

225229142 SWETHA JENIFER

Question1. Perform CRUD operations on Student Table as outlined in the reference

(<https://medium.com/analytics-vidhya/programming-with-databases-in-python-using-sqlite4cecbef51ab9>
(<https://medium.com/analytics-vidhya/programming-with-databases-in-python-using-sqlite4cecbef51ab9>).

In [1]:

```
import sqlite3
conn = sqlite3.connect('students.db')
cursor = conn.cursor()
cursor.execute("DROP TABLE IF EXISTS STUDENT")
query = """CREATE TABLE STUDENT(
            ID INT PRIMARY KEY NOT NULL,
            NAME CHAR(20) NOT NULL,
            ROLL CHAR(20),
            ADDRESS CHAR(50),
            CLASS CHAR(20) )"""
cursor.execute(query)
conn.commit()
conn.close()
```

In [2]:

```
import sqlite3
conn = sqlite3.connect('students.db')
conn.execute("INSERT INTO STUDENT (ID,NAME,ROLL,ADDRESS,CLASS) "
"VALUES (1, 'jeni', '001', 'Bangalore', 'M.Sc.DS')")
conn.execute("INSERT INTO STUDENT (ID,NAME,ROLL,ADDRESS,CLASS) "
"VALUES (2, 'mini', '002', 'Hyd', 'B.SC.CS')")
conn.execute("INSERT INTO STUDENT (ID,NAME,ROLL,ADDRESS,CLASS) "
"VALUES(3, 'hari','003', 'Delhi','B.Sc.Maths')")
conn.commit()
conn.close()
```

In [3]:

```
import sqlite3
conn = sqlite3.connect('students.db')
cursor = conn.execute("SELECT * from STUDENT")
print(cursor.fetchall())
conn.close()
```

```
[(1, 'jeni', '001', 'Bangalore', 'M.Sc.DS'), (2, 'mini', '002', 'Hyd', 'B.S
C.CS'), (3, 'hari', '003', 'Delhi', 'B.Sc.Maths')]
```

In [4]:

```
import sqlite3
conn = sqlite3.connect('students.db')
conn.execute("UPDATE STUDENT set ROLL = 005 where ID = 1")
conn.commit()
cursor = conn.execute("SELECT * from STUDENT")
print(cursor.fetchall())
conn.close()
```

```
[(1, 'jeni', '5', 'Bangalore', 'M.Sc.DS'), (2, 'mini', '002', 'Hyd', 'B.SC.C
S'), (3, 'hari', '003', 'Delhi', 'B.Sc.Maths')]
```

In [5]:

```
import sqlite3
conn = sqlite3.connect('students.db')
conn.execute("DELETE from STUDENT where ID = 3;")
conn.commit()
cursor = conn.execute("SELECT * from STUDENT")
print(cursor.fetchall())
conn.close()
```

```
[(1, 'jeni', '5', 'Bangalore', 'M.Sc.DS'), (2, 'mini', '002', 'Hyd', 'B.SC.C
S')]
```

Question2. Open the table MyRestaurants.db that you have created for NoSQL course

In []:

```
!pip install cx_Oracle
```

In []:

```
import cx_Oracle
conn=cx_Oracle.connect("c:\users\lenovo\anaconda3\lib\site-packages (8.3.0)")
import cx_Oracle")
sql="select * from myrestaurants"
cursor=conn.cursor()
cursor.execute(sql)
for row in cursor.execute("select * from myrestaurants"):
    print("name",row[0])
    print("foodtype",row[1])
    print("distance",row[2])
    print("lastvisit",row[3])
    print("ilike",row[4])
conn.commit()
conn.close()
```

Question3. Write a SQL query that returns all restaurants in your table MyRestaurants.db.

In []:

```
import cx_Oracle
conn=cx_Oracle.connect("system/admin")
sql="select * from myrestaurants"
cursor=conn.cursor()
cursor.execute(sql)
for row in cursor.execute("select * from myrestaurants"):
    print("Name : ",row[0])
conn.commit()
conn.close()
```

Question4. Write a SQL query that returns the names of restaurants in descending order that makes Chinese foods

In []:

```
import cx_Oracle
conn=cx_Oracle.connect("system/admin")
sql="select * from myrestaurants"
cursor=conn.cursor()
cursor.execute(sql)
for row in cursor.execute("select name,foodtype from myrestaurants where foodtype='chinese'"):
    print("name",row[0])
    print("foodtype",row[1])
    print("\n")
conn.commit()
conn.close()
```