

APLIKASI MANAJEMEN BARANG PERTOKOAN

Disusun untuk Memenuhi Tugas Matakuliah Pemrograman Desktop yang Dibimbing oleh

Moh. Kautsar Sophan, , S.Kom., M.MT



Disusun oleh :

Mohammad Al-Fayed Mada Fauzi 190411100033

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS TEKNIK

UNIVERSITAS TRUNOJOYO MADURA

2020

DAFTAR ISI

DAFTAR ISI	i
1. Deskripsi Aplikasi	1
2. Usecase Diagram	1
3. Database Aplikasi	3
4. Daftar File Python	4
5. Capture Hasil Runnning Aplikasi	4
6. Evaluasi	15
7. Listingan Code Program	16

PEMBAHASAN

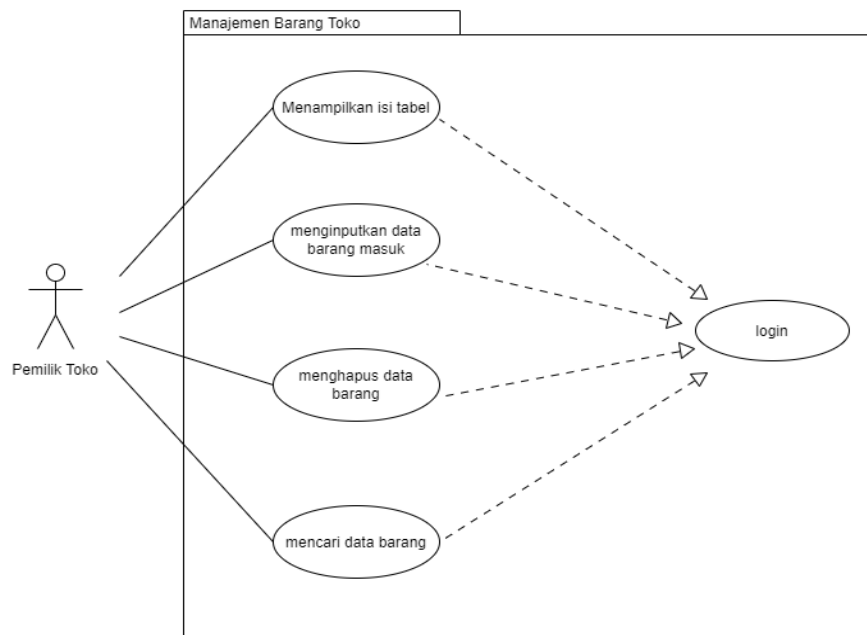
1. Deskripsi Aplikasi

Ide pembuatan aplikasi ini dilatar belakangi oleh situasi yang kerap saya jumpai saat berbelanja di toko kecil atau toko kelontong di dekat rumah, dimana pemilik toko tersebut yang masih melakukan pencatatan barang yang masuk secara manual menggunakan kertas, hal ini menyebabkan ketidakberaturan catatan barang dan membutuhkan waktu lebih ketika mengecek data barang karena harus mencari data satu persatu, belum lagi saat kondisi toko ramai pembeli harus menunggu dikarenakan pemilik sibuk memeriksa catatan barang yang ada, sehingga menimbulkan antrian yang sangat beresiko disaat pandemi seperti ini dimana kita dilarang untuk berkerumun serta wajib menjaga jarak antara satu sama lain. Selain itu bila catatan hilang atau rusak maka pemilik toko harus meluangkan waktu lagi untuk melakukan pendataan ulang pada barang yang ada, dimana hal itu tidak efisien dan justru membuang waktu pemilik toko.

Oleh karena itulah saya membuat aplikasi ini yang sejatinya ditujukan untuk para pemilik toko kelontong agar dapat melakukan pendataan barang dengan mudah dan efisien. Hanya dengan memasukkan data barang yang masuk kedalam aplikasi, maka secara otomatis data tersebut akan terekam dalam aplikasi tersebut dan dapat dilihat sewaktu-waktu, serta tidak perlu takut catatan hilang atau rusak seperti saat melakukan pendataan secara manual.

2. Use Case Diagram

Use case diagram merupakan suatu gambaran atau representasi dari interaksi yang terjadi antara sistem dengan lingkungannya(pemakainya). Use case termasuk kedalam salah satu tipe dari diagram UML(Unified Modeling Language), dimana use case didefinisikan dari sudut pandang aktor yang terlibat. Istilah aktor disini merujuk pada orang atau alat yang menggunakan sistem tersebut atau bisa juga siapapun yang berinteraksi dengan sistem.

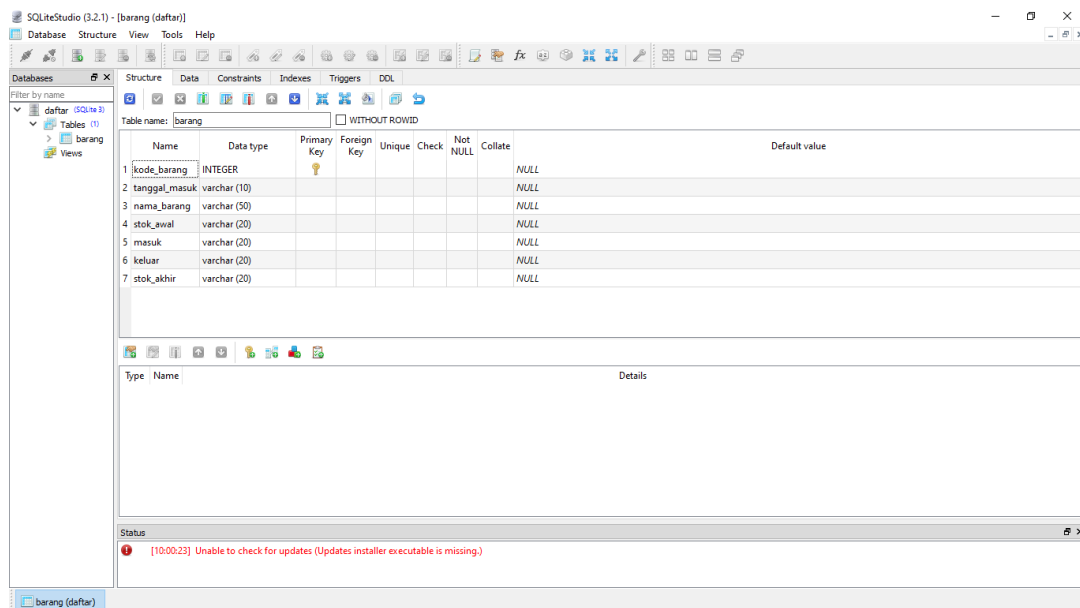


Use Case Diagram

Jadi dalam aplikasi yang saya buat, yang bertindak sebagai aktor adalah pemilik toko itu sendiri dan untuk dapat menjalankan aplikasi serta mengakses seluruh fitur yang ada dalam aplikasi ini, maka sang aktor(pemilik toko) harus melakukan login terlebih dahulu baru setelah itu dapat mengakses fitur aplikasi yang diantaranya:

- Menampilkan isi tabel : menampilkan seluruh data isian yang terdapat pada tabel, dengan cara menklik button yang telah disiapkan.
- Menginputkan data barang masuk : menginputkan data dengan format yang diinputkan yaitu kode barang, tanggal masuk barang, nama barang, jumlah stok awal(stok sebelum barang masuk lagi), jumlah barang yang masuk, jumlah barang yang keluar dan jumlah stok akhir(setelah sisa stok ditambah jumlah barang yang masuk).
- Menghapus data barang : menghapus data barangp ada tabel dengan memasukkan id dari barang yang akan dihapus.
- Mencari data barang : melakukan seacrhing data barang dengan menginputkan nama barang yanga dimaksud.

3. Database Aplikasi



Untuk database aplikasi yang saya buat, saya menggunakan sqlite database dengan nama database daftar yang didalamnya berisi 1 tabel bernama barang untuk menyimpan data barang dengan yang terdiri dari 7 kolom, yang strukturnya sebagai berikut :

- Kolom kode barang : menggunakan tipe data integer dengan setting primary key (bersifat unik/tidak boleh memiliki kesamaan), tabel ini digunakan untuk menyimpan kode dari barang yang diinputkan.
- Kolom tanggal_masuk : ,menggunakan tipe data varchar dengan panjang inputan maksimal 10 karakter. Tabel ini digunakan untuk menyimpan data tanggal saat barang masuk ke toko.
- Kolom nama_barang : meggunakan tipe data varchar dengan panjang inputan maksimal 50 karakter. Tabel ini digunakan untuk menyimpan nama dari barang.
- Kolom stok_awal : menggunakan tipe data varchar dengan panjang inputan maksimal 20 karakter. Tabel ini digunakan untuk menyimpan jumlah stok awal(stok barang yang dimiliki sebelum stok baru masuk).
- Kolom keluar : menggunakan tipe data varchar dengan panjang inputan maksimal 20 karakter. Tabel ini digunakan untuk menyimpan jumlah stok barang yang keluar sebelum ada barang baru yang masuk.
- Kolom masuk : menggunakan tipe data varchar dengan panjang inputan maksimal 20 karakter. Tabel ini digunakan untuk menyimpan jumlah stok baru dari barang yang masuk.

- Kolom `stok_akhir` : menggunakan tipe data `varchar` dengan panjang inputan maksimal 20 karakter. Tabel ini digunakan untuk menyimpan jumlah stok akhir dari suatu barang, setelah stok awal dikurang dengan jumlah barang yang keluar dan ditambah dengan jumlah barang baru yang masuk.

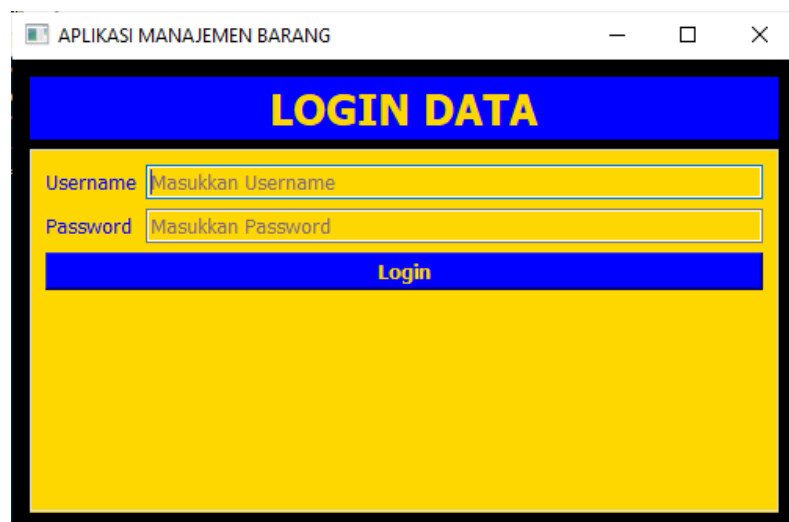
4. File Python

File dari aplikasi yang saya buat berjumlah 1 file yaitu file dengan nama `aplikasi.py` yang didalamnya menampung seluruh tampilan dari aplikasi yang saya buat. Berikut penjelasannya dari isi file yang saya buat ;

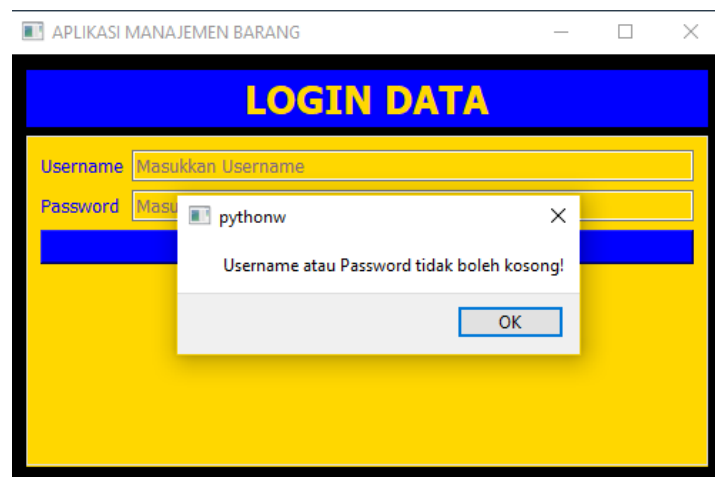
- Class `login()` : menampung tampilan (window) login aplikasi, tampilan ini ialah tampilan yang pertama kali muncul saat menjalankan aplikasi nantinya.
- Class `tampilan_utama()` : digunakan untuk membuat tampilan utama dari aplikasi yang didalamnya terdapat table database yang telah dibuat. Selain itu didalamnya juga terdapat fitur menampilkan seluruh data tabel serta search data pada tabel.
- Class `tambah()` : digunakan untuk membuat window baru yang nantinya menjalankan fitur penambahan atau insert data pada tabel.
- Class `hapus()` : digunakan untuk membuat window baru yang nantinya menjalankan penghapusan data pada tabel.

5. Capture Hasil Running

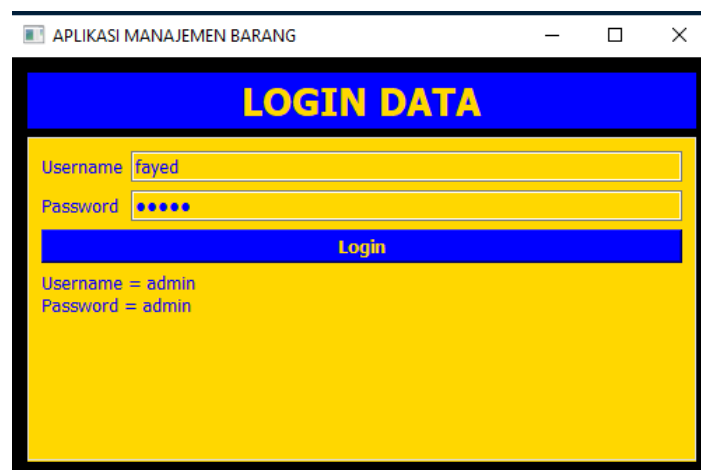
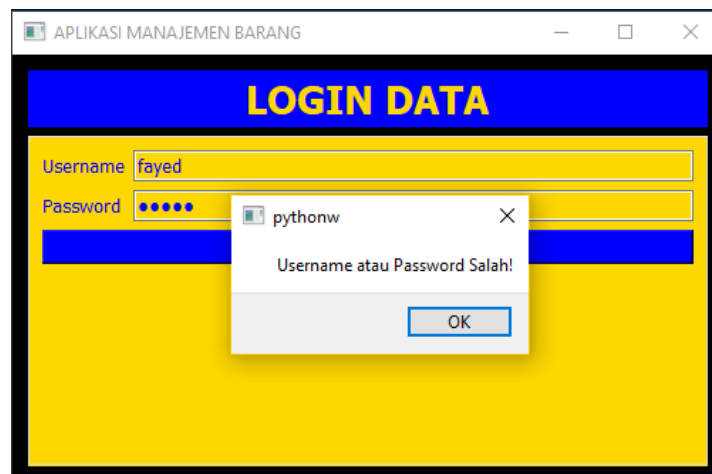
- Halaman login



- Saat username dan password kosong



- Saat username/password salah, maka akan muncul pesan peringatan dan akan diberitahuakan username serta password yang benar.



- Apabila username dan password benar, maka akan diarahkan menuju ke window utama.



- Tampilan utama aplikasi. Untuk menampilkan tabel daftar barang anda perlu meng-klik tombol “tampilan data barang” setelah itu secara otomatis tabel akan ditampilkan.

APLIKASI MANAJEMEN BARANG

DATA BARANG MASUK

Cari Barang **Cari**

hitung jumlah barang masuk perhari **hitung jumlah seluruh data barang**

↓

Tambah Data Barang **Hapus Data Barang** **Tampilkan Daftar Barang**

APLIKASI MANAJEMEN BARANG

DATA BARANG MASUK

Cari Barang **Cari**

hitung jumlah barang masuk perhari **hitung jumlah seluruh data barang**

	kode_barang	tanggal_masuk	nama_barang	stok_awal	masuk	keluar	stok_akhir
1	1	03/11/2020	indomie goreng	3 kardus	2 kardus	2 kardus	3 kardus
2	2	03/11/2020	indomie soto	2 kardus	2 kardus	2 kardus	2 kardus
3	3	03/11/2020	gula pasir	10 kg	3 kg	5 kg	8 kg
4	4	03/11/2020	tabung gas	20 buah	11 buah	10 buah	19 buah
5	5	03/11/2020	galon air aqua	10 buah	3 buah	5 buah	8 buah
6	6	03/11/2020	kecap sedap	10 botol	7 botol	6 botol	11 botol
7	7	03/11/2020	garam	2 kardus	1 kardus	1 kardus	2 kardus
8	8	06/11/2020	telur ayam	9 kotak	10 kotak	8 kotak	11 kotak
9	9	06/11/2020	aqua botol	3 kardus	1 kardus	2 kardus	2 kardus
10	10	06/11/2020	sambal abc	1 kardus	1 kardus	1 kardus	1 kardus
11	11	06/11/2020	tepung terigu	10 kg	6 kg	9 kg	7 kg
12	12	06/11/2020	beras biasa	25 kg	13 kg	15 kg	23 kg
13	13	06/11/2020	sosis so good	2 kardus	3 kardus	2 kardus	3 kardus
14	14	06/11/2020	mie sedap ...	2 kardus	1 kardus	1 kardus	2 kardus

Tambah Data Barang **Hapus Data Barang** **Tampilkan Daftar Barang**

- Searching data pada tabel. Untuk melakukan searching data, anda tinggal memasukkan nama barang yang dicari pada kolom pencarian. Contohnya seperti dibawah ini :

The screenshot shows a web application titled "APLIKASI MANAJEMEN BARANG" with a main heading "DATA BARANG MASUK". Below the heading is a search bar labeled "Cari Barang" with the text "beras" entered. To the right of the search bar is a "Cari" button. Below the search bar are two buttons: "hitung jumlah barang masuk perhari" and "hitung jumlah seluruh data barang". Below these buttons is a table with 8 columns: kode_barang, tanggal_masuk, nama_barang, stok_awal, masuk, keluar, and stok_akhir. The table contains two rows of data for "beras biasa".

	kode_barang	tanggal_masuk	nama_barang	stok_awal	masuk	keluar	stok_akhir
1	12	06/11/2020	beras biasa	25 kg	13 kg	15 kg	23 kg
2	56	10/12/2020	beras biasa	23 kg	15 kg	18 kg	20 kg

At the bottom of the application window are three buttons: "Tambah Data Barang", "Hapus Data Barang", and "Tampilkan Daftar Barang".

- Window tambah data barang. Untuk menambahkan data barang anda perlu mengklik tombol “tambah data barang” lalu mengisi data barang yang akan diinputkan.

The screenshot shows the same application window, but the search bar is empty and the table contains a list of various goods. A red arrow points to the "Tambah Data Barang" button at the bottom.

	kode_barang	tanggal_masuk	nama_barang	stok_awal	masuk	keluar	stok_akhir
67	67	18/12/2020	permen karet	-	2 pack	-	2 pack
68	68	18/12/2020	tepung terigu	7 kg	10 kg	7 kg	10 kg
69	69	18/12/2020	gula pasir	8 kg	10 kg	7 kg	11 kg
70	70	18/12/2020	sunlight	2 kardus	3 kardus	2 kardus	3 kardus
71	71	18/12/2020	rinso	10 rentengan	7 rentengan	8 rentengan	9 rentengan
72	72	18/12/2020	tissue paseo	2 kardus	2 kardus	2 kardus	2 kardus
73	73	18/12/2020	garam	2 kardus	2 kardus	2 kardus	2 kardus
74	74	18/12/2020	kecap sedap	11 botol	8 botol	10 botol	9 botol
75	75	20/12/2020	kertas hvs	6 pack	5 pack	5 pack	6 pack
76	76	20/12/2020	kertas folio	7 pack	5 pack	7 pack	5 pack
77	77	20/12/2020	bulpoin pilot1	3 kotak	2 kotak	2 kotak	3 kotak
78	78	20/12/2020	gudang garam	2 kotak	2 kotak	2 kotak	2 kotak
79	79	20/12/2020	rokok LA	2 kotak	1 kotak	1 kotak	1 kotak
80	80	20/12/2020	skm indomilk	15 kaleng	10 kaleng	13 kaleng	12 kaleng

At the bottom of the application window are three buttons: "Tambah Data Barang", "Hapus Data Barang", and "Tampilkan Daftar Barang".

TAMBAH DATA BARANG

TAMBAH DATA

Kode Barang

Tgl. Masuk

Nama Barang

Stok Awal

Masuk

Keluar

Stok Akhir

Tambah

- Saat kode barang terindikasi sama dengan yang sudah diinputkan sebelumnya atau telah digunakan sebelumnya, maka akan ada pesan pemberitahuan dan anda anda dianjurkan mengganti kode barangnya. Contoh input data :

TAMBAH DATA BARANG

TAMBAH DATA

Kode Barang

Tgl. Masuk

Nama Barang

Stok Awal

Masuk

Keluar

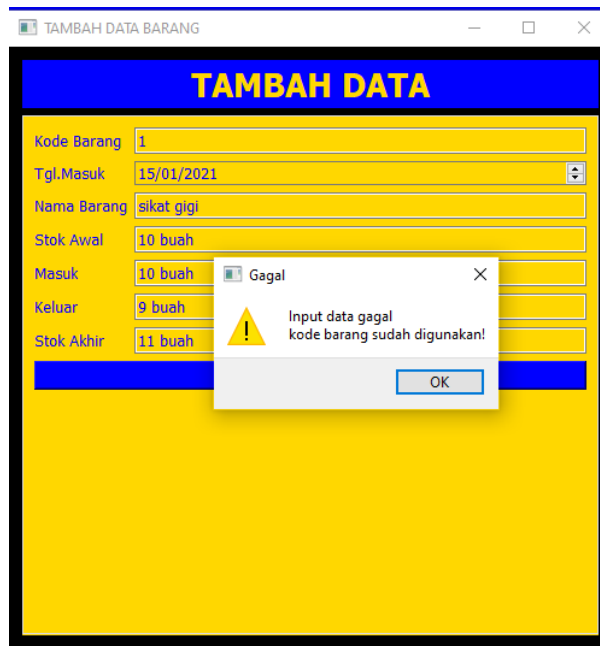
Stok Akhir

Tambah

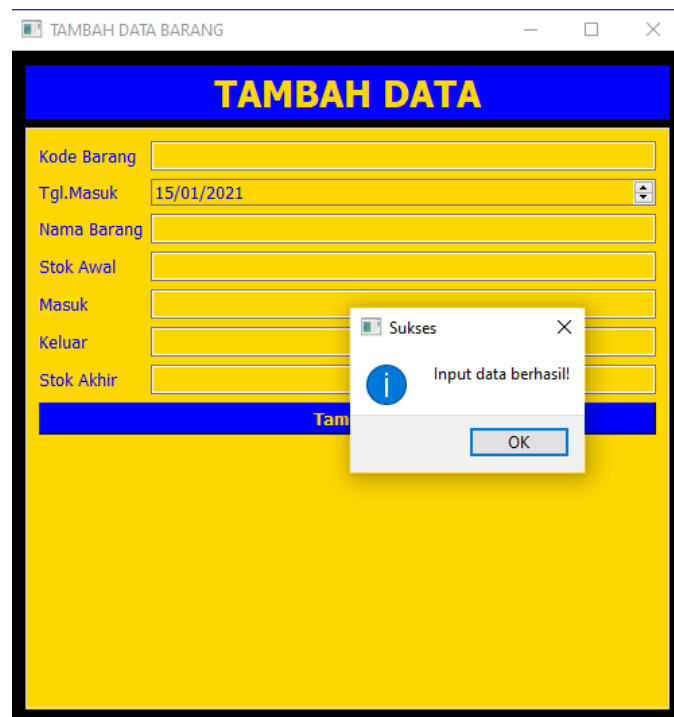
Question

Yakin data sudah benar?

Yes **Cancel**



- Setelah berhasil menginputkan data, maka untuk menampilkan data baru yang telah diinputkan tadi perlu melakukan load tabel dengan cara meng-klik lagi tombol “tampilkan data barang” nantinya tabel akan otomatis terefres dan data barang yang diinputkan akan tampil pada tabel.



Sebelum data tabel di load, data baru yang diinputkan tadi tidak muncul pada tabel.

APLIKASI MANAJEMEN BARANG

DATA BARANG MASUK

Cari Barang **Cari**

hitung jumlah barang masuk perhari

hitung jumlah seluruh data barang

	kode_barang	tanggal_masuk	nama_barang	stok_awal	masuk	keluar	stok_akhir
67	67	18/12/2020	permen karet	-	2 pack	-	2 pack
68	68	18/12/2020	tepung terigu	7 kg	10 kg	7 kg	10 kg
69	69	18/12/2020	gula pasir	8 kg	10 kg	7 kg	11 kg
70	70	18/12/2020	sunlight	2 kardus	3 kardus	2 kardus	3 kardus
71	71	18/12/2020	rinso	10 rentengan	7 rentengan	8 rentengan	9 rentengan
72	72	18/12/2020	tissue paseo	2 kardus	2 kardus	2 kardus	2 kardus
73	73	18/12/2020	garam	2 kardus	2 kardus	2 kardus	2 kardus
74	74	18/12/2020	kecap sedap	11 botol	8 botol	10 botol	9 botol
75	75	20/12/2020	kertas hvs	6 pack	5 pack	5 pack	6 pack
76	76	20/12/2020	kertas folio	7 pack	5 pack	7 pack	5 pack
77	77	20/12/2020	bulpoin pilot1	3 kotak	2 kotak	2 kotak	3 kotak
78	78	20/12/2020	gudang garam	2 kotak	2 kotak	2 kotak	2 kotak
79	79	20/12/2020	rokok LA	2 kotak	1 kotak	1 kotak	1 kotak
80	80	20/12/2020	skm indomilk	15 kaleng	10 kaleng	13 kaleng	12 kaleng

Tambah Data Barang

Hapus Data Barang

Tampilkan Daftar Barang

Setelah tabel diload, maka data tadi akan otomatis ditampilkan.

APLIKASI MANAJEMEN BARANG

DATA BARANG MASUK

Cari Barang **Cari**

hitung jumlah barang masuk perhari

hitung jumlah seluruh data barang

	kode_barang	tanggal_masuk	nama_barang	stok_awal	masuk	keluar	stok_akhir
68	68	18/12/2020	tepung terigu	7 kg	10 kg	7 kg	10 kg
69	69	18/12/2020	gula pasir	8 kg	10 kg	7 kg	11 kg
70	70	18/12/2020	sunlight	2 kardus	3 kardus	2 kardus	3 kardus
71	71	18/12/2020	rinso	10 rentengan	7 rentengan	8 rentengan	9 rentengan
72	72	18/12/2020	tissue paseo	2 kardus	2 kardus	2 kardus	2 kardus
73	73	18/12/2020	garam	2 kardus	2 kardus	2 kardus	2 kardus
74	74	18/12/2020	kecap sedap	11 botol	8 botol	10 botol	9 botol
75	75	20/12/2020	kertas hvs	6 pack	5 pack	5 pack	6 pack
76	76	20/12/2020	kertas folio	7 pack	5 pack	7 pack	5 pack
77	77	20/12/2020	bulpoin pilot1	3 kotak	2 kotak	2 kotak	3 kotak
78	78	20/12/2020	gudang garam	2 kotak	2 kotak	2 kotak	2 kotak
79	79	20/12/2020	rokok LA	2 kotak	1 kotak	1 kotak	1 kotak
80	80	20/12/2020	skm indomilk	15 kaleng	10 kaleng	13 kaleng	12 kaleng
81	81	15/01/2021	sikat gigi	10 buah	10 buah	9 buah	11 buah

Tambah Data Barang

Hapus Data Barang

Tampilkan Daftar Barang

- Window hapus data barang. Untuk menuju halaman tersebut anda harus meng-klik tombol “hapus data barang” setelah itu anda akan diarahkan menuju halaman tersebut dan anda dapat melakukan penghapusan data barang dengan menginputkan kode barang dari data barang yang akan dihapus.

APLIKASI MANAJEMEN BARANG

DATA BARANG MASUK

Cari Barang **Cari**

hitung jumlah barang masuk perhari
hitung jumlah seluruh data barang

	kode_barang	tanggal_masuk	nama_barang	stok_awal	masuk	keluar	stok_akhir
68	68	18/12/2020	tepung terigu	7 kg	10 kg	7 kg	10 kg
69	69	18/12/2020	gula pasir	8 kg	10 kg	7 kg	11 kg
70	70	18/12/2020	sunlight	2 kardus	3 kardus	2 kardus	3 kardus
71	71	18/12/2020	rinso	10 rentengan	7 rentengan	8 rentengan	9 rentengan
72	72	18/12/2020	tissue paseo	2 kardus	2 kardus	2 kardus	2 kardus
73	73	18/12/2020	garam	2 kardus	2 kardus	2 kardus	2 kardus
74	74	18/12/2020	kecap sedap	11 botol	8 botol	10 botol	9 botol
75	75	20/12/2020	kertas hvs	6 pack	5 pack	5 pack	6 pack
76	76	20/12/2020	kertas folio	7 pack	5 pack	7 pack	5 pack
77	77	20/12/2020	bulpoin pilot1	3 kotak	2 kotak	2 kotak	3 kotak
78	78	20/12/2020	gudang garam	2 kotak	2 kotak	2 kotak	2 kotak
79	79	20/12/2020	rokok LA	2 kotak	1 kotak	1 kotak	1 kotak
80	80	20/12/2020	skm indomilk	15 kaleng	10 kaleng	13 kaleng	12 kaleng
81	81	15/01/2021	sikat gigi	10 bu	10 buah	9 buah	11 buah

Tambah Data Barang
Hapus Data Barang
Tampilkan Daftar Barang

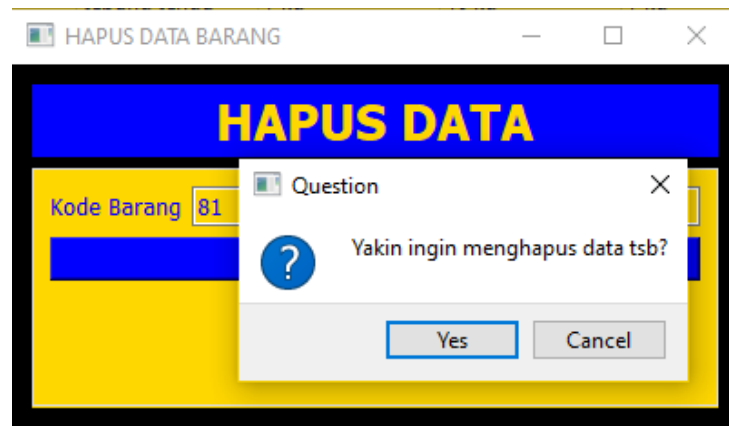
HAPUS DATA BARANG

HAPUS DATA

Kode Barang

Hapus

Contoh penghapusan data pada tabel :



Apabila ingin mengecek apakah data sudah terhapus atau belum, anda perlu me-load tabel terlebih dahulu. Tampilan dibawah merupakan tampilan sebelum tabel diload, dimana data barang yang kita hapus tadi masih ada pada tabel.

APLIKASI MANAJEMEN BARANG

DATA BARANG MASUK

Cari Barang Cari

hitung jumlah barang masuk perhari
hitung jumlah seluruh data barang

	kode_barang	tanggal_masuk	nama_barang	stok_awal	masuk	keluar	stok_akhir
68	68	18/12/2020	tepung terigu	7 kg	10 kg	7 kg	10 kg
69	69	18/12/2020	gula pasir	8 kg	10 kg	7 kg	11 kg
70	70	18/12/2020	sunlight	2 kardus	3 kardus	2 kardus	3 kardus
71	71	18/12/2020	rinso	10 rentengan	7 rentengan	8 rentengan	9 rentengan
72	72	18/12/2020	tissue paseo	2 kardus	2 kardus	2 kardus	2 kardus
73	73	18/12/2020	garam	2 kardus	2 kardus	2 kardus	2 kardus
74	74	18/12/2020	kecap sedap	11 botol	8 botol	10 botol	9 botol
75	75	20/12/2020	kertas hvs	6 pack	5 pack	5 pack	6 pack
76	76	20/12/2020	kertas folio	7 pack	5 pack	7 pack	5 pack
77	77	20/12/2020	bulpoin pilot1	3 kotak	2 kotak	2 kotak	3 kotak
78	78	20/12/2020	gudang garam	2 kotak	2 kotak	2 kotak	2 kotak
79	79	20/12/2020	rokok LA	2 kotak	1 kotak	1 kotak	1 kotak
80	80	20/12/2020	skm indomilk	15 kaleng	10 kaleng	13 kaleng	12 kaleng
81	81	15/01/2021	sikat gigi	10 buah	10 buah	9 buah	11 buah

Tambah Data Barang
Hapus Data Barang
Tampilkan Daftar Barang

Tampilan setelah tabel diload. Saat data disearch, data sudah tidak ditemukan yang berarti data tersebut sukses dihapus.

APLIKASI MANAJEMEN BARANG

DATA BARANG MASUK

Cari Barang **Cari**

hitung jumlah barang masuk perhari				hitung jumlah seluruh data barang			
	kode_barang	tanqqal_masuk	nama_barang	stok_awal	masuk	keluar	stok_akhir
67	67	18/12/2020	permen karet	-	2 pack	-	2 pack
68	68	18/12/2020	tepung terigu	7 kg	10 kg	7 kg	10 kg
69	69	18/12/2020	gula pasir	8 kg	10 kg	7 kg	11 kg
70	70	18/12/2020	sunlight	2 kardus	3 kardus	2 kardus	3 kados
71	71	18/12/2020	rinso	10 rentengan	7 rentengan	8 rentengan	9 rentengan
72	72	18/12/2020	tissue paseo	2 kardus	2 kardus	2 kardus	2 kardus
73	73	18/12/2020	garam	2 kardus	2 kardus	2 kardus	2 kardus
74	74	18/12/2020	kecap sedap	11 botol	8 botol	10 botol	9 botol
75	75	20/12/2020	kertas hvs	6 pack	5 pack	5 pack	6 pack
76	76	20/12/2020	kertas folio	7 pack	5 pack	7 pack	5 pack
77	77	20/12/2020	bulpoin pilot1	3 kotak	2 kotak	2 kotak	3 kotak
78	78	20/12/2020	gudang garam	2 kotak	2 kotak	2 kotak	2 kotak
79	79	20/12/2020	rokok LA	2 kotak	1 kotak	1 kotak	1 kotak
80	80	20/12/2020	skm indomilk	15 kaleng	10 kaleng	13 kaleng	12 kaleng

Tambah Data Barang **Hapus Data Barang** **Tampilkan Daftar Barang**

- Mempilkan berapa jumlah data nama barang yang diinputkat sesuai tanggalnya.

APLIKASI MANAJEMEN BARANG

DATA BARANG MASUK

Cari Barang **Cari**

hitung jumlah barang masuk perhari		hitung jumlah seluruh data barang	
	tanqqal_masuk	jumlah_data_masuk	
1	01/12/2020	4	
2	03/11/2020	7	
3	06/11/2020	7	
4	06/12/2020	4	
5	09/11/2020	7	
6	10/12/2020	8	
7	12/11/2020	7	
8	15/12/2020	8	
9	16/11/2020	4	
10	18/12/2020	9	
11	20/11/2020	5	
12	20/12/2020	6	
13	26/11/2020	4	

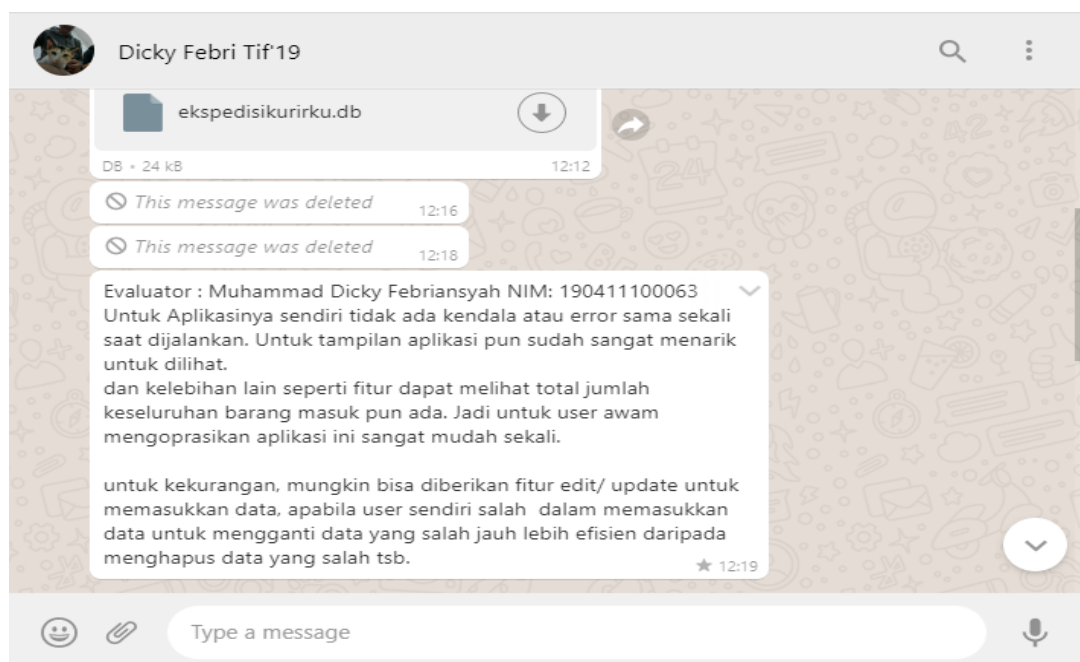
Tambah Data Barang **Hapus Data Barang** **Tampilkan Daftar Barang**

- Menampilkan jumlah seluruh data nama barang yang telah diinputkan pada tabel.



6. Evaluasi

Untuk evaluasi saya lakukan lewat media whatsapp, dimana sebelum itu say mengirimkan file program saya kepada teman yang akan melakukan evaluasi terhadap program yang saya buat. Disini saya menunjuk satu teman untuk mengevaluasi program saya, yaitu Muhammad Dicky Ferbriansyah, dengn NIM 190411100063



7. Listing Code Program

```
import sys
from PyQt5 import QtWidgets
from PyQt5.QtWidgets import *
from PyQt5.QtCore import*
from PyQt5.QtSql import QSqlQueryModel
from PyQt5 import QSql
import sqlite3

#----- LOGIN & REGISTRASI APLIKASI
-----

class login(QWidget):
    def __init__(self):
        super().__init__()
        self.lay=QGridLayout()
        self.setLayout(self.lay)

        #judul
        label=QLabel("LOGIN DATA")
        label.setAlignment(Qt.AlignCenter)
        label.setMaximumHeight(40)
        label.setStyleSheet("background-color:blue; font:bold
20pt; color:gold;")
        self.lay.addWidget(label,0,0)

        #login
        gb=QGroupBox()
        self.lay1=QFormLayout()
        gb.setLayout(self.lay1)
        gb.setStyleSheet("background-color:gold; color:blue;
font:10pt;")
        self.lay.addWidget(gb,1,0)
        lb1=QLabel("Username")
        lb2=QLabel("Password")
        self.dt1=QLineEdit()
        self.dt2=QLineEdit()
        self.dt2.setEchoMode(QLineEdit.Password)
        self.dt1.setPlaceholderText("Masukkan Username")
        self.dt2.setPlaceholderText("Masukkan Password")
```

```

        self.btn1=QPushButton("Login")
        self.label=QLabel()
        self.btn1.setStyleSheet("background-color: blue; font :
bold 10pt; color:gold;")
        self.btn1.clicked.connect(self.klik)
        self.lay1.addRow(lb1,self.dt1)
        self.lay1.addRow(lb2,self.dt2)
        self.lay1.addRow(self.btn1)
        self.lay1.addRow(self.label)

    def klik(self):
        pesan = QMessageBox()
        user=self.dt1.text()
        pw=self.dt2.text()
        if user==" " and pw==" ":
            pesan.setText("Username atau Password tidak boleh
kosong!")
            pesan.exec()
        elif user=="admin" and pw=="admin":
            pesan.setText("Login berhasil!")
            pesan.exec()
            self.dt1.setText("")
            self.dt2.setText("")
            self.utama=tampilan_utama()
            self.utama.show()
        else:
            pesan.setText("Username atau Password Salah!")
            pesan.exec()
            self.label.setText("Username = admin \nPassword =
admin")

#----- TAMPILAN UTAMA APLIKASI -----
-----
class tampilan_utama(QWidget):
    def __init__(self):
        super().__init__()
        self.setGeometry(100,100,700,600)
        self.setWindowTitle("APLIKASI MANAJEMEN BARANG")
        self.setStyleSheet("background-color:black;")

```

```

self.lay=QGridLayout()
self.setLayout(self.lay)

self.konekdb = sqlite3.connect("daftar.db")
self.kon = self.konekdb.cursor()
self.kon.execute("CREATE TABLE IF NOT EXISTS
barang(kode_barang INTEGER PRIMARY KEY AUTOINCREMENT,
tanggal_masuk varchar(10), nama_barang varchar(50), stok_awal
varchar(20), masuk varchar(20), keluar varchar(20), stok_akhir
varchar(20))")
self.kon.close()

#judul
label=QLabel("DATA BARANG MASUK")
label.setAlignment(Qt.AlignCenter)
label.setMaximumHeight(40)
label.setStyleSheet("background-color:blue; font:20pt;
color:gold; font:bold;")
self.lay.addWidget(label,0,0)

#search data
hlay1=QHBoxLayout()
lbl=QLabel("Cari Barang")
self.ln=QLineEdit()
self.btn=QPushButton("Cari")
lbl.setStyleSheet("font:11pt; color:gold; font:bold;")
self.ln.setStyleSheet("font:10pt; color:gold")
self.ln.setPlaceholderText("Masukkan nama barang yang
dicari")
self.btn.setStyleSheet("background-color: blue; font :
bold 10pt; color:gold;")
self.btn.clicked.connect(self.cari)
hlay1.addWidget(lbl)
hlay1.addWidget(self.ln)
hlay1.addWidget(self.btn)
self.lay.addLayout(hlay1,1,0)

#count
hlay4=QHBoxLayout()
self.bt1=QPushButton("hitung jumlah barang masuk perhari")
self.bt2=QPushButton("hitung jumlah seluruh data barang")

```

```

        self.bt1.setStyleSheet("background-color: blue; font :
bold 10pt; color:gold;")
        self.bt2.setStyleSheet("background-color: blue; font :
bold 10pt; color:gold;")
        self.bt1.clicked.connect(self.report2)
        self.bt2.clicked.connect(self.report3)
        hlay4.addWidget(self.bt1)
        hlay4.addWidget(self.bt2)
        self.lay.addLayout(hlay4,2,0)

#table view
self.tv= QTableView()
self.tv.setStyleSheet("background-color:gold; border:1px
solid black;")
self.lay.addWidget(self.tv,3,0)

#button
hlay3=QHBoxLayout()
self.button1=QPushButton("Tambah Data Barang")
self.button2=QPushButton("Hapus Data Barang")
self.button3=QPushButton("Tampilkan Daftar Barang")
self.button1.setStyleSheet("background-color: blue; font :
bold 10pt; color:gold;")
self.button2.setStyleSheet("background-color: blue; font :
bold 10pt; color:gold;")
self.button3.setStyleSheet("background-color: blue; font :
bold 10pt; color:gold;")
hlay3.addWidget(self.button1)
hlay3.addWidget(self.button2)
hlay3.addWidget(self.button3)
self.lay.addLayout(hlay3,4,0)
self.button1.clicked.connect(self.add)
self.button2.clicked.connect(self.hps)
self.button3.clicked.connect(self.open)

def add(self):
    self.insert=tambah()
    self.insert.show()

def hps(self):
    self.delete=hapus()

```

```

        self.delete.show()

    def open(self):
        db = QSql.QSqlDatabase.addDatabase('QSQLITE')
        db.setDatabaseName('daftar.db')
        if not db.open():
            QMessageBox.warning(QMessageBox(), 'Gagal', "Gagal
Menampilkan Daftar Barang!")
            self.button3.setEnabled(False)
        else:
            self.button3.setEnabled(True)
            self.read_data()
            pass

    def read_data(self):
        model = QSql.QSqlTableModel()
        model.setTable('barang')
        model.setEditStrategy(QSql.QSqlTableModel.OnFieldChange)
        model.select()
        self.tv.setModel(model)

    def cari(self):
        model= QSql.QSqlQueryModel()
        cari = self.ln.text()
        model.setQuery("SELECT * FROM barang WHERE nama_barang
LIKE '%{}%'".format(cari))
        self.tv.setModel(model)
        self.tv.show()

    def report2(self):
        model3= QSql.QSqlQueryModel()
        model3.setQuery("SELECT tanggal_masuk, COUNT(*) AS
jumlah_data_masuk FROM barang GROUP BY tanggal_masuk")
        self.tv.setModel(model3)
        self.tv.show()

    def report3(self):
        model4= QSql.QSqlQueryModel()
        model4.setQuery("SELECT COUNT(*) AS jumlah_seluruh_data
FROM barang")
        self.tv.setModel(model4)

```

```

        self.tv.show()

#----- MENAMBAHKAN DATA -----
-----

class tambah(QWidget):
    def __init__(self):
        super().__init__()
        self.setGeometry(100,100,500,500)
        self.setWindowTitle("TAMBAH DATA BARANG")
        self.setStyleSheet("background-color:black;")
        self.lay=QGridLayout()
        self.setLayout(self.lay)

        #judul
        label=QLabel("TAMBAH DATA")
        label.setAlignment(Qt.AlignCenter)
        label.setMaximumHeight(40)
        label.setStyleSheet("background-color:blue;          font:20pt;
color:gold; font:bold;")
        self.lay.addWidget(label,0,0)

        #tambah data
        gb=QGroupBox()
        gb.setStyleSheet("background-color:gold;          color:blue;
font:10pt;")
        self.lay.addWidget(gb,1,0)
        self.lay1=QFormLayout()
        gb.setLayout(self.lay1)
        lb1=QLabel("Kode Barang")
        lb2=QLabel("Tgl.Masuk")
        lb3=QLabel("Nama Barang")
        lb4=QLabel("Stok Awal")
        lb5=QLabel("Masuk")
        lb6=QLabel("Keluar")
        lb7=QLabel("Stok Akhir")
        self.dt1=QLineEdit()
        self.dt2=QDateEdit()
        self.dt3=QLineEdit()
        self.dt4=QLineEdit()
        self.dt5=QLineEdit()

```

```

        self.dt6=QLineEdit()
        self.dt7=QLineEdit()
        self.btn1=QPushButton("Tambah")
        self.btn1.setStyleSheet("background-color: blue; font :
bold 10pt; color:gold;")
        self.btn1.clicked.connect(self.tambah_data)
        self.lay1.addRow(lb1,self.dt1)
        self.lay1.addRow(lb2,self.dt2)
        self.lay1.addRow(lb3,self.dt3)
        self.lay1.addRow(lb4,self.dt4)
        self.lay1.addRow(lb5,self.dt5)
        self.lay1.addRow(lb6,self.dt6)
        self.lay1.addRow(lb7,self.dt7)
        self.lay1.addRow(self.btn1)

def tambah_data(self):
    kode=self.dt1.text()
    tgl=self.dt2.text()
    nama=self.dt3.text()
    awal=self.dt4.text()
    masuk=self.dt5.text()
    keluar=self.dt6.text()
    akhir=self.dt7.text()

    alert=QMessageBox.question(QMessageBox(), 'Question',
"Yakin data sudah benar?",QMessageBox.Yes, QMessageBox.Cancel)
    if alert==QMessageBox.Yes:
        try:
            self.konekdb = sqlite3.connect("daftar.db")
            self.kon = self.konekdb.cursor()
            self.kon.execute("""INSERT INTO
barang(kode_barang,tanggal_masuk,nama_barang,stok_awal,masuk,kelua
r,stok_akhir) values ({},'{}','{}','{}','{}','{}','{}') """).
            format(kode,tgl,nama,awal,masuk,keluar,akhir))
            self.konekdb.commit()
            self.dt1.setText("")
            self.dt3.setText("")
            self.dt4.setText("")
            self.dt5.setText("")
            self.dt6.setText("")
            self.dt7.setText("")

```



```

        QMessageBox.information(QMessageBox(), 'Sukses',
        "Input data berhasil!")
    except:
        QMessageBox.warning(QMessageBox(), 'Gagal', 'Input
data gagal \nkode barang sudah digunakan!')

#----- MENGHAPUS DATA -----
-----

class hapus(QWidget):
    def __init__(self):
        super().__init__()
        self.setGeometry(100,100,400,200)
        self.setWindowTitle("HAPUS DATA BARANG")
        self.setStyleSheet("background-color:black;")
        self.lay=QGridLayout()
        self.setLayout(self.lay)

        #judul
        label=QLabel("HAPUS DATA")
        label.setAlignment(Qt.AlignCenter)
        label.setMaximumHeight(40)
        label.setStyleSheet("background-color:blue;          font:20pt;
color:gold; font:bold;")
        self.lay.addWidget(label,0,0)

        #hapus
        gb2=QGroupBox()
        gb2.setStyleSheet("background-color:gold;          color:blue;
font:10pt;")
        self.lay.addWidget(gb2,1,0)
        lay2=QFormLayout()
        gb2.setLayout(lay2)
        lb1=QLabel("Kode Barang")
        self.ln1=QLineEdit()
        self.btn=QPushButton("Hapus")
        self.ln1.setPlaceholderText("Masukkan kode barang dari
baris yang akan dihapus")
        self.btn.setStyleSheet("background-color:  blue;  font  :
bold 10pt; color:gold;")
        self.btn.clicked.connect(self.hapus_data)

```

```

        lay2.addRow(lb1, self.ln1)
        lay2.addRow(self.btn)

    def hapus_data(self):
        pesan=QMessageBox.question(QMessageBox(), 'Question',
        "Yakin      ingin      menghapus      data      tsb?", QMessageBox.Yes,
        QMessageBox.Cancel)
        if pesan==QMessageBox.Yes:
            hps=self.ln1.text()
            self.konekdb = sqlite3.connect("daftar.db")
            self.kon = self.konekdb.cursor()
            self.kon.execute(f"DELETE      FROM      barang      WHERE
kode_barang={hps}")
            self.konekdb.commit()

if __name__ == "__main__":
    app = QApplication(sys.argv)
    window=login()
    window.setGeometry(100,100,500,300)
    window.setWindowTitle("APLIKASI MANAJEMEN BARANG")
    window.setStyleSheet("background-color:black;")
    window.show()
    sys.exit(app.exec_())

```