NAMA : RENGGA FITNANDA

NPM : 5230411158

KELAS : IF-C

Soal 1 creat

import sqlite3

koneksi = sqlite3.connect("database\_fauna.db")

# insert data ke tabel

koneksi.execute('''

                CREATE TABLE FAUNA (

                    id\_fauna INTEGER PRIMARY KEY AUTOINCREMENT,

                    nama\_fauna VARCHAR(50),

                    jenis VARCHAR(50),

                    asal VARCHAR(50),

                    jml\_skrng INTEGER(10),

                    thn\_ditemukan INTEGER(10)

                )

               ''')

koneksi.close()

soal no 2 creat

import sqlite3

koneksi = sqlite3.connect("database\_fauna.db")

koneksi.execute('''

                    INSERT INTO FAUNA(nama\_fauna, jenis, asal, jml\_skrng,thn\_ditemukan)

                    VALUES('Harimau Jawa','Mamalia','Jawa','40','2019')

                    ''')

koneksi.execute('''

                    INSERT INTO FAUNA(nama\_fauna, jenis, asal, jml\_skrng,thn\_ditemukan)

                    VALUES('Kuskus Beruang','Mamalia','Sulawesi','30','2021')

                    ''')

koneksi.execute('''

                    INSERT INTO FAUNA(nama\_fauna, jenis, asal, jml\_skrng,thn\_ditemukan)

                    VALUES('Beruang Madu','Mamalia','Sumatera','1000','2020')

                    ''')

koneksi.execute('''

                    INSERT INTO FAUNA(nama\_fauna, jenis, asal, jml\_skrng,thn\_ditemukan)

                    VALUES('Pesut Mahakam','Mamalia','Kalimantan','100','2021')

                    ''')

koneksi.execute('''

                    INSERT INTO FAUNA(nama\_fauna, jenis, asal, jml\_skrng,thn\_ditemukan)

                    VALUES('Burung Maleo','Burung','Sulawesi','7000','2023')

                    ''')

koneksi.execute('''

                    INSERT INTO FAUNA(nama\_fauna, jenis, asal, jml\_skrng,thn\_ditemukan)

                    VALUES('Macan Dahan','Mamalia','Sumatera','400','2020')

                    ''')

koneksi.execute('''

                    INSERT INTO FAUNA(nama\_fauna, jenis, asal, jml\_skrng,thn\_ditemukan)

                    VALUES('Kancil','Mamalia','Jawa','60','2019')

                    ''')

koneksi.execute('''

                    INSERT INTO FAUNA(nama\_fauna, jenis, asal, jml\_skrng,thn\_ditemukan)

                    VALUES('Gajah Kalimantan','Mamalia','Kalimantan','1500','2021')

                    ''')

koneksi.execute('''

                    INSERT INTO FAUNA(nama\_fauna, jenis, asal, jml\_skrng,thn\_ditemukan)

                    VALUES('Elang Jawa','Burung','Jawa','200','2021')

                    ''')

koneksi.execute('''

                    INSERT INTO FAUNA(nama\_fauna, jenis, asal, jml\_skrng,thn\_ditemukan)

                    VALUES('Katak Borneo','Amfibi','Kalimantan','2000','2023')

                    ''')

koneksi.commit()

koneksi.close()

A screenshot of a computer

Description automatically generated

Soal no 3 select.all

import sqlite3

koneksi = sqlite3.connect("database\_fauna.db")

kursor = koneksi.cursor()

#mengambil swmua data dalam tabel dan tampilkan

kursor.execute("SELECT \* FROM FAUNA")

# Tampilkan dalam bentuk baris

baris\_tabel = kursor.fetchall()

# membuat format table dengan method format

print("DATA FAUNA")

print("="\*90)

print("{:<5} {:18} {:10} {:10} {:15} {:>20}".format("ID","Nama\_Fauna","Jenis","Asal","Jumlah Saat Ini","Tahun Terakhir"))

print("-"\*90)

# tampilkan data sesuai format tabel dg perulangan

for baris in baris\_tabel :

    print("{:<5} {:18} {:10} {:10} {:15} {:>20}".format(baris[0],baris[1],baris[2],baris[3],baris[4],baris[5]))

koneksi.close()

A screenshot of a computer

Description automatically generated

Soal no 4 select­­ where

import sqlite3

koneksi = sqlite3.connect("database\_fauna.db")

kursor = koneksi.cursor()

#mengambil swmua data dalam tabel dan tampilkan

kursor.execute("SELECT \* FROM FAUNA WHERE jenis = 'Mamalia' AND jml\_skrng <'1000'")

# Tampilkan dalam bentuk baris

baris\_tabel = kursor.fetchall()

# membuat format table dengan method format

print("DATA FAUNA")

print("="\*100)

print("{:<5} {:18} {:10} {:10} {:15} {:>20}".format("ID","Nama\_Fauna","Jenis","Asal","Jumlah Saat Ini","Tahun Terakhir"))

print("-"\*100)

# tampilkan data sesuai format tabel dg perulangan

for baris in baris\_tabel :

    print("{:<5} {:18} {:10} {:10} {:15} {:>20}".format(baris[0],baris[1],baris[2],baris[3],baris[4],baris[5]))

koneksi.close()

A screenshot of a computer

Description automatically generated

Soal no 5 select where

import sqlite3

koneksi = sqlite3.connect("database\_fauna.db")

kursor = koneksi.cursor()

#mengambil swmua data dalam tabel dan tampilkan

kursor.execute("SELECT \* FROM FAUNA WHERE jenis = 'Mamalia' AND asal= 'Sulawesi' ")

# Tampilkan dalam bentuk baris

baris\_tabel = kursor.fetchall()

# membuat format table dengan method format

print("DATA FAUNA")

print("="\*85)

print("{:<5} {:18} {:10} {:10} {:15} {:>20}".format("ID","Nama\_Fauna","Jenis","Asal","Jumlah Saat Ini","Tahun Terakhir"))

print("-"\*85)

# tampilkan data sesuai format tabel dg perulangan

for baris in baris\_tabel :

    print("{:<5} {:18} {:10} {:10} {:15} {:>20}".format(baris[0],baris[1],baris[2],baris[3],baris[4],baris[5]))

koneksi.close()

A computer screen shot of a message

Description automatically generated

Soal no 6 select\_where

import sqlite3

koneksi = sqlite3.connect("database\_fauna.db")

kursor = koneksi.cursor()

#mengambil swmua data dalam tabel dan tampilkan

kursor.execute("SELECT \* FROM FAUNA WHERE asal = 'Sumatera' OR jml\_skrng =  '>500' ")

# Tampilkan dalam bentuk baris

baris\_tabel = kursor.fetchall()

# membuat format table dengan method format

print("DATA FAUNA")

print("="\*85)

print("{:<5} {:18} {:10} {:10} {:15} {:>20}".format("ID","Nama\_Fauna","Jenis","Asal","Jumlah Saat Ini","Tahun Terakhir"))

print("-"\*85)

# tampilkan data sesuai format tabel dg perulangan

for baris in baris\_tabel :

    print("{:<5} {:18} {:10} {:10} {:15} {:>20}".format(baris[0],baris[1],baris[2],baris[3],baris[4],baris[5]))

koneksi.close()

A screenshot of a computer

Description automatically generated

No 7 selct\_sum

import sqlite3

koneksi = sqlite3.connect("database\_fauna.db")

kursor = koneksi.cursor()

kursor.execute("SELECT SUM (jml\_skrng) FROM FAUNA")

total = kursor.fetchone()[0]

print(f"Total Populasi :{total}")

koneksi.close()

A screenshot of a computer

Description automatically generated

No 8 select\_order

import sqlite3

# Membuat koneksi ke database atau membuat database baru jika belum ada

koneksi = sqlite3.connect('database\_fauna.db')

kursor = koneksi.cursor()

# Menjalankan query SELECT dengan ORDER BY

kursor.execute("SELECT \* FROM FAUNA ORDER BY nama\_fauna ASC") #ASC|DESC

baris\_table = kursor.fetchall()

print("Data Fauna:")

print("==============================================================")

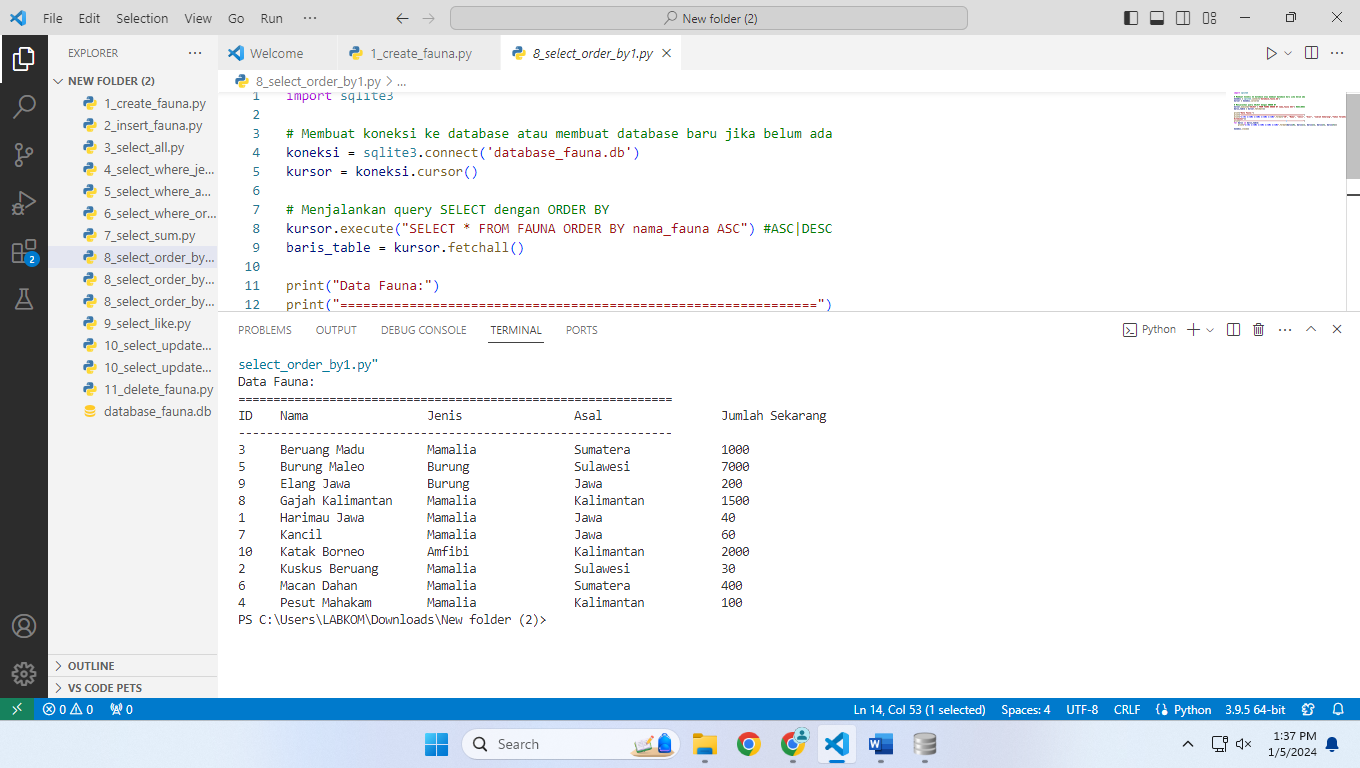
print("{:<5} {:<20} {:<20} {:<20} {:<10}".format("ID", "Nama", "Jenis", "Asal", "Jumlah Sekarang","Tahun Terakhir Ditemukan"))

print("--------------------------------------------------------------")

for baris in baris\_table:

    print("{:<5} {:<20} {:<20} {:<20} {:<10}".format(baris[0], baris[1], baris[2], baris[3], baris[4]))

koneksi.close()



Soal no 8 select\_order

import sqlite3

# Membuat koneksi ke database atau membuat database baru jika belum ada

koneksi = sqlite3.connect('database\_fauna.db')

kursor = koneksi.cursor()

# Menjalankan query SELECT dengan ORDER BY

kursor.execute("SELECT \* FROM FAUNA ORDER BY jml\_skrng DESC") #ASC|DESC

baris\_table = kursor.fetchall()

print("Data Fauna:")

print("==============================================================")

print("{:<5} {:<20} {:<20} {:<20} {:<10}".format("ID", "Nama", "Jenis", "Asal", "Jumlah Sekarang","Tahun Terakhir Ditemukan"))

print("--------------------------------------------------------------")

for baris in baris\_table:

    print("{:<5} {:<20} {:<20} {:<20} {:<10}".format(baris[0], baris[1], baris[2], baris[3], baris[4]))

koneksi.close()

\A computer screen shot of a computer screen

Description automatically generated

Soal no 8 select\_order

import sqlite3

# Membuat koneksi ke database atau membuat database baru jika belum ada

koneksi = sqlite3.connect('database\_fauna.db')

kursor = koneksi.cursor()

# Menjalankan query SELECT dengan ORDER BY

kursor.execute("SELECT \* FROM FAUNA ORDER BY thn\_ditemukan ASC") #ASC|DESC

baris\_table = kursor.fetchall()

print("Data Fauna:")

print("==============================================================================================")

print("{:<5} {:18} {:10} {:10} {:15} {:>20}".format("ID", "Nama", "Jenis", "Asal", "Jumlah Sekarang","Tahun Terakhir"))

print("----------------------------------------------------------------------------------------------")

for baris in baris\_table:

    print("{:<5} {:18} {:10} {:10} {:15} {:>20}".format(baris[0], baris[1], baris[2], baris[3], baris[4],baris[5]))

koneksi.close()

A screenshot of a computer

Description automatically generated

Soal no 9 select\_like

import sqlite3

# Membuat koneksi ke database atau membuat database baru jika belum ada

koneksi = sqlite3.connect('database\_fauna.db')

kursor = koneksi.cursor()

# Menjalankan query SELECT dengan LIKE

nama = 'B%'  # Mencari nama

kursor.execute(f"SELECT \* FROM FAUNA WHERE nama\_fauna LIKE ?", (nama,))

baris\_table = kursor.fetchall()

print("Data Fauna :")

print("==============================================================")

print("{:<5} {:18} {:10} {:10} {:15} {:>20}".format("ID", "Nama", "Jenis", "Asal", "Jumlah Sekarang","Tahun Terakhir"))

print("----------------------------------------------------------------------------------------------")

for baris in baris\_table:

    print("{:<5} {:18} {:10} {:10} {:15} {:>20}".format(baris[0], baris[1], baris[2], baris[3], baris[4],baris[5]))

koneksi.close()

A screenshot of a computer

Description automatically generated

Soal no 10 select\_update

# UPDATE table\_nameA

# SET column1 = value1, column2 = value2, ...

# WHERE condition;

import sqlite3

# Membuat koneksi ke database atau membuat database baru jika belum ada

conn = sqlite3.connect('database\_fauna.db')

cursor = conn.cursor()

# Data yang ingin diubah

id\_fauna =10

jml\_baru = 650

# Menjalankan query UPDATE

cursor.execute(f"UPDATE FAUNA SET jml\_skrng = {jml\_baru} WHERE id\_fauna = {id\_fauna}")

conn.commit()

# Menampilkan pesan setelah update berhasil

if cursor.rowcount > 0:

    print(f"Data fauna dengan ID {id\_fauna} berhasil diupdate.")

else:

    print(f"Tidak ada data fauna dengan ID {id\_fauna}.")

# Menutup koneksi

conn.close()

A computer screen shot of a computer

Description automatically generated

Soal no 10 select\_update

# UPDATE table\_name

# SET column1 = value1, column2 = value2, ...

# WHERE condition;

import sqlite3

# Membuat koneksi ke database atau membuat database baru jika belum ada

conn = sqlite3.connect('database\_fauna.db')

cursor = conn.cursor()

# Data yang ingin diubah

id\_fauna = 4

asal\_baru = "Kalimantan Timur"

# Menjalankan query UPDATE

cursor.execute(f"UPDATE FAUNA SET asal = {asal\_baru} WHERE id\_fauna = {id\_fauna}")

conn.commit()

# Menampilkan pesan setelah update berhasil

if cursor.rowcount > 0:

    print(f"Data fauna dengan ID {id\_fauna} berhasil diupdate.")

else:

    print(f"Tidak ada data fauna dengan ID {id\_fauna}.")

# Menutup koneksi

conn.close()

soal no 11 delete\_fauna

import sqlite3

# Membuat koneksi ke database atau membuat database baru jika belum ada

conn = sqlite3.connect('database\_fauna.db')

cursor = conn.cursor()

# Menjalankan query DELETE

asal = "Kalimantan"  # Ganti dengan kriteria yang sesuai untuk data yang ingin dihapus

cursor.execute("DELETE FROM FAUNA WHERE asal = ?", (asal,))

conn.commit()

# Menampilkan pesan setelah penghapusan berhasil

if cursor.rowcount > 0:

    print(f"Data fauna dengan Asal {asal} berhasil dihapus.")

else:

    print(f"Tidak ada data fauna dengan Asal {asal}.")

# Menutup koneksi

conn.close()

A screenshot of a computer

Description automatically generated