

# 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.pgml3;

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 if it 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

