Program 13: Database application using SQLite

Activity main.xml

<?xml version="1.0" encoding="utf-8"?>  
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textAppearance="?android:attr/textAppearanceLarge"  
 android:text="Name"  
 android:id="@+id/textView"  
 android:layout\_alignParentTop="true"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true" />  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textAppearance="?android:attr/textAppearanceLarge"  
 android:text="Surname"  
 android:id="@+id/textView2"  
 android:layout\_below="@+id/editText\_name"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true" />  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textAppearance="?android:attr/textAppearanceLarge"  
  
 android:text="Marks"  
 android:id="@+id/textView3"  
 android:layout\_below="@+id/editText\_surname"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true" />  
 <EditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/editText\_name"  
 android:layout\_alignTop="@+id/textView"  
 android:layout\_toRightOf="@+id/textView"  
 android:layout\_toEndOf="@+id/textView" />  
 <EditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/editText\_surname"  
 android:layout\_alignTop="@+id/textView2"  
 android:layout\_toRightOf="@+id/textView2"  
 android:layout\_toEndOf="@+id/textView2" />  
 <EditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/editText\_Marks"  
 android:layout\_below="@+id/editText\_surname"  
 android:layout\_toRightOf="@+id/textView3"  
 android:layout\_toEndOf="@+id/textView3" />  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
  
 android:text="Add Data"  
 android:id="@+id/button\_add"  
 android:layout\_below="@+id/editText\_Marks"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_marginTop="76dp" />  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="View All"  
 android:id="@+id/button\_viewAll"  
 android:layout\_above="@+id/button\_update"  
 android:layout\_centerHorizontal="true" />  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Update"  
 android:id="@+id/button\_update"  
 android:layout\_below="@+id/button\_add"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true" />  
 <Button  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Delete"  
 android:id="@+id/button\_delete"  
 android:layout\_centerVertical="true"  
 android:layout\_below="@+id/button\_viewAll"  
 android:layout\_alignLeft="@+id/button\_viewAll"  
  
 android:layout\_alignStart="@+id/button\_viewAll" />  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textAppearance="?android:attr/textAppearanceLarge"  
 android:text="id"  
 android:id="@+id/textView\_id"  
 android:layout\_below="@+id/editText\_Marks"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true" />  
 <EditText  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:id="@+id/editText\_id"  
 android:layout\_alignTop="@+id/textView\_id"  
 android:layout\_toRightOf="@+id/textView3"  
 android:layout\_toEndOf="@+id/textView3" />  
</RelativeLayout>

Mainactivity.java

package com.example.pgm13;  
  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
import android.database.Cursor;  
import android.os.Bundle;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
  
public class MainActivity extends AppCompatActivity {  
 DatabaseHelper myDb;  
 EditText editName,editSurname,editMarks ,editTextId;  
 Button btnAddData;  
 Button btnviewAll;  
 Button btnDelete;  
 Button btnviewUpdate;  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 myDb = new DatabaseHelper(this);  
 editName = (EditText)findViewById(R.id.*editText\_name*);  
 editSurname =  
 (EditText)findViewById(R.id.*editText\_surname*);  
  
 editMarks = (EditText)findViewById(R.id.*editText\_Marks*);  
 editTextId = (EditText)findViewById(R.id.*editText\_id*);  
 btnAddData = (Button)findViewById(R.id.*button\_add*);  
 btnviewAll = (Button)findViewById(R.id.*button\_viewAll*);  
 btnviewUpdate= (Button)findViewById(R.id.*button\_update*);  
 btnDelete= (Button)findViewById(R.id.*button\_delete*);  
 AddData();  
 viewAll();  
 UpdateData();  
 DeleteData();  
 }  
 public void DeleteData() {  
 btnDelete.setOnClickListener(  
 new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Integer deletedRows = myDb.deleteData(editTextId.getText().toString());  
 if(deletedRows > 0)  
 Toast.*makeText*(MainActivity.this,"Data Deleted",Toast.*LENGTH\_LONG*).show();  
 else  
 Toast.*makeText*(MainActivity.this,"Data not Deleted",Toast.*LENGTH\_LONG*).show();  
 }  
 }  
 );  
 }  
 public void UpdateData() {  
 btnviewUpdate.setOnClickListener(  
  
 new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 boolean isUpdate = myDb.updateData(editTextId.getText().toString(), editName.getText().toString(), editSurname.getText().toString(),editMarks.getText().toString());  
 if(isUpdate == true)  
 Toast.*makeText*(MainActivity.this,"Data Update",Toast.*LENGTH\_LONG*).show();  
 else  
 Toast.*makeText*(MainActivity.this,"Data not Updated",Toast.*LENGTH\_LONG*).show();  
 }  
 }  
 );  
 }  
 public void AddData() {  
 btnAddData.setOnClickListener(  
 new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 boolean isInserted = myDb.insertData(editName.getText().toString(), editSurname.getText().toString(), editMarks.getText().toString() );  
  
 if(isInserted == true)  
 Toast.*makeText*(MainActivity.this,"Data Inserted",Toast.*LENGTH\_LONG*).show();  
 else  
 Toast.*makeText*(MainActivity.this,"Data not Inserted",Toast.*LENGTH\_LONG*).show();  
 }  
 }  
 );  
 }  
  
 public void viewAll() {  
 btnviewAll.setOnClickListener(  
 new View.OnClickListener() {  
 @Override  
 public void onClick(View v) {  
 Cursor res = myDb.getAllData();  
 if(res.getCount() == 0) {  
// show message  
 showMessage("Error","Nothing found");  
  
 return;  
 }  
  
 StringBuffer buffer = new StringBuffer();  
 while (res.moveToNext()) {  
 buffer.append("Id :"+  
 res.getString(0)+"\n");  
 buffer.append("Name :"+  
 res.getString(1)+"\n");  
 buffer.append("Surname :"+  
 res.getString(2)+"\n");  
 buffer.append("Marks :"+  
 res.getString(3)+"\n\n");  
 }  
  
// Show all data  
 showMessage("Data",buffer.toString());  
 }  
 }  
 );  
 }  
  
 public void showMessage(String title,String Message){  
 AlertDialog.Builder builder = new AlertDialog.Builder(this);  
 builder.setCancelable(true);  
 builder.setTitle(title);  
 builder.setMessage(Message);  
 builder.show();  
 }  
  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
// Inflate the menu; this adds items to the action bar ifit is present.  
//getMenuInflater().inflate(R.menu.menu\_main, menu);  
 return true;  
 }  
  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
  
 int id = item.getItemId();  
  
 return super.onOptionsItemSelected(item);  
 }  
}

Databasehelper.java

package com.example.pgm13;  
  
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 {  
 public static final String *DATABASE\_NAME* = "Student.db";  
 public static final String *TABLE\_NAME* = "student\_table";  
 public static final String *COL\_1* = "ID";  
 public static final String *COL\_2* = "NAME";  
 public static final String *COL\_3* = "SURNAME";  
 public static final String *COL\_4* = "MARKS";  
  
 public DatabaseHelper(Context context) {  
 super(context, *DATABASE\_NAME*, null, 1);  
 }  
  
 @Override  
 public void onCreate(SQLiteDatabase db) {  
 db.execSQL("create table " + *TABLE\_NAME* +" (ID INTEGER PRIMARY KEY AUTOINCREMENT,NAME TEXT,SURNAME TEXT,MARKS INTEGER)");  
 }  
  
 @Override  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int  
 newVersion) {  
 db.execSQL("DROP TABLE IF EXISTS "+*TABLE\_NAME*);  
  
 onCreate(db);  
 }  
  
 public boolean insertData(String name,String surname,String marks) {  
 SQLiteDatabase db = this.getWritableDatabase();  
 ContentValues contentValues = new ContentValues();  
 contentValues.put(*COL\_2*,name);  
 contentValues.put(*COL\_3*,surname);  
 contentValues.put(*COL\_4*,marks);  
 long result = db.insert(*TABLE\_NAME*,null ,contentValues);  
 if(result == -1)  
 return false;  
 else  
 return true;  
 }  
  
 public Cursor getAllData() {  
 SQLiteDatabase db = this.getWritableDatabase();  
 Cursor res = db.rawQuery("select \* from "+*TABLE\_NAME*,null);  
 return res;  
 }  
  
 public boolean updateData(String id,String name,String surname,String  
 marks) {  
 SQLiteDatabase db = this.getWritableDatabase();  
 ContentValues contentValues = new ContentValues();  
 contentValues.put(*COL\_1*,id);  
 contentValues.put(*COL\_2*,name);  
 contentValues.put(*COL\_3*,surname);  
 contentValues.put(*COL\_4*,marks);  
 db.update(*TABLE\_NAME*, contentValues, "ID = ?",new String[]  
 { id });  
  
 return true;  
 }  
  
 public Integer deleteData (String id) {  
 SQLiteDatabase db = this.getWritableDatabase();  
 return db.delete(*TABLE\_NAME*, "ID = ?",new String[] {id});  
 }  
}

Output

