SQLite Database- Student table with at least 5 fields of different data types, perform the following operations: create, insert, show all records, search, delete (based on name of all the student), update:

Code:

activity_main.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="16dp">

    <EditText android:id="@+id/edtName" android:hint="Name"
android:layout_width="match_parent" android:layout_height="wrap_content"/>
    <EditText android:id="@+id/edtAge" android:hint="Age" android:inputType="number"
android:layout_width="match_parent" android:layout_height="wrap_content"/>
    <EditText android:id="@+id/edtGender" android:hint="Gender"
android:layout_width="match_parent" android:layout_height="wrap_content"/>
    <EditText android:id="@+id/edtGrade" android:hint="Grade"
android:inputType="numberDecimal" android:layout_width="match_parent"
android:layout_height="wrap_content"/>

    <Button android:id="@+id/btnInsert" android:text="Insert"
android:layout_width="match_parent" android:layout_height="wrap_content"/>
    <Button android:id="@+id/btnShowAll" android:text="Show All"
android:layout_width="match_parent" android:layout_height="wrap_content"/>

    <EditText android:id="@+id/edtSearch" android:hint="Search by Name"
android:layout_width="match_parent" android:layout_height="wrap_content"/>
    <Button android:id="@+id/btnSearch" android:text="Search"
android:layout_width="match_parent" android:layout_height="wrap_content"/>
    <Button android:id="@+id/btnUpdate" android:text="Update"
android:layout_width="match_parent" android:layout_height="wrap_content"/>
    <Button android:id="@+id/btnDelete" android:text="Delete"
android:layout_width="match_parent" android:layout_height="wrap_content"/>

    <TextView android:id="@+id/txtResults" android:layout_width="match_parent"
android:layout_height="wrap_content" android:padding="8dp"/>

</LinearLayout>
```

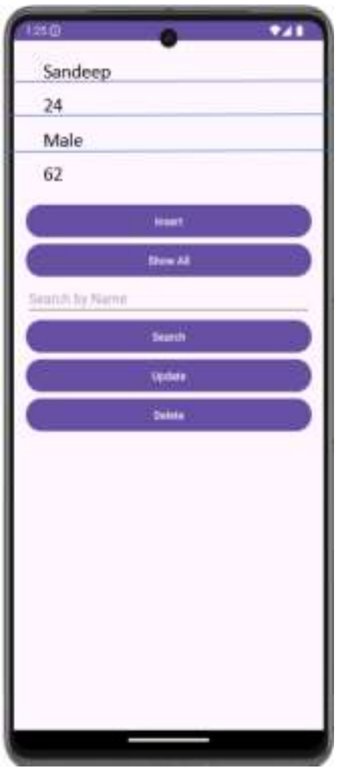
MainActivity.java:

```
package com.example.studentdatabaseapp;

import androidx.appcompat.app.AppCompatActivity;
import android.database.Cursor;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class MainActivity extends AppCompatActivity {

    private DatabaseHelper dbHelper;
    private EditText edtName, edtAge, edtGender, edtGrade, edtSearch;
    private TextView txtResults;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        dbHelper = new DatabaseHelper(this);
        edtName = findViewById(R.id.edtName);
        edtAge = findViewById(R.id.edtAge);
        edtGender = findViewById(R.id.edtGender);
        edtGrade = findViewById(R.id.edtGrade);
        edtSearch = findViewById(R.id.edtSearch);
        txtResults = findViewById(R.id.txtResults);

        // Insert
        findViewById(R.id.btnInsert).setOnClickListener(v -> {
            String name = edtName.getText().toString();
            int age = Integer.parseInt(edtAge.getText().toString());
            String gender = edtGender.getText().toString();
            double grade = Double.parseDouble(edtGrade.getText().toString());

            dbHelper.insertStudent(name, age, gender, grade);
            txtResults.setText("Inserted: " + name);
        });

        // Show All
        findViewById(R.id.btnShowAll).setOnClickListener(v -> showAllStudents());

        // Search
        findViewById(R.id.btnSearch).setOnClickListener(v -> {
            String name = edtSearch.getText().toString();
            searchStudentByName(name);
        });

        // Update
        findViewById(R.id.btnUpdate).setOnClickListener(v -> {
            String oldName = edtSearch.getText().toString();
            String newName = edtName.getText().toString();
            int newAge = Integer.parseInt(edtAge.getText().toString());
            String newGender = edtGender.getText().toString();
            double newGrade = Double.parseDouble(edtGrade.getText().toString());

            dbHelper.updateStudent(oldName, newName, newAge, newGender, newGrade);
            txtResults.setText("Updated: " + oldName + " to " + newName);
        });

        // Delete
        findViewById(R.id.btnDelete).setOnClickListener(v -> {
            String name = edtSearch.getText().toString();
            dbHelper.deleteStudentByName(name);
            txtResults.setText("Deleted: " + name);
        });
    }

    // Show all students
    private void showAllStudents() {
        Cursor cursor = dbHelper.getAllStudents();
        StringBuilder result = new StringBuilder();
        while (cursor.moveToNext()) {
            result.append("ID: ").append(cursor.getInt(0)).append(", ")
                    .append("Name: ").append(cursor.getString(1)).append(", ")
                    .append("Age: ").append(cursor.getInt(2)).append(", ")
                    .append("Gender: ").append(cursor.getString(3)).append(", ")
                    .append("Grade: ").append(cursor.getDouble(4)).append("\n");
        }
        txtResults.setText(result.toString());
        cursor.close();
    }

    // Search student by name
    private void searchStudentByName(String name) {
        Cursor cursor = dbHelper.searchStudentByName(name);
        StringBuilder result = new StringBuilder();
        if (cursor.moveToFirst()) {
            result.append("ID: ").append(cursor.getInt(0)).append(", ")
                    .append("Name: ").append(cursor.getString(1)).append(", ")
                    .append("Age: ").append(cursor.getInt(2)).append(", ")
                    .append("Gender: ").append(cursor.getString(3)).append(", ")
                    .append("Grade: ").append(cursor.getDouble(4)).append("\n");
        } else {
            result.append("No student found with name: ").append(name);
        }
        txtResults.setText(result.toString());
        cursor.close();
    }
}
```

DatabaseHelper.java:

```
package com.example.studentdatabaseapp;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

public class DatabaseHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "StudentDB";
    private static final int DATABASE_VERSION = 1;
    private static final String TABLE_NAME = "Student";

    // Define table columns
    private static final String COLUMN_ID = "id";
    private static final String COLUMN_NAME = "name";
    private static final String COLUMN_AGE = "age";
    private static final String COLUMN_GENDER = "gender";
    private static final String COLUMN_GRADE = "grade";

    public DatabaseHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        String CREATE_TABLE = "CREATE TABLE " + TABLE_NAME + "("
                + COLUMN_ID + " INTEGER PRIMARY KEY AUTOINCREMENT, "
                + COLUMN_NAME + " TEXT, "
                + COLUMN_AGE + " INTEGER, "
                + COLUMN_GENDER + " TEXT, "
                + COLUMN_GRADE + " REAL)";
        db.execSQL(CREATE_TABLE);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
        onCreate(db);
    }

    // Insert a new student
    public void insertStudent(String name, int age, String gender, double grade) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COLUMN_NAME, name);
        values.put(COLUMN_AGE, age);
        values.put(COLUMN_GENDER, gender);
        values.put(COLUMN_GRADE, grade);
        db.insert(TABLE_NAME, null, values);
        db.close();
    }

    // Retrieve all students
    public Cursor getAllStudents() {
        SQLiteDatabase db = this.getReadableDatabase();
        return db.rawQuery("SELECT * FROM " + TABLE_NAME, null);
    }

    // Search for students by name
    public Cursor searchStudentByName(String name) {
        SQLiteDatabase db = this.getReadableDatabase();
        return db.rawQuery("SELECT * FROM " + TABLE_NAME + " WHERE " +
COLUMN_NAME + "=?", new String[]{name});
    }

    // Update student details
    public void updateStudent(String oldName, String newName, int newAge, String newGender,
double newGrade) {
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(COLUMN_NAME, newName);
        values.put(COLUMN_AGE, newAge);
        values.put(COLUMN_GENDER, newGender);
        values.put(COLUMN_GRADE, newGrade);
        db.update(TABLE_NAME, values, COLUMN_NAME + "=?", new String[]{oldName});
        db.close();
    }

    // Delete student by name
    public void deleteStudentByName(String name) {
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete(TABLE_NAME, COLUMN_NAME + "=?", new String[]{name});
        db.close();
    }
}
```

OUTPUT:

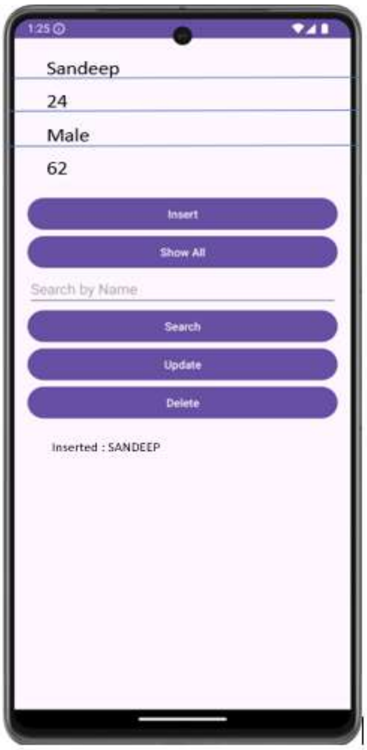

Sandeep

24

Male

62

RECORD INSERTED:


Sandeep

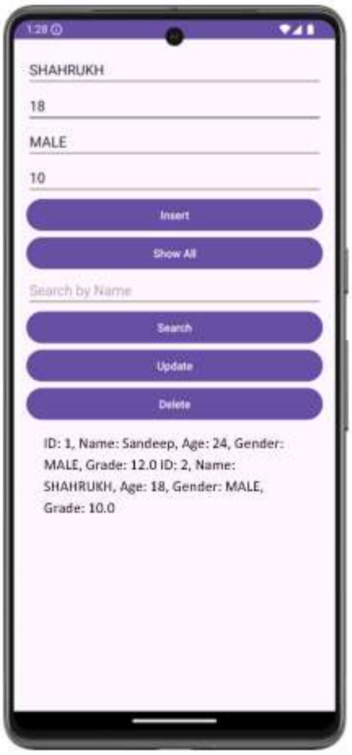
24

Male

62

Inserted : SANDEEP

SHOWS ALL RECORDS:


SHAHRUKH

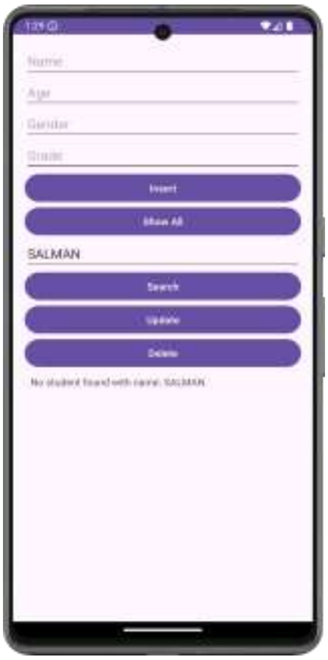
18

MALE

10

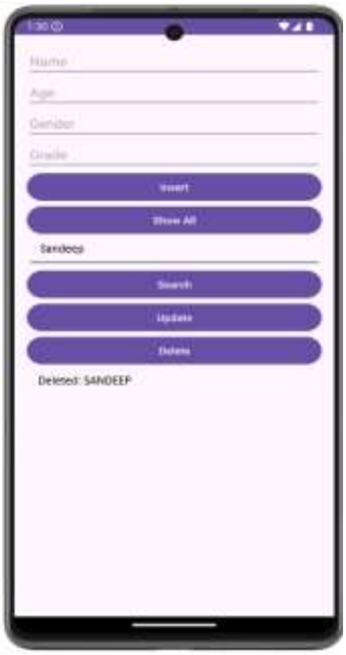
ID: 1, Name: Sandeep, Age: 24, Gender: MALE, Grade: 12.0 ID: 2, Name: SHAHRUKH, Age: 18, Gender: MALE, Grade: 10.0

SEARCHING FOR RECORDS:


SALMAN

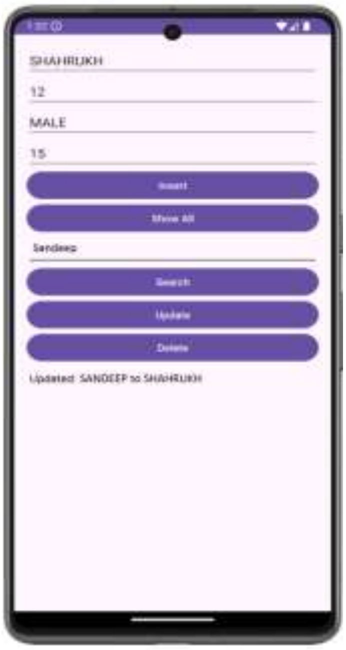
No student found with name: SALMAN

DELETING RECORDS:


Sandeep

Deleted: SANDEEP

UPDATING RECORDS:


SHAHRUKH

12

MALE

15

Sandeep

Updated: SANDEEP to SHAHRUKH