

**TUGAS UJIAN TENGAH SEMESTER
PRAKTIK PEMROGRAMAN PYTHON**



Disusun oleh:

Widi Suryo Nugroho

(V3922047)

Dosen:

Yusuf Fadlila Rachman, S. Kom., M. Kom

**PS D-III TEKNIK INFORMATIKA
SEKOLAH VOKASI
UNIVERSITAS SEBELAS MARET
2023**

HASIL DAN PEMBAHASAN

1. Langkah yang pertama yaitu, mengimpor modul sqlite3 ke dalam program Python. Setelah itu pada baris kedua program akan membuat koneksi dengan basis data SQLite yang disimpan di file test.db. Pada baris ketiga, program mencetak pesan "Opened database successfully".

```
In [1]: import sqlite3

conn = sqlite3.connect('test.db')

print("Opened database successfully")
```

Opened database successfully

2. Langkah yang kedua yaitu membuat database. Pada baris pertama, program tersebut mengimpor modul mysql.connector ke dalam program Python. Pada baris kedua hingga keempat, program membuat koneksi dengan basis data MySQL yang terletak di localhost. Parameter user digunakan untuk memberikan nama pengguna yang akan mengakses basis data MySQL. Parameter passwd digunakan untuk memberikan kata sandi pengguna tersebut. Jika kata sandi kosong, maka parameter passwd dapat diisi dengan tanda kutip kosong (""). Pada baris keenam, cursorObject yang akan digunakan untuk mengeksekusi perintah SQL pada basis data MySQL. Pada baris yang terakhir, program mengeksekusi perintah CREATE DATABASE db_sales_V3922047 yang berarti program membuat database yang bernama db_sales_V3922047.

```
In [2]: import mysql.connector

dataBase = mysql.connector.connect(
    host = "localhost",
    user = "root",
    passwd = ""
)

#preparing a cursor
cursorObject = dataBase.cursor()

#createdb
cursorObject.execute("CREATE DATABASE db_sales_V3922047")
```

3. Selanjutnya yaitu membuat tabel di dalam databse. Pada baris kedua hingga kelima, program membuat koneksi dengan basis data MySQL yang terletak di localhost. Parameter user digunakan untuk memberikan nama pengguna yang akan mengakses basis data MySQL. Parameter passwd digunakan untuk memberikan kata sandi pengguna tersebut. Jika kata sandi kosong, maka parameter passwd dapat diisi dengan tanda kutip kosong (""). Parameter database digunakan untuk memberikan nama database yang akan digunakan. Pada baris kesembilan hingga delapan belas, program mengeksekusi perintah SQL CREATE TABLE data_stok_barang. Perintah tersebut akan membuat sebuah tabel baru dengan nama data_stok_barang di dalam databse MySQL. Pada baris kedua puluh, program menutup koneksi dengan basis data MySQL menggunakan metode close().

```
In [3]: import mysql.connector

dataBase = mysql.connector.connect(
    host='localhost',
    user='root',
    passwd='',
    database='db_sales_V3922047'
)

# preparing cursor object
cursorObject = dataBase.cursor()

# creating table
studentRecord = """CREATE TABLE data_stok_barang (
    id_barang VARCHAR(18) NOT NULL PRIMARY KEY,
    nama_barang VARCHAR(255) NOT NULL,
    harga_barang INT,
    stok_awal INT,
    barang_masuk INT,
    barang_keluar INT,
    stok_akhir INT
)"""

# table created
cursorObject.execute(studentRecord)

# disconnect from server
dataBase.close()
```

4. Berikutnya yaitu membuat programnya.

Pilih terlebih dahulu lokasi database yang akan dibuat programnya.

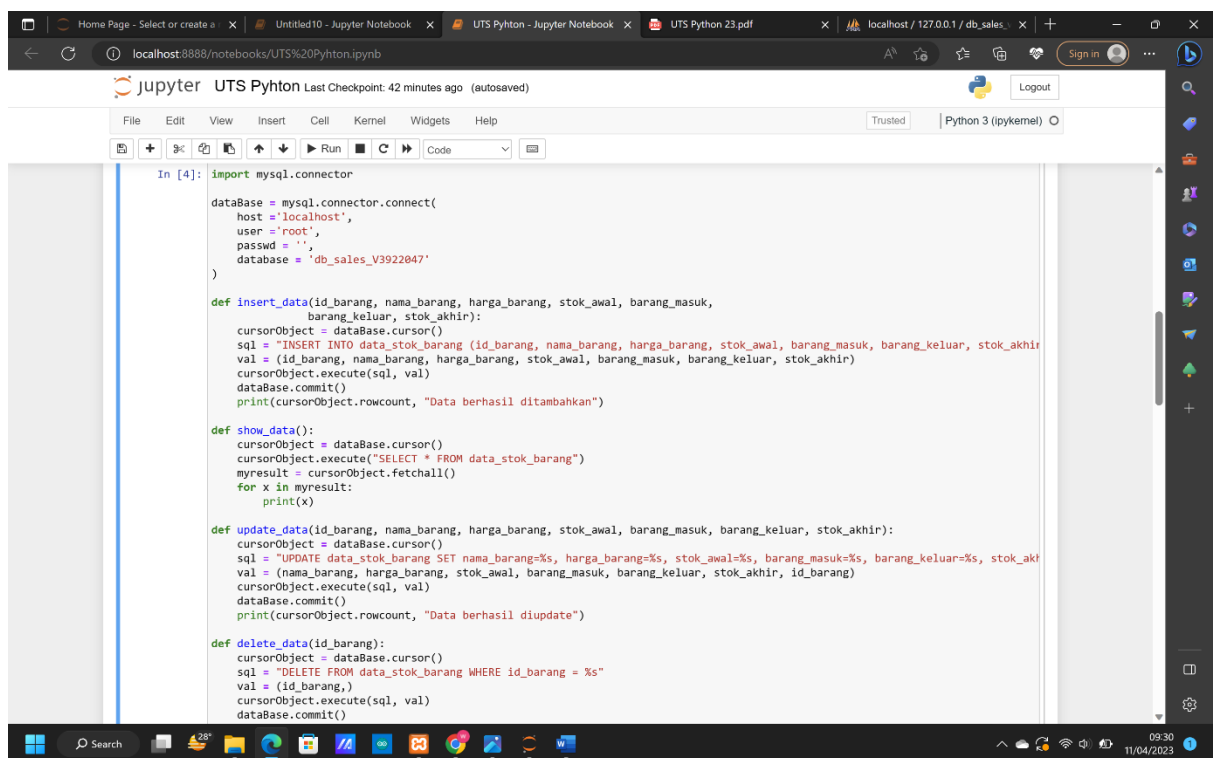
Untuk insert_data() digunakan untuk memasukkan data dalam database. Atribut dari insert_data() yaitu, id_barang, nama_barang, stok_awal, barang_masuk, barang_keluar, dan stok_akhir.

Untuk show_data() digunakan untuk menampilkan data yang telah dimasukkan ke dalam database.

Untuk update_data() digunakan untuk mengedit dan mengupdate salah satu data yang dipilih dalam database.

Untuk delete_data() digunakan untuk menghapus data yang terdapat di dalam database.

Untuk search_data() digunakan untuk mencari data yang terdapat di dalam database.



The screenshot shows a Jupyter Notebook titled "UTS Python" running on a local host. The code defines a MySQL connector and several functions for database operations. The functions are: insert_data, show_data, update_data, and delete_data. The insert_data function takes id_barang, nama_barang, harga_barang, stok_awal, barang_masuk, barang_keluar, and stok_akhir as arguments and inserts them into the data_stok_barang table. The show_data function displays all data from the data_stok_barang table. The update_data function updates the data in the data_stok_barang table based on the id_barang. The delete_data function deletes the data from the data_stok_barang table based on the id_barang.

```
In [4]: import mysql.connector

dataBase = mysql.connector.connect(
    host = 'localhost',
    user = 'root',
    passwd = '',
    database = 'db_sales_V3922847'
)

def insert_data(id_barang, nama_barang, harga_barang, stok_awal, barang_masuk,
    barang_keluar, stok_akhir):
    cursorObject = dataBase.cursor()
    sql = "INSERT INTO data_stok_barang (id_barang, nama_barang, harga_barang, stok_awal, barang_masuk, barang_keluar, stok_akhir"
    val = (id_barang, nama_barang, harga_barang, stok_awal, barang_masuk, barang_keluar, stok_akhir)
    cursorObject.execute(sql, val)
    dataBase.commit()
    print(cursorObject.rowcount, "Data berhasil ditambahkan")

def show_data():
    cursorObject = dataBase.cursor()
    cursorObject.execute("SELECT * FROM data_stok_barang")
    myresult = cursorObject.fetchall()
    for x in myresult:
        print(x)

def update_data(id_barang, nama_barang, harga_barang, stok_awal, barang_masuk, barang_keluar, stok_akhir):
    cursorObject = dataBase.cursor()
    sql = "UPDATE data_stok_barang SET nama_barang=%s, harga_barang=%s, stok_awal=%s, barang_masuk=%s, barang_keluar=%s, stok_akhir=%s"
    val = (nama_barang, harga_barang, stok_awal, barang_masuk, barang_keluar, stok_akhir, id_barang)
    cursorObject.execute(sql, val)
    dataBase.commit()
    print(cursorObject.rowcount, "Data berhasil diupdate")

def delete_data(id_barang):
    cursorObject = dataBase.cursor()
    sql = "DELETE FROM data_stok_barang WHERE id_barang = %s"
    val = (id_barang,)
    cursorObject.execute(sql, val)
    dataBase.commit()
```

Menu 0 digunakan untuk keluar dan sistem selesai.
Menu 1 digunakan untuk melakukan insert data,
Menu 2 digunakan untuk melakukan tampilkan data.
Menu 3 digunakan untuk melakukan update data.
Menu 4 digunakan untuk melakukan hapus data.
Menu 5 digunakan untuk melakukan cari data.

```
localhost:8888/notebooks/UTS%20Python.ipynb
jupyter UTS Python Last Checkpoint: 42 minutes ago (autosaved)
Python 3 (ipykernel)

print(cursorObject.rowcount, "Data berhasil dihapus")

def search_data(keyword):
    cursorObject = DataBase.cursor()
    sql = "SELECT * FROM data_stok_barang WHERE nama_barang LIKE %s"
    val = ("% " + keyword + "%",)
    cursorObject.execute(sql, val)
    myresult = cursorObject.fetchall()
    for x in myresult:
        print(x)

print("=== Aplikasi Database Python ===")
print("1. Insert Data")
print("2. Tampilkan Data")
print("3. Update Data")
print("4. Hapus Data")
print("5. Cari Data")
print("0. Keluar")
print("-----")

menu = input("Pilih Menu: ")

while menu != "0":
    if menu == "1":
        id_barang = input("Masukkan ID Barang: ")
        nama_barang = input("Masukkan Nama Barang: ")
        harga_barang = int(input("Masukkan Harga Barang: "))
        stok_awal = int(input("Masukkan Stok Awal Barang: "))
        barang_masuk = int(input("Masukkan Barang Masuk: "))
        barang_keluar = int(input("Masukkan Barang Keluar: "))
        stok_akhir = stok_awal + barang_masuk - barang_keluar

        insert_data(id_barang, nama_barang, harga_barang, stok_awal, barang_masuk, barang_keluar, stok_akhir)

    elif menu == "2":
        show_data()

    elif menu == "3":
```

```
localhost:8888/notebooks/UTS%20Python.ipynb
jupyter UTS Python Last Checkpoint: 43 minutes ago (autosaved)
Python 3 (ipykernel)

        elif menu == "3":
            id_barang = input("Masukkan ID Barang yang akan diupdate: ")
            nama_barang = input("Masukkan Nama Barang Baru: ")
            harga_barang = int(input("Masukkan Harga Barang Baru: "))
            stok_awal = int(input("Masukkan Stok Awal Barang Baru: "))
            barang_masuk = int(input("Masukkan Barang Masuk Baru: "))
            barang_keluar = int(input("Masukkan Barang Keluar Baru: "))
            stok_akhir = stok_awal + barang_masuk - barang_keluar

            update_data(id_barang, nama_barang, harga_barang, stok_awal, barang_masuk, barang_keluar, stok_akhir)

        elif menu == "4":
            id_barang = input("Masukkan ID Barang yang akan dihapus: ")
            delete_data(id_barang)

        elif menu == "5":
            keyword = input("Masukkan Kata Kunci: ")
            search_data(keyword)

        else:
            print("Menu tidak tersedia")

        print("-----")
        menu = input("Pilih Menu: ")

print("Program selesai")
```

5. Berikut merupakan hasil saat program dijalankan.

Pada saat memilih menu 1, maka akan memasukkan ID barang, nama barang, harga barang, stok awal barang, barang masuk, dan barang keluar.

Pada saat memilih menu 2, maka akan menampilkan data-data yang berada di dalam database.

Pada saat memilih menu 3, maka akan diminta memasukkan ID barang yang akan diupdate, nama barang baru, harga barang baru, stok awal barang baru, barang masuk baru, dan barang keluar baru.

```
=== Aplikasi Database Python ===
1. Insert Data
2. Tampilkan Data
3. Update Data
4. Hapus Data
5. Cari Data
0. Keluar
-----
Pilih Menu: 1
Masukkan ID Barang: 121212
Masukkan Nama Barang: Buku
Masukkan Harga Barang: 15000
Masukkan Stok Awal Barang: 5
Masukkan Barang Masuk: 10
Masukkan Barang Keluar: 3
1 Data berhasil ditambahkan
-----
Pilih Menu: 1
Masukkan ID Barang: 232323
Masukkan Nama Barang: Pensil
Masukkan Harga Barang: 3000
Masukkan Stok Awal Barang: 15
Masukkan Barang Masuk: 10
Masukkan Barang Keluar: 7
1 Data berhasil ditambahkan
-----
Pilih Menu: 2
('121212', 'Buku', 15000, 5, 10, 3, 12)
('232323', 'Pensil', 3000, 15, 10, 7, 18)
-----
Pilih Menu: 3
Masukkan ID Barang yang akan diupdate: 121212
Masukkan Nama Barang Baru: Buku
Masukkan Harga Barang Baru: 17000
Masukkan Stok Awal Barang Baru: 20
Masukkan Barang Masuk Baru: 7
Masukkan Barang Keluar Baru: 3
1 Data berhasil diupdate
-----
```

Pada saat memilih menu 4, maka akan diminta ID barang untuk dihapus dari database.
Pada saat memilih menu 5, maka akan diminta memasukkan kata kunci untuk mencari barang yang dituju.

Pilih Menu: 4

Masukkan ID Barang yang akan dihapus: 232323

1 Data berhasil dihapus

Pilih Menu: 5

Masukkan Kata Kunci: Buku

('121212', 'Buku', 17000, 20, 7, 3, 24)

Pilih Menu: 0

Program selesai