Name:Priyansh Salian

Batch:C31|Roll No:2003148

**EXPERIMENT NO: 4**

**XML FILE:**

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android" xmlns:app="http://schemas.android.com/apk/res-auto" xmlns:tools="http://schemas.android.com/tools" android:layout\_width="match\_parent" android:layout\_height="match\_parent" tools:context=".MainActivity" android:orientation="vertical"

> <EditText

android:id="@+id/student\_id" android:layout\_width="match\_parent" android:layout\_height="0dp" android:layout\_weight="1" android:ems="10" android:inputType="number" android:hint="Student ID"

/>

<EditText

android:id="@+id/student\_name" android:layout\_width="match\_parent" android:layout\_height="0dp" android:layout\_weight="1" android:ems="10"

android:inputType="textPersonName" android:hint="Student Name"

/>

<Button

android:layout\_width="match\_parent" android:layout\_height="0dp" android:layout\_weight="1" android:text="Load All Students" android:onClick="loadStudents"

/>

<TextView android:id="@+id/result" android:layout\_width="match\_parent" android:layout\_height="0dp" android:layout\_weight="1" android:hint="Result" android:textSize="30dp"

/> <Button android:layout\_width="match\_parent" android:layout\_height="0dp" android:layout\_weight="1" android:onClick="addStudent" android:text="ADD" /> <Button android:layout\_width="match\_parent" android:layout\_height="0dp" android:layout\_weight="1" android:onClick="updateStudent" android:text="UPDATE"

/> <Button android:layout\_width="match\_parent" android:layout\_height="0dp" android:layout\_weight="1" android:onClick="deleteStudent" android:text="DELETE By Id"

/>

</LinearLayout>

**MAIN ACTIVITY FILE:**

package com.programtown.example; import android.os.Bundle;

import android.text.method.ScrollingMovementMethod; import android.view.View; import android.widget.EditText; import android.widget.TextView; import androidx.appcompat.app.AppCompatActivity; public class MainActivity extends AppCompatActivity {

TextView resultText;

EditText studentId;

EditText studentName;

MyDBHandler dbHandler; @Override

protected void onCreate(Bundle savedInstanceState) { super.onCreate(savedInstanceState); setContentView(R.layout.activity\_main); resultText = (TextView) findViewById(R.id.result); studentId = (EditText) findViewById(R.id.student\_id); studentName = (EditText) findViewById(R.id.student\_name); resultText.setMovementMethod(new ScrollingMovementMethod()); dbHandler= new MyDBHandler(this);

}

public void loadStudents(View view) { resultText.setText(dbHandler.loadHandler()); studentId.setText(""); studentName.setText("");

}

public void addStudent (View view) {

if(!studentId.getText().toString().isEmpty() && !studentName.getText().toString().isEmpty())

{

int id = Integer.parseInt(studentId.getText().toString());

String name = studentName.getText().toString(); Student student = new Student(id, name); long insertId=dbHandler.addHandler(student); if(insertId==-1){

resultText.setText("Record already exists");

}

else{

studentId.setText(""); studentName.setText(""); resultText.setText("Record added");

}

}

else{

resultText.setText("Please fill correct id and name");

}

}

public void updateStudent(View view) {

if( !studentId.getText().toString().isEmpty() && !studentName.getText().toString().isEmpty())

{

boolean result = dbHandler.updateHandler(Integer.parseInt( studentId.getText().toString()), studentName.getText().toString()); if (result) { studentId.setText(""); studentName.setText(""); resultText.setText("Record Updated");

} else {

resultText.setText("No Record Found");

}

}

else{

resultText.setText("Please fill correct id and name");

}

}

public void deleteStudent(View view) { if(!studentId.getText().toString().isEmpty()) { boolean result = dbHandler.deleteHandler(Integer.parseInt( studentId.getText().toString())); if (result) { studentId.setText(""); studentName.setText(""); resultText.setText("Record Deleted");

} else {

resultText.setText("No Record Found");

}

} else{

resultText.setText("Please fill correct id");

} }

@Override protected void onDestroy() { super.onDestroy(); dbHandler.close();

}

}

# MY DBHANDLER CLASS

package com.programtown.example; import android.content.ContentValues; import android.content.Context; import android.database.Cursor; import android.database.sqlite.SQLiteDatabase; import android.database.sqlite.SQLiteOpenHelper; public class MyDBHandler extends SQLiteOpenHelper { private static final int DATABASE\_VERSION = 1; private static final String DATABASE\_NAME = "studentDB.db"; private static final String TABLE\_STUDENTS = "students"; private static final String COLUMN\_ID = "StudentID"; private static final String COLUMN\_NAME = "StudentName";

MyDBHandler(Context context)

{

super(context, DATABASE\_NAME, null, DATABASE\_VERSION);

}

@Override

public void onCreate(SQLiteDatabase db) {

String CREATE\_STUDENT\_TABLE = "CREATE TABLE " +

TABLE\_STUDENTS + "(" + COLUMN\_ID + " INTEGER PRIMARY KEY," + COLUMN\_NAME

+ " TEXT " + ")";

db.execSQL(CREATE\_STUDENT\_TABLE);

}

@Override

public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {

db.execSQL("DROP TABLE IF EXISTS " + TABLE\_STUDENTS); onCreate(db);

}

String loadHandler() {

String result = "";

String query = "Select\*FROM " + TABLE\_STUDENTS;

SQLiteDatabase db = this.getWritableDatabase(); Cursor cursor = db.rawQuery(query, null); while (cursor.moveToNext()) { int result\_0 = cursor.getInt(0); String result\_1 = cursor.getString(1); result += String.valueOf(result\_0) + " " + result\_1 +

System.getProperty("line.separator");

}

cursor.close(); db.close(); if(result.equals("")) result="No Record Found"; return result;

}

long addHandler(Student student) { long id;

ContentValues values = new ContentValues(); values.put(COLUMN\_ID, student.getID()); values.put(COLUMN\_NAME, student.getStudentName()); SQLiteDatabase db = this.getWritableDatabase(); id = db.insert(TABLE\_STUDENTS, null, values); db.close(); return id;

}

boolean updateHandler(int ID, String name) {

SQLiteDatabase db = this.getWritableDatabase(); ContentValues args = new ContentValues(); args.put(COLUMN\_ID, ID); args.put(COLUMN\_NAME, name);

return db.update(TABLE\_STUDENTS, args, COLUMN\_ID + "=" + ID, null) > 0;

}

boolean deleteHandler(int ID) { boolean result = false;

String query = "Select\*FROM " + TABLE\_STUDENTS + " WHERE " + COLUMN\_ID + " = '" + String.valueOf(ID) + "'";

SQLiteDatabase db = this.getWritableDatabase();

Cursor cursor = db.rawQuery(query, null); Student student = new Student(); if (cursor.moveToFirst()) {

student.setID(Integer.parseInt(cursor.getString(0))); db.delete(TABLE\_STUDENTS, COLUMN\_ID + "=?", new String[] {

String.valueOf(student.getID())

});

cursor.close(); result = true;

}

db.close(); return result;

}

}

**OUTPUT:**



