Practical no 6 : Sql implementation by Parth Gawad, Roll no : 62

Connecting to Database

```
import sqlite3

connection = sqlite3.connect('./genericDatabase.db')
cursor = connection.cursor();
```

Creating a Table

```
cursor.execute('''Create table students(
               id Integer primary key autoincrement,
               name text not null,
               age integer not null
               )''')
connection.commit()
```

CRUD Operations

Create(Insertion)

```
def create_student(name, age):
    cursor.execute('''
        INSERT INTO students (name, age)
        VALUES (?, ?)
    ''', (name, age))
    connection.commit()
    print("Record added successfully!")

create_student("Griffith", 20)
create_student("Guts", 22)
```

```
Record added successfully!
Record added successfully!
```

Read (Retrieve)

```
def read_students():
    cursor.execute('SELECT * FROM students')
    rows = cursor.fetchall()
    print("Student Records:")
    for row in rows:
        print(row)
read_students()
```

```
Student Records:
(1, 'Griffith', 20)
(2, 'Guts', 22)
```

```
(3, 'Griffith', 20)
(4, 'Guts', 22)
```

Update

```
def update_student_age(student_id, new_age):
    cursor.execute('''
        update students set age=? where id=?
    ''', (new_age, student_id))
    connection.commit()
    print("Student age updated successfully!")

update_student_age(1, 21)  # Updating Alice's age to 21
read_students()
```

```
Student age updated successfully!
Student Records:
(1, 'Griffith', 21)
(2, 'Guts', 22)
(3, 'Griffith', 20)
(4, 'Guts', 22)
```

Delete

```
def delete_student(student_id):
    cursor.execute('''
          Delete from students where id=?
    ''', (student_id,))
    connection.commit()
    print("Student deleted successfully!")

delete_student(2)  # Deleting Bob's record
read_students()
```

```
Student deleted successfully!
Student Records:
(1, 'Griffith', 21)
(3, 'Griffith', 20)
(4, 'Guts', 22)
```

Closing the database connection

```
connection.close();
```