package com.example.nagbar;  
  
import android.database.Cursor;  
import android.os.Bundle;  
import android.util.Log;  
import android.widget.Toast;  
  
import androidx.activity.EdgeToEdge;  
import androidx.appcompat.app.AppCompatActivity;  
import androidx.core.graphics.Insets;  
import androidx.core.view.ViewCompat;  
import androidx.core.view.WindowInsetsCompat;  
  
public class MainActivity extends AppCompatActivity {  
  
  
 DBClass dbClass;  
 Cursor cursor;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 EdgeToEdge.*enable*(this);  
 setContentView(R.layout.*activity\_main*);  
  
 DBClass dbClass=new DBClass(this);  
 dbClass.addinfo("kanwal","12345");  
 dbClass.addinfo("ahmad","12345");  
 dbClass.addinfo("akram","12345");  
 dbClass.addinfo("saleem","12345");  
 dbClass.addinfo("muraad","12345");  
  
 //curser fatch data  
  
 cursor = dbClass.getAllData();  
  
 if (cursor.getCount() == 0) {  
 Toast.*makeText*(this, "No Data Found", Toast.*LENGTH\_SHORT*).show();  
 } else {  
 while (cursor.moveToNext()) {  
 int id = cursor.getInt(0);  
 String name = cursor.getString(1);  
 String phone = cursor.getString(2);  
  
 Log.*d*("DB\_FETCH", "ID: " + id + ", Name: " + name + ", Phone: " + phone);  
 }  
 }  
  
 boolean isUpdated = dbClass.updateData(4, "New Name", "1234567890");  
  
 if (isUpdated) {  
 Log.*d*("DB\_UPDATE", "Data updated successfully!");  
 } else {  
 Log.*d*("DB\_UPDATE", "Failed to update data.");  
 }  
  
 boolean isDeleted = dbClass.deleteData(1);  
  
 if (isDeleted) {  
 Log.*d*("DB\_DELETE", "Data deleted successfully!");  
 } else {  
 Log.*d*("DB\_DELETE", "Failed to delete data.");  
 }  
  
 cursor.close();  
 dbClass.close();  
  
 }  
  
}

package com.example.nagbar;  
  
import android.content.ContentValues;  
import android.content.Context;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
  
import androidx.annotation.Nullable;  
  
public class DBClass extends SQLiteOpenHelper {  
 private static final String *NAME*="MYDATABASE";  
 private static final int *VERSION*=1;  
 private static final String *TABLE\_NAME*="studentInfo";  
  
 private static final String *KEY\_ID*="Id";  
 private static final String *KEY\_NAME*="Name";  
 private static final String *KEY\_PHONE\_NUMBER*="phoneNumber";  
 public DBClass(Context context) {  
 super(context, *NAME*, null, *VERSION*);  
  
  
 }  
 //SQLiteDatabase database=this.getReadableDatabase();  
  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
  
 String createTableQuery = "CREATE TABLE " + *TABLE\_NAME* + " ( "  
 + *KEY\_ID* + " INTEGER PRIMARY KEY AUTOINCREMENT, "  
 + *KEY\_NAME* + " TEXT, "  
 + *KEY\_PHONE\_NUMBER* + " TEXT)";  
 db.execSQL(createTableQuery);  
 }  
 public void addinfo(String name,String Phonenumber){  
 SQLiteDatabase database=this.getWritableDatabase();  
  
 ContentValues value=new ContentValues();  
 value.put(*KEY\_NAME*,name);  
 value.put(*KEY\_PHONE\_NUMBER*,Phonenumber);  
  
 database.insert(*TABLE\_NAME*,null,value);  
 database.close();  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
 db.execSQL("DROP TABLE IF EXISTS " + *TABLE\_NAME*);  
 onCreate(db);  
  
 }  
 public Cursor getAllData() {  
 SQLiteDatabase database = this.getReadableDatabase();  
 return database.rawQuery("SELECT \* FROM " + *TABLE\_NAME*, null);  
 }  
  
 public boolean updateData(int id, String name, String phoneNumber) {  
 SQLiteDatabase database = this.getWritableDatabase();  
 ContentValues values = new ContentValues();  
  
 values.put(*KEY\_NAME*, name);  
 values.put(*KEY\_PHONE\_NUMBER*, phoneNumber);  
  
 int result = database.update(*TABLE\_NAME*, values, *KEY\_ID* + "=?", new String[]{String.*valueOf*(id)});  
  
 return result > 0; // Returns true if at least one row was updated  
 }  
  
 public boolean deleteData(int id) {  
 SQLiteDatabase database = this.getWritableDatabase();  
 int result = database.delete(*TABLE\_NAME*, *KEY\_ID* + "=?", new String[]{String.*valueOf*(id)});  
  
 return result > 0; // Returns true if at least one row was deleted  
 }  
}