CRUD Operations in Python:

C--> Creates database and insert values into database

R--> Reading data from database

U--> Updating the values already present in the table.

D--> Delete complete row.

```
import sqlite3
#Connecting to the Database
con = sqlite3.connect('database.sqlite3')
#Points to database
cur = con.cursor()


#create the table
cur.execute("""
CREATE TABLE IF NOT EXISTS employees (
    name VARCHAR(50),
    empid VARCHAR(10),
    age INT,
    department VARCHAR(50),
    salary INT
)
""")
con.commit()
# Close the connection
con.close()

print("Table created successfully.")
```

```
Table created successfully.
```

Inserting values into the table:

```
import sqlite3
#Connecting to the Database
con = sqlite3.connect('database.sqlite3')
#Points to database
cur = con.cursor()
# Insert values into the employees table
cur.executemany("""
INSERT INTO employees VALUES (?, ?, ?,?,?)
""", [('John', 'E_ID001', 50000,21,'IT'),
      ("Ramu",'E_ID002',90000,35,'AI'),
      ("Sonu",'E_ID003',60000,27,'CSE')])


# Commit the transaction
con.commit()

# Close the connection
con.close()

print("Data inserted successfully.")
```

```
Data inserted successfully.
```

Reading Data From DataBase: There are many ways to display the contents of database

1. By using for loop
2. By using fetchall() function
3. By using fetchmany() function

```
import sqlite3
# Connect to the SQLite database
con = sqlite3.connect("database.sqlite3")
# Create a cursor object
cur = con.cursor()
# Query to retrieve all rows from the employees table
rows = cur.execute("SELECT * FROM employees")

# Display the data
print("employees Data:")
for row in rows:
    print(row)

# Close the connection
#con.close()
```

```
employees Data:
('John', 'E_ID001', 50000, '21', 'IT')
('Ramu', 'E_ID002', 90000, '35', 'AI')
('Sonu', 'E_ID003', 60000, '27', 'CSE')
```

Double-click (or enter) to edit

```
import sqlite3
# Connect to the SQLite database
con = sqlite3.connect("database.sqlite3")
# Create a cursor object
cur = con.cursor()
# Query to retrieve all rows from the employees table
rows = cur.execute("SELECT * FROM employees")

#By using fetchall
# Fetch all rows from the result set
rows =cur.fetchall()
print(rows,type(rows))

#By using fetchmany
#rows =cur.fetchmany(2)
```

```
[('John', 'E_ID001', 50000, '21', 'IT'), ('Ramu', 'E_ID002', 90000, '35', 'AI'), ('Sonu', 'E_ID003', 60000, '
```

Updating: Updating values already present in the table.

syntax: UPDATE table name SET attribute1 = value, attribute = value2....

where condition:It must be included else all the records int the table will be updated.

```
import sqlite3
# Connect to the database
con = sqlite3.connect('database.sqlite3')
cur = con.cursor()
# Update the salary for employee with empid 'E_ID001'
cur.execute("UPDATE employees SET salary = ? WHERE empid = ?",
            (55000, 'E_ID001'))

# Commit the changes
con.commit()

# Verify the update
cur.execute("SELECT * FROM employees WHERE empid = 'E_ID001'")
```

```
rows = cur.fetchall()
for row in rows:
   print(row)

# Close the connection
con.close()
```

```
('John', 'E_ID001', 50000, '21', 55000)
```

Delete: To delete complete row.

```
import sqlite3
# Connect to the SQLite database
con = sqlite3.connect('database.sqlite3')
# Create a cursor object
cur = con.cursor()

# 1. Perform the DELETE operation
# Deleting the employee with the name 'Sonu'
cur.execute("DELETE FROM employees WHERE name = 'Sonu'")

# Commit the changes
con.commit()

# 2. Verify the deletion by checking if 'Sonu' is still in the table
cur.execute("SELECT * FROM employees WHERE name = 'Sonu'")

deleted_row = cur.fetchone()
if deleted_row is None:
    print("Sonu is no longer in the table.")
else:
    print("Sonu is still in the table.")

# Close the connection
#con.close()
```

```
Sonu is no longer in the table.
```

```
# Query to retrieve all rows from the employees table
cur.execute("SELECT * FROM employees")
# Fetch all rows from the result set
rows = cur.fetchall()
# Display the data
print("employees Data:")
for row in rows:
    print(row)
```

```
employees Data:
('John', 'E_ID001', 50000, '21', 55000)
('Ramu', 'E_ID002', 90000, '35', 'AI')
```

Convert the result of sql query into DataFrame by using pandas.read_sql_query()

```
import sqlite3
import pandas as pd

# Connect to the SQLite database
con = sqlite3.connect('database.sqlite3')

# Query to select all rows from the employees table
query = "SELECT * FROM employees"

# Convert the result of the query into a pandas DataFrame
df = pd.read_sql_query(query, con)

# Display the DataFrame
```

```
print(df)

# Close the connection
con.close()
```

```
    name    empid    age department salary
0  John  E_ID001  50000         21  55000
1  Ramu  E_ID002  90000         35     AI
```

1. connect(): we can connect to database
2. cursor(): cursor to database
3. execute():excecutes any query like creating table,inserting values like.
4. commit(): commit current ransactions.
5. close(): it is used to close the database.
6. rollback(): rollback any changes to the database.