")
    continue

```

3.7 source code menu admin

```

while True:
    menuadmin()
    try:

```



```

        opsi = int(input("masukan opsi yang ingin di pilih:
"))
    except ValueError:
        print ("opsi yang ada masukan harus berupa angka
tidak boleh string, coba lagi")
    if opsi == 1:
        print("OPSI 1 = MENAMBAHKAN DATA PENJUALAN TIKET")
        print(" di opsi ini anda diminta untuk menambahkan
data penjualan terbaru")
        nama_konser_baru = input("masukan nama konser yang
baru : ")
        kategori_tiket_baru = input("masukan kategori tiket
baru : ")
        tempat_konser_baru = input("masukan tempat konser
baru : ")
        harga_tiket_baru = 0
        stock_tiket_baru = 0
        input_valid = True
        try:
            harga_tiket_baru = int(input("masukan harga
tiket : "))
            stock_tiket_baru = int(input("masukan stock
tiket terbaru: "))
        except ValueError:
            print("harga dan stock yang anda masukan itu
harus berupa angka bukan string, coba lagi")
            input_valid = False
        if input_valid:
            keys_baru = len(data_tiket_konser) + 1
            data_tiket_konser[keys_baru] = {"nama konser":
nama_konser_baru,
            "kategori tiket" : kategori_tiket_baru,
            "tempat" : tempat_konser_baru,
            "harga tiket" : harga_tiket_baru,
            "stock" : stock_tiket_baru
            }
            print("data tiket konser berhasil di tambahkan")
        elif opsi == 2:
            print("OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET")
            table_baru = data_tiket_tersedia()
            print(table_baru)
        elif opsi == 3:
            print("OPSI 3 = MENGUBAH DATA PENJUALAN TIKET")
            table_baru = data_tiket_tersedia()
            print(table_baru)
            nomor_tiket = int(input("masukan nomor tiket yang
ingin di ubah: "))
            while True:

```

```

        if nomor_tiket in data_tiket_konser:
            print("pilih data tiket yang ingin di ubah: ")

            print("[1] = nama konser")
            print("[2] = kategori tiket")
            print("[3] = tempat")
            print("[4] = harga tiket")
            print("[5] = stock")
            try:
                pilihan = int(input("masukan pilihan
data yang ingin di ubah: "))
            except ValueError:
                ("pilihan yang anda masukan harus berupa
angka bukan string, coba lagi ya")
            data_tiket_yang_di_ubah = ""
            if pilihan == 1:
                data_tiket_yang_di_ubah = "nama konser"
            elif pilihan == 2:
                data_tiket_yang_di_ubah = "kategori tiket"
            elif pilihan == 3:
                data_tiket_yang_di_ubah = "tempat"
            elif pilihan == 4:
                data_tiket_yang_di_ubah = "harga tiket"
            elif pilihan == 5:
                data_tiket_yang_di_ubah = "stock"
            if data_tiket_yang_di_ubah != "":
                data_tiket_baru = input(f"masukan data tiket
baru sesuai sama pilihan yang di pilih '{data_tiket_yang_di_ubah}': ")
                if data_tiket_yang_di_ubah == "harga tiket"
or data_tiket_yang_di_ubah == "stock":
                    try:
                        data_tiket_baru =
int(data_tiket_baru)
                    except ValueError:
                        print("harga dan stock yang di
masukan harus berupa angka tidak boleh string, coba lagi")

                data_tiket_konser[nomor_tiket][data_tiket_yang_di_ubah] = data_tiket_baru
                pilihan = input("apakah masih ingin mengubah
data? ya/tidak: ")

                if pilihan == "ya":
                    print("silahkan masukan data di pilihan
yang ingin di ubah lagi sesuai dengan nomor tiket yang di pilih di awal")
                elif pilihan == "tidak":
                    print("selsai mengubah data tiket
konser")

                    break
            else:

```

```

        print("nomor tiket tidak ada, silahkan coba
lagi")

    elif opsi == 4:
        print("OPSI 4 = MENGHAPUS DATA PENJUALAN TIKET")
        table_baru = data_tiket_tersedia()
        try:
            nomor_tiket = int(input("masukan nomor tiket
yang akan di hapus: "))
        except ValueError:
            print("nomor tiket yang anda masukan harus
berupa angka yaa bukan string, coba lagi")
            if nomor_tiket in data_tiket_konser:
                del data_tiket_konser[nomor_tiket]
                print ("data tiket berhasil di hapus")
            else:
                print("nomor tiket tidak di temukan dalam daftar
tiket")

    elif opsi == 5:
        print("berhasil keluar dari menu admin")
        input("ketik enter untuk kembali ke halaman awal: ")
        break
    else:
        print("opsi tidak tersedia, silahkan pilih opsi
1-5")

        continue

```

3.8 source code menu pengguna

```

elif pilihan == "daftar":
    nama_baru = input("masukan username baru : ")
    pw_baru = input ("masukan password baru : ")
    data_admin = False
    for admin_data_baru in admin_terdaftar.values():
        if admin_data_baru ["nama"] == nama_baru:
            data_admin = True
            print ("username/password sudah ada")
            exit ()
    else:
        keys_baru = 5
        keys_baru = len(admin_terdaftar.keys()) + 1
        admin_terdaftar [keys_baru] = {"nama" : nama_baru,
"pw" : pw_baru}
        print("anda adalah pengguna")
        print(140*"=")

```

```

def menupengguna():
    print("~~~~~MENU PENGGUNA~~~~~")
    print("OPSI 1 = UNTUK MELIHAT DATA TIKET KONSER")
    print("OPSI 2 = MENAMBAHKAN DATA PEMBELIAN TIKET")
    print("OPSI 3 = MENAMPILKAN PEMBELIAN TIKET")
    print("OPSI 4 = KELUAR")
while True:
    menupengguna()
    try:
        opsi = int(input("masukan opsi yang ingin di
pilih: "))
    except ValueError:
        print("opsi yang anda masukan harus berupa angka
bukan string, coba lagi")
    if opsi == 1:
        print("OPSI 1 = UNTUK MELIHAT DATA TIKET KONSER")
        table_baru = data_tiket_tersedia()
        print(table_baru)
    elif opsi == 2:
        print("OPSI 2 = MENAMBAHKAN DATA PEMBELIAN TIKET")
        try:
            nama_konser_yang_ingin_di_pesan =
input("masukan nama konser yang ingin di pesan: ")
            jumlah_tiket_yang_ingin_di_pesan =
int(input("masukan jumlah tiket yang ingin di pesan: "))
        except ValueError:
            print("ini harus sesuai nama konser itu string
perhatikan juga huruf besar kecilnya, dan jumlah itu angka bukan
string yaa, coba lagi")
            jumlah_tiket = 0
            nama_konser = ""
            input_valid = True
            if input_valid:
                total_harga_pembelian = 0
                pembelian_berhasil = False
                nomor_tiket_terpilih = -1
                for key, data_tiket in
data_tiket_konser.items():
                    if data_tiket ["nama konser"] ==
nama_konser_yang_ingin_di_pesan:
                        nomor_tiket_terpilih = key
                        if

```

```

cek_stock_tiket(data_tiket["stock"],
jumlah_tiket_yang_ingin_di_pesan):

data_tiket_konser[nomor_tiket_terpilih]["stock"]-=
jumlah_tiket_yang_ingin_di_pesan
total_harga_pembelian =
hitung_total_harga(data_tiket["harga tiket"],
jumlah_tiket_yang_ingin_di_pesan)
pembelian_berhasil = True
break
else:
    print("stock tiket tidak cukup")
    if pembelian_berhasil:
        keys_baru = len(data_pembelian_tiket) + 1
        data_pembelian_tiket[keys_baru] = { "nama
konser" : nama_konser_yang_ingin_di_pesan,
                                           "jumlah tiket"
: jumlah_tiket_yang_ingin_di_pesan,
                                           "total harga"
: total_harga_pembelian
        }
        print("data pembelian telah di tambahkan")

    else:
        if nomor_tiket_terpilih == -1:
            print(f"nama konser
'{nama_konser_yang_ingin_di_pesan}' tidak di temukan, periksa
tulisan nama konsernya huruf besar atau kecil")
        elif opsi == 3:
            print("OPSI 3 = MENAMPILKAN PEMBELIAN TIKET")
            table_pembelian = tampilan_data_pembelian_tiket()
            if len(data_pembelian_tiket) > 0:
                print(table_pembelian)
                total_pendapatan_semua = 0
                for beli in data_pembelian_tiket.values():
                    total_pendapatan_semua += beli['total
harga']

            else:
                print("belum ada data pembelian")
        elif opsi == 4:
            print("kamu keluar dari menu pengguna")
            input("tekan enter untuk kembali ke halaman awal")

```

```
        break
    else:
        print("opsi tidak tersedia, silahkan masukan opsi
1-4")
        continue
```

4. Hasil Output

4.1 login

```
h:\chibi live\documents\praktikum apd-b2-23\posttest\post-test-apd-8\post-test-apd-8.py
***** SELAMAT DATANG *****
di dalam
PROGRAM PENJUALAN TIKET KONSER
total admin terdaftar: 4
login/daftar/keluar : login
masukan username: intan
masukan password: 2354
anda adalah admin
=====
MENU ADMIN
OPSI 1 = MENAMBAHKAN DATA PENJUALAN TIKET
OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET
OPSI 3 = MENGUBAH DATA PENJUALAN TIKET
OPSI 4 = MENGHAPUS DATA PENJUALAN TIKET
OPSI 5 = KELUAR
masukan opsi yang ingin di pilih: 1
```

4.2 menu admin opsi 1 (menambahkan data tiket)

```
MENU ADMIN
OPSI 1 = MENAMBAHKAN DATA PENJUALAN TIKET
OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET
OPSI 3 = MENGUBAH DATA PENJUALAN TIKET
OPSI 4 = MENGHAPUS DATA PENJUALAN TIKET
OPSI 5 = KELUAR
masukan opsi yang ingin di pilih: 1
OPSI 1 = MENAMBAHKAN DATA PENJUALAN TIKET
di opsi ini anda diminta untuk menambahkan data penjualan terbaru
masukan nama konser yang baru : H2H
masukan kategori tiket baru : reguler
masukan tempat konser baru : GBK
masukan harga tiket : 900000
masukan stock tiket terbaru: 190
data tiket konser berhasil di tambahkan
```

4.3 menu admin opsi 2 (menampilkan data tiket)

```
MENU ADMIN
OPSI 1 = MENAMBAHKAN DATA PENJUALAN TIKET
OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET
OPSI 3 = MENGUBAH DATA PENJUALAN TIKET
OPSI 4 = MENGHAPUS DATA PENJUALAN TIKET
OPSI 5 = KELUAR
masukan opsi yang ingin di pilih: 2
OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET
+-----+
| no | nama konser | kategori tiket | tempat | harga tiket | stock |
+-----+
| 1 | NCT | reguler | BSD | 1200000 | 17 |
| 2 | AESPA | vip | GBK | 1500000 | 10 |
| 3 | TXT | vvip | JIS | 1900000 | 15 |
| 4 | H2H | reguler | GBK | 900000 | 190 |
+-----+
```

4.4 menu admin opsi 3 (mengubah data tiket)

```

#####MENU ADMIN#####
OPSI 1 = MENAMBAHKAN DATA PENJUALAN TIKET
OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET
OPSI 3 = MENGUBAH DATA PENJUALAN TIKET
OPSI 4 = MENGHAPUS DATA PENJUALAN TIKET
OPSI 5 = KELUAR
masukan opsi yang ingin di pilih: 3
OPSI 3 = MENGUBAH DATA PENJUALAN TIKET
+-----+
| no | nama konser | kategori tiket | tempat | harga tiket | stock |
+-----+
| 1 | NCT | reguler | BSD | 1200000 | 17 |
| 2 | AESPA | vip | GBK | 1500000 | 10 |
| 3 | TXT | vvip | JIS | 1900000 | 15 |
| 4 | H2H | reguler | GBK | 900000 | 190 |
+-----+
masukan nomor tiket yang ingin di ubah: 2
pilih data tiket yang ingin di ubah:
[1] = nama konser
[2] = kategori tiket
[3] = tempat
[4] = harga tiket
[5] = stock
masukan pilihan data yang ingin di ubah: 2
masukan data tiket baru sesuai sama pilihan yang di pilih 'kategori tiket': reguler
apakah masih ingin mengubah data? ya/tidak: ya
silahkan masukan data di pilihan yang ingin di ubah lagi sesuai dengan nomor tiket yang di pilih di awal
pilih data tiket yang ingin di ubah:
[1] = nama konser
[2] = kategori tiket
[3] = tempat
[4] = harga tiket
[5] = stock
masukan pilihan data yang ingin di ubah: 4
masukan data tiket baru sesuai sama pilihan yang di pilih 'harga tiket': 1200000

apakah masih ingin mengubah data? ya/tidak: ya
silahkan masukan data di pilihan yang ingin di ubah lagi sesuai dengan nomor tiket yang di pilih di awal
pilih data tiket yang ingin di ubah:
[1] = nama konser
[2] = kategori tiket
[3] = tempat
[4] = harga tiket
[5] = stock
masukan pilihan data yang ingin di ubah: 127
nomor tiket tidak ada, silahkan coba lagi
pilih data tiket yang ingin di ubah:
[1] = nama konser
[2] = kategori tiket
[3] = tempat
[4] = harga tiket
[5] = stock
masukan pilihan data yang ingin di ubah: 5
masukan data tiket baru sesuai sama pilihan yang di pilih 'stock': 123
apakah masih ingin mengubah data? ya/tidak: tidak
selesai mengubah data tiket konser

+-----+
| no | nama konser | kategori tiket | tempat | harga tiket | stock |
+-----+
| 1 | NCT | reguler | BSD | 1200000 | 17 |
| 2 | AESPA | reguler | GBK | 1200000 | 123 |
| 3 | TXT | vvip | JIS | 1900000 | 15 |
| 4 | H2H | reguler | GBK | 900000 | 190 |
+-----+

```

4.5 menu admin opsi 4 (menghapus data)


```

MENU ADMIN
OPSI 1 = MENAMBAHKAN DATA PENJUALAN TIKET
OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET
OPSI 3 = MENGUBAH DATA PENJUALAN TIKET
OPSI 4 = MENGHAPUS DATA PENJUALAN TIKET
OPSI 5 = KELUAR
masukan opsi yang ingin di pilih: 4
OPSI 4 = MENGHAPUS DATA PENJUALAN TIKET
masukan nomor tiket yang akan di hapus: 1
data tiket berhasil di hapus
MENU ADMIN
OPSI 1 = MENAMBAHKAN DATA PENJUALAN TIKET
OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET
OPSI 3 = MENGUBAH DATA PENJUALAN TIKET
OPSI 4 = MENGHAPUS DATA PENJUALAN TIKET
OPSI 5 = KELUAR
masukan opsi yang ingin di pilih: 2
OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET

```

no	nama konser	kategori tiket	tempat	harga tiket	stock
2	AESPA	reguler	GBK	1200000	123
3	TXT	vvip	JIS	1900000	15
4	H2H	reguler	GBK	900000	190

4.6 menu admin opsi 5 (keluar)

```

MENU ADMIN
OPSI 1 = MENAMBAHKAN DATA PENJUALAN TIKET
OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET
OPSI 3 = MENGUBAH DATA PENJUALAN TIKET
OPSI 4 = MENGHAPUS DATA PENJUALAN TIKET
OPSI 5 = KELUAR
masukan opsi yang ingin di pilih: 5
berhasil keluar dari menu admin
ketik enter untuk kembali ke halaman awal:
login(daftar/keluar) :

```

4.7 daftar

```

login(daftar/keluar) : daftar
masukan username baru : mahasiswa
masukan password baru : 1234
anda adalah pengguna
=====
MENU PENGGUNA
OPSI 1 = UNTUK MELIHAT DATA TIKET KONSER
OPSI 2 = MENAMBAHKAN DATA PEMBELIAN TIKET
OPSI 3 = MENAMPILKAN PEMBELIAN TIKET
OPSI 4 = KELUAR
masukan opsi yang ingin di pilih:

```

4.8 menu pengguna opsi 1 (menampilkan data tiket yang telah di perbarui admin)

```

MENU PENGGUNA
OPSI 1 = UNTUK MELIHAT DATA TIKET KONSER
OPSI 2 = MENAMBAHKAN DATA PEMBELIAN TIKET
OPSI 3 = MENAMPILKAN PEMBELIAN TIKET
OPSI 4 = KELUAR
masukan opsi yang ingin di pilih: 1
OPSI 1 = UNTUK MELIHAT DATA TIKET KONSER

```

no	nama konser	kategori tiket	tempat	harga tiket	stock
2	AESPA	reguler	GBK	1200000	123
3	TXT	vvip	JIS	1900000	15
4	H2H	reguler	GBK	900000	190

4.9 menu pengguna opsi 2 (menambahkan data pembelian)

```

MENU PENGGUNA
OPSI 1 = UNTUK MELIHAT DATA TIKET KONSER
OPSI 2 = MENAMBAHKAN DATA PEMBELIAN TIKET
OPSI 3 = MENAMPILKAN PEMBELIAN TIKET
OPSI 4 = KELUAR
masukan opsi yang ingin di pilih: 2
OPSI 2 = MENAMBAHKAN DATA PEMBELIAN TIKET
masukan nama konser yang ingin di pesan: H2H
masukan jumlah tiket yang ingin di pesan: 4
data pembelian telah di tambahkan

```

4.10 menu pengguna opsi 3 (menampilkan data tiket yang di pesan dan menjumlahkan)

```
~~~~~MENU PENGGUNA~~~~~
OPSI 1 = UNTUK MELIHAT DATA TIKET KONSER
OPSI 2 = MENAMBAHKAN DATA PEMBELIAN TIKET
OPSI 3 = MENAMPILKAN PEMBELIAN TIKET
OPSI 4 = KELUAR
masukan opsi yang ingin di pilih: 3
OPSI 3 = MENAMPILKAN PEMBELIAN TIKET
+---+---+---+---+
| no | nama konser | jumlah tiket | total harga |
+---+---+---+---+
| 1  | H2H         | 4            | 3600000     |
+---+---+---+---+
```

4.11 menu pengguna opsi 4 (keluar)

```
~~~~~MENU PENGGUNA~~~~~
OPSI 1 = UNTUK MELIHAT DATA TIKET KONSER
OPSI 2 = MENAMBAHKAN DATA PEMBELIAN TIKET
OPSI 3 = MENAMPILKAN PEMBELIAN TIKET
OPSI 4 = KELUAR
masukan opsi yang ingin di pilih: 4
kamu keluar dari menu pengguna
tekan enter untuk kembali ke halaman awal
login/daftar/keluar : 
```

4.12 keluar

```
login/daftar/keluar : keluar
anda keluar dari program
bye bye, jumpa lagi
PS C:\Users\indah\OneDrive\Documents\praktikum apd-82-25> 
```

5. Langkah-langkah GIT

5.1 GIT Add

```
PS C:\Users\indah\OneDrive\Documents\praktikum apd-B2-25> git add .
PS C:\Users\indah\OneDrive\Documents\praktikum apd-B2-25> git
```

fungsinya untuk menambahkan file kita ke repository

5.2 GIT Commit

```
PS C:\Users\indah\OneDrive\Documents\praktikum apd-B2-25> git commit -m "upload PT APD 8"
[main d836587] upload PT APD 8
12 files changed, 437 insertions(+)
create mode 100644 kelas/pertemuan 8/pertemuan8.py
create mode 100644 posttest/post-test-apd 8/2509106088_Intan-Indah-Sari-PT-APD-8.drawio.png
create mode 100644 posttest/post-test-apd 8/carakerja.py
create mode 100644 posttest/post-test-apd 8/data.py
create mode 100644 posttest/post-test-apd 8/main.py
create mode 100644 posttest/post-test-apd 8/modular/carakerja.py
create mode 100644 posttest/post-test-apd 8/modular/data.py
create mode 100644 posttest/post-test-apd 8/modular/main.py
create mode 100644 posttest/post-test-apd 8/modular/pengguna-admin-keluar.py
create mode 100644 posttest/post-test-apd 8/pengguna-admin-keluar.py
create mode 100644 posttest/post-test-apd 8/post-test-apd-8.py
PS C:\Users\indah\OneDrive\Documents\praktikum apd-B2-25>
```

Fungsinya untuk menulis pesan di repository

5.3 GIT Push

```
PS C:\Users\indah\OneDrive\Documents\praktikum apd-B2-25> git push -u origin main
Enumerating objects: 18, done.
Counting objects: 100% (18/18), done.
Delta compression using up to 2 threads
Compressing objects: 100% (11/11), done.
Writing objects: 100% (13/13), 2.44 MiB | 310.00 KiB/s, done.
Total 13 (delta 4), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Resolving deltas: 100% (4/4), completed with 4 local objects.
To https://github.com/intanindahsari/praktikum_APD.git
 05577ef..d836587 main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\indah\OneDrive\Documents\praktikum apd-B2-25>
```

Fungsinya git push adalah untuk mengupload file kita ke github