

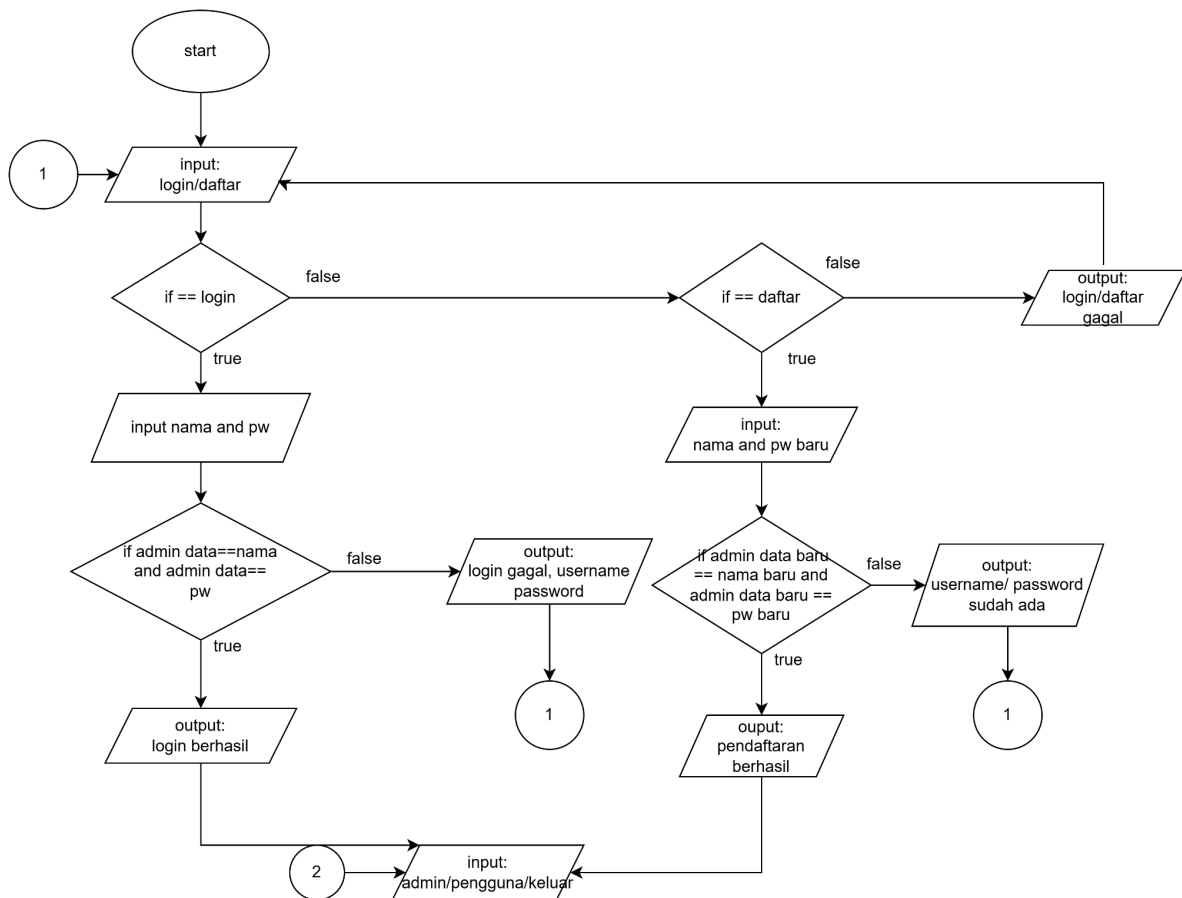
**LAPORAN PRAKTIKUM**  
**POSTTEST (7)**  
**ALGORITMA PEMROGRAMAN DASAR**



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

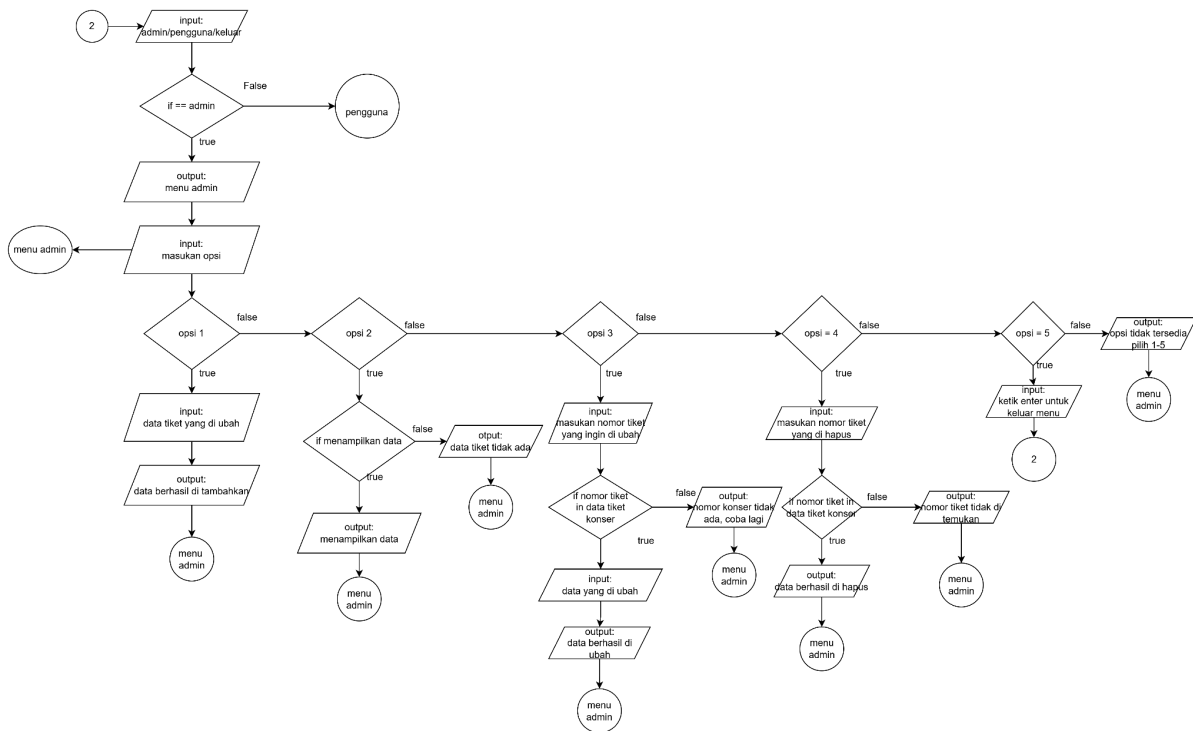
**PROGRAM STUDI INFORMATIKA**  
**UNIVERSITAS MULAWARMAN**  
**SAMARINDA**  
**2025**

## 1. Flowchart



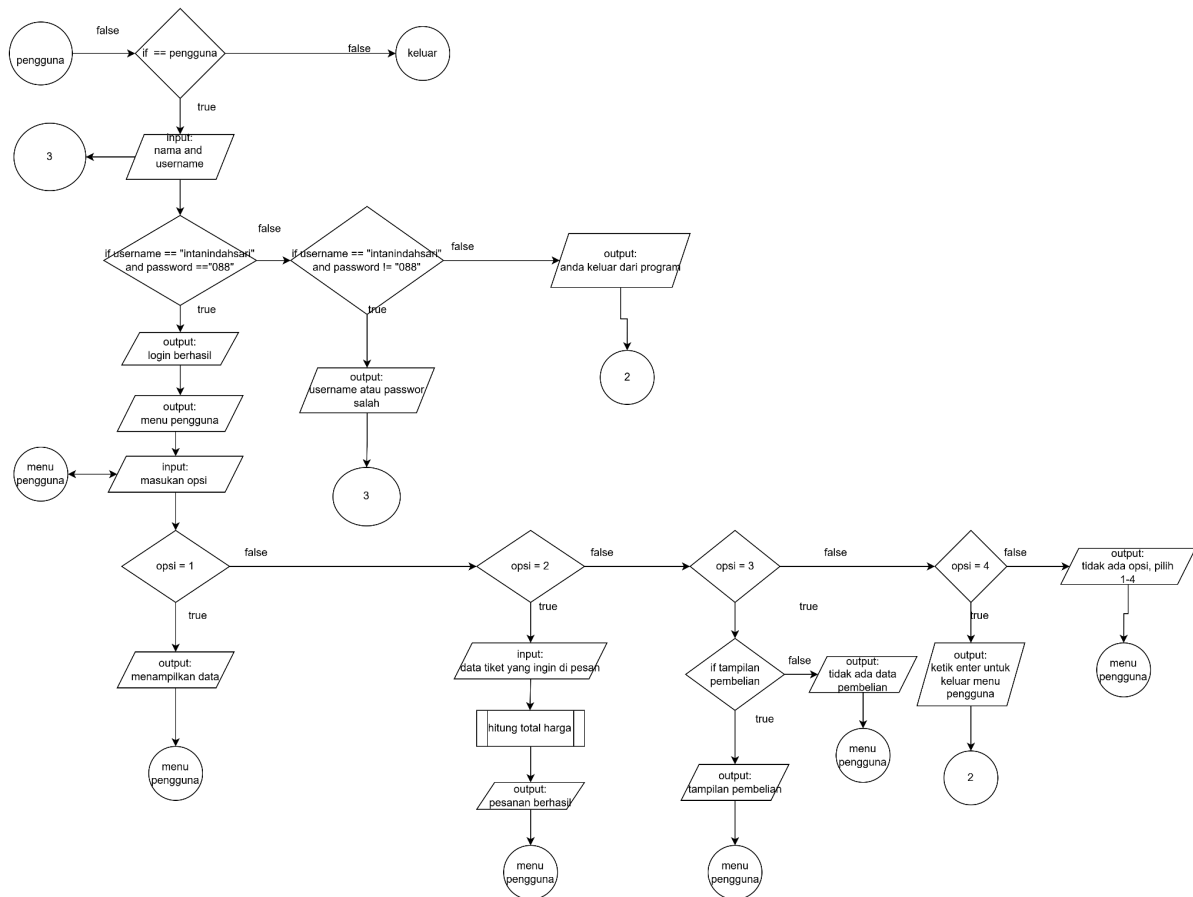
Gambar 1.1 flowchart register

Pada bagian ini di minta untuk memilih login/daftar jika login maka dia harus memasukan nama dan pw yang sudah terdaftar, dan jika memilih daftar maka dia harus memasukan nama atau pw baru yang belum ada dalam data admin terdaftar agar programnya bisa berjalan ke langkah selanjutnya.



Gambar 1.2 flowchart menu admin

Bagian ini adalah program setelah login atau daftar jika berhasil dapat memilih sebagai apa admin/pengguna lalu masuk jika dia memilih admin maka menu tampilannya akan menampilkan bagian admin dalam menu admin ini fiturnya adalah dengan menggunakan CRUD untuk mengolah data penjualan tiket konsernya. Masing- masing opsi ini dia mencakup bagian dari CRUD dan jika memilih opsi terakhir maka dia akan kemali ke halaman input admin/pengguna.



Gambar 1.3 flowchart menu pengguna

Bagian ini adalah bagian menu pengguna jika di awal yang di masukan status sebagai pengguna atau setelah memilih opsi terakhir dari menu admin maka, akan muncul menu pengguna dan di menu pengguna ini sebelum masuk ke dalam menunya program meminta input username dan password yang telah di tentukan untuk melanjutkan programnya ke menu pengguna dan di menu ini juga ada 4 opsi, menu pengguna ini berbeda fiturnya dengan menu admin di menu pengguna hanya menampilkan tiket dan juga menambah data pembelian serta membantu menghitung total harga dari tiket yang di pesan.



Gambar 1.4 flowchart keluar

Ini adalah bagian terakhir yaitu keluar dimana jika memilih keluar maka program otomatis berhenti menjalankan programnya.

## **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 Posedur

source code

```
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 Fungsi tanpa parameter

source code

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

def data_tiket_tersedia():
    data_tiket_baru = {}
    for nomor_tiket, tiket in data_tiket_konser.items():
        tabel = {}
```

```

    tabel["nomor tiket"] = str(nomor_tiket)
    tabel["nama konser"] = tiket["nama konser"]
    tabel["kategori tiket"] = tiket["kategori tiket"]
    tabel["tempat"] = tiket["tempat"]
    tabel["harga tiket"] = tiket["harga tiket"]
    tabel["stock"] = tiket["stock"]
    data_tiket_baru[nomor_tiket] = tabel
    return data_tiket_baru

```

### 3.3 Fungsi dengan parameter

source code

```

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 Fitur register

source code

```

sapaan_awal(judul_program)
print(f" total admin terdaftar: {jumlah_admin_terdaftar()}")
pilihan = input("login/daftar : ")
if pilihan == "login":
    login_berhasil = False
    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("login berhasil")
            print(140*"=")
            login_berhasil = True
            break
    else:
        print("login gagal, username/password yang anda masukan salah")

```



```

        exit ()
elif pilihan == "daftar":
    nama_baru = input("masukan nama baru : ")
    pw_baru = input ("masukan pw 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("pendaftaran berhasil")

```

### 3.5 Fitur multiuser

source code

```

while True:
    status = input("admin/pengguna/keluar : ")
    if status == "admin":
        login_berhasil = True
        if login_berhasil == True:
            elif status == "pengguna":
                print(" mohon maaf untuk status pengguna anda harus")
                print("masukan username = nama dan password = nim")
                print("karena ini adalah ketentuan programnya")
                username = "intanindahsari"
                password = "088"
                try:
                    username = input("masukan username: ")
                    password = input("masukan password: ")
                except ValueError:
                    print("username dan password yang di masukan itu harus sesuai
yaa, coba lagi")
                    if username == "intanindahsari" and password == "088":
                        print("login berhasil")
            elif status == "keluar":
                print("anda telah keluar dari PROGRAM PENJUALAN TIKET KONSER")
                print("bye bye, sampai jumpa lagi")

```

```
exit()
```

### 3.6 Menu admin

source code

```
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:
```

### 3.7 Menu admin opsi 1,2,3,4,5

source code

```
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 lagii")
        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}
```

```

: 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")
    menampilkan_data = data_tiket_tersedia()
    if menampilkan_data:
        print(tabulate(menampilkan_data.values(),
headers="keys", tablefmt='grid'))
    else:
        print("data tiket tidak ada")
elif opsi == 3:
    print("OPSI 3 = MENGUBAH DATA PENJUALAN TIKET")
    menampilkan_data = data_tiket_tersedia()
    print(tabulate(menampilkan_data.values(),
headers="keys", tablefmt='grid'))
    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
            try:
                pilihan = input("apakah masih ingin mengubah
data? ya/tidak: ")

            except ValueError:
                print("pilihan yang anda masukan itu harus
ini (ya/tidak) tidak boleh yang lain dan juga bukan angka ya, coba lagi")
                print("data berhasil di ubah")
                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")
        menampilkan_data = 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")

```

### 3.8 Menu pengguna

source code

```

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")

```

### 3.9 Menu pengguna opsi 1,2,3,4

source code

```

if opsi == 1:
    print("OPSI 1 = UNTUK MELIHAT DATA TIKET KONSER")
    menampilkan_data = data_tiket_tersedia()
    if menampilkan_data:
        print(tabulate(menampilkan_data.values(),
headers="keys", tablefmt='grid'))
    else:
        print("data tiket tidak ada")
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")
                        tampilan_pembelian = {}
                        for nomor_pembelian, beli in
data_pembelian_tiket.items():
                            nama_konser_yang_disimpan = beli["nama konser"]
                            jumlah_tiket_konser_yang_disimpan = beli["jumlah
tiket"]

                            data_pembelian_yang_disimpan = beli["total harga"]
                            tampilan_pembelian[nomor_pembelian] = { "nomor

```

```

tiket" : str(nomor_pembelian),
                                                    "nama konser" :
nama_konser_yang_disimpan,
                                                    "jumlah tiket" :
jumlah_tiket_konser_yang_disimpan,
                                                    "total harga" :
f"Rp{data_pembelian_yang_disimpan}"
    }
    if len (tampilan_pembelian) > 0:
        print(tabulate(tampilan_pembelian.values(),
headers="keys", tablefmt='grid'))
    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 pilih 1-4")

```

### 3.10 Variabel Global

source code

```

admin_terdaftar = { 1: {"nama" : "intan",
                        "pw" : "2354"},
                    2 : {"nama" : "indah",
                        "pw" : "kuning"},
                    3 : {"nama" : "sari",
                        "pw" : "yellow"},
                    4 : {"nama" : "lily",
                        "pw" : "sedapmalam"}
}
data_pembelian_tiket = {}
data_tiket_konser = {
    1 : {"nama konser": "NCT",
        "kategori tiket" : "reguler",
        "tempat" : "BSD",
        "harga tiket" : 1200000,
        "stock" : 17
    },
    2 : { "nama konser": "AESPA",
        "kategori tiket" : "vip",
        "tempat" : "GBK",
        "harga tiket" : 1500000,
        "stock" : 10
    }
}

```

```
    },  
    3 : { "nama konser": "TXT",  
          "kategori tiket" : "vvip",  
          "tempat" : "JIS",  
          "harga tiket" : 1900000,  
          "stock" : 15  
        }  
  }  
}
```



## 4. Hasil Output

### 4.1 register (login)

```
***** SELAMAT DATANG *****
di dalam
PROGRAM PENJUALAN TIKET KONSER
total admin terdaftar: 4
login/daftar : login
masukan username: lily
masukan password: sedapmalam
login berhasil
```

### 4.2 register (daftar)

```
***** SELAMAT DATANG *****
di dalam
PROGRAM PENJUALAN TIKET KONSER
total admin terdaftar: 4
login/daftar : daftar
masukan nama baru : bumi
masukan pw baru : 1234
pendaftaran berhasil
```

### 4.3 status sebagai admin

```
admin/pengguna/keluar : 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:
```

### 4.4 menu admin (opsi 1 = menambahkan data)

```
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 : reg
masukan tempat konser baru : BSD
masukan harga tiket : 1100000
masukan stock tiket terbaru: 102
data tiket konser berhasil di tambahkan
```

### 4.5 menu admin (opsi 2 = menampilkan 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: 2
OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET
```

nomor tiket	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	reg	BSD	1100000	102

#### 4.6 menu admin (opsi 3 = mengubah data)

```
masukan opsi yang ingin di pilih: 3
OPSI 3 = MENGUBAH DATA PENJUALAN TIKET

+-----+-----+-----+-----+-----+-----+
| nomor tiket | 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 | reg | BSD | 1100000 | 102 |
+-----+-----+-----+-----+-----+-----+

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
? ya/tidak: ya
data berhasil di ubah
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: 5
masukan data tiket baru sesuai sama pilihan yang di pilih 'stock': 25
apakah masih ingin mengubah data? ya/tidak: tidak
data berhasil di ubah
selsai mengubah data tiket konser
```

#### tampilan hasil ubah data

```
OPSI 2 = MENAMPILKAN DATA PENJUALAN TIKET

+-----+-----+-----+-----+-----+-----+
| nomor tiket | nama konser | kategori tiket | tempat | harga tiket | stock |
+-----+-----+-----+-----+-----+-----+
| 1 | NCT | reguler | BSD | 1200000 | 17 |
+-----+-----+-----+-----+-----+-----+
| 2 | AESPA | reg | GBK | 900000 | 25 |
+-----+-----+-----+-----+-----+-----+
| 3 | TXT | vvip | JIS | 1900000 | 15 |
+-----+-----+-----+-----+-----+-----+
| 4 | H2H | reg | BSD | 1100000 | 102 |
+-----+-----+-----+-----+-----+-----+
```

#### 4.7 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

+-----+-----+-----+-----+-----+-----+
| nomor tiket | nama konser | kategori tiket | tempat | harga tiket | stock |
+-----+-----+-----+-----+-----+-----+
| 2 | AESPA | reg | GBK | 900000 | 25 |
+-----+-----+-----+-----+-----+-----+
| 3 | TXT | vvip | JIS | 1900000 | 15 |
+-----+-----+-----+-----+-----+-----+
| 4 | H2H | reg | BSD | 1100000 | 102 |
+-----+-----+-----+-----+-----+-----+
```

#### 4.8 menu admin (opsi 5 = keluar menu 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: 5
berhasil keluar dari menu admin
ketik enter untuk kembali ke halaman awal:
admin/pengguna/keluar : █
```

#### 4.9 status sebagai pengguna (username dan passwordnya sudah di tentukan)

```
admin/pengguna/keluar : pengguna
mohon maaf untuk status pengguna anda harus
masukan username = nama dan password = nim
karena ini adalah ketentuan programnya
masukan username: intanindahsari
masukan password: 088
login berhasil
~~~~~MENU PENGGUNA~~~~~
OPSI 1 = UNTUK MELIHAT DATA TIKET KONSER
OPSI 2 = MENAMBAHKAN DATA PEMBELIAN TIKET
OPSI 3 = MENAMPILKAN PEMBELIAN TIKET
OPSI 4 = KELUAR
```

#### 4.10 menu pengguna (opsi 1 = melihat data tiket yang tersedia)

```
masukan opsi yang ingin di pilih: 1
OPSI 1 = UNTUK MELIHAT DATA TIKET KONSER
+-----+-----+-----+-----+-----+-----+
| nomor tiket | nama konser | kategori tiket | tempat | harga tiket | stock |
+-----+-----+-----+-----+-----+-----+
|          2 | AESPA      | reg            | GBK    | 900000      | 25    |
+-----+-----+-----+-----+-----+-----+
|          3 | TXT        | vvip           | JIS    | 1900000     | 15    |
+-----+-----+-----+-----+-----+-----+
|          4 | H2H        | reg            | BSD    | 1100000     | 102   |
+-----+-----+-----+-----+-----+-----+
```

#### 4.11 menu pengguna (opsi 2 = menampilkan pembelian tiket)

```
~~~~~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: 5
data pembelian telah di tambahkan
```

#### 4.12 menu admin (opsi 3 = menampilkan jumlah tiket yang di beli)

```
~~~~~MENU ADMIN~~~~~
masukan opsi yang ingin di pilih: 3
OPSI 3 = MENAMPILKAN PEMBELIAN TIKET
+-----+-----+-----+-----+
| nomor tiket | nama konser | jumlah tiket | total harga |
+-----+-----+-----+-----+
|          1 | H2H        | 5            | Rp5500000   |
+-----+-----+-----+-----+
```

#### 4.13 menu pengguna (opsi 4 = keluar dari menu 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
kamu keluar dari menu pengguna
tekan enter untuk kembali ke halaman awal
admin/pengguna/keluar : █
```

#### 4.14 status keluar program

```
admin/pengguna/keluar : keluar
anda telah keluar dari PROGRAM PENJUALAN TIKET KONSER
bye bye, sampai jumpa lagi
PS C:\Users\indah\OneDrive\Documents\praktikum apd-B2-25> █
```

## 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>
```

fungsi git add adalah untuk menambah semua file baru atau yang di ubah di repository

### 5.2 GIT Commit

```
PS C:\Users\indah\OneDrive\Documents\praktikum apd-B2-25> git commit -m "upload pt apd 7"  
[main d2149d1] upload pt apd 7  
3 files changed, 456 insertions(+)  
rename kelas/{petemuan 6 => pertemuan 6}/pertemuan6.py (100%)  
create mode 100644 kelas/pertemuan 7/pertemuan7.py  
create mode 100644 posttest/post test apd 7/post-test-apd-7.py  
PS C:\Users\indah\OneDrive\Documents\praktikum apd-B2-25>
```

fungsi git commit adalah untuk menampilkan pesan

### 5.3 GIT Push

```
PS C:\Users\indah\OneDrive\Documents\praktikum apd-B2-25> git push -f origin main  
Enumerating objects: 49, done.  
Counting objects: 100% (49/49), done.  
Delta compression using up to 2 threads  
Compressing objects: 100% (39/39), done.  
Writing objects: 100% (49/49), 3.88 MiB | 487.00 KiB/s, done.  
Total 49 (delta 10), reused 0 (delta 0), pack-reused 0 (from 0)  
remote: Resolving deltas: 100% (10/10), done.  
To https://github.com/intanindahsari/praktikum_APD.git  
+ 63428e9...d2149d1 main -> main (forced update)  
PS C:\Users\indah\OneDrive\Documents\praktikum apd-B2-25>
```

fungsi git push adalah untuk melakukan upload file kita ke github