

YASH KASARE AI&DS 24

✓ Importing

```
import sqlite3
```

✓ Connecting to Database

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

✓ Create Table

```
cursor.execute('''
    CREATE TABLE IF NOT EXISTS 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)
```

✓ 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)
```

▼ 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)
```

▼ Closing the database connection

```
connection.close ()
```