



MODUL 9

Penggunaan Database



Pada modul ini, akan dibahas mengenai penggunaan database SQLite dalam aplikasi. Secara khusus penggunaan database dibatasi pada paket yang sama.

Firman Hidayah

A. Tujuan Pembelajaran

1. Memahami pengertian database
2. Membuat aplikasi yang melibatkan database

B. Petunjuk

- ✓ Awali setiap aktivitas dengan doa, semoga berkah dan mendapat kemudahan.
- ✓ Pahami tujuan, dasar teori, dan latihan-latihan praktikum dengan baik.
- ✓ Kerjakan tugas-tugas praktikum dengan baik, sabar, dan jujur.
- ✓ Tanyakan kepada asisten/dosen apabila ada hal-hal yang kurang jelas.

C. Dasar Teori

1. Pengertian Database

Database adalah suatu pengorganisasian data dengan tujuan agar data dapat dimanipulasi dan diperoleh dengan cepat dan mudah. Terkait dengan database, Android menyediakan SQLite. Database tersebut bersifat relasional. Data yang disusun dalam bentuk tabel yang berisi field dan record.

Secara bawaan, suatu database hanya dapat digunakan oleh aplikasi yang menciptakannya. Dalam hal ini database biasanya disimpan di dalam folder `/data/data/nama_paket/database`

Secara umum, operasi yang dilakukan di database meliputi:

- Pembuatan database
- Pembukaan database
- Penyisipan data dalam tabel
- Pengubahan data dalam tabel
- Penghapusan data dalam tabel
- Penutupan database

Sesuai yang tersirat di namanya, SQLite menggunakan *Structured Query Language* (SQL) untuk mengakses database. SQL merupakan standar untuk mengakses data di dalam database. Beberapa perintah SQL adalah sebagai berikut:

- CREATE TABLE (untuk menciptakan tabel)

- INSERT INTO (untuk menyisipkan data ke dalam tabel)
- UPDATE (untuk menguubah data dalam tabel)
- DELETE FROM (menghapus data dari suatu tabel)

2. Membuat Aplikasi yang Melibatkan Database

Untuk membuat aplikasi yang melibatkan penggunaan database langkah yang umum digunakan adalah sebagai berikut:


- 1) Penyiapan aplikasi yang melibatkan database
- 2) Menyiapkan konstanta untuk pengaksesan database
- 3) Membuat Class DBHelper
- 4) Menyiapkan kode untuk mengakses database
- 5) Menuliskan data di MainActivity.java

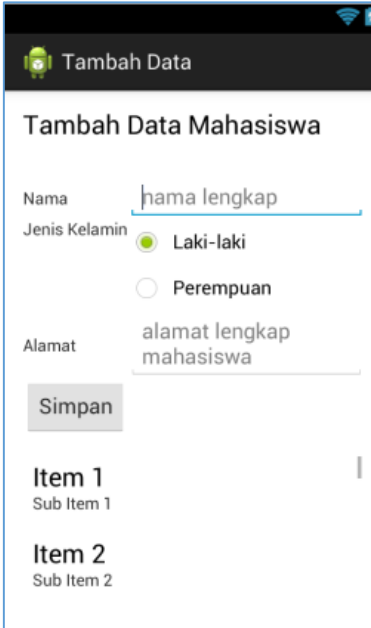
D. Praktikum

1. Praktikum 1 – Membuat aplikasi Tambah Data Mahasiswa

Untuk mempraktikkan aplikasi yang melibatkan database, silahkan ikuti pedoman berikut ini. Berikut adalah langkah-langkahnya:

a. Menyiapkan Aplikasi

- 1) Pilih menu **File** kemudian klik **Close All** untuk menutup semua file yang sedang terbuka.
- 2) Klik icon **New** [] pada toolbar
- 3) Buka folder **Android** kemudian klik ganda pada **Android Application Project**
- 4) Isikan tulisan **Tambah Data** pada **Application Name** dan isikan field- field yang lain seperti biasanya.
- 5) Klik tombol **Next**, sebanyak empat kali
- 6) Klik tombol **Finish**, sehingga tampil desain aplikasi
- 7) Buatlah layout aplikasi dengan tampilan seperti berikut ini:



- 8) Melalui panel properties lakukan perubahan sejumlah properti dari tiap komponen yang ada pada **activity_main.xml** seperti berikut ini:

Kontrol	Property	Value
editText1	id	editTextNama
radio0	id	radioLaki
	text	Laki-laki
radio1	id	radioPerempuan
	text	Perempuan
editText2	id	editTextAlamat
button1	id	buttonSimpan
	text	Simpan
listView1	id	listViewDb

- 9) Periksa **activity_main.xml** sehingga seperti berikut ini:

```
<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"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.tambahdata.MainActivity" >

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:text="Tambah Data Mahasiswa"
        android:textAppearance="?android:attr/textAppearanceLarge" />

    <TableLayout
        android:id="@+id/tableLayout1"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_below="@+id/textView1"
        android:layout_marginTop="30dp" >

        <TableRow
            android:id="@+id/tableRow1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" >
```

```

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Nama" />

<EditText
    android:id="@+id/editTextNama"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_weight="1"
    android:hint="nama lengkap" >

    <requestFocus />
</EditText>
</TableRow>

<TableRow
    android:id="@+id/tableRow2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" >

    <TextView
        android:id="@+id/textView3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Jenis Kelamin" />

    <RadioGroup
        android:id="@+id/radioGroup1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" >

        <RadioButton
            android:id="@+id/radioLaki"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:checked="true"
            android:text="Laki-laki" />

        <RadioButton
            android:id="@+id/radioPerempuan"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Perempuan" />
    </RadioGroup>
</TableRow>

<TableRow
    android:id="@+id/tableRow3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" >

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Alamat" />

    <EditText
        android:id="@+id/editTextAlamat"

```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:ems="10"
        android:layout_weight="1"
        android:hint="alamat lengkap mahasiswa"
        android:inputType="textMultiLine" />

</TableRow>

<TableRow
    android:id="@+id/tableRow4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" >

    <Button
        android:id="@+id/buttonSimpan"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Simpan" />

</TableRow>
</TableLayout>

<ListView
    android:id="@+id/listViewDb"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/tableLayout1"
    android:layout_marginTop="14dp" >
</ListView>
</RelativeLayout>

```

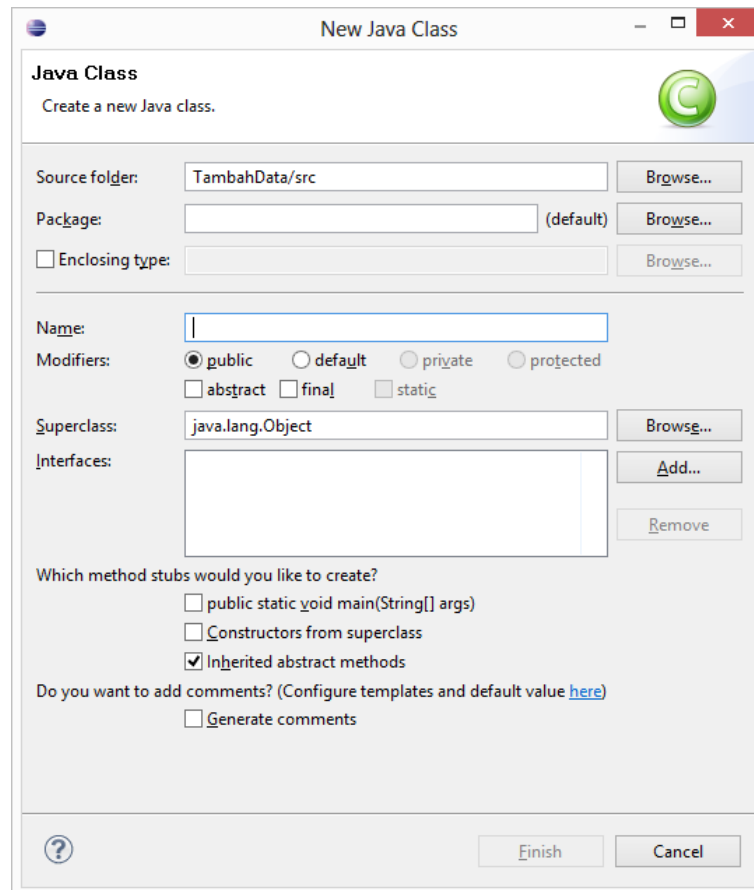
10) Simpan project

b. Menyiapkan Konstanta untuk Pengaksesan Database

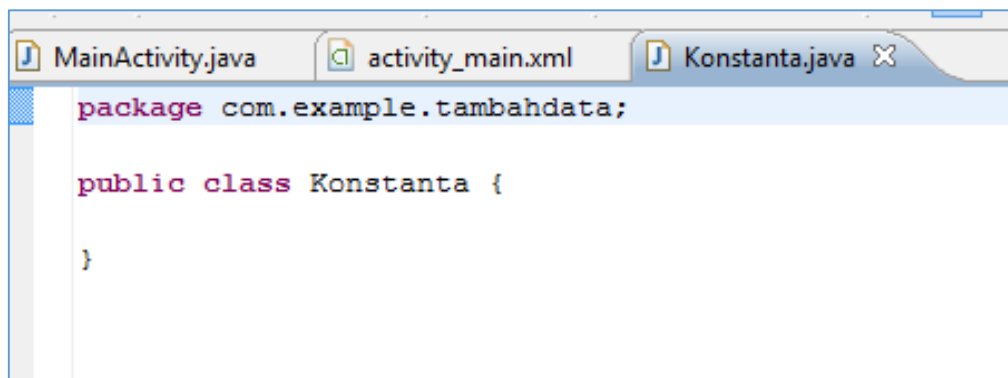
Tahap berikutnya adalah menyiapkan file `Konstanta.java` yang berisi sebuah kelas bernama `Konstanta`. Kelas tersebut digunakan untuk mendefinisikan sejumlah konstanta yang dipakai untuk mengakses database.

Langkah-langkah yang perlu dilakukan adalah sebagai berikut:

- 1) Klik pada folder **TambahData** di panel Package Explorer
- 2) Klik pada menu **File**
- 3) Sorot pada **New – Class**, sehingga tampil kotak dialog berikut:



- 4) Isikan `com.example.tambahdata` di field **Package**
- 5) Ketikkan **Konstanta** di field **Name**
- 6) Klik pada tombol **Finish** kemudian akan terdapat tampilan berikut ini:



- 7) Lengkapilah isi file `Konstanta.java` menjadi seperti berikut ini:

```
package com.example.tambahdata;

public class Konstanta {
    public final static String NAMA_DB = "mahasiswa.db";
    public final static int VERSI_DB = 1;
    public final static String NAMA_TABEL = "tbl_mhs";
    public final static String ID_MHS = "id_mhs";
}
```



```

public final static String NAMA_MHS = "nama_mhs";
public final static String JK_MHS = "jk_mhs";
public final static String ALAMAT_MHS = "alamat_mhs";
}

```

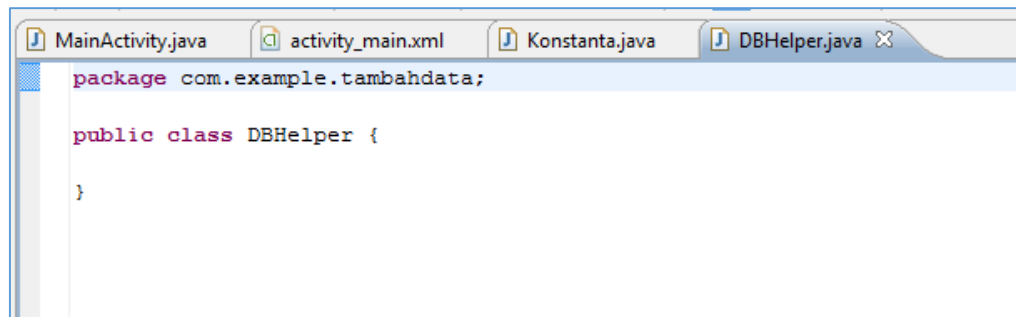
8) Simpan project

c. Menyiapkan Kelas DBHelper

Kelas berikutnya yang perlu dibuat adalah DBHelper. Kelas ini mewarisi SQLiteOpenHelper dan ditujukan untuk menangani pembuatan database dan pembuatan tabel.

Langkah-langkah yang perlu dilakukan adalah sebagai berikut:

- 1) Klik pada folder **TambahData** di panel Package Explorer
- 2) Klik pada menu **File**
- 3) Sorot pada **New – Class**, sehingga tampil kotak dialog New Java Class
- 4) Isikan `com.example.tambahdata` di field **Package**
- 5) Ketikkan **DBHelper** di field **Name**
- 6) Klik pada tombol **Finish** kemudian akan terdapat tampilan berikut ini:



7) Lengkapilah isi file DBHelper.java menjadi seperti berikut ini:

```

package com.example.tambahdata;
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteException;
import android.database.sqlite.SQLiteOpenHelper;
import android.database.sqlite.SQLiteDatabase.CursorFactory;

public class DBHelper extends SQLiteOpenHelper {
    //berisi perintah SQL untuk menciptakan tabem mahasiswa
    private final static String BUAT_TABEL = "create table" +
        Konstanta.NAMA_TABEL + " (" +
        Konstanta.ID_MHS + "integer primary key autoincrement, " +
        Konstanta.NAMA_MHS + "text not null, " +
        Konstanta.JK_MHS + "text not null" +

```

```

        Konstanta.ALAMAT_MHS + "text not null);";

        //memanggil konstruktor super kelas
        public DBHelper(Context konteks, String nama, CursorFactory f, int
versi){
            super (konteks, nama, f, versi);
        }

        //menciptakan tabel
        public void onCreate (SQLiteDatabase db){
            try{
                db.execSQL(BUAT_TABEL);
            }
            catch(SQLiteException e){
            }
        }

        @Override
        //upgrade tabel, jika sudah ada tabel maka akan dihapus dan buat
baru
        public void onUpgrade(SQLiteDatabase db, int versiLama, int
versiBaru){
            db.execSQL("drop table if exist " + Konstanta.NAMA_TABEL);
            onCreate(db);
        }
    }
}

```

d. Menyiapkan Kelas untuk mengakses database

Untuk kepentingan memudahkan dalam mengakses data, kelas bernama **Dbase** akan dibuat. Kelas inilah yang akan menyediakan sejumlah metode untuk memudahkan pembuatan aplikasi yang mengakses database.

Langkah-langkah yang perlu dilakukan adalah sebagai berikut:

- 1) Klik pada folder **TambahData** di panel Package Explorer
- 2) Klik pada menu **File**
- 3) Sorot pada **New – Class**, sehingga tampil kotak dialog New Java Class
- 4) Isikan `com.example.tambahdata` di field **Package**
- 5) Ketikkan **Dbase** di field **Name**
- 6) Lengkapilah isi file **Dbase.java** menjadi seperti berikut ini:

```

package com.example.tambahdata;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteException;

public class Dbase {
    private SQLiteDatabase db;
    private final Context konteks;
    private final DBHelper dbhelper;
}

```

```

//konstruktor
public Dbase (Context k){
    konteks = k;
    dbhelper = new DBHelper(konteks, Konstanta.NAMA_DB, null,
Konstanta.VERSI_DB);
}

//membuka database
public void open() throws SQLiteException {
    try{
        db = dbhelper.getWritableDatabase();
    }
    catch(SQLiteException e){
        db = dbhelper.getReadableDatabase();
    }
}

//menutup database
public void close(){
    db.close();
}

//menyisipkan data
public long insertData(String nama_mhs, String jk_mhs, String
alamat_mhs){
    try{
        ContentValues dataBaru = new ContentValues();
        dataBaru.put(Konstanta.NAMA_MHS, nama_mhs);
        dataBaru.put(Konstanta.JK_MHS, jk_mhs);
        dataBaru.put(Konstanta.ALAMAT_MHS, alamat_mhs);

        return db.insert(Konstanta.NAMA_TABEL, null, dataBaru);
    }
    catch (SQLiteException e){
        return -1;
    }
}

//mengambil seluruh data
public Cursor getAllData(){
    return db.query(Konstanta.NAMA_TABEL,
        new String[]{
            Konstanta.ID_MHS,
            Konstanta.NAMA_MHS,
            Konstanta.JK_MHS,
            Konstanta.ALAMAT_MHS},
        null, null, null, null, null);
}
}

```

7) Simpan project:

e. Menuliskan kode di MainActivity.java

Tahap berikut ini adalah menuliskan kode program di MainActivity.java. Di sini lah pemanfaatan hal-hal yang telah dilakukan sebelumnya akan dimulai.

Langkah-langkah yang perlu dilakukan adalah sebagai berikut:

1) Buka MainActivity.java kemudian ketikkan kode berikut ini:

```
package com.example.tambahdata;

import android.app.Activity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.ListView;
import android.widget.Button;
import android.widget.Toast;
import android.widget.AdapterView;
import android.database.Cursor;
import java.util.ArrayList;

public class MainActivity extends Activity {
    Dbase db;
    Button tombolSimpan;
    EditText editNama, editAlamat;
    RadioButton radioLaki, radioPerempuan;
    ListView listViewData;

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

        //buat objek database
        db = new Dbase(this);

        //objek komponen
        editNama = (EditText) findViewById(R.id.editTextNama);
        radioLaki = (RadioButton) findViewById(R.id.radioLaki);
        radioPerempuan = (RadioButton) findViewById(R.id.radioPerempuan);
        editAlamat = (EditText) findViewById(R.id.editTextAlamat);
        listViewData = (ListView) findViewById(R.id.listViewDb);
        tombolSimpan = (Button) findViewById(R.id.buttonSimpan);

        tombolSimpan.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                String jk = "";
                if (radioLaki.isChecked())
                {
                    jk = "L";
                }

                if (radioPerempuan.isChecked())
                {
                    jk = "P";
                }
                db.open();
                db.insertData(editNama.getText().toString(), jk,
editAlamat.getText().toString());
                db.close();
            }
        });
    }
}
```

```

        Toast.makeText(getBaseContext(), "Data berhasil
disimpan", Toast.LENGTH_SHORT).show();

        //reset field
        editNama.setText("");
        radioLaki.setChecked(false);
        radioPerempuan.setChecked(false);
        editAlamat.setText("");

        tampilkanData();
    }
});
tampilkanData();
}

public void tampilkanData() {
    db.open();
    Cursor c = db.getAllData();

    //buat array dinamis
    ArrayList<String> larik = new ArrayList<String>();
    if (c.moveToFirst()) {
        do {
            int id_mhs          = c.getInt(0);
            String nama_mhs     = c.getString(1);
            String jk_mhs       = c.getString(2);
            String alamat_mhs   = c.getString(3);

            larik.add(Integer.toString(id_mhs) + ": " +
nama_mhs + "(" + jk_mhs + ")\n" + alamat_mhs);

        } while (c.moveToNext());
    }
    db.close();

    //letakkan larik ke ListView
    ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
android.R.layout.simple_list_item_1, larik);
    listViewData.setAdapter(adapter);

}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is
present.
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();
    if (id == R.id.action_settings) {
        return true;
    }
    return super.onOptionsItemSelected(item);
}

```

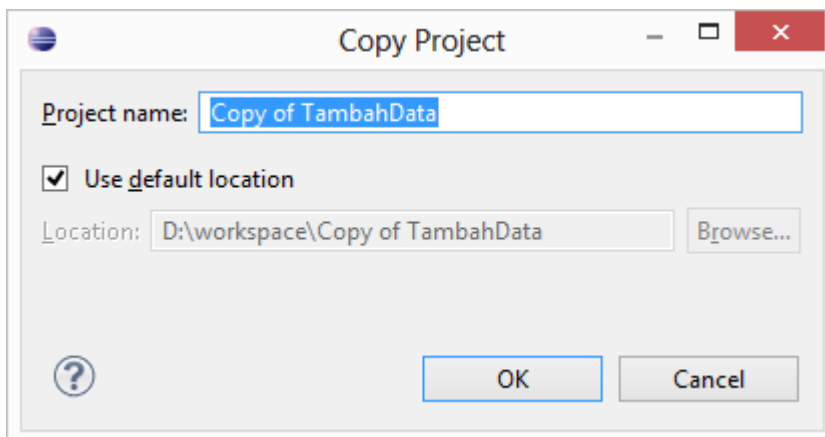
}

- 2) Simpan project dan jalankan program

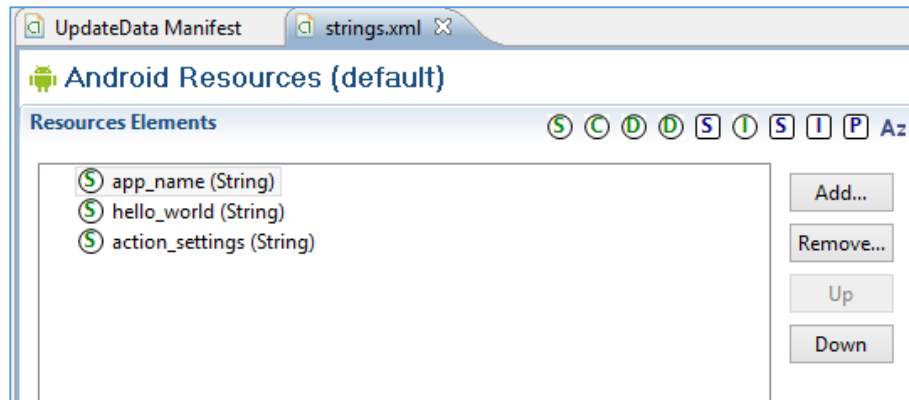
2. Praktikum 2 – Melakukan Update Data

Pada praktikum kali ini akan dipraktikkan cara membuat aplikasi yang melibatkan database dengan operasi update data.

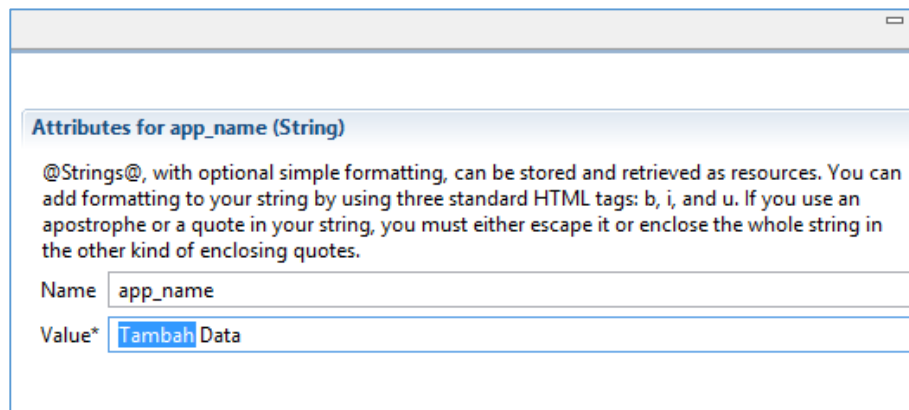
- 1) Pilih menu **File** kemudian klik **Close All** untuk menutup semua file yang sedang terbuka.
- 2) Letakkan cursor mouse ke folder **TambahData** di Panel Package Explorer.
- 3) **Klik kanan** kemudian klik **Copy**
- 4) Letakkan Cursor di area kosong Package Explorer
- 5) **Klik kanan** kemudian klik **Paste** maka akan tampil kotak dialog seperti berikut ini:



- 6) Ketikkan **Update Data** pada **Project Name**
- 7) Klik Tombol **OK**
- 8) Untuk mengubah nama aplikasi (app_name) lakukan langkah berikut:
 - a. buka **strings.xml** di folder /res/values
 - b. Klik pada app_name



c. Ubah value Tambah Data menjadi **Update Data**



d. Simpan project (Ctrl+S) kemudian close Strings.xml

9) Ubahlah desain layoutnya menjadi seperti berikut ini dengan menambahkan field untuk **id** [perhatikan instruksi dari instruktur]:



Ubah properti dari kontrol ini menjadi seperti berikut ini:

Property	Value
id	editTextId
hint	id mahasiswa

Jadikan kontrol ini menjadi focus pertama [perhatikan instruksi dari instruktur]

10) Periksa **activity_main.xml** sehingga seperti berikut ini:

```

<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"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context="com.example.tambahdata.MainActivity" >

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:text="Ubah Data Mahasiswa"
        android:textAppearance="?android:attr/textAppearanceLarge" />

    <TableLayout
        android:id="@+id/tableLayout1"
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_below="@+id/textView1"
        android:layout_marginTop="30dp" >

        <TableRow
            android:id="@+id/tableRow5"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" >

            <TextView
                android:id="@+id/textView5"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="ID" />

            <EditText
                android:id="@+id/editTextId"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:ems="10"
                android:hint="id mahasiswa" />
            <requestFocus />

        </TableRow>

        <TableRow
            android:id="@+id/tableRow1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" >

            <TextView
                android:id="@+id/textView2"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Nama" />

```



```

        <EditText
            android:id="@+id/editTextNama"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:hint="nama lengkap" >
        </EditText>
    </TableRow>

    <TableRow
        android:id="@+id/tableRow2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" >

        <TextView
            android:id="@+id/textView3"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Jenis Kelamin" />

        <RadioGroup
            android:id="@+id/radioGroup1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" >

            <RadioButton
                android:id="@+id/radioLaki"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:checked="true"
                android:text="Laki-laki" />

            <RadioButton
                android:id="@+id/radioPerempuan"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Perempuan" />
        </RadioGroup>
    </TableRow>

    <TableRow
        android:id="@+id/tableRow3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" >

        <TextView
            android:id="@+id/textView4"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Alamat" />

        <EditText
            android:id="@+id/editTextAlamat"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:ems="10"
            android:layout_weight="1"
            android:hint="alamat lengkap mahasiswa"
            android:inputType="textMultiLine" />

```

```

</TableRow>

<TableRow
    android:id="@+id/tableRow4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content" >

    <Button
        android:id="@+id/buttonSimpan"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Simpan" />

</TableRow>

</TableLayout>

<ListView
    android:id="@+id/listViewDb"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_below="@+id/tableLayout1"
    android:layout_marginTop="14dp" >
</ListView>

</RelativeLayout>

```

11) Simpan project

12) Lakukan modifikasi kode pada file **Dbase.java** dengan menambahkan metode `updateData` sehingga kode menjadi seperti berikut ini (perhatikan kode yang diarsir):

```

package com.example.tambahdata;

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteException;

public class Dbase {
    private SQLiteDatabase db;
    private final Context konteks;
    private final DBHelper dbHelper;

    //konstruktor
    public Dbase (Context k){
        konteks = k;
        dbHelper = new DBHelper(konteks, Konstanta.NAMA_DB, null,
        Konstanta.VERSI_DB);
    }

    //membuka database
    public void open() throws SQLiteException {
        try{
            db = dbHelper.getWritableDatabase();
        }
        catch (SQLiteException e){

```

```

        db = dbHelper.getReadableDatabase();
    }

    //menutup database
    public void close(){
        db.close();
    }

    //menyisipkan data
    public long insertData(String nama_mhs, String jk_mhs, String
alamat_mhs){
        try{
            ContentValues dataBaru = new ContentValues();
            dataBaru.put(Konstanta.NAMA_MHS, nama_mhs);
            dataBaru.put(Konstanta.JK_MHS, jk_mhs);
            dataBaru.put(Konstanta.ALAMAT_MHS, alamat_mhs);

            return db.insert(Konstanta.NAMA_TABEL, null, dataBaru);
        }
        catch (SQLException e){
            return -1;
        }
    }

    //mengambil seluruh data
    public Cursor getAllData(){
        return db.query(Konstanta.NAMA_TABEL,
            new String[]{
                Konstanta.ID_MHS,
                Konstanta.NAMA_MHS,
                Konstanta.JK_MHS,
                Konstanta.ALAMAT_MHS},
            null, null, null, null, null);
    }

    //mengambil data berdasarkan kunci
    public Cursor getData(long kunci){
        Cursor c = db.query(Konstanta.NAMA_TABEL,
            new String[]{
                Konstanta.ID_MHS,
                Konstanta.NAMA_MHS,
                Konstanta.JK_MHS,
                Konstanta.ALAMAT_MHS},
            Konstanta.ID_MHS + "=" + kunci, null, null, null,
null);
        if(c != null){
            c.moveToFirst();
        }
        return c;
    }

    //mengubah data mahasiswa
    public boolean updateData(long kunci, String nama_mhs, String jk_mhs,
String alamat_mhs){
        ContentValues cv = new ContentValues();
        cv.put(Konstanta.NAMA_MHS, nama_mhs);
        cv.put(Konstanta.JK_MHS, nama_mhs);
        cv.put(Konstanta.ALAMAT_MHS, nama_mhs);

        return db.update(Konstanta.NAMA_TABEL, cv, Konstanta.ID_MHS +
"=" + kunci, null) > 0;
    }

```

```
}
```

13) Lakukan modifikasi pada kode di MainActivity.java dan perhatikan kode yang diarsir seperti berikut ini:

```
package com.example.tambahdata;

import android.app.Activity;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.ListView;
import android.widget.Button;
import android.widget.Toast;
import android.widget.AdapterView;
import android.database.Cursor;
import java.util.ArrayList;

public class MainActivity extends Activity {
    Dbase db;
    Button tombolSimpan;
    EditText editId, editNama, editAlamat;
    RadioButton radioLaki, radioPerempuan;
    ListView listViewData;

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

        //buat objek database
        db = new Dbase(this);

        //objek komponen
        editId = (EditText) findViewById(R.id.editTextId);
        editNama = (EditText) findViewById(R.id.editTextNama);
        radioLaki = (RadioButton) findViewById(R.id.radioLaki);
        radioPerempuan = (RadioButton) findViewById(R.id.radioPerempuan);
        editAlamat = (EditText) findViewById(R.id.editTextAlamat);
        listViewData = (ListView) findViewById(R.id.listViewDb);
        tombolSimpan = (Button) findViewById(R.id.buttonSimpan);

        tombolSimpan.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                String jk = "";
                if (radioLaki.isChecked())
                {
                    jk = "L";
                }
            }
        });
    }
}
```

```

        if (radioPerempuan.isChecked())
        {
            jk = "P";
        }
        db.open();

        //cek keberadaan kunci
        Cursor c =
db.getData(Long.parseLong(editId.getText().toString()));
        if (!c.moveToFirst()){
            Toast.makeText(getBaseContext(), "Data tidak
ditemukan. Ganti ID lain!", Toast.LENGTH_SHORT).show();
            db.close();
            return;
        }

        //lakukan update data

        db.updateData(Long.valueOf(editId.getText().toString()),
editNama.getText().toString(), jk, editAlamat.getText().toString());
        db.close();
        Toast.makeText(getBaseContext(), "Data berhasil
diupdate!", Toast.LENGTH_SHORT).show();
        tampilkanData();
    }
});
tampilkanData();
}

public void tampilkanData(){
    db.open();
    Cursor c = db.getAllData();

    //buat array dinamis
    ArrayList<String> larik = new ArrayList<String>();
    if (c.moveToFirst()){
        do{
            int id_mhs          = c.getInt(0);
            String nama_mhs     = c.getString(1);
            String jk_mhs       = c.getString(2);
            String alamat_mhs   = c.getString(3);

            larik.add(Integer.toString(id_mhs) + ": " +
nama_mhs + "(" + jk_mhs + ")\n" + alamat_mhs);

        }while(c.moveToNext());
    }
    db.close();

    //letakkan larik ke ListView
    ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
android.R.layout.simple_list_item_1, larik);
    listViewData.setAdapter(adapter);
}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is
    present.
    getMenuInflater().inflate(R.menu.main, menu);
}

```

```

        return true;
    }

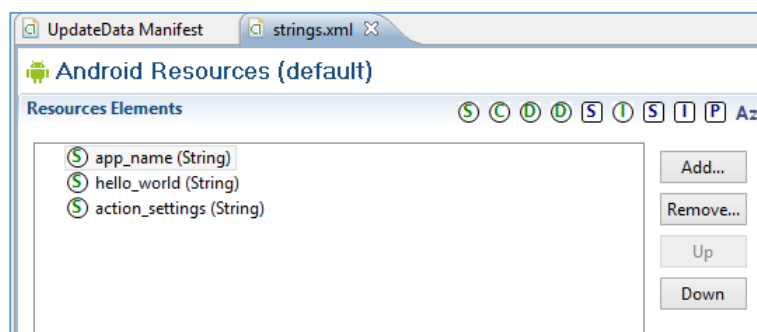
    @Override
    public boolean onOptionsItemSelected(MenuItem item) {
        // Handle action bar item clicks here. The action bar will
        // automatically handle clicks on the Home/Up button, so long
        // as you specify a parent activity in AndroidManifest.xml.
        int id = item.getItemId();
        if (id == R.id.action_settings) {
            return true;
        }
        return super.onOptionsItemSelected(item);
    }
}

```

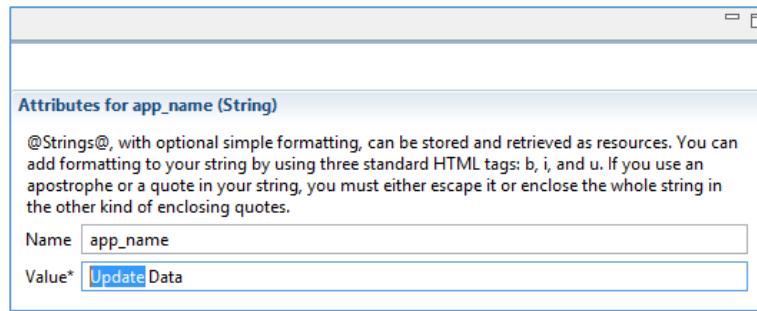
3. Praktikum 2 – Melakukan Update Data

Pada praktikum kali ini akan dipraktikkan cara membuat aplikasi yang melibatkan database dengan operasi update data.

- 1) Pilih menu **File** kemudian klik **Close All** untuk menutup semua file yang sedang terbuka.
- 2) Letakkan cursor mouse ke folder **UpdateData** di Panel Package Explorer.
- 3) **Klik kanan** kemudian klik **Copy**
- 4) Letakkan Cursor di area kosong Package Explorer
- 5) **Klik kanan** kemudian klik **Paste** maka akan tampil kotak dialog **Copy Project**
- 6) Ketikkan **Delete Data** pada **Project Name**
- 7) Klik Tombol **OK**
- 8) Untuk mengubah nama aplikasi (app_name) lakukan langkah berikut:
 - a. buka **strings.xml** di folder /res/values
 - b. Klik pada app_name

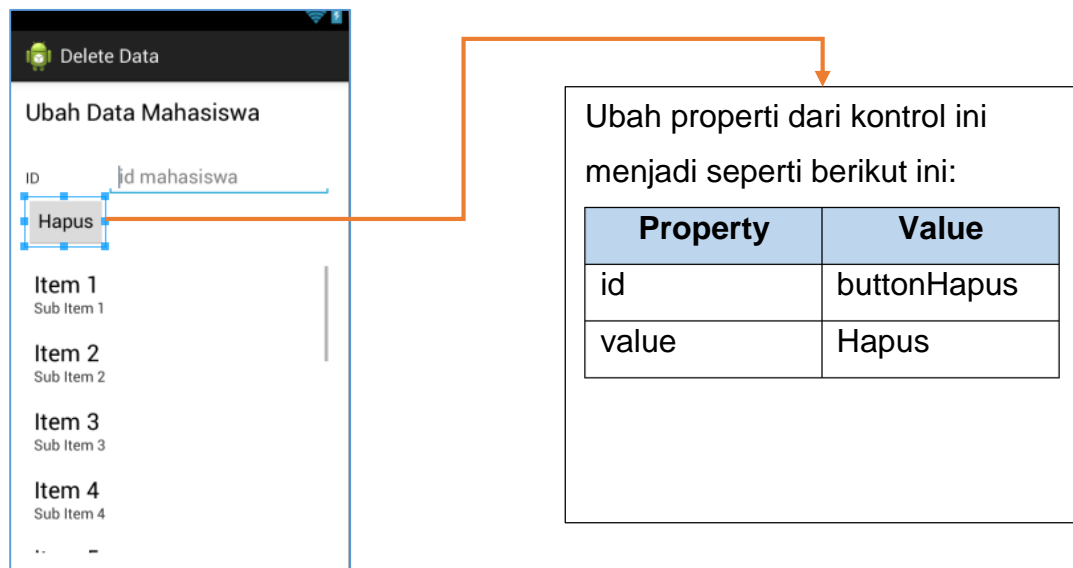


- c. Ubah value Update Data menjadi **Delete Data**



d. Simpan project (Ctrl+S) kemudian close Strings.xml

- 9) Ubahlah desain layoutnya menjadi seperti berikut ini dengan menghapus beberapa komponen yang sudah ada:



- 10) Periksa **activity_main.xml** sehingga seperti berikut ini:

```
<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"
android:paddingBottom="@dimen/activity_vertical_margin"
android:paddingLeft="@dimen/activity_horizontal_margin"
android:paddingRight="@dimen/activity_horizontal_margin"
android:paddingTop="@dimen/activity_vertical_margin"
tools:context="com.example.tambahdata.MainActivity" >

<TextView
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentLeft="true"
    android:layout_alignParentTop="true"
    android:text="Ubah Data Mahasiswa"
    android:textAppearance="?android:attr/textAppearanceLarge" />

<TableLayout
    android:id="@+id/tableLayout1"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
```

```

        android:layout_alignParentLeft="true"
        android:layout_below="@+id/textView1"
        android:layout_marginTop="30dp" >

        <TableRow
            android:id="@+id/tableRow5"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" >

            <TextView
                android:id="@+id/textView5"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="ID" />

            <EditText
                android:id="@+id/editTextId"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:layout_weight="1"
                android:ems="10"
                android:hint="id mahasiswa" />
            <requestFocus />

        </TableRow>

        <TableRow
            android:id="@+id/tableRow4"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content" >

            <Button
                android:id="@+id/buttonHapus"
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:text="Hapus" />

        </TableRow>

    </TableLayout>

    <ListView
        android:id="@+id/listViewDb"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_below="@+id/tableLayout1"
        android:layout_marginTop="14dp" >
    </ListView>

</RelativeLayout>

```

11) Simpan project

12) Lakukan modifikasi kode pada file **Dbase.java** dengan menambahkan metode `deleteData` sehingga kode menjadi seperti berikut ini (perhatikan kode yang diarsir):

```
package com.example.tambahdata;
```



```

import android.content.ContentValues;
import android.content.Context;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteException;

public class Dbase {
    private SQLiteDatabase db;
    private final Context konteks;
    private final DBHelper dbHelper;

    //konstruktor
    public Dbase (Context k){
        konteks = k;
        dbHelper = new DBHelper(konteks, Konstanta.NAMA_DB, null,
        Konstanta.VERSI_DB);
    }

    //membuka database
    public void open() throws SQLiteException {
        try{
            db = dbHelper.getWritableDatabase();
        }
        catch (SQLiteException e){
            db = dbHelper.getReadableDatabase();
        }
    }

    //menutup database
    public void close(){
        db.close();
    }

    //menyisipkan data
    public long insertData(String nama_mhs, String jk_mhs, String
    alamat_mhs){
        try{
            ContentValues dataBaru = new ContentValues();
            dataBaru.put(Konstanta.NAMA_MHS, nama_mhs);
            dataBaru.put(Konstanta.JK_MHS, jk_mhs);
            dataBaru.put(Konstanta.ALAMAT_MHS, alamat_mhs);

            return db.insert(Konstanta.NAMA_TABEL, null, dataBaru);
        }
        catch (SQLiteException e){
            return -1;
        }
    }

    //mengambil seluruh data
    public Cursor getAllData(){
        return db.query(Konstanta.NAMA_TABEL,
            new String[]{
                Konstanta.ID_MHS,
                Konstanta.NAMA_MHS,
                Konstanta.JK_MHS,
                Konstanta.ALAMAT_MHS},
            null, null, null, null, null);
    }

    //mengambil data berdasarkan kunci

```

```

public Cursor getData(long kunci){
    Cursor c = db.query(Konstanta.NAMA_TABEL,
        new String[]{
            Konstanta.ID_MHS,
            Konstanta.NAMA_MHS,
            Konstanta.JK_MHS,
            Konstanta.ALAMAT_MHS},
        Konstanta.ID_MHS + "=" + kunci, null, null, null,
null);
    if(c != null){
        c.moveToFirst();
    }
    return c;
}
//mengubah data mahasiswa
public boolean updateData(long kunci, String nama_mhs, String jk_mhs,
String alamat_mhs){
    ContentValues cv = new ContentValues();
    cv.put(Konstanta.NAMA_MHS, nama_mhs);
    cv.put(Konstanta.JK_MHS, nama_mhs);
    cv.put(Konstanta.ALAMAT_MHS, nama_mhs);

    return db.update(Konstanta.NAMA_TABEL, cv, Konstanta.ID_MHS +
"=" + kunci, null) > 0;
}

//Menghapus data menurut kunci
public boolean deleteData(long kunci){
    return db.delete(Konstanta.NAMA_TABEL, Konstanta.ID_MHS + "=" +
kunci, null)>0;
}
}

```

13) Lakukan modifikasi pada kode di MainActivity.java dan perhatikan kode yang diarsir seperti berikut ini:

```

package com.example.tambahdata;

import android.app.Activity;
import android.app.AlertDialog;
import android.os.Bundle;
import android.view.Menu;
import android.view.MenuItem;
import android.view.View;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.ListView;
import android.widget.Button;
import android.widget.Toast;
import android.widget.AdapterView;
import android.content.DialogInterface;
import android.database.Cursor;
import java.util.ArrayList;

public class MainActivity extends Activity {
    Dbase db;
    Button tombolHapus;
    EditText editId;
    RadioButton radioLaki, radioPerempuan;
    ListView listViewData;

```

```

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

    //buat objek database
    db = new Dbase(this);

    //objek komponen
    editId = (EditText) findViewById(R.id.editTextId);
    listViewData = (ListView) findViewById(R.id.listViewDb);
    tombolHapus = (Button) findViewById(R.id.buttonHapus);

    tombolHapus.setOnClickListener(new View.OnClickListener() {

        @Override
        public void onClick(View arg0) {
            // TODO Auto-generated method stub
            db.open();

            //cek keberadaan kunci
            Cursor c =
db.getData(Long.parseLong(editId.getText().toString()));
            if (!c.moveToFirst()) {
                Toast.makeText(getBaseContext(), "Data tidak
ditemukan. Ganti ID lain!", Toast.LENGTH_SHORT).show();
                db.close();
                return;
            }

            // konfirmasi untuk menghapus
            String nama_mhs = c.getString(0);
            db.close();

            AlertDialog.Builder kotakDialog = new
AlertDialog.Builder(MainActivity.this);
            kotakDialog.setCancelable(false);
            kotakDialog.setMessage("Apakah anda yakin akan
menghapus data " + nama_mhs + "?");
            kotakDialog.setTitle("Konfirmasi");

            //tombol Ya
            kotakDialog.setPositiveButton("Ya", new
DialogInterface.OnClickListener() {
                @Override
                public void onClick(DialogInterface dialog,
int id) {

                    // TODO Auto-generated method stub
                    db.open();

                    db.deleteData(Long.parseLong(editId.getText().toString()));
                    Toast.makeText(getBaseContext(), "Data
berhasil dihapus!", Toast.LENGTH_SHORT).show();
                    db.close();
                    tampilkanData();
                }
            });
        }
    });
}

```

```

        //tombol Tidak
        kotakDialog.setNegativeButton("Tidak", new
DialogInterface.OnClickListener() {
            @Override
            public void onClick(DialogInterface dialog,
int id) {

                // TODO Auto-generated method stub
                dialog.cancel();

            }
        });

        //menampilkan kotak dialog
        kotakDialog.create().show();
    }
});
tampilkanData();
}

public void tampilkanData(){
    db.open();
    Cursor c = db.getAllData();

    //buat array dinamis
    ArrayList<String> larik = new ArrayList<String>();
    if (c.moveToFirst()){
        do{
            int id_mhs          = c.getInt(0);
            String nama_mhs     = c.getString(1);
            String jk_mhs       = c.getString(2);
            String alamat_mhs   = c.getString(3);

            larik.add(Integer.toString(id_mhs) + ": " +
nama_mhs + "(" + jk_mhs + ")\n" + alamat_mhs);

        }while(c.moveToNext());
    }
    db.close();

    //letakkan larik ke ListView
    ArrayAdapter<String> adapter = new ArrayAdapter<String>(this,
android.R.layout.simple_list_item_1,larik);
    listViewData.setAdapter(adapter);

}

@Override
public boolean onCreateOptionsMenu(Menu menu) {
    // Inflate the menu; this adds items to the action bar if it is
present.
    getMenuInflater().inflate(R.menu.main, menu);
    return true;
}

@Override
public boolean onOptionsItemSelected(MenuItem item) {
    // Handle action bar item clicks here. The action bar will
    // automatically handle clicks on the Home/Up button, so long
    // as you specify a parent activity in AndroidManifest.xml.
    int id = item.getItemId();
    if (id == R.id.action_settings) {
        return true;
    }
}

```

```
    }  
    return super.onOptionsItemSelected(item);  
  }  
}
```

14) Simpan project dan jalankan aplikasi