

BAB VII

SQLITE DATABASE

7.1. Tujuan Pembelajaran

- a. Mahasiswa dapat mengerti apa yang dimaksud *SQLite*
- b. Mahasiswa dapat mengetahui konsep *SQLite*
- c. Mahasiswa dapat menerapkan *SQLite* kedalam Aplikasi

7.2. Software yang dibutuhkan

- a. Java JDK
- b. Android Studio 4.2
- c. SDK API 26

7.3. SQLite

SQLite adalah sebuah engine database SQL yang langsung tertanam atau pada aplikasi. Tidak seperti kebanyakan database SQL lainnya, SQLite tidak memiliki server yang terpisah dari aplikasi. SQLite membaca dan menulis langsung ke file disk biasa. Database SQLite memiliki fitur lengkap dengan banyak tabel, indexes, trigger, dan tampilan, serta tersimpan pada satu file tunggal dalam hard-disk. Format file databasenya bersifat crossplatform. Sehingga kita dapat dengan bebas menyalin database antara sistem 32-bit dan 64bit atau antara arsitektur yang berbeda platform. Fitur-fitur ini membuat SQLite menjadi pilihan populer sebagai Application File Format.

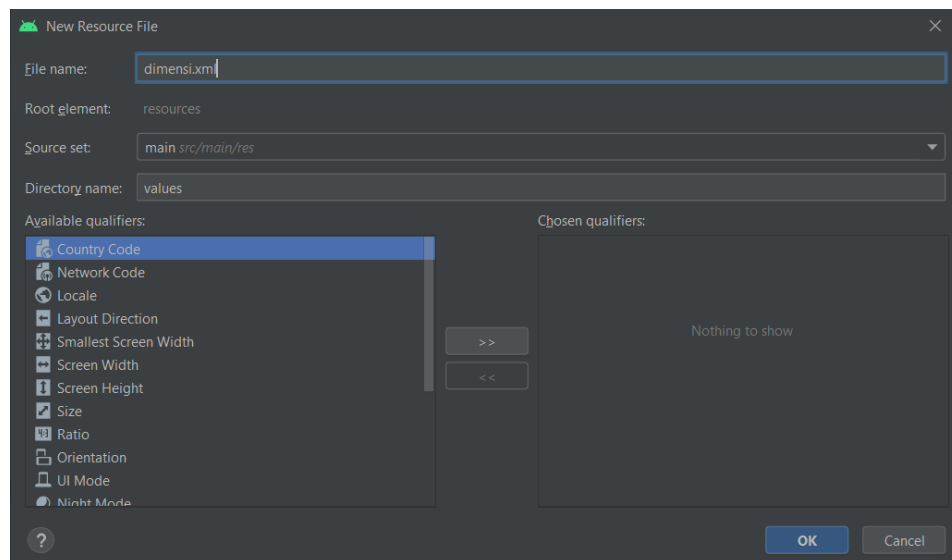
SQLite telah didukung oleh banyak bahasa pemrograman seperti C, C ++, BASIC, C#, Python, Java dan Delphi. Dukungan dari COM (ActiveX) membuat SQLite dapat diakses ke bahasa scripted di Windows seperti VB Script dan JavaScript, sehingga akan menambah kemampuan untuk aplikasi HTML. Dukungan terhadap database ini juga tersedia di sistem operasi embedded seperti iOS, Android, Symbian OS, Maemo, Blackberry dan WebOS karena ukurannya yang kecil dan mudah digunakan. Berikut ini beberapa keunggulan SQLite :

1. SQLite tidak memerlukan proses atau sistem server yang terpisah untuk beroperasi (Serverless).
2. SQLite hadir dengan zero-configuration, yang berarti tidak ada setup atau administrasi yang dibutuhkan.
3. Database SQLite yang lengkap disimpan dalam file tunggal yang tersimpan dalam disk serta bersifat cross-platform.

4. SQLite sangat kecil dan ringan, kurang dari 400KiB untuk konfigurasi lengkap atau kurang dari 250KiB dengan fitur opsional dihilangkan.
5. SQLite bersifat mandiri, yang berarti tidak ada dependensi eksternal.
6. Transaksi SQLite sepenuhnya sesuai dengan ACID, memungkinkan akses yang aman dari banyak proses.
7. SQLite mendukung sebagian besar fitur bahasa query yang ditemukan dalam standar SQL92 (SQL2).
8. SQLite ditulis dalam ANSI-C dan menyediakan API yang sederhana dan mudah digunakan.
9. SQLite tersedia di semua sistem operasi baik ini UNIX (Linux, Mac OS-X, Android, iOS) dan Windows (Win32, WinCE, WinRT).

7.4. Langkah – langkah praktikum

1. Buatlah project baru dengan “Empty Activity” sebagai starter seperti pada praktikum sebelumnya
2. Buatlah resource pada folder values bernama dimensi.xml dengan cara res >> values >> Klik kanan >> New Resource Values



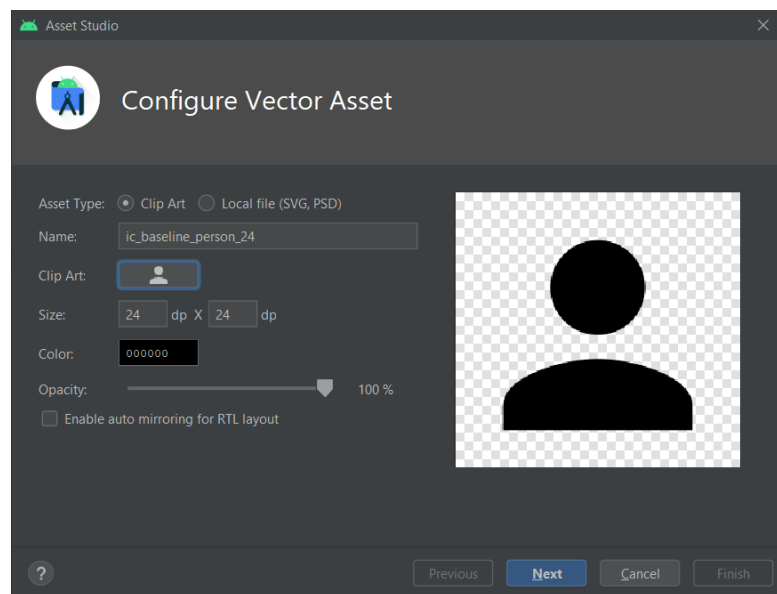
3. Isilah source code pada dimensi.xml seperti pada dibawah :

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <dimen name="activity_horizontal_margin">16dp</dimen>
    <dimen name="activity_vertical_margin">16dp</dimen>
</resources>
```

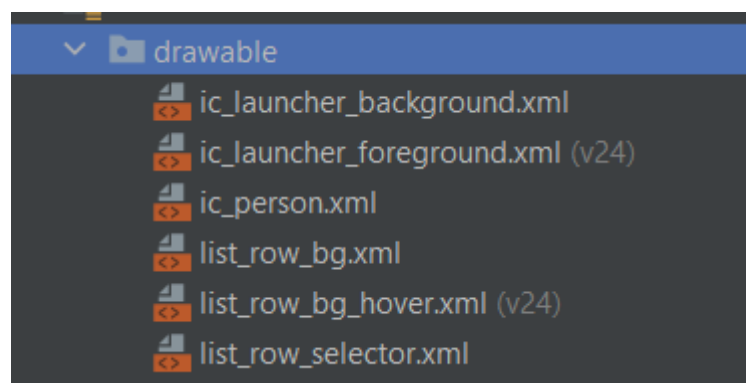
4. Buka file colors.xml pada res >> values >> colors.xml lalu tambahkan beberapa baris kode dibawah :

```
//tambah warna
<color name="list_row_start_color">#FFBB86FC</color>
<color name="list_row_end_color">#FF6200EE</color>
<color name="list_row_hover_start_color">#FF6200EE</color>
<color name="list_row_hover_end_color">#FFBB86FC</color>
```

5. Buatlah vector assets pada drawable dengan cara klik kanan pada folder drawable lalu pilih New >> Vector Assets
6. Carilah clip art yang sesuai seperti pada gambar lalu simpan dengan nama ic_person



7. Klik kanan pada folder drawable lalu pilih New >> Drawable Resource File lalu buatlah 3 file dengan nama masing-masing adalah list_row_bg, list_row_bg_hover, dan list_row_selector



8. Isikan masing source code dibawah pada masing-masing file

list_row_bg.xml

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle">
    <gradient
        android:startColor="@color/list_row_start_color"
        android:endColor="@color/list_row_end_color"
        android:angle="270"/>
    </shape>
```

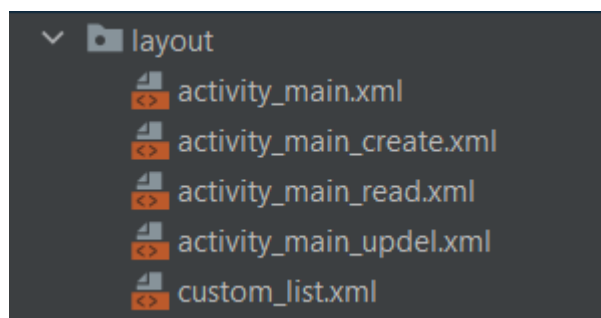
list_row_bg_hover.xml

```
<?xml version="1.0" encoding="utf-8"?>
<shape xmlns:android="http://schemas.android.com/apk/res/android"
    android:shape="rectangle">
    <gradient
        android:angle="270"
        android:endColor="@color/list_row_hover_end_color"
        android:startColor="@color/list_row_hover_start_color"/>
    </shape>
```

list_row_selector.xml

```
<?xml version="1.0" encoding="utf-8"?>
<selector
    xmlns:android="http://schemas.android.com/apk/res/android">
    <item android:drawable="@drawable/list_row_bg"
        android:state_pressed="false" android:state_selected="false"/>
    <item android:drawable="@drawable/list_row_bg_hover"
        android:state_pressed="true"/>
    <item android:drawable="@drawable/list_row_bg_hover"
        android:state_pressed="false" android:state_selected="true"/>
</selector>
```

9. Buatlah beberapa layout resource file dengan cara klik kanan pada file layout >> New >> Layout Resource File lalu buatlah beberapa file sehingga memiliki file seperti berikut



10. Isikan source code pada tiap-tiap resource

activity_main.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
        android:orientation="vertical"
        xmlns:tools="http://schemas.android.com/tools"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:paddingBottom="@dimen/activity_vertical_margin"
        android:paddingLeft="@dimen/activity_horizontal_margin"
        android:paddingRight="@dimen/activity_horizontal_margin"
        android:paddingTop="@dimen/activity_vertical_margin"
        android:gravity="center"
        tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello Teman - Teman!" />

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="create"
        android:onClick="create"
        android:layout_marginTop="20dp"/>

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="read"
        android:onClick="read"
        android:layout_marginTop="10dp"/>

</LinearLayout>
```

activity_main_create.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:id="@+id/activity_main"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:paddingBottom="@dimen/activity_vertical_margin"
        android:paddingLeft="@dimen/activity_horizontal_margin"
        android:paddingRight="@dimen/activity_horizontal_margin"
        android:paddingTop="@dimen/activity_vertical_margin"
        android:gravity="center"
        android:orientation="vertical"
        tools:context=".MainCreate">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Hello welcome to Create" />

</LinearLayout>
```

```

        <EditText
            android:id="@+id/create_nama"
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginTop="20dp"
            android:hint="Nama" />

        <EditText
            android:id="@+id/create_kelas"
            android:hint="Kelas"
            android:layout_width="match_parent"
            android:layout_height="wrap_content" />

        <Button
            android:id="@+id/create_btn"
            android:layout_marginTop="20dp"
            android:text="CREATE"
            android:layout_width="match_parent"
            android:layout_height="wrap_content" />

    </LinearLayout>

```

activity_main_read.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:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:orientation="vertical"
    tools:context=".MainRead">

    <ListView
        android:id="@+id/list_mahasiswa"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>

</LinearLayout>

```

activity_main_updel.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:id="@+id/activity_main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    android:gravity="center"
    android:orientation="vertical"
    tools:context=".MainUpdel">

```

```

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Hello welcome to Update and Delete" />

<EditText
    android:id="@+id/updel_nama"
    android:layout_marginTop="20dp"
    android:hint="Nama"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

<EditText
    android:id="@+id/updel_kelas"
    android:hint="Kelas"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

<Button
    android:id="@+id/btn_up"
    android:layout_marginTop="20dp"
    android:text="UPDATE"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

<Button
    android:id="@+id/btn_del"
    android:text="DELETE"
    android:layout_width="match_parent"
    android:layout_height="wrap_content" />

</LinearLayout>

```

custom_list.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    android:orientation="horizontal"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:background="@drawable/list_row_selector"
    android:padding="5dp">

    <LinearLayout
        android:layout_weight="2"
        android:layout_width="0dp"
        android:layout_height="wrap_content">
        <ImageView
            android:layout_width="80dp"
            android:layout_height="80dp"
            android:id="@+id/iconid"
            android:src="@drawable/ic_person"/>
    </LinearLayout>

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_gravity="center"
        android:layout_weight="5"
        android:orientation="vertical">

```

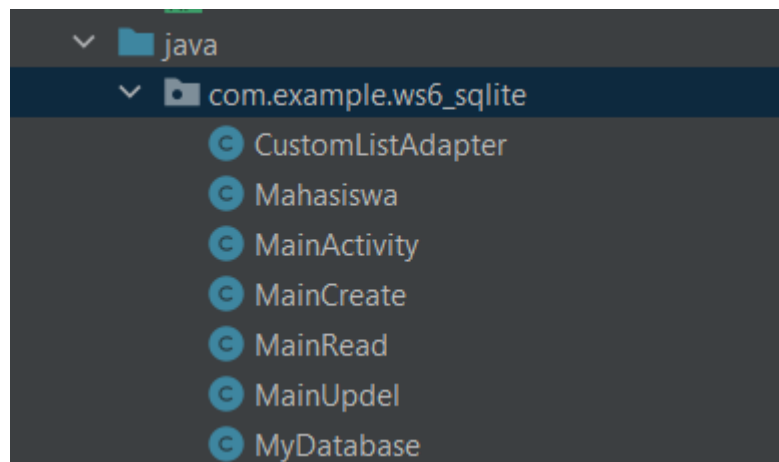
```

<TextView
    android:id="@+id/text_nama"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignStart="@+id/rating"
    android:layout_alignLeft="@+id/rating"
    android:text="NAMA"
    android:textColor="@color/black"
    android:textStyle="bold" />

<TextView
    android:id="@+id/text_kelas"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:layout_below="@+id/text_nama"
    android:layout_toEndOf="@+id/iconid"
    android:layout_toRightOf="@+id/iconid"
    android:text="KELAS"
    android:textColor="@color/black" />
</LinearLayout>
</LinearLayout>

```

11. Buatlah beberapa class baru pada folder project dengan cara klik folder project utama
 >> Klik kanan >> New >> Java Class lalu buat beberapa class sehingga kita mempunyai class seperti dibawah



12. Isikan tiap-tiap source code pada class seperti dibawah :

CustomListAdapter.java

```

public class CustomListAdapter extends BaseAdapter {
    private Activity activity;
    private LayoutInflater inflater;
    private List<Mahasiswa> Mahasiswa;

    public CustomListAdapter(Activity activity, List<Mahasiswa>
    Mahasiswa) {
        this.activity = activity;
        this.Mahasiswa = Mahasiswa;
    }

    @Override
    public int getCount() {

```



```

        return Mahasiswa.size();
    }

    @Override
    public Object getItem(int location) {
        return Mahasiswa.get(location);
    }

    @Override
    public long getItemId(int position) {
        return position;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup
parent) {

        if (inflater == null)
            inflater = (LayoutInflater) activity
.getSystemService(Context.LAYOUT_INFLATER_SERVICE);
        if (convertView == null)
            convertView = inflater.inflate(R.layout.custom_list,
null);

        TextView nama = (TextView)
convertView.findViewById(R.id.text_nama);
        TextView kelas = (TextView)
convertView.findViewById(R.id.text_kelas);

        ImageView imageView = (ImageView)
convertView.findViewById(R.id.iconid);

        Mahasiswa m = Mahasiswa.get(position);

        nama.setText("Nama : " + m.get_nama());
        kelas.setText("Kelas : " + m.get_kelas());

        return convertView;
    }
}

```

Mahasiswa.java

```

public class Mahasiswa {
    private String _id, _nama, _kelas;

    public Mahasiswa (String id, String nama, String kelas) {
        this._id = id;
        this._nama = nama;
        this._kelas = kelas;
    }

    public Mahasiswa() {
    }

    public String get_id() {
        return _id;
    }

    public void set_id(String _id) {
        this._id = _id;
    }
}

```

```

    public String get_nama() {
        return _nama;
    }

    public void set_nama(String _nama) {
        this._nama = _nama;
    }

    public String get_kelas() {
        return _kelas;
    }

    public void set_kelas(String _kelas) {
        this._kelas = _kelas;
    }
}

```

MainActivity.java

```

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    public void create(View view){
        Intent a = new
Intent(MainActivity.this,MainCreate.class);
        startActivity(a);
    }

    public void read(View view){
        Intent b = new Intent(MainActivity.this,MainRead.class);
        startActivity(b);
    }

}

```

MainCreate.java

```

public class MainCreate extends AppCompatActivity {
    private MyDatabase db;
    private EditText Enama, Ekelas;
    private String Snama, Skelas;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main_create);
        db = new MyDatabase(this);

        Enama = (EditText) findViewById(R.id.create_nama);
        Ekelas = (EditText) findViewById(R.id.create_kelas);

        Button btnCreate = (Button)
findViewById(R.id.create_btn);
        btnCreate.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {

                Snama = String.valueOf(Enama.getText());
                Skelas = String.valueOf(Ekelas.getText());
            }
        });
    }
}

```

```

        if (Snama.equals("")) {
            Enama.requestFocus();
            Toast.makeText(MainCreate.this, "Silahkan isi
nama",
                        Toast.LENGTH_SHORT).show();
        }
        else if (Skelas.equals("")) {
            Ekelas.requestFocus();
            Toast.makeText(MainCreate.this, "Silahkan isi
kelas",
                        Toast.LENGTH_SHORT).show();
        }
        else {
            Enama.setText("");
            Ekelas.setText("");
            Toast.makeText(MainCreate.this, "Data telah
ditambah",
                        Toast.LENGTH_SHORT).show();
            db.CreateMahasiswa(new Mahasiswa(null, Snama,
Skelas));

            Intent a = new Intent(MainCreate.this,
MainActivity.class);
            startActivity(a);
        }
    }
});
}
}

```

MainRead.java

```

public class MainRead extends AppCompatActivity implements
AdapterView.OnItemClickListener{
    private ListView mListView;
    private CustomListAdapter adapter_off;
    private MyDatabase db;
    private List<Mahasiswa> ListMahasiswa = new
ArrayList<Mahasiswa>();

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main_read);
        db = new MyDatabase(this);

        adapter_off = new CustomListAdapter(this, ListMahasiswa
);

        mListView = (ListView) findViewById(R.id.list_mahasiswa);
        mListView.setAdapter(adapter_off);
        mListView.setOnItemClickListener(this);
        mListView.setClickable(true);
        ListMahasiswa.clear();

        List<Mahasiswa> mahasiswa = db.ReadMahasiswa();
        for (Mahasiswa mhs : mahasiswa) {
            Mahasiswa daftar = new Mahasiswa();
            daftar.set_id(mhs.get_id());
            daftar.set_nama(mhs.get_nama());
            daftar.set_kelas(mhs.get_kelas());
            ListMahasiswa.add(daftar);
        }
    }
}

```

```

        if ((ListMahasiswa.isEmpty()))
            Toast.makeText(MainRead.this, "Tidak ada data",
                Toast.LENGTH_SHORT).show();
        else {
        }
    }
}

@Override
public void onItemClick(AdapterView<?> parent, View view, int
i, long l) {
    Object o = mListview.getItemAtPosition(i);
    Mahasiswa detailMhs = (Mahasiswa)o;

    String Sid = detailMhs.get_id();
    String Snama = detailMhs.get_nama();
    String Skelas = detailMhs.get_kelas();

    Intent goUpdel = new Intent(MainRead.this,
MainUpdel.class);
    goUpdel.putExtra("Iid", Sid);
    goUpdel.putExtra("Inama", Snama);
    goUpdel.putExtra("Ikelas", Skelas);
    startActivity(goUpdel);
}

@Override
protected void onResume() {
    super.onResume();
    ListMahasiswa.clear();
    mListview.setAdapter(adapter_off);

    List<Mahasiswa> mahasiswa = db.ReadMahasiswa();
    for (Mahasiswa mhs : mahasiswa) {
        Mahasiswa daftar = new Mahasiswa();
        daftar.set_id(mhs.get_id());
        daftar.set_nama(mhs.get_nama());
        daftar.set_kelas(mhs.get_kelas());
        ListMahasiswa.add(daftar);

        if ((ListMahasiswa.isEmpty()))
            Toast.makeText(MainRead.this, "Tidak ada data",
                Toast.LENGTH_SHORT).show();
        else {
        }
    }
}
}

```

MainUpdel.java

```

import androidx.appcompat.app.AppCompatActivity;

public class MainUpdel extends AppCompatActivity {
    private MyDatabase db;
    private String Sid, Snama, Skelas;
    private EditText Enama, Ekelas;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
    }
}

```

```

setContentView(R.layout.activity_main_updel);
db = new MyDatabase(this);

Intent i = this getIntent();
Sid = i.getStringExtra("Iid");
Snama = i.getStringExtra("Inama");
Skelas = i.getStringExtra("Ikelas");

Enama = (EditText) findViewById(R.id.updel_nama);
Ekelas = (EditText) findViewById(R.id.updel_kelas);

Enama.setText(Snama);
Ekelas.setText(Skelas);

Button btnUpdate = (Button) findViewById(R.id.btn_up);
btnUpdate.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Snama = String.valueOf(Enama.getText());
        Skelas = String.valueOf(Ekelas.getText());
        if (Snama.equals("")) {
            Enama.requestFocus();
            Toast.makeText(MainUpdel.this, "Silahkan isi
nama",
                                Toast.LENGTH_SHORT).show();
        } else if (Skelas.equals("")) {
            Ekelas.requestFocus();
            Toast.makeText(MainUpdel.this, "Silahkan isi
kelas",
                                Toast.LENGTH_SHORT).show();
        } else {
            db.UpdateMahasiswa(new Mahasiswa(Sid, Snama,
Skelas));
            Toast.makeText(MainUpdel.this, "Data telah
diupdate",
                                Toast.LENGTH_SHORT).show();
            finish();
        }
    }
});

Button btnDelete = (Button) findViewById(R.id.btn_del);
btnDelete.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        db.DeleteMahasiswa(new Mahasiswa(Sid, Snama,
Skelas));
        Toast.makeText(MainUpdel.this, "Data telah
dihapus",
                                Toast.LENGTH_SHORT).show();
        finish();
    }
});
}
}

```

MyDatabase.java

```
public class MyDatabase extends SQLiteOpenHelper {
    private static int DATABASE_VERSION = 1;
    private static String DATABASE_NAME = "db_kampus";

    private static final String tb_mahasiswa = "tb_mahasiswa";

    private static final String tb_mahasiswa_id = "id";
    private static final String tb_mahasiswa_nama = "nama";
    private static final String tb_mahasiswa_kelas = "kelas";

    private static final String CREATE_TABLE_MAHASISWA = "CREATE
TABLE " +
        tb_mahasiswa + "("
        + tb_mahasiswa_id + " INTEGER PRIMARY KEY ,"
        + tb_mahasiswa_nama + " TEXT ,"
        + tb_mahasiswa_kelas + " TEXT " + ")";

    public MyDatabase (Context context){
        super(context, DATABASE_NAME, null , DATABASE_VERSION);
    }

    @Override
    public void onCreate(SQLiteDatabase db) {
        db.execSQL(CREATE_TABLE_MAHASISWA);
    }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion, int
newVersion) {

    }

    public void CreateMahasiswa(Mahasiswa data){
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(tb_mahasiswa_id, data.get_id());
        values.put(tb_mahasiswa_nama, data.get_nama());
        values.put(tb_mahasiswa_kelas, data.get_kelas());
        db.insert(tb_mahasiswa, null, values);
        db.close();
    }

    public List<Mahasiswa> ReadMahasiswa() {
        List<Mahasiswa> listMhs = new ArrayList<Mahasiswa>();
        String selectQuery = "SELECT * FROM " + tb_mahasiswa;
        SQLiteDatabase db = this.getWritableDatabase();
        Cursor cursor = db.rawQuery(selectQuery, null);

        if (cursor.moveToFirst()) {
            do {
                Mahasiswa data = new Mahasiswa();
                data.set_id(cursor.getString(0));
                data.set_nama(cursor.getString(1));
                data.set_kelas(cursor.getString(2));
                listMhs.add(data);
            } while (cursor.moveToNext());
        }
        db.close();
        return listMhs;
    }
}
```

```

    }

    public int UpdateMahasiswa (Mahasiswa data){
        SQLiteDatabase db = this.getWritableDatabase();
        ContentValues values = new ContentValues();
        values.put(tb_mahasiswa_nama, data.get_nama());
        values.put(tb_mahasiswa_kelas, data.get_kelas());

        return db.update(tb_mahasiswa, values, tb_mahasiswa_id +
            " = ?",
                new String[]{String.valueOf(data.get_id())});
    }

    public void DeleteMahasiswa(Mahasiswa data){
        SQLiteDatabase db = this.getWritableDatabase();
        db.delete(tb_mahasiswa, tb_mahasiswa_id+ " = ?",
            new String[]{String.valueOf(data.get_id())});
        db.close();
    }
}

```

13. Deklarasikan activity yang sudah dibuat pada androidManifest.xml

```

<activity android:name=".MainRead" android:exported="true"/>
<activity android:name=".MainCreate" android:exported="true"/>
<activity android:name=".MainUpdel" android:exported="true"/>

```

14. Pastikan tidak ada yang error lalu bisa memulai proses build aplikasi

7.5. Tugas Rumah

Buatlah sebuah program sederhana yang mengimplementasikan SQLite sesuai kreativitas masing – masing. Diharapkan setiap mahasiswa memiliki tema yang berbeda