

Nama : Isep Lutpi Nur  
NPM : 2113191079  
Kelas : Informatika / A2  
Mata Kuliah : Mobile Programming  
Semester : 5 (Ganjil)  
Tugas : Minggu 11 Pemrograman Database pada Android Studio

---

## CRUD SQLite Android Aplikasi Data Mahasiswa

SQLite adalah perpustakaan yang prosesnya dilakukan secara mandiri / independent / self-contained / stand-alone, dalam artian memiliki sedikit ketergantungan. Mampu menjalankan transaksi yang mengimplementasikan engine SQL database, tanpa server, tanpa konfigurasi.



### Implementasi SQLite di Android Studio

SQLite mampu berjalan di sistem operasi apapun, bahkan embedded system sekalipun. Seluruh perpustakaan SQLite dienkapsulasi dalam kode sumber tunggal yang tidak memerlukan fasilitas atau tools khusus untuk membangunnya.

SQLite mendukung semua fitur database relasional, sehingga untuk mengakses sebuah database Anda tidak perlu membuat sebuah koneksi seperti halnya saat Anda mengimplementasikan driver JDBC, ODBC, dll.

SQLite bersifat opensource dengan domain public, dengan demikian Anda bebas menggunakannya untuk tujuan apapun baik komersial maupun non-komersial. SQLite telah digunakan oleh banyak aplikasi di seluruh dunia, bahkan tak terhitung lagi, mulai dari proyek level rendah sampai level tinggi.

## Contoh Kasus

Saya akan membuat sebuah aplikasi untuk menampung informasi client berupa identitas KTP seperti NIK, Nama, TTL, Jenis Kelamin, Alamat, Agama, Status, Pekerjaan, dan Kewarganegaraan. Seperti biasa, hal yang perlu dilakukan adalah menerapkan konsep dasar pengelolaan database yaitu CRUD (Create, Read, Update, Delete).

## Cara Android Mengelola Data di Database

1. Mendefinisikan Skema dan Kontrak
2. Membuat Database Menggunakan SQL Helper
3. Memasukkan Informasi ke Database
4. Membaca Informasi dari Database
5. Memperbarui Database
6. Menghapus Informasi dari Database

### 1. Mendefinisikan Skema dan Kontrak

Salah satu prinsip database SQL adalah skema, Penting mendeklarasikan tentang bagaimana membuat database yang terorganisir. Skema ini merefleksikan statement SQL yang akan Anda gunakan untuk membuat database. Anda akan merasakan manfaatnya ketika membuat companion class (kelas pendamping), yang disebut juga contract class, yang secara eksplisit menetapkan layout skema Anda dalam sistem dan terdokumentasikan dengan sendirinya.

Field	Data Type	Key	
-----	-----	-----	
ID	integer	Primary	
firstname	text unique		
lastname	text		

### 2. Membuat Database Menggunakan SQL Helper

Tip: Selalu gunakan short key seperti CTRL+SPACE atau ALT+ENTER untuk mempermudah melakukan debugging saat Anda mulai mengadopsi kode-kode di bawah ini.

#### Langkah 1:

Open Android Studio

#### Langkah 2:

Pilih New Project → buat nama project suka-suka, misalnya "Mahasiswa" → Next.

### Langkah 3:

Pilih Target Device → Check Phone and Tablet → API14: Android 4.0 (IceCreamSandwich) → Next.

### Langkah 4:

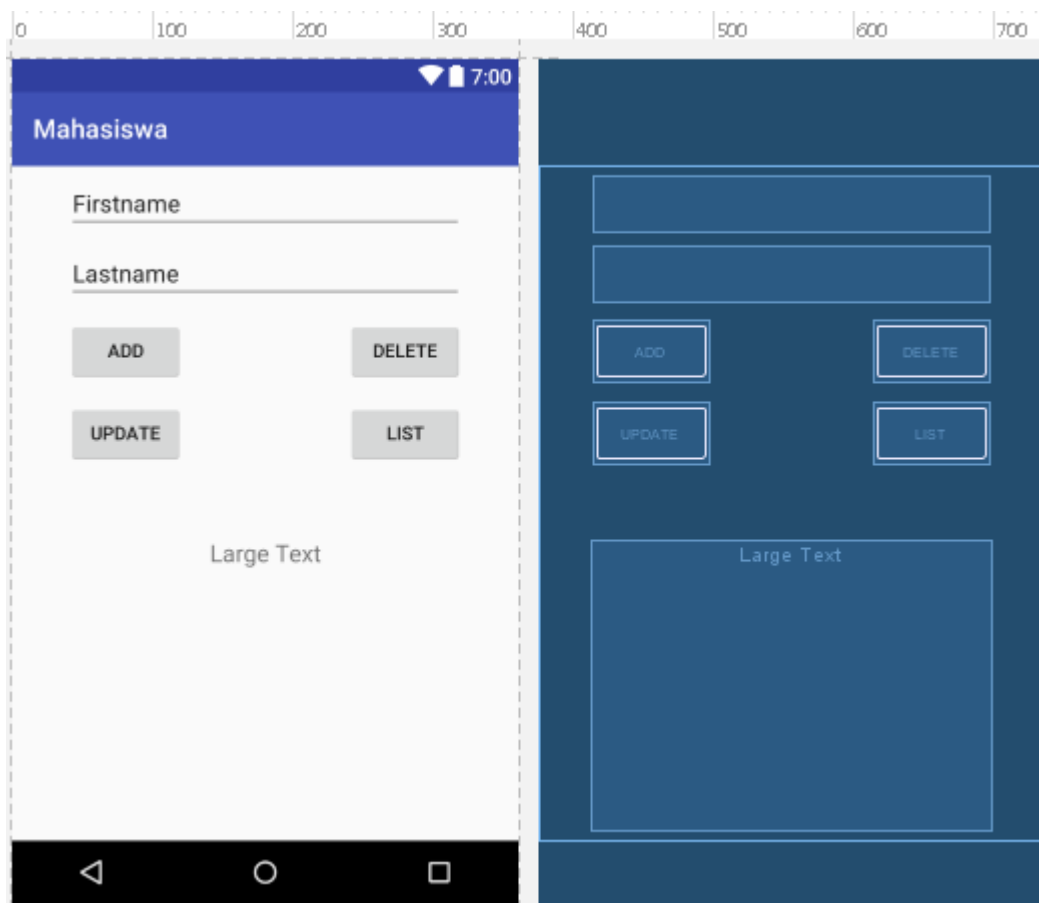
Pilih Empty Activity, sebagai layout utama → Sesuaikan nama Activity & Layout / (default) → Finish.

### Langkah 5:

Pilih tabulasi activity\_main.xml → mode Design.

### Langkah 6:

Drag & Drop Plain Text, TextView dan 4 Button ke Layout → sesuaikan posisinya seperti gambar di bawah ini. Kemudian atur juga masing-masing propertinya baik untuk ID, Text, dan Sizenya.



*Layout Utama Demo SQLite Android*

Atau jika Anda tidak suka dengan mode grafis(Design), Anda bisa gunakan mode Text dan sesuaikan kodenya seperti berikut,

## activity\_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout
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="com.gatewan.mahasiswa.MainActivity">

    <EditText
        android:id="@+id/firstname_input"
        android:layout_width="300dp"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        android:layout_marginLeft="8dp"
        android:layout_marginRight="8dp"
        android:layout_marginTop="8dp"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Firstname"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintHorizontal_bias="0.503"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.0" />

    <EditText
        android:id="@+id/lastname_input"
        android:layout_width="300dp"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        android:layout_marginLeft="8dp"
        android:layout_marginRight="8dp"
        android:layout_marginTop="8dp"
        android:ems="10"
        android:inputType="textPersonName"
        android:text="Lastname"
        app:layout_constraintBottom_toBottomOf="parent"
        app:layout_constraintHorizontal_bias="0.503"
        app:layout_constraintLeft_toLeftOf="parent"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent"
        app:layout_constraintVertical_bias="0.117" />

    <Button
        android:id="@+id/btn_delete"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginBottom="8dp"
        android:layout_marginLeft="8dp"
        android:layout_marginRight="8dp"
        android:layout_marginTop="8dp"
        android:onClick="btn_click"
        android:text="Delete"
```

```
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintHorizontal_bias="0.878"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.243" />
```

<Button

```
android:id="@+id/btn_add"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginBottom="8dp"
android:layout_marginLeft="8dp"
android:layout_marginRight="8dp"
android:layout_marginTop="8dp"
android:onClick="btn_click"
android:text="Add"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintHorizontal_bias="0.121"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.243" />
```

<Button

```
android:id="@+id/btn_update"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginBottom="8dp"
android:layout_marginLeft="8dp"
android:layout_marginRight="8dp"
android:layout_marginTop="8dp"
android:onClick="btn_click"
android:text="Update"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintHorizontal_bias="0.121"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.382" />
```

<Button

```
android:id="@+id/btn_list"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginBottom="8dp"
android:layout_marginLeft="8dp"
android:layout_marginRight="8dp"
android:layout_marginTop="8dp"
android:onClick="btn_click"
android:text="List"
app:layout_constraintBottom_toBottomOf="parent"
app:layout_constraintHorizontal_bias="0.878"
app:layout_constraintLeft_toLeftOf="parent"
app:layout_constraintRight_toRightOf="parent"
app:layout_constraintTop_toTopOf="parent"
app:layout_constraintVertical_bias="0.382" />
```

```

<TextView
    android:id="@+id/textView"
    android:layout_width="304dp"
    android:layout_height="220dp"
    android:layout_marginBottom="8dp"
    android:layout_marginLeft="8dp"
    android:layout_marginRight="8dp"
    android:layout_marginTop="8dp"
    android:text="Large Text"
    android:textAlignment="center"
    android:textSize="18sp"
    app:layout_constraintBottom_toBottomOf="parent"
    app:layout_constraintLeft_toLeftOf="parent"
    app:layout_constraintRight_toRightOf="parent"
    app:layout_constraintTop_toTopOf="parent"
    app:layout_constraintVertical_bias="1.0"
    tools:layout_editor_absoluteY="283dp" />
</android.support.constraint.ConstraintLayout>

```

## Langkah 7

Buka folder app → java → klik kanan folder (dimana MainActivity berada) → new → Java Class, beri nama "DB\_Controller", kemudian sesuaikan kodenya sebagaimana berikut, (jika Anda menggunakan nama project, package, class, dan variabel yang berbeda, maka silakan disesuaikan sendiri dengan apa yang sudah Anda perbuat).

DB\_Controller.java

```

package com.gatewan.mahasiswa;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;
import android.widget.TextView;

/**
 * Created by WawanBeneran on 6/12/2017.
 * NIM.12131294
 */

public class DB_Controller extends SQLiteOpenHelper {
    public DB_Controller(Context context, String name, SQLiteDatabase.CursorFactory
factory, int version) {
        super(context, "TEST.DB", factory, version);
    }

    @Override
    public void onCreate(SQLiteDatabase sqLiteDatabase) {
        sqLiteDatabase.execSQL("CREATE TABLE STUDENTS(ID INTEGER PRIMARY KEY
AUTOINCREMENT, FIRSTNAME TEXT UNIQUE, LASTNAME TEXT);");
    }

    @Override
    public void onUpgrade(SQLiteDatabase sqLiteDatabase, int i, int i1) {
        sqLiteDatabase.execSQL("DROP TABLE IF EXISTS STUDENTS;");
    }
}

```

```

        onCreate(sqliteDatabase);
    }

    public void insert_student(String firstname, String lastname){
        ContentValues contentValues = new ContentValues();
        contentValues.put("FIRSTNAME", firstname);
        contentValues.put("LASTNAME", lastname);
        this.getWritableDatabase().insertOrThrow("STUDENTS", "", contentValues);
    }

    public void delete_student(String firstname){
        this.getWritableDatabase().delete("STUDENTS", "FIRSTNAME='"+firstname+"'", null);
    }

    public void update_student(String old_firstname, String new_firstname){
        this.getWritableDatabase().execSQL("UPDATE STUDENTS SET
FIRSTNAME='"+new_firstname+"' WHERE FIRSTNAME='"+old_firstname+"'");
    }

    public void list_all_students(Textview textView){
        Cursor cursor = this.getReadableDatabase().rawQuery("SELECT * FROM STUDENTS",
null);
        textView.setText("");
        while (cursor.moveToNext()){
            textView.append(cursor.getString(1)+" "+cursor.getString(2)+"\n");
        }
    }
}

```

## Langkah 8

Sesuaikan kode MainActivity.java sebagaimana berikut, (jika Anda menggunakan nama project, package, class, dan variabel yang berbeda, maka silakan disesuaikan sendiri dengan apa yang sudah Anda perbuat).

### MainActivity.java

```

package com.gatewan.mahasiswa;

import android.content.DialogInterface;
import android.database.sqlite.SQLiteException;
import android.os.Bundle;
import android.support.v7.app.AlertDialog;
import android.support.v7.app.AppCompatActivity;
import android.view.View;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

public class MainActivity extends AppCompatActivity {
    EditText firstname, lastname;
    TextView textView;
    DB_Controller controller;

    @Override
    protected void onCreate(Bundle savedInstanceState) {

```

```

        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        firstname = (EditText)findViewById(R.id.firstname_input);
        lastname = (EditText)findViewById(R.id.lastname_input);
        textView = (TextView)findViewById(R.id.textView);

        controller = new DB_Controller(this,"",null,1);
    }

    public void btn_click(View view){
        switch(view.getId()){
            case R.id.btn_add:
                try {
                    controller.insert_student(firstname.getText().toString(),lastname.getText().toString());
                }catch (SQLException e){
                    Toast.makeText(MainActivity.this, "ALREADY EXIST",
                        Toast.LENGTH_SHORT).show();
                }

                break;
            case R.id.btn_delete:
                controller.delete_student(firstname.getText().toString());
                break;
            case R.id.btn_update:
                AlertDialog.Builder dialog = new AlertDialog.Builder(MainActivity.this);
                dialog.setTitle("ENTER NEW FIRSTNAME ");

                final EditText new_firstname = new EditText(this);
                dialog.setView(new_firstname);

                dialog.setPositiveButton("OK", new DialogInterface.OnClickListener(){
                    @Override
                    public void onClick(DialogInterface dialogInterface, int i){
                        controller.update_student(firstname.getText().toString(),new_firstname.getText().toString());
                    }
                });
                dialog.show();
                break;
            case R.id.btn_list:
                controller.list_all_students(textView);
                break;
        }
    }
}

```

## Langkah 9

Jalankan project Anda menggunakan AVD