

LAPORAN PRAKTIKUM POSTTEST 5
ALGORITMA DAN PEMROGRAMAN DASAR



Muhammad Riva Fachrodhiya

2409106053

PROGRAM STUDI INFORMATIKA
UNIVERSITAS MULAWARMAN
SAMARINDA

2024

LATAR BELAKANG

LIST

List adalah salah satu tipe data kolektif yang ada di Python, mirip dengan array di bahasa pemrograman lain namun lebih fleksibel.

TUPLE

Tuple pada python adalah struktur data yang digunakan untuk menyimpan sekumpulan data. Tuple bersifat immutable, artinya isi tuple tidak bisa kita ubah dan hapus. Namun, dapat kita isi dengan berbagai macam nilai objek. Tuple adalah salah satu struktur data di Python yang mampu menyimpan sekumpulan nilai dalam satu variabel.

SOLUSI

INSTRUKSI:

Buatlah program dengan ketentuan berikut :

Bisa melakukan operasi Create, Read, Update, dan Delete (CRUD) menggunakan List atau Tuple

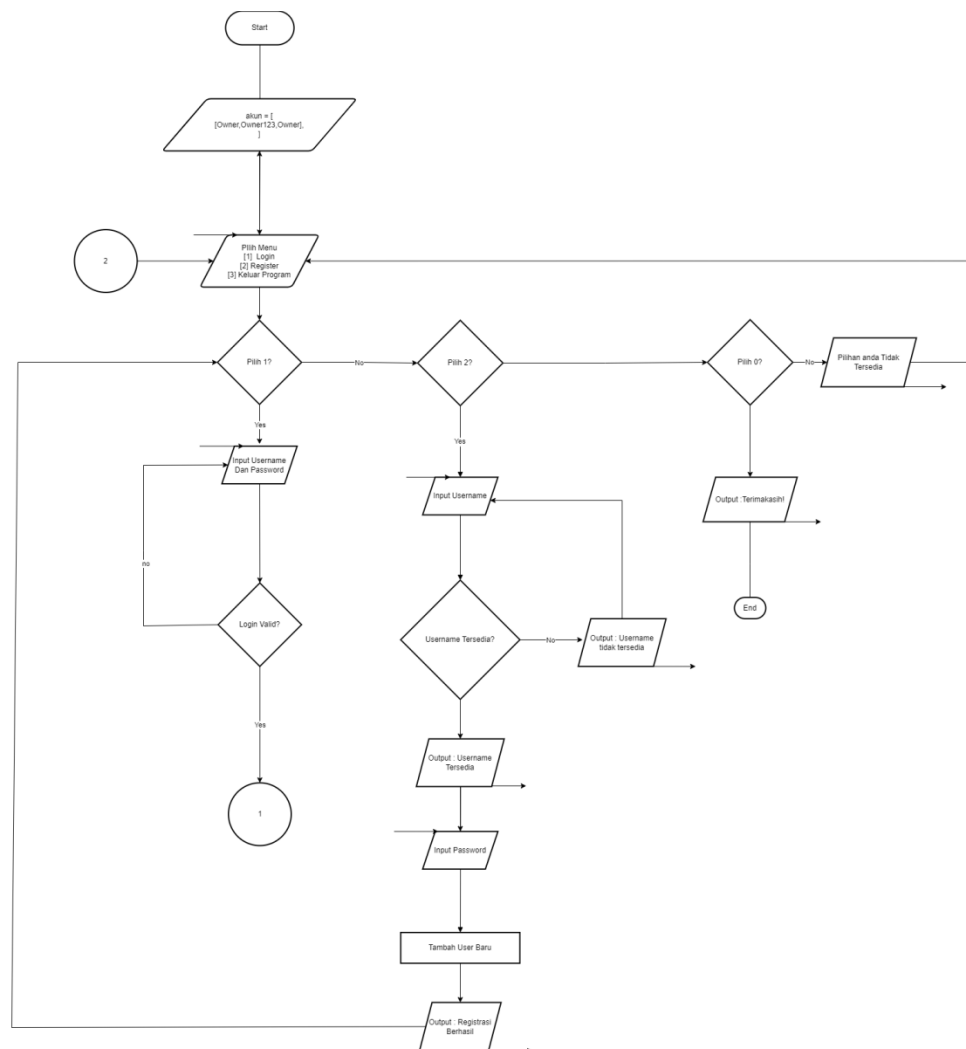
Poin (+):

- Multiuser (admin dan pengguna biasa)
- Error Handling
- Dapat Melakukan Register
- Menggunakan Nested List

Buat juga flowchart dari program yang kalian buat

Saya diberi tugas untuk posttest ke 5 dengan instruksi diatas. Sebelumnya, saya mengambil judul “Manajemen Keuntungan dari Suatu Perusahaan ATK”. Berikut solusi saya mengenai tugas posttest ke 5:

FLOWCHART



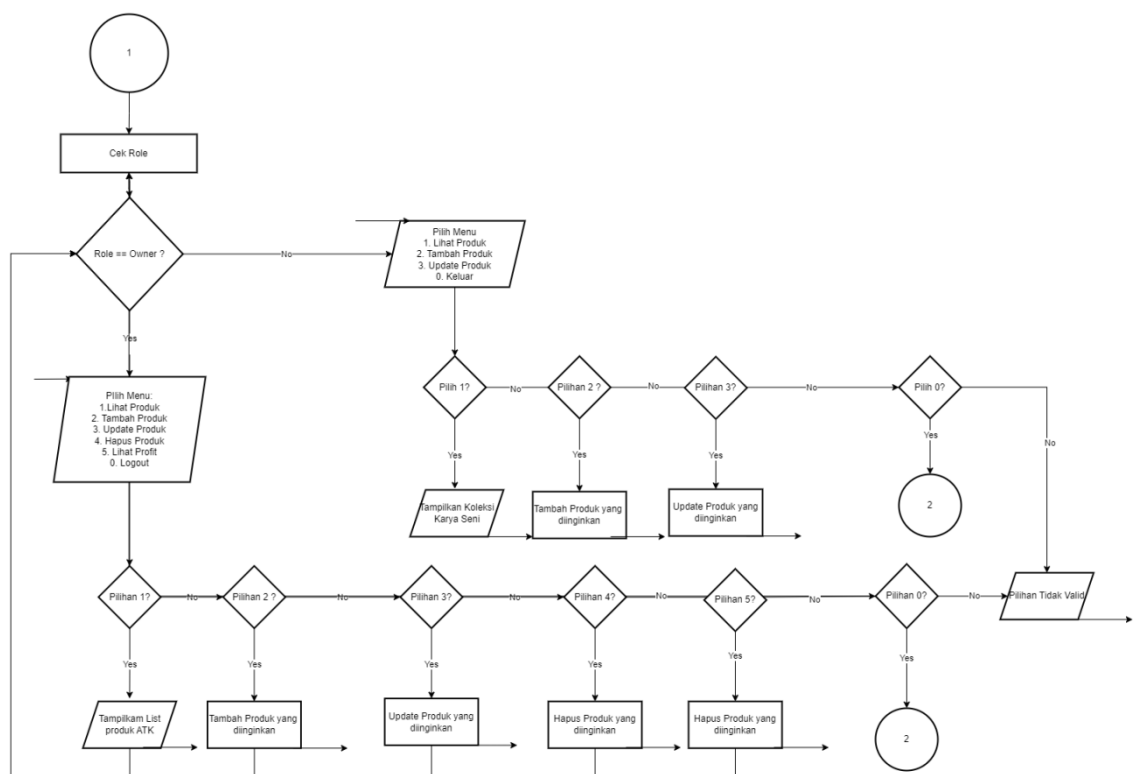
Saya akan menjelaskan flowchart saya. Di gambar atas ini merupakan flowchart bagian Autentikasi Data. Pertama, pengguna akan disuguhkan menu login, register, atau keluar program. Sebelumnya, di flowchart ini sudah terdaftar 1 akun, yaitu akun dari Owner program ini.

Jika pengguna memilih menu login dan memasukkan username beserta password sesuai dengan akun Owner, maka pengguna akan login sebagai Owner.

Jika pengguna memilih menu register, maka pengguna akan disuruh memasukkan username baru, namun jika username tersebut tidak tersedia (Contoh : Owner), maka pengguna akan disuruh memasukkan username

baru. Sesudah memasukkan username baru, pengguna akan disuruh memasukkan password baru. Saat proses registrasi ini sudah selesai, pengguna akan dilempar ke menu awal. Pada proses ini, pengguna sudah bisa login menggunakan akun hasil registrasi tersebut.

Jika pengguna memilih menu keluar dari program, maka pengguna akan keluar dari program tersebut.



Masuk di menu utama, pertama-tama pengguna akan di cek role nya sebagai Owner atau bukan. Saya akan menjelaskan menu utama di bagian Owner terlebih dahulu.

Pertama, pengguna akan di cek apakah role akun yang digunakan pengguna adalah Owner atau bukan, jika iya maka pengguna langsung disuguhkan beberapa menu dari program ini. Beberapa menu tersebut ialah:

1. Lihat Produk,
2. Tambah Produk,
3. Update Produk,

4. Hapus Produk,
5. Lihat Profit, dan
0. Logout.

Jika pengguna memilih menu 1, maka pengguna akan ditampilkan list produk ATK.

Jika pengguna memilih menu 2, maka pengguna dapat menambahkan produk baru dalam list produk ATK tersebut.

Jika pengguna memilih menu 3, maka pengguna dapat mengupdate atau memperbarui produk dalam list produk ATK tersebut.

Jika pengguna memilih menu 4, maka pengguna dapat menghapus produk dalam list produk ATK tersebut.

Jika pengguna memilih menu 5, maka pengguna dapat melihat profit dari perusahaan ATK tersebut.

Jika pengguna memilih menu 0, maka pengguna akan logout dari akun dan akan kembali ke menu awal.

Lalu, jika pengguna mendapatkan role Admin, maka menu yang akan muncul ialah:

1. Lihat Produk,
2. Tambah Produk,
3. Update Produk, dan
0. Logout

Jika pengguna memilih menu 1, maka pengguna akan ditampilkan list produk ATK.

Jika pengguna memilih menu 2, maka pengguna dapat menambahkan produk pada list produk ATK tersebut.

Jika pengguna memilih menu 3, maka pengguna dapat mengupdate atau memperbarui produk didalam list produk ATK tersebut.

Jika pengguna memilih menu 0, maka pengguna akan logout dari akun dan akan kembali ke menu awal.

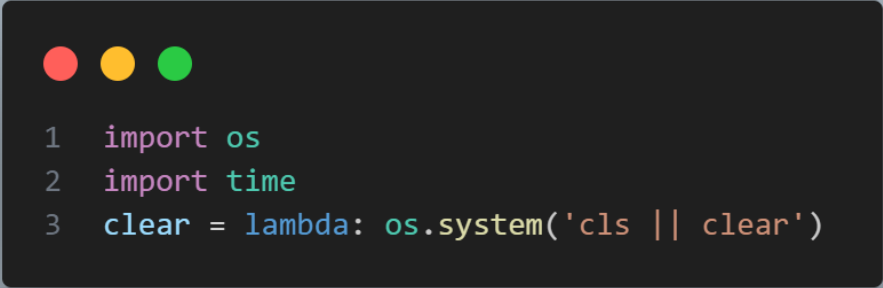
Alasan saya membedakan menu antara Owner dan Admin karena menurut saya Admin tidak dapat menghapus atau menghilangkan produk di perusahaan manapun, dan Admin tidak dapat melihat profit dari perusahaan tersebut.

PROGRAM

Saya akan menjelaskan program CRUD saya dengan judul “Manajemen Keuntungan dari Suatu Perusahaan ATK.

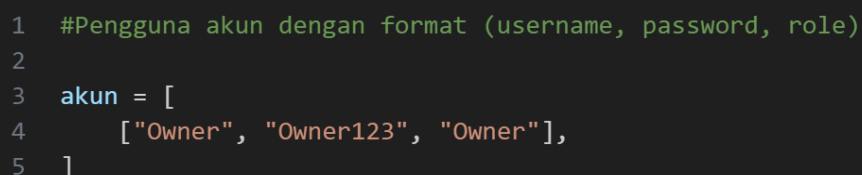
Pertama, saya menggunakan “import os”, dengan command “os.system(‘cls || clear’). Fungsinya adalah untuk membersihkan terminal dari program saya.

Kedua, saya menggunakan “import time” dengan command “time.sleep(detik)”. Fungsinya adalah agar program saya bisa delay dengan waktu yang ditentukan (Contoh : time.sleep(5), command ini dapat ngedelay program dengan waktu 5 detik).



```
1 import os
2 import time
3 clear = lambda: os.system('cls || clear')
```

Dibagian ini, saya membuat list akun. Disini, sudah ada akun yang teregistrasi yaitu akun dari owner, namun di list akun ini bisa menambah akun baru dengan role yang pasti, yaitu role Admin.



```
1 #Pengguna akun dengan format (username, password, role)
2
3 akun = [
4     ["Owner", "Owner123", "Owner"],
5 ]
```

Dibagian ini merupakan list produk-produk ATK, di gambar ini sudah dijelaskan format dari list tersebut.

```
1 #Produk-produk ATK dengan format (no, barang, harga jual, stok , harga beli)
2
3 produkATK = [
4     [1, "Kertas Folio", 1000, 100, 500,],
5     [2, "Keras HVS \t", 1000, 500, 500],
6     [3, "Buku Tulis", 3000, 120, 2000],
7     [4, "Pensil", 2000, 50, 1000],
8     [5, "Pulpen", 2000, 50, 1000],
9 ]
```

Sebelumnya, saya menggunakan fungsi def untuk beberapa bagian agar saya tidak perlu lagi mengetik code dari bagian tersebut (Contoh : Login).

Dibagian login ini, saya menggunakan perulangan While dengan syarat True. Lalu pengguna akan memasukkan username dan password. Selanjutnya saya menggunakan perulangan For untuk memeriksa apakah username dan password sudah sama seperti yang di list akun tadi. Setelah berhasil memasukkan username dan password, ada fungsi return yaitu untuk memeriksa role dari akun yang baru login tersebut.

```
1 #Bagian untuk login
2 def login():
3     clear()
4     while True:
5         print("=== SILAHKAN LOGIN ===")
6         username = (input("Masukkan username : "))
7         password = input("Masukkan password : ")
8
9         for user in akun:
10             if user[0] == username and user[1] == password:
11                 print("Anda berhasil login!")
12                 input("Tekan Enter untuk melanjutkan.....")
13                 return user[2] #Buat kasih role
14         print("Username atau Password salah!")
15         print()
16         input("Tekan Enter untuk melanjutkan...")
17         clear()
```

Dibagian registrasi, ada fungsi try yaitu cara untuk "mencoba" menjalankan kode dan "menangkap" error tanpa membuat program berhenti. Fungsi except digunakan untuk menangkap error yang terjadi.

Pertama-tama pengguna akan disuruh memasukkan username, lalu ada perulangan for untuk memeriksa apakah sudah ada username yang sama atau tidak, jika ada username yang sama, maka pengguna akan memasukkan username baru. Setelah berhasil memasukkan username, pengguna akan memasukkan password. Setelah memasukkan password, ada fungsi append yaitu untuk menambah data baru dari suatu list. (Contoh : `akun.append([username, password, "Admin"])`). Jadi akun yang teregistrasi sudah pasti mendapatkan role Admin.

```
1 def registrasi():
2     clear()
3     try:
4         print("\n=== SILAHKAN REGISTRASI ===")
5         username = input("Masukkan username anda: ")
6
7         # Cek apakah username sudah ada
8         for user in akun:
9             if user[0] == username:
10                print("Username sudah digunakan!")
11                return
12            password = input("Masukkan password anda: ")
13            akun.append([username, password, "Admin"])
14            print("Registrasi berhasil! Silahkan kembali login!")
15            input("Tekan Enter untuk melanjutkan....")
16            return
17     except Exception as e:
18         print(f"Terjadi kesalahan saat registrasi: {str(e)}")
```

Dibagian tampilan produk ini merupakan tampilan untuk role Owner, ada command “:<”, fungsinya adalah membuat space kosong agar tampilan di terminal menjadi rapi. Lalu ada perulangan for untuk memasukkan list produk ATK dengan print agar lebih mudah.

```
1 #Tampilan Produk Owner
2 def displayprodukOwner():
3     clear()
4     print("\n\t\t\t\t\t----- DAFTAR PRODUK -----")
5     print("=" * 75)
6     print(f"{'NO':<5}{'Nama':<20}{'Harga Jual':<15}{'Stok':<10}{'Harga Beli':<15}{'Profit':<10}")
7     for produk in produkATK:
8         profit = produk[2] - produk[4]
9         print(f"{produk[0]:<5}{produk[1]:<20}{produk[2]:<15}{produk[3]:<10}{produk[4]:<15}{profit:<10}")
10    print("=" * 75)
```



```

----- DAFTAR PRODUK -----
=====
NO    Nama            Harga Jual    Stok    Harga Beli    Profit
1     Kertas Folio      1000          100     500           500
2     Keras HVS          1000          500     500           500
3     Buku Tulis         3000          120     2000          1000
4     Pensil             2000          50      1000          1000
5     Pulpen             2000          50      1000          1000
=====
Tekan Enter untuk kembali ke menu...

```

Dibagian tambah produk, owner dapat menambahkan produk pada list produk-produk ATK tadi. (Contoh: barang = Tipes, hargaBeli = 5000, hargaJual = 7000, stok = 100).

```

1  #Tambah Produk
2  def TambahProduk():
3      clear()
4      try:
5          print("\n=== TAMBAH PRODUK ===")
6          barang = input("Masukkan nama produk: ")
7          hargaBeli = int(input("Masukkan harga beli: "))
8          hargaJual = int(input("Masukkan harga jual: "))
9          stok = int(input("Masukkan jumlah stok: "))
10
11         nomorBaru = max([p[0] for p in produkATK]) + 1
12         produkATK.append([nomorBaru, barang, hargaJual, stok, hargaBeli])
13         print("Produk berhasil ditambahkan!")
14     except ValueError:
15         print("Mohon masukkan angka yang valid untuk harga dan stok!")
16     except Exception as e:
17         print(f"Terjadi kesalahan: {(e)}")

```

```

----- DAFTAR PRODUK -----
=====
NO    Nama            Harga Jual    Stok    Harga Beli
1     Kertas Folio      1000          100     500
2     Keras HVS        1000          500     500
3     Buku Tulis       3000          120     2000
4     Pensil           2000          50      1000
5     Pulpen           2000          50      1000
6     Tipes            7000          100     5000
=====
Tekan Enter untuk kembali ke menu...

```

Di bagian update produk, pengguna dapat merubah atau memperbarui produk yang ada didalam list produk-produk ATK tersebut. (Contoh : Saya ingin merubah harga produk tipex, maka saya akan memasukkann harga harga yang baru.)

```
1 #Update Produk
2 def updateProduk():
3     clear()
4     try:
5         displayprodukAdmin()
6         idProduk = int(input("\nMasukkan ID produk yang akan diupdate: "))
7
8         for produk in produkATK:
9             if produk[0] == idProduk:
10                 print(f"Mengupdate produk: {produk[1]}")
11                 produk[1] = input("Masukkan nama baru (kosongkan jika tidak diubah): ") or produk[1]
12                 produk[2] = int(input("Masukkan harga beli baru (0 jika tidak diubah): ")) or produk[2]
13                 produk[3] = int(input("Masukkan stok baru (0 jika tidak diubah): ")) or produk[3]
14                 produk[4] = int(input("Masukkan harga jual baru (0 jika tidak diubah): ")) or produk[4]
15                 clear()
16                 print("Produk sedang diupdate! Mohon tunggu 5 detik.....")
17                 time.sleep(5)
18                 clear()
19                 input("Tekan Enter untuk melanjutkan.....")
20                 clear()
21                 displayprodukAdmin()
22                 return
23             print("ID produk tidak ditemukan!")
24         except ValueError:
25             print("Mohon masukkan ID yang valid!")
26         except Exception as e:
27             print(f"Terjadi kesalahan: {str(e)}")
```

----- DAFTAR PRODUK -----

NO	Nama	Harga Jual	Stok	Harga Beli
1	Kertas Folio	1000	100	500
2	Keras HVS	1000	500	500
3	Buku Tulis	3000	120	2000
4	Pensil	2000	50	1000
5	Pulpen	2000	50	1000
6	Tipex	7000	100	5000

```
Masukkan ID produk yang akan diupdate: 6
Mengupdate produk: Tipex
Masukkan nama baru (kosongkan jika tidak diubah):
Masukkan harga beli baru (0 jika tidak diubah): 7000
Masukkan stok baru (0 jika tidak diubah): 0
Masukkan harga jual baru (0 jika tidak diubah): 10000
```

----- DAFTAR PRODUK -----

NO	Nama	Harga Jual	Stok	Harga Beli	Profit
1	Kertas Folio	1000	100	500	500
2	Keras HVS	1000	500	500	500
3	Buku Tulis	3000	120	2000	1000
4	Pensil	2000	50	1000	1000
5	Pulpen	2000	50	1000	1000
6	Tipex	10000	100	7000	3000

Tekan Enter untuk kembali ke menu...

Gambar diatas ini adalah hasil update terbaru.

Dibawah ini adalah bagian hapus produk. Menggunakan command .remove untuk menghapus produk didalam list produk-produk ATK.

```

1  #Hapus Produk
2  def hapusProduk():
3      clear()
4      try:
5          displayprodukOwner()
6          idProduk = int(input("\nMasukkan nomor produk yang akan dihapus: "))
7
8          for produk in produkATK:
9              if produk[0] == idProduk:
10                 produkATK.remove(produk)
11                 clear()
12                 print("Produk sedang dihapus! Mohon tunggu 5 detik....")
13                 time.sleep(5)
14                 input("Produk berhasil dihapus! Tekan Enter untuk melanjutkan....")
15                 clear()
16                 displayprodukOwner()
17                 return
18             print("ID produk tidak ditemukan!")
19         except ValueError:
20             print("Mohon masukkan ID yang valid!")
21         except Exception as e:
22             print(f"Terjadi kesalahan: {str(e)}")

```

----- DAFTAR PRODUK -----

NO	Nama	Harga Jual	Stok	Harga Beli	Profit
1	Kertas Folio	1000	100	500	500
2	Keras HVS	1000	500	500	500
3	Buku Tulis	3000	120	2000	1000
4	Pensil	2000	50	1000	1000
5	Pulpen	2000	50	1000	1000
6	Tipex	10000	100	7000	3000

Masukkan nomor produk yang akan dihapus: 6

Output yang dihasilkan adalah

```
----- DAFTAR PRODUK -----  
=====
```

NO	Nama	Harga Jual	Stok	Harga Beli	Profit
1	Kertas Folio	1000	100	500	500
2	Keras HVS	1000	500	500	500
3	Buku Tulis	3000	120	2000	1000
4	Pensil	2000	50	1000	1000
5	Pulpen	2000	50	1000	1000

```
=====
```

Lalu, dibagian terakhir, ada menu lihat profit.

```
1 #Total profit  
2 def totalProfit():  
3     total_profit = 0  
4     for produk in produkATK:  
5         # harga jual - harga beli  
6         profitPcs = produk[2] - produk[4]  
7         # profit per pieces dikali stok produk  
8         total_profit += profitPcs * produk[3]  
9     return total_profit
```

```
Total keuntungan anda sebesar Rp520000  
Tekan Enter untuk kembali ke menu...
```

Lalu untuk role Worker, yang membedakan selain beberapa menu adalah output yang keluar yaitu di menu tampilan. Di menu ini tidak ditampilkan profitnya berapa.

```
1 #Tampilan Produk Admin
2 def displayprodukAdmin():
3     clear()
4     print("\t\t\t\t\t----- DAFTAR PRODUK -----")
5     print("=" * 75)
6     print(f"{'NO':<5}{'Nama':<20}{'Harga Jual':<15}{'Stok':<10}{'Harga Beli':<15}")
7     for produk in produkATK:
8         print(f"{produk[0]:<5}{produk[1]:<20}{produk[2]:<15}{produk[3]:<10}{produk[4]:<15}")
9     print("=" * 75)
```

```
----- DAFTAR PRODUK -----
=====
NO  Nama                Harga Jual  Stok  Harga Beli
1   Kertas Folio        1000       100   500
2   Keras HVS           1000       500   500
3   Buku Tulis          3000       120   2000
4   Pensil              2000        50   1000
5   Pulpen              2000        50   1000
=====
Tekan Enter untuk kembali ke menu... 
```

Lalu, masuk ke fungsi main, yaitu menu utama dari program ini, berikut adalah lampiran dari kode program ini:

```

1  def main():
2      while True:
3          clear()
4          print("=" * 75)
5          print("\tPROGRAM MANAJEMEN KEUNTUNGAN DARI SUATU PERUSAHAAN ATK")
6          print("=" * 75)
7          print("[1] Login\n[2] Register\n[3] Keluar Program")
8          print("=" * 75)
9
10         pilihan = input("Pilih Menu (1-3): ")
11
12         if pilihan == "1":
13             role = login()
14             while True:
15                 clear()
16                 if role == "Owner":
17                     print(f"=== Menu {role.title()} ===")
18                     print("1. Lihat Produk\n2. Tambah Produk\n3. Update Produk\n4. Hapus Produk\n5. Lihat Profit\n0. Logout")
19                     pilihanMenu = input("Pilih menu : ")
20
21                     if pilihanMenu == "1":
22                         clear()
23                         displayprodukOwner()
24                         input("Tekan Enter untuk kembali ke menu...")
25                     elif pilihanMenu == "2":
26                         clear()
27                         TambahProduk()
28                         input("Tekan Enter untuk kembali ke menu...")
29                     elif pilihanMenu == "3":
30                         clear()
31                         updateProduk()
32                         input("Tekan Enter untuk kembali ke menu...")
33                     elif pilihanMenu == "4":
34                         clear()
35                         hapusProduk()
36                         input("Tekan Enter untuk kembali ke menu...")
37                     elif pilihanMenu == "5":
38                         clear()
39                         total_profit = totalProfit()
40                         print(f"Total keuntungan anda sebesar Rp{total_profit}")
41                         input("Tekan Enter untuk kembali ke menu...")
42                     elif pilihanMenu == "0":
43                         clear()
44                         print("Anda telah logout dari program!")
45                         input("Tekan Enter untuk melanjutkan....")
46                         break
47
48                 elif role == "Admin":
49                     clear()
50                     print(f"=== Menu {role.title()} ===")
51                     print("1. Lihat Produk\n2. Tambah Produk\n3. Update Produk\n0. Logout")
52                     pilihanMenu = input("Pilih menu : ")
53
54                     if pilihanMenu == "1":
55                         clear()
56                         displayprodukAdmin()
57                         input("Tekan Enter untuk kembali ke menu...")
58                     elif pilihanMenu == "2":
59                         clear()
60                         TambahProduk()
61                         input("Tekan Enter untuk kembali ke menu...")
62                     elif pilihanMenu == "3":
63                         clear()
64                         updateProduk()
65                         input("Tekan Enter untuk kembali ke menu...")
66                     elif pilihanMenu == "0":
67                         clear()
68                         print("Anda telah logout dari program!")
69                         input("Tekan Enter untuk melanjutkan....")
70                         break
71
72             elif pilihan == "2":
73                 registrasi()
74             elif pilihan == "3":
75                 print("Terima kasih telah menggunakan program ini!")
76                 exit()

```

Di menu main ini, pengguna akan diarahkan untuk login terlebih dahulu

```

=====
                        PROGRAM MANAJEMEN KEUNTUNGAN DARI SUATU PERUSAHAAN ATK
=====
[1] Login
[2] Register
[3] Keluar Program
=====
Pilih Menu (1-3): 

```

Setelah berhasil login, pengguna akan diperiksa rolenya, dan jika rolenya adalah Owner, maka output yang dihasilkan adalah:

```
=== Menu Owner ===  
1. Lihat Produk  
2. Tambah Produk  
3. Update Produk  
4. Hapus Produk  
5. Lihat Profit  
0. Logout  
Pilih menu : 
```

Dan jika role yang diperiksa ialah Admin, maka output yang dihasilkan ialah:

```
=== Menu Admin ===  
1. Lihat Produk  
2. Tambah Produk  
3. Update Produk  
0. Logout  
Pilih menu : 
```

Berikut adalah output ketika pengguna keluar dari program:

```
Terima kasih telah menggunakan program ini!
```