4 SQLite Database

Introduction

In this lab, we will be learning how to save data to a database for repeating or structured data, such as contact information. This experiment assumes that you are familiar with SQL databases in general and helps you get started with SQLite databases on Android.
For more information about SQL, see the following link.

Objectives

At the end of this lab you will be expected to know.

- How to create your own database using SQLite.
- How to create custom Views from scratch to suit a specific need.
- How to create Confirmation Dialogs.

Users List Application

For this lab, we will be creating a "Students List" application. It is a simple app that allows a user to view and add a list of students names and ages to SQLite database.

Create a new Android Project:

1. In Android Studio, create a new project:
    - If you don't have a project opened, in the **Welcome to Android Studio** window, click **Start a new Android Studio project**.
    - If you have a project opened, select **File > New Project**.
2. In the New Project screen, enter the following values:
    - Application Name: "Student List"
    - Company Domain: "birzeit.edu"
3. Click **Next**.
4. In the **Target Android Devices** screen, keep the default values and click **Next**.
5. In the **Add an Activity to Mobile** screen, select **Empty Activity** and click **Next**.
6. In the **Customize the Activity** screen, keep the default values and click **Next**.

Creating Sqlite Database:

SQLite is a **opensource** SQL database that stores data to a text file on a device. Android comes in with built in SQLite database implementation.
SQLite supports all the relational database features. In order to access this database, you don't need to establish any kind of connections for it like JDBC,ODBC e.t.c.
The following steps show how to create database:

1. Implement the UI of the application using the Linear Layout as shown in figure 1.

```xml
<?xml version="1.0" encoding="utf-8"?>
    <LinearLayout
        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"
        android:orientation="vertical"
        tools:context="edu.birzeit.studentlist.MainActivity">
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <EditText
            android:layout_width="200dp"
            android:layout_height="100dp"
            android:id="@+id/enterName"
            android:background="#89fad4"
            android:text="Enter Name" />
        <EditText
            android:layout_width="200dp"
            android:layout_height="100dp"
            android:id="@+id/enterAge"
            android:background="#5aaef4"
            android:text="Enter Age" />
    </LinearLayout>
        <Button
            android:layout_width="match_parent"
            android:layout_height="100dp"
            android:text="add Studen"
            android:id="@+id/add" />
        <Button
            android:layout_width="match_parent"
            android:layout_height="100dp"
            android:text="show Students"
            android:id="@+id/show" />
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:id="@+id/display"
            android:text="Students Names:"
            android:textColor="#4acd36"
            android:textSize="20dp" />
    </LinearLayout>
```

Figure 1 activity_main.xml file

2. Create the **DBHelper** as java class under **edu.birzeit.studentlist** package that exist in **java** folder. Then do the following:
    - Extend **SQLiteOpenHelper** and import “android.database.sqlite.*” library.
    - Implement two basic Methods: **onCreate** and **onUpgrade** and add constructor.
    - Add the following constant:
        - DATABASE_NAME = "student.db"
        - DATABASE_VERSION = 1
        - TABLE_NAME = "student_name"
        - COLUMN_ID = "id"
        - COLUMN_NAME = "name"
        - COLUMN_AGE = "age”

```
package edu.birzeit.studentlist;
import android.content.Context;
import android.database.sqlite.*;
/*Created by Modallal on 2/3/2017*/
public class DBHelper extends SQLiteOpenHelper {
    public static final String DATABASE_NAME = "student.db";
    private static final int DATABASE_VERSION = 1;
    public static final String TABLE_NAME = "student_name";
    public static final String COLUMN_ID = "id";
    public static final String COLUMN_NAME = "name";
    public static final String COLUMN_AGE = "age";

    public DBHelper(Context context) {
        super(context, DATABASE_NAME , null, DATABASE_VERSION);
    }
    @Override
    public void onCreate(SQLiteDatabase sqLiteDatabase) {
    }
    @Override
    public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {
    }
}
```

- Rewrite **onCreate** and **onUpgrade** methods:

```
@Override
public void onCreate(SQLiteDatabase db) {
    db.execSQL("CREATE TABLE " + TABLE_NAME + "(" + COLUMN_ID + " INTEGER PRIMARY KEY, "+
 COLUMN_NAME + " TEXT, " +COLUMN_AGE+ " INTEGER)");
}
@Override
public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
    db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
    onCreate(db);
}
```

After this you can create your "**student**" database and create "**student_name**" table inside it by making an object of type **DBHelper.**

DBHelper **db**= **new** DBHelper(MainActivity.**this**);

- **Write Insert, Update, Get, and Delete methods inside DBHelper class**

  In this part, basic methods will be implemented inside **DBHelper** class.

  - insertStudent method:

```
public void insertStudent(String name,String age) {
    SQLiteDatabase db = getWritableDatabase();
    ContentValues contentValues = new ContentValues();
    contentValues.put(COLUMN_NAME, name);
    contentValues.put(COLUMN_AGE, age);
    db.insert(TABLE_NAME, null, contentValues);
}
```

  - updateStudent method:

```
public void updateStudent(Integer id, String name,String age) {
    SQLiteDatabase db = this.getWritableDatabase();
    ContentValues contentValues = new ContentValues();
    contentValues.put(COLUMN_NAME, name);
    contentValues.put(COLUMN_AGE, age);
    db.update(TABLE_NAME, contentValues, COLUMN_ID + " = ? ", new String[] {
Integer.toString(id) } );
}
```

- getAllStudents method:

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

- deleteStudent method:

```
public Integer deleteStudent(Integer id) {
    SQLiteDatabase db = this.getWritableDatabase();
    return db.delete(TABLE_NAME,
            COLUMN_ID + " = ? ",
            new String[] { Integer.toString(id) });
}
```

DBHellper.java file:

```
package edu.birzeit.studentlist;
import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.*;
/*Created by Modallal on 2/3/2017*/
public class DBHelper extends SQLiteOpenHelper {
    public static final String DATABASE_NAME = "student.db";
    private static final int DATABASE_VERSION = 1;
    public static final String TABLE_NAME = "student_name";
    public static final String COLUMN_ID = "id";
    public static final String COLUMN_NAME = "name";
    public static final String COLUMN_AGE = "age";
    public DBHelper(Context context) {
        super(context, DATABASE_NAME , null, DATABASE_VERSION);
    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL("CREATE TABLE " + TABLE_NAME + "(" + COLUMN_ID + " INTEGER PRIMARY KEY, " +
COLUMN_NAME + " TEXT, " +COLUMN_AGE+ " INTEGER)");
    }
    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {
        db.execSQL("DROP TABLE IF EXISTS " + TABLE_NAME);
        onCreate(db);}
    public void insertStudent(String name,String age) {
        SQLiteDatabase db = getWritableDatabase();
        ContentValues contentValues = new ContentValues();
        contentValues.put(COLUMN_NAME, name);
        contentValues.put(COLUMN_AGE, age);
        db.insert(TABLE_NAME, null, contentValues);}
    public void updateStudent(Integer id, String name,String age) {
      SQLiteDatabase db = this.getWritableDatabase();
      ContentValues contentValues = new ContentValues();
      contentValues.put(COLUMN_NAME, name);
      contentValues.put(COLUMN_AGE, age);
      db.update(TABLE_NAME, contentValues, COLUMN_ID + " = ? ", new String[] Integer.toString(id)});}
    public Cursor getAllStudents() {
        SQLiteDatabase db = this.getReadableDatabase();
        Cursor res = db.rawQuery( "SELECT * FROM " + TABLE_NAME, null );
        return res;}
    public Integer deleteStudent(Integer id) {
        SQLiteDatabase db = this.getWritableDatabase();
        return db.delete(TABLE_NAME,
                COLUMN_ID + " = ? ",
                new String[] { Integer.toString(id) });
    }
}
```

- **Insert studens names and ages**

Add the following code to **MainActivity.java** file:

```
package edu.birzeit.studentlist;
import android.database.Cursor;
import android.support.v7.app.AppCompatActivity;
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 {
 DBHelper db;
 Cursor c;
 TextView text;
 EditText enterName,enterAge;
 Button add,show;
 @Override
 protected void onCreate(Bundle savedInstanceState) {
  super.onCreate(savedInstanceState);
  setContentView(R.layout.activity_main);
  text=(TextView)findViewById(R.id.display);
  enterName=(EditText)findViewById(R.id.enterName);
  enterAge=(EditText)findViewById(R.id.enterAge);
  add=(Button)findViewById(R.id.add);
  show=(Button)findViewById(R.id.show);
  db=new DBHelper(MainActivity.this);
  add.setOnClickListener(new View.OnClickListener() {
  @Override
  public void onClick(View view) {
   db.insertStudent(enterName.getText().toString(),
   enterAge.getText().toString());
   enterName.setText("Enter Name");
   enterAge.setText("Enter Age");}});
  show.setOnClickListener(new View.OnClickListener() {
  @Override
    public void onClick(View view) {
     String names="Students Name:\n";
     c=db.getAllStudents();
     try {
      while (c.moveToNext()) {
       names=names+c.getString(1)+"    "+c.getString(2)+"\n";
           text.setText(names);
      }
     } finally {
      c.close();
     }
       db.close();
     }
            });
    }
}
```

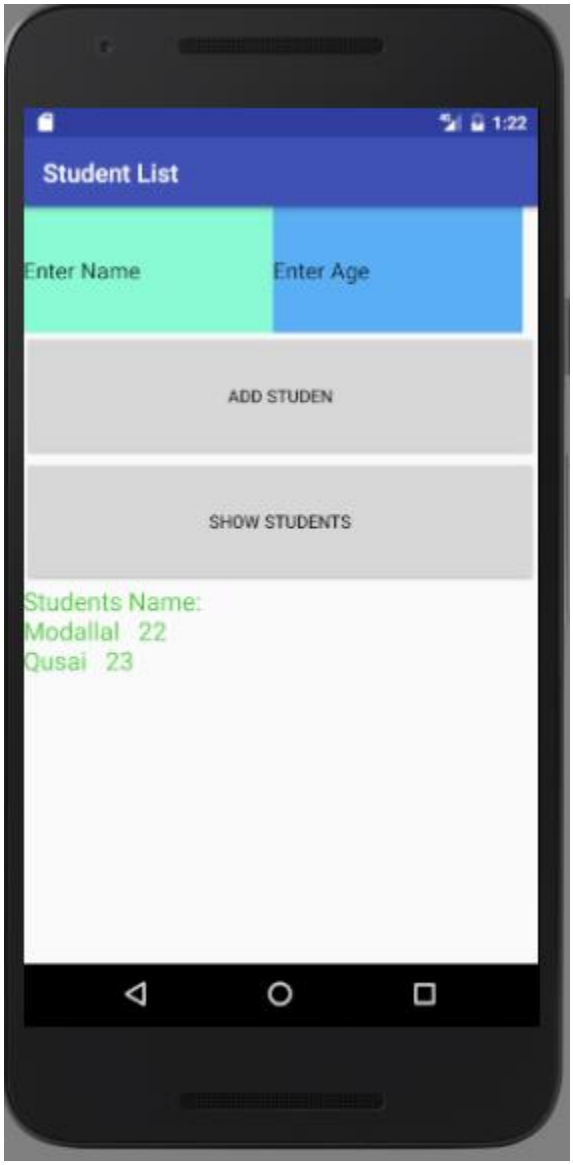

Figure **2**

- Run the application on the emulator (See Figure 2).

Task:

Add button that enable the user to delete all students in the database.

ToDo

This part will be given to you by the teacher assistant in the lab time.