**BAB III**

**ANDROID STUDIO #1**

1. **Tujuan**

1. Praktikan mampu mengetahui lingkungan pengembangan Android

2. Praktikan memahami siklus activity pada aplikasi Android

3. Praktikan memahami penggunaan layout – layout pada Android Studio

4. Praktikan dapat memahami penggunaan intent

1. **Alat dan Bahan**

1. PC / Notebook (minimal RAM 4GB)

2. Android Studio 3.2.1, JDK, Android SDK

3. Device Android dan kabel USB (wajib)

1. **Dasar Teori**

**3.3.1 Android**

Android adalah sistem operasi untuk telepon seluler yang berbasiskan Linux. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri sehingga dapat digunakan oleh bermacam peranti penggerak.

Android tersedia secara open source bagi manufaktur perangkat keras untuk memodifikasi sesuai kebutuhan. Meskipun konfigurasi perangkat Android tidak sama antara satu dengan yang lain, namun Android sendiri mendukung beberapa fitur berikut ini:

a. Penyimpanan (Storage) - menggunakan SQLite yang merupakan basis data relasional yang ringan untuk menyimpan data.

b. Koneksi(Connectivity) - Mendukung GSM/EDGE, IDEN, CDMA, EVDO, UMTS, Bluetooth, WiFi, LTE, dan WiMAX.

c. Pesan (Messaging) Mendukung SMS dan MMS

d. Web Browser - Menggunakan open source WebKit termasuk di dalamnya engine Chrome V8 JavaScript.

e. Media - media di dukung antara lain : H.263, H.264, MPEG-4 SP, AMR, AMR-WB, AAC, HE-AAC, MP3, MIDI, Ogg Vorbis, JPEG, PNG, GIF, dan BMP.

f. Perangkat keras - terdapat Accelerometer Sensor, kamera, kompas digital, sensor proximity, dan GPS.

g. Multi-touch - mendukung layar multi-touch

h. Multi-tasking - mendukung aplikasi multi-tasking.

i. Dukungan Flash - Android 2.3 mendukung Flash 10.1

**3.3.2 Android Studio**



**Gambar 3.1** Android Studio

**Android Studio** adalah sebuah **IDE** untuk Android Development yang dikenalkan pihak google pada acara Google I/O di tahun 2013. Android Studio merupakan suatu pengembangan dari Eclipse IDE, dan dibuat berdasarkan IDE Java populer, yaitu **IntelliJ IDEA.** Android Studio merupakan IDE resmi untuk pengembangan aplikasi Android. Berbeda dengan Eclipse yang menggunakan Ant, Android Studio menggunakan Gradle sebagai **build environment.**

Fitur-fitur lainnya yang sudah disematkan pada Android Studio adalah sebagai berikut :

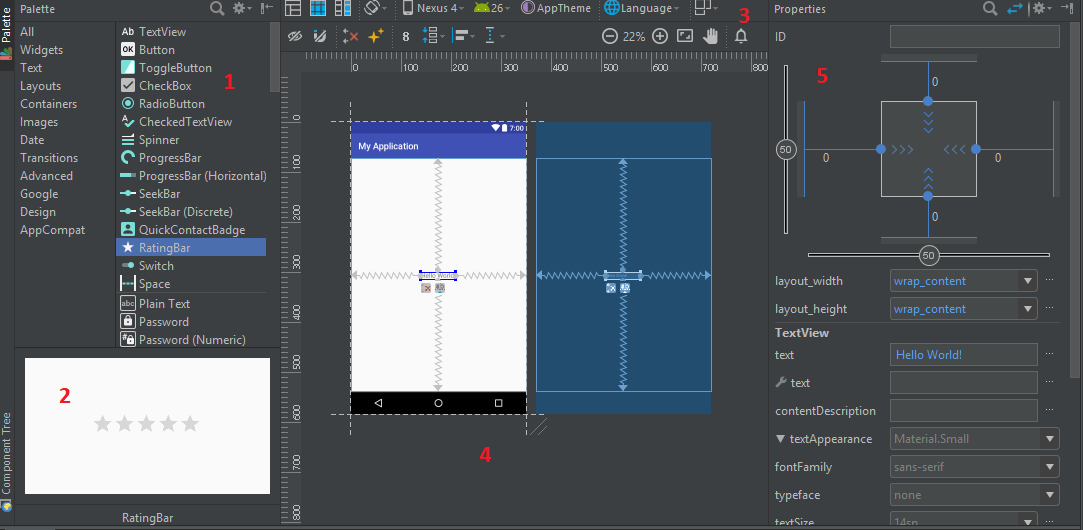
1. Menggunakan Gradle-based build system yang fleksibel.
2. Bisa mem-build multiple APK .
3. Template support untuk Google Services dan berbagai macam tipe perangkat.
4. Layout editor yang lebih bagus.
5. Built-in support untuk Google Cloud Platform, sehingga mudah untuk integrasi dengan Google Cloud Messaging dan App Engine.
6. Import library langsung dari Maven repository dan sebagainya.

**Sumber** : *Pengenalan Android Studio,* [*http://www.jadibaru.com/android/pengenalan-android-studio-2/*](http://www.jadibaru.com/android/pengenalan-android-studio-2/)

**3.3.3 Android Studio**

1. Layout

Pada Android Studio layout di definisikan menggunakan bahasa XML, layout di bedakan menjadi dua yaitu design dan text. Pada design, sama halnya Frame GUI di Java, menampilkan layout untuk mengatur komponen UI pada aplikasi yang kita buat. Kita dapat men drag and drop komponen yang tersedia pada tab Palette dan mengatur ukuran komponen di properties.

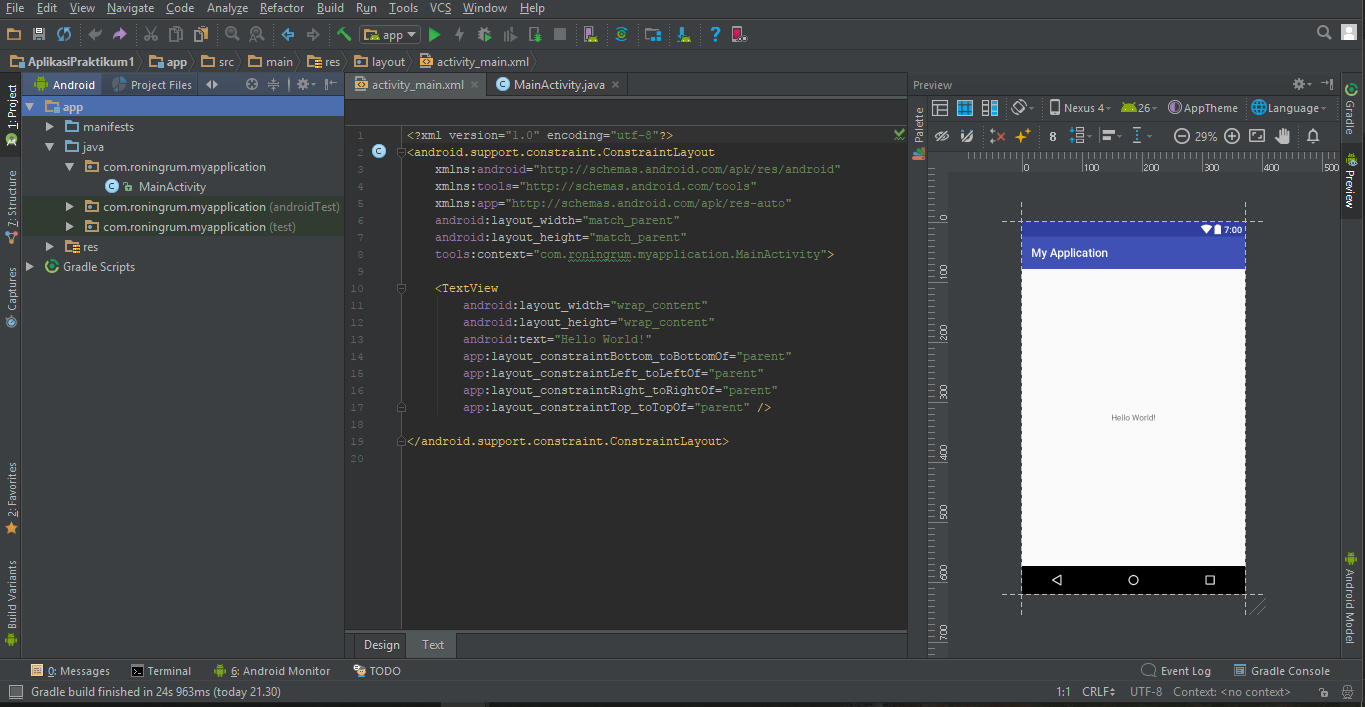


**Gambar 3.2** Layout Editor Mode Design pada Android Studio

Region atau daerah pada yang ada pada layout editor design dibagi menjadi 5 area yaitu:

1. Palette: Menyediakan daftar widget dan layout yang bisa Anda seret ke dalam layout di editor.
2. Component Tree: Menampilkan hierarki tampilan untuk layout Anda. Klik sebuah item di sini untuk melihatnya dipilih dalam editor.
3. Toolbar: Menyediakan tombol untuk mengonfigurasi penampilan layout di editor dan untuk mengedit properti layout.
4. Design Editor: Menampilkan layout Anda bersama tampilan Design dan Blueprint.
5. Properties: Menyediakan kontrol properti untuk tampilan yang dipilih saat ini.

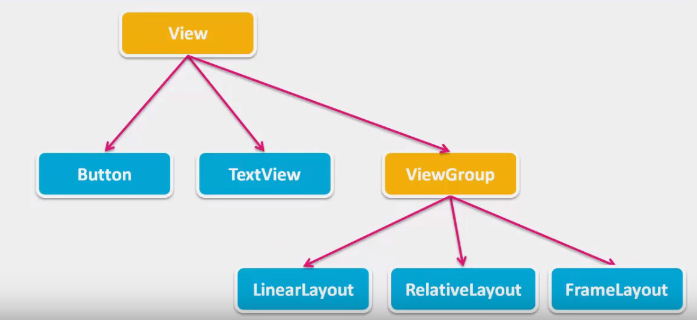
Sementara pada mode text, akan menampilkan barisan kode atau barisan susunan kode dari text XML nya. dalam Android studio sudah memiliki fitur **Live Preview**, jadi dengan adanya fitur ini, kalian tidak perlu bolak-balik ke mode design hanya untuk melihat hasil kode XML yang sudah dibuat atau edit.



**Gambar 3.3** Layout editor XML Mode Text pada Android Studio

Pada Layout di android studio setiap komponen GUInya disebut View, View adalah: basic class atau kelas dasar dari setiap komponen User Interface pada Android termasuk tata letak (layout).

Terdapat sebuah subclass dalam view yaitu viewgroup, view group merupakan class yang diperuntukan untuk layout saja.



**Gambar 3.4** Struktur View

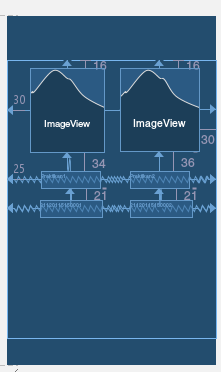
Layout adalah struktur visual untuk antarmuka pengguna, seperti User Interface sebuah [aktivitas](https://developer.android.com/guide/components/activities.html?hl=id) atau [widget aplikasi](https://developer.android.com/guide/topics/appwidgets/index.html?hl=id). Mendeklarasikan layout dengan dua cara:

* **Deklarasikan elemen UI dalam XML**. Android menyediakan sebuah kosakata XML sederhana yang sesuai dengan kelas dan subkelas View, seperti halnya untuk widget dan layout.
* **Buat instance elemen layout saat waktu proses**. Aplikasi Anda bisa membuat objek View dan ViewGroup (dan memanipulasi propertinya) lewat program.

Pada Layout terdapat kelas – kelas yang sering digunakan yaitu

1. Constraint Layout

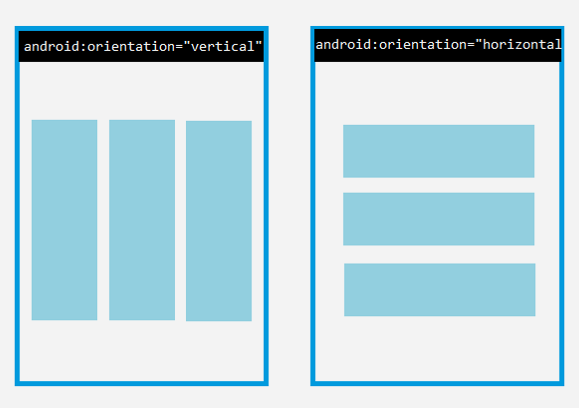
Constraint Layout adalah tampilan grup yang tersedia di pustaka Constraint Layout, yang telah disertakan bersama Android Studio 2.2 dan yang lebih tinggi. Layout ini dibangun dari nol bersama Layout Editor, jadi apa saja dapat diakses dari Design editor dan tidak perlu mengedit XML secara manual. Yang paling penting, sistem layout berbasis batasan ini memungkinkan untuk membuat hampir semua layout tanpa menambahkan tampilan grup apa pun.



**Gambar 3.5** Contoh Constraint Layout

1. Linear Layout

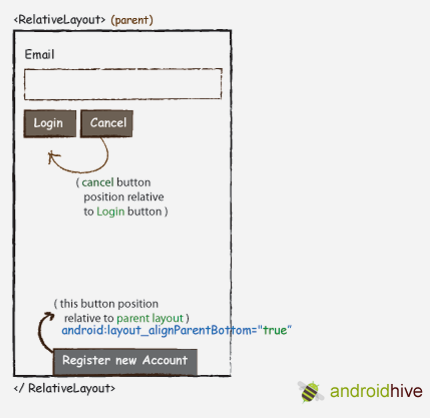
LinearLayout **a**kan menempatkan komponen-komponen didalamnya secara horizontal atau vertical ( menyamping atau menurun ). LinearLayout memiliki atribut weight untuk masing-masing child view yang terdapat didalam LinearLayout yang berguna untuk mengontrol porsi ukuran view secara Relatif dalam sebuah ruang (space) yang tersedia.



**Gambar 3.6** Linear Layout

1. Relative Layout

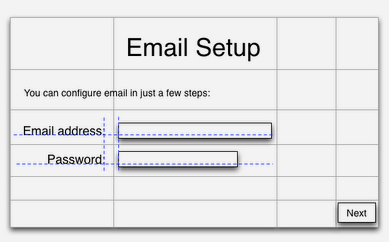
RelativeLayoutadalah layout yang menampilkan elemen-elemen view dalam posisi yang relatif. Posisi dari sebuah view yang dapat diletakkan relatif terhadap posisi elemen view di sekitarnya atau relatif terhadap area layout utama. Sebuah desain tampilan aplikasi bisa dibuat lebih mudah dan sederhana menggunakan RelativeLayout dibandingkan dengan LinearLayout.



**Gambar 3.7** Relative Layout

1. Grid Layout

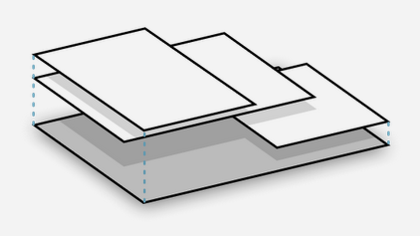
Diperkenalkan pada API level 14 (Android 4.o / Ice Cream Sandwich), layout ini akan memberikan kemudahan dengan mengakomodir komponen didalamnya ke dalam bentuk Grid (Kolom dan Baris). Dalam sebuah referensi, GridLayout merupakan komponen layout yang sangat flexibel dan dapat dimanfaatkan untuk menyederhanakan pembuatan Layout UI yang bersifat kompleks dan bersarang yang terdapat di komponen Layout lainnya.



**Gambar 3.8** Grid Layout

1. Frame Layout

Layout ini adalah layout yang paling sederhana. Layout ini akan membuat komponen yang ada didalamnya menjadi menumpuk atau saling menutupi satu dengan yang lainnya (*layering*). Komponen yang paling pertama pada layout ini akan berada dibawah komponen-komponen diatasnya. Pada materi penggunaan fragment di materi sebelumnya, FrameLayout memiliki kemampuan untuk menjadi container untuk fragment-fragment didalam sebuah Activity. Berikut ilustrasi dari penggunaan FrameLayout terhadap child view yang dimiliki didalamnya.



**Gambar 3.9** Frame Layout

**Sumber** : Android Layout,

[*https://blog.dicoding.com/android-module-5-android-layout/*](https://blog.dicoding.com/android-module-5-android-layout/)

3.3.4 SQLite

SQLite merupakan sistem manajemen database relasional yang dibangun dalam sebuah library bahasa pemrograman C. Berbeda dengan kebanyakan sistem manajemen database lainnya, SQLite bukanlah mesin database client-server melainkan berupa file tunggal atau beberapa file saja.

* + 1. Shared Preference

Sebuah penyimpanan data yang ada di dalam Android yang bersifat data primtif pada file internal aplikasi dengan pasangan key-value.

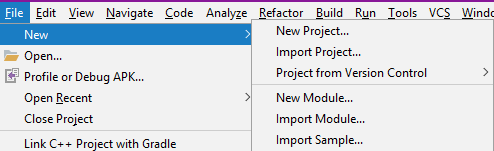
3.3.6 Intent

Intent

Intent adalah sebuah kelas dalam programming Android yang berfungsi untuk perpindahan halaman.Intent juga merupakan suatu objek yang terdapat dalam suatu activity dimana objek tersebut dapat komunikasi dengan activity yang lain, baik activity pada fungsi internal android misal seperti memanggil activity dalam satu package atau beda package yang masih berada dalam satu project.Intent merupakan objek tipe android.content.Intent. Melalui metode startActivity() yang digunakan untuk memulai sebuah activity lain.

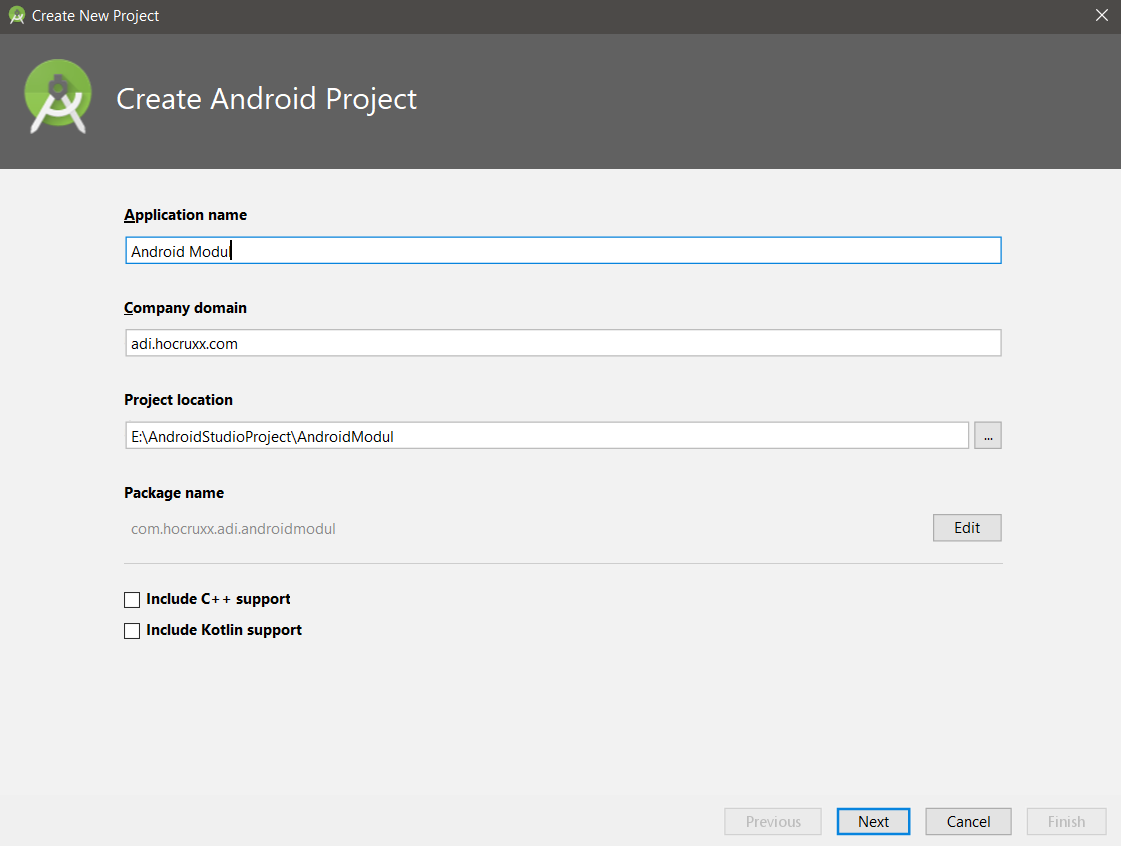
Intent dibagi menjadi 2, yaitu :

1. Explicit Intent berfungsi untuk mengaktifkan komponen-komponen dalam satu aplikasi yang sama. Misalnya seperti : Berpindah Activity.
2. Implicit Intent berfungsi untuk memanggil fungsi activity yang sudah ada di fungsi internal android seperti Dial Number, Open Browser dan lainnya.
3. **Langkah Percobaan**
   * 1. **Mengatur Device Android**
4. Pengaturan  Opsi Pengembang  Ubah menjadi “aktif” aktifkan Debugging USB
   * 1. **Membuat Project Aplikasi Android**
5. Buka Android Studio
6. Klik menu File  New  New Project



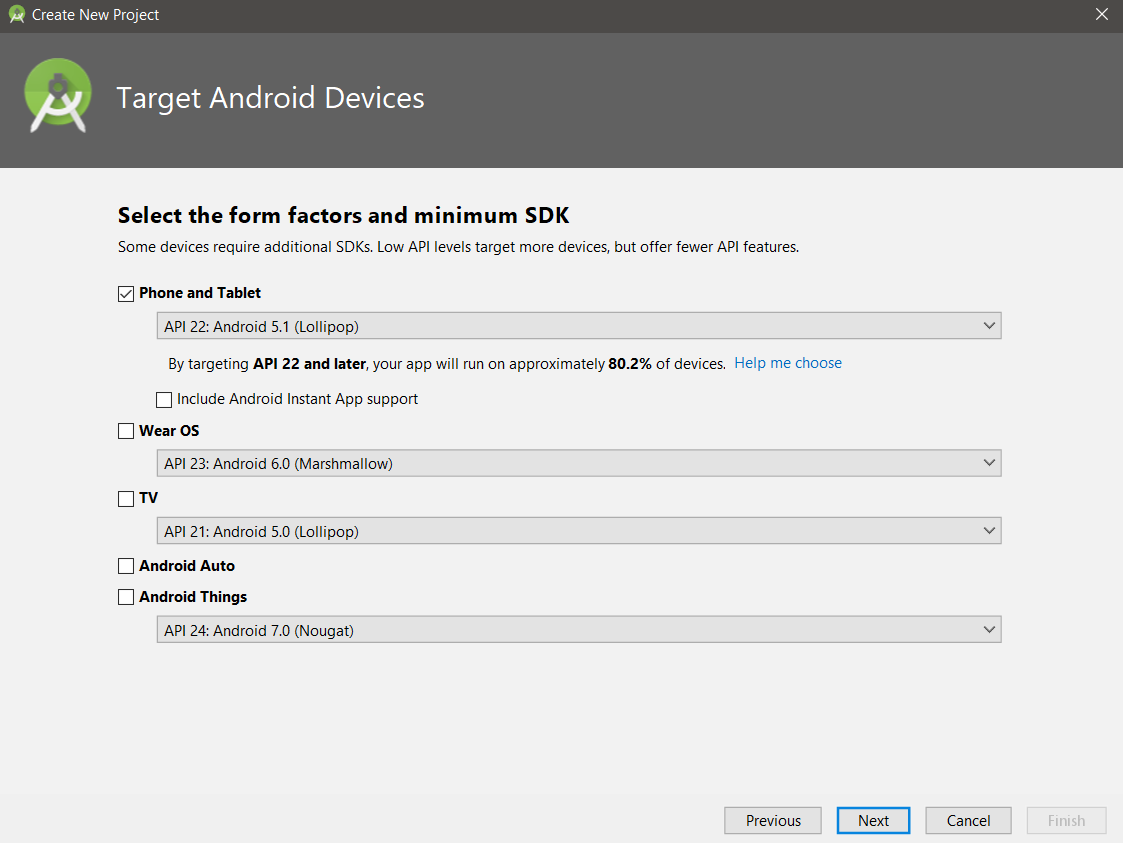
**Gambar 3.19** New Project

1. Buat Project Baru dengan nama AplikasiModul2Kelxx



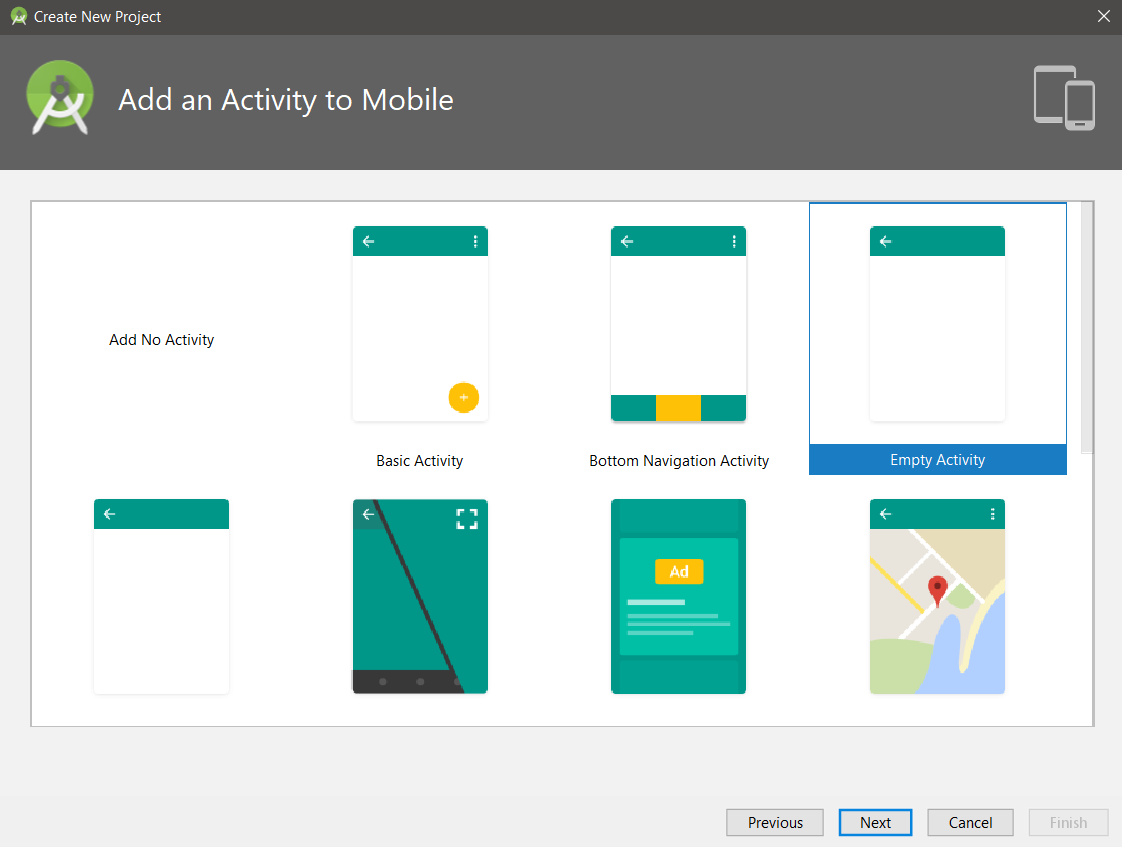
**Gambar 3.20** Tab Create New Project

1. Kemudian pilih min.SDK sesuai dengan kebutuhan pengguna. Namun, pada praktikum, menggunakan min SDK 22 yaitu Android Lollipop.



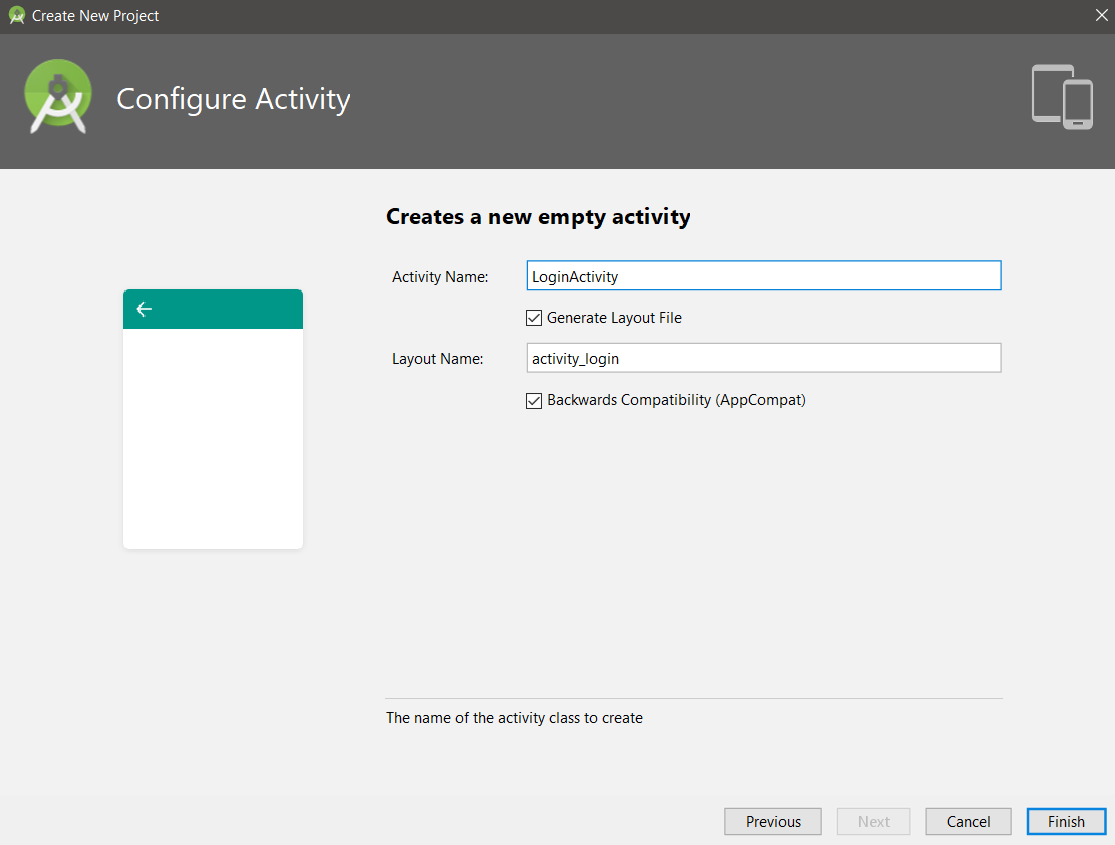
**Gambar 3.21** Target Android Device

1. Kemudian pilih jenis activity yaitu empty activity

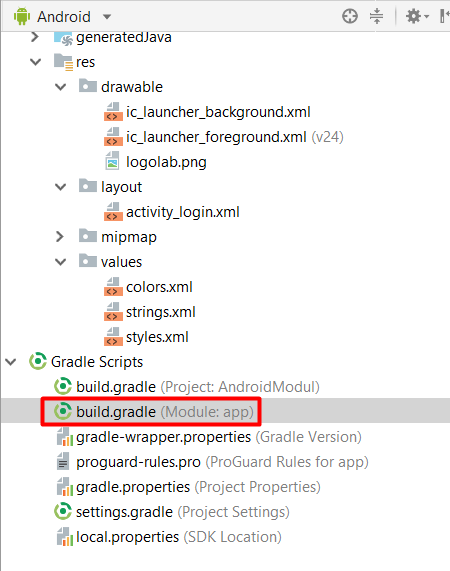


**Gambar 3.22** Pilihan Activity

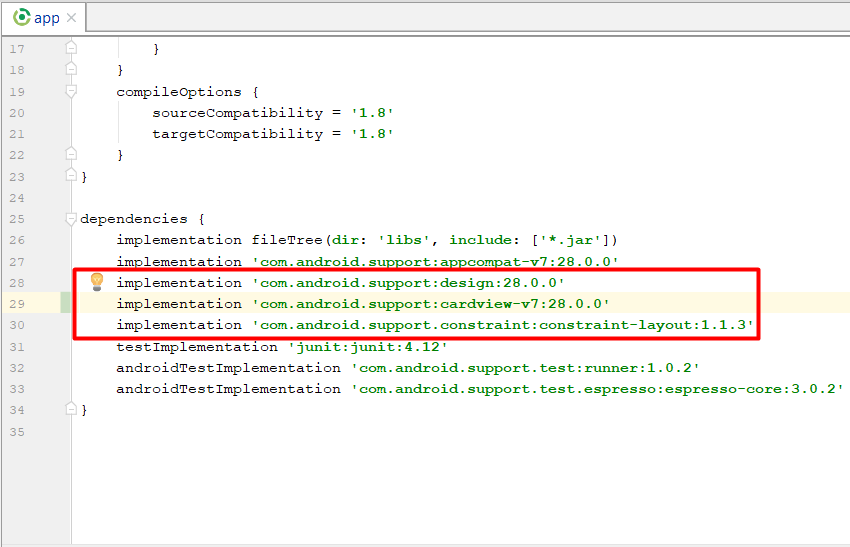
1. Kemudian, beri nama activity dengan nama ‘Login Activity’ lalu klik finish. Kemudian, tunggu sampai proses gradle selesai.



**Gambar 3.23** Isi Nama Activity

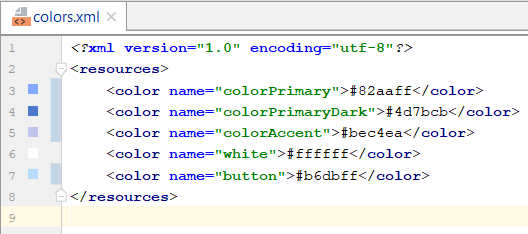


1. Setelah itu kita cari build.gradle(Module:app) untuk men-copy dependency material, berikut *sourcecode* nya:



1. Tambahkan beberapa code di build.gradle yang telah kita *click* sebelumnya. Kurang lebih seperti pada gambar di atas.
2. Pertama, kita akan desain tampilan colors, strings, dan styles nya untuk color string dan style xml terletak di folder res – value di panel tree pada android studio

. Berikut *sourcecode*  dari colors.xml :



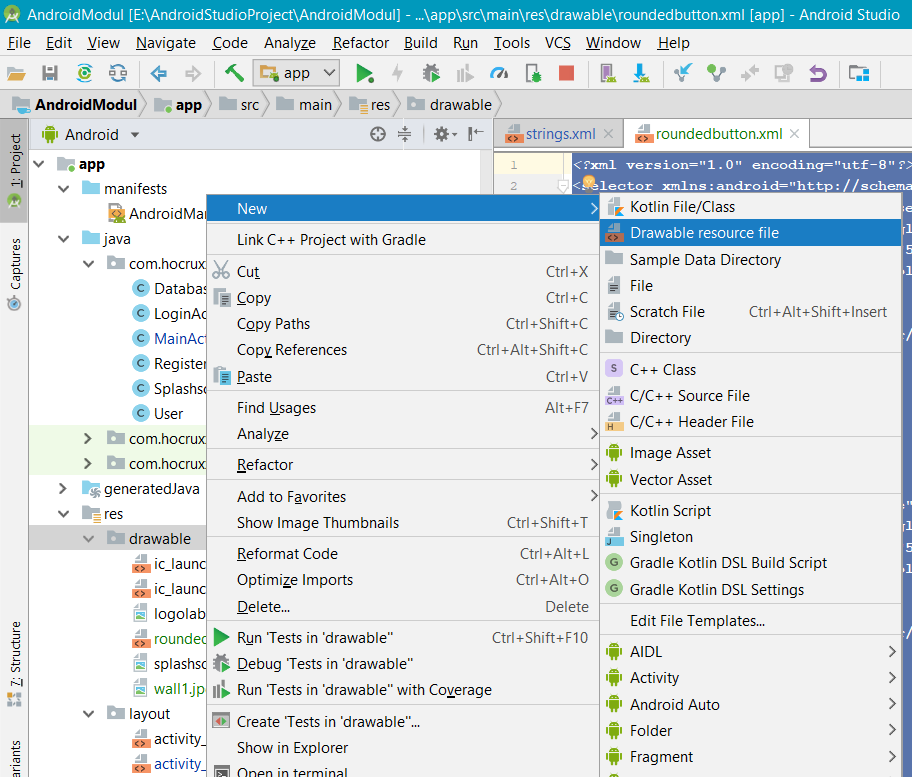
1. Setelah itu kita masuk ke strings.xml. Sesuai Namanya file xml ini berguna untuk menyimpan nilai string, berikut *sourcecode* nya:

<**resources**>  
 <**string name="app\_name"**>Android Modul</**string**>  
 <**string name="tittle"**>PRAKTIKUM MDP MODUL ANDROID</**string**>  
 <**string name="regtittle"**>REGISTRASI DATA</**string**>  
 <**string name="user"**>username</**string**>  
 <**string name="password"**>password</**string**>  
 <**string name="login"**>masuk</**string**>  
 <**string name="register"**>register</**string**>  
 <**string name="logout"**>Logout</**string**>  
 <**string name="Header"**>Selamat datang Di Praktikum MDP 2018</**string**>  
 <**string name="name"**>Nama Kamu</**string**>  
 <**string name="description"**>Selanjutnya akan di jelaskan di modul android berikutnya Mengenai API dan MVP</**string**>  
</**resources**>

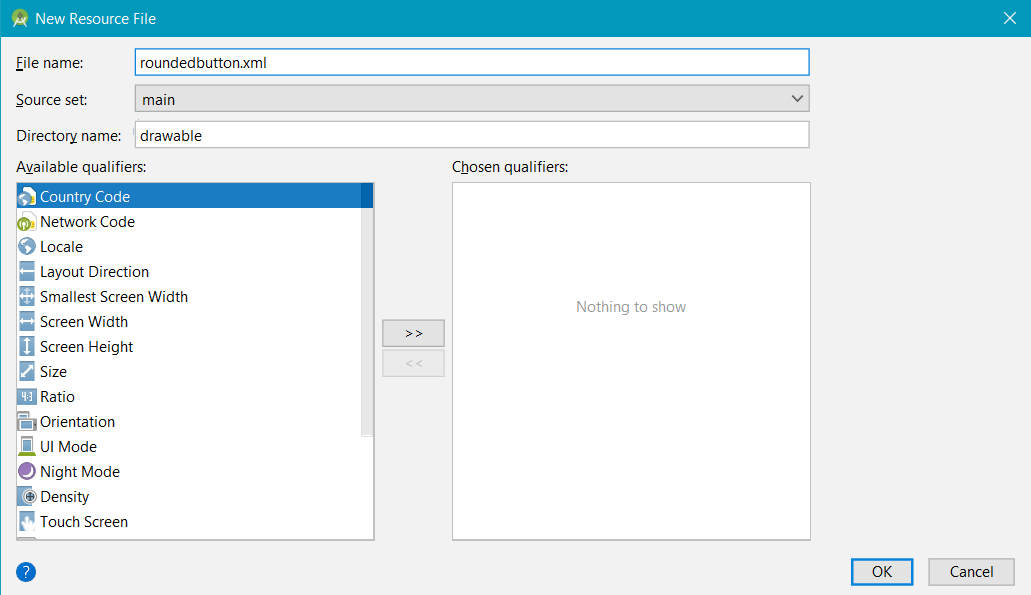
1. Kemudian di dalam styles.xml, di dalam *values* ini mengatur memprsentasikan *style* atau ketika dalam bahas pemrograman web, seperti file css (*cascading style sheet*) yang mengatur style tampilan aplikasi android studio, berikut *sourcecode* nya :

<**resources**>  
  
 *<!-- Base application theme. -->* <**style name="AppTheme" 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**>

1. Kemudian buat drawable resource file untuk membuat desain tombol di aplikasi melengkung, dengan mengklik kanan folder drawable dan pilih new kemudian klik menu drawable resource file.



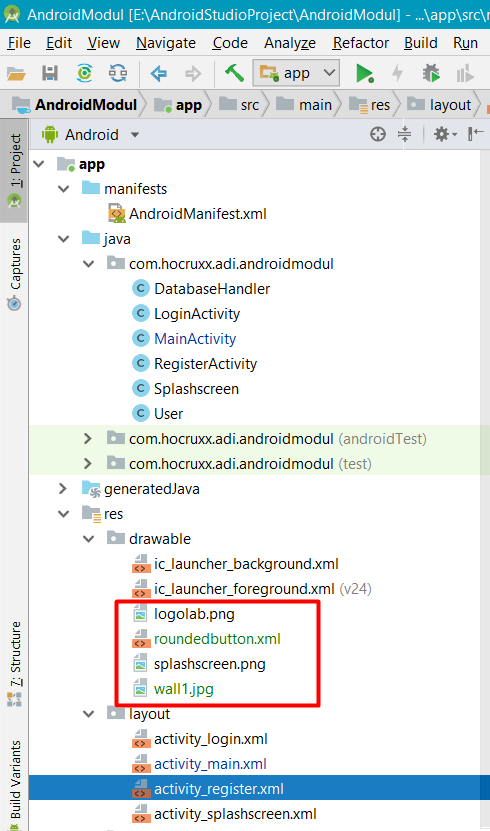
1. Buat file dengan nama roundedbutton.xml



1. Buke file tersebut dan copy kan source codenya

*<?***xml version="1.0" encoding="utf-8"***?>*<**selector xmlns:android="http://schemas.android.com/apk/res/android"**>  
 <**item android:state\_pressed="false"**>  
 <**shape android:shape="rectangle"**>  
 <**corners android:radius="5dp"** />  
 <**solid android:color="@color/button"** />  
 <**stroke  
 android:width="1dp"  
 android:color="@color/button"** />  
 <**padding  
 android:bottom="4dp"  
 android:left="4dp"  
 android:right="4dp"  
 android:top="4dp"** />  
 </**shape**>  
 </**item**>  
 <**item android:state\_pressed="true"**>  
 <**shape android:shape="rectangle"**>  
 <**corners android:radius="5dp"** />  
 <**solid android:color="@color/button"** />  
 <**stroke  
 android:width="1dp"  
 android:color="@color/button"** />  
 <**padding  
 android:bottom="4dp"  
 android:left="4dp"  
 android:right="4dp"  
 android:top="4dp"** />  
 </**shape**>  
 </**item**>  
</**selector**>

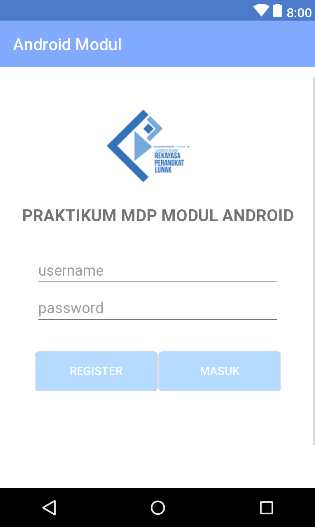
1. Setelah mengatur di folder values dan drawable, kemudian kita akan memasukkan sebuah gambar gambar yang akan kita gunakan nantinya untuk tampilan aplikasinya. Kita masukkan logonya di folder res🡪drawable ( drag and drop di folder ini ).



1. Masuk ke bagian design activity yang akan kita atur melalui folder res ->layout, yaitu activiy\_login.xml. Berikut *sourcecode* nya :

*<?***xml version="1.0" encoding="utf-8"***?>*<**ScrollView 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="@color/white"  
 tools:context=".LoginActivity"**>  
  
 <**RelativeLayout  
 android:id="@+id/rlLogin"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="60dp"**>  
  
 <**ImageView  
 android:id="@+id/ivLogin"  
 android:layout\_width="100dp"  
 android:layout\_height="100dp"  
 android:layout\_gravity="center\_horizontal"  
 android:layout\_marginLeft="130dp"  
 android:src="@drawable/logolab"  
 android:contentDescription="LOGO"** />  
  
 <**android.support.v7.widget.AppCompatTextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="@string/tittle"  
 android:textAlignment="center"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:layout\_marginTop="120dp"** />  
  
  
 <**LinearLayout  
 android:id="@+id/llLogin"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerInParent="true"  
 android:layout\_marginTop="320dp"  
 android:layout\_marginEnd="50dp"  
 android:orientation="vertical"  
 android:padding="18dp"**>  
  
 <**EditText  
 android:id="@+id/etUsernameLogin"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="@string/user"  
 android:inputType="textEmailAddress"  
 android:maxLines="1"  
 android:singleLine="true"** />  
  
 <**EditText  
 android:id="@+id/etPasswdLogin"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="30dp"  
 android:hint="@string/password"  
 android:imeActionId="6"  
 android:imeActionLabel="Sign In"  
 android:imeOptions="actionUnspecified"  
 android:inputType="textPassword"  
 android:maxLines="1"  
 android:singleLine="true"** />  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
  
 <**Button  
 android:id="@+id/btnSignUpLogin"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:background="@drawable/roundedbutton"  
 android:text="@string/register"  
 android:textColor="@color/white"**/>  
  
 <**LinearLayout  
 android:layout\_width="2dp"  
 android:layout\_height="2dp"  
 android:layout\_weight="0"  
 android:orientation="vertical"**></**LinearLayout**>  
  
 <**Button  
 android:id="@+id/btnSignInLogin"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:background="@drawable/roundedbutton"  
 android:text="@string/login"  
 android:textColor="@color/white"** />  
 </**LinearLayout**>  
 </**LinearLayout**>  
 </**RelativeLayout**>  
</**ScrollView**>

1. Maka tampilannya akan seperti gambar dibawah ini

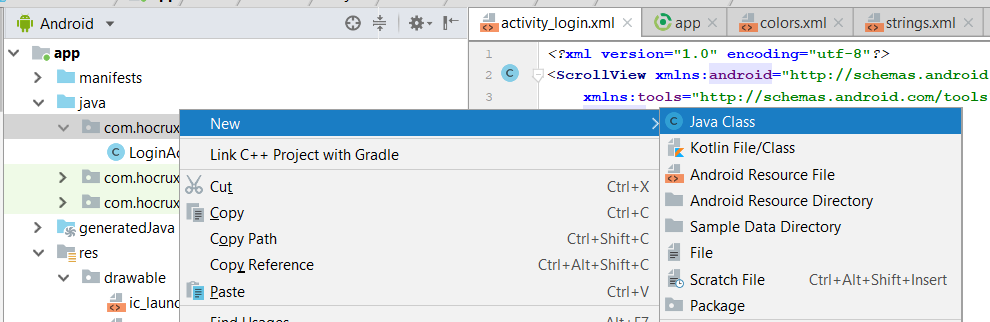


**Gambar 3.24** Layout activity\_login.xml

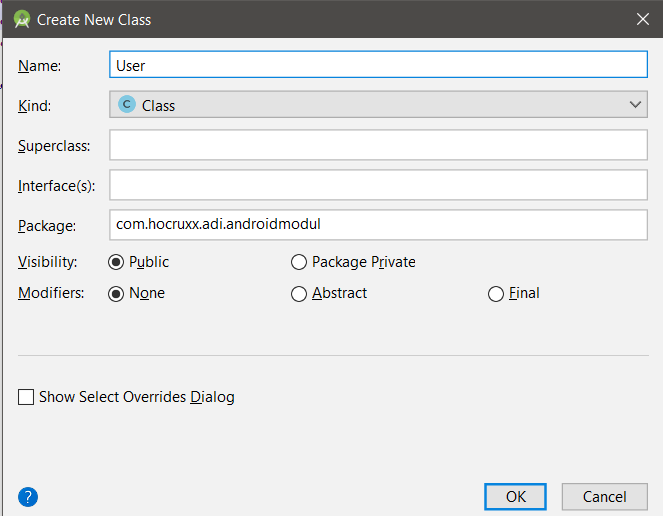
1. Setelah itu kita masukkan *sourcecode* LoginActiviy.java yang digunakan untuk masuk ke dalam aplikasi nya. Kali ini kita menggunakan SharedPreferences untuk menyimpan database username dan password yang kita masukkan. Sifat SharedPreferences hamper sama dengan kebanyakan database layaknya SQLite tapi hanya bisa menyimpan bentuk data yang sederhana dan lebih kecil.

**import** android.content.Context;  
**import** android.content.Intent;  
**import** android.content.SharedPreferences;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.text.TextUtils;  
**import** android.util.Log;  
**import** android.util.Printer;  
**import** android.view.KeyEvent;  
**import** android.view.View;  
**import** android.view.inputmethod.EditorInfo;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
**import** android.widget.TextView;  
**import** android.widget.Toast;  
  
**import** java.security.PrivateKey;  
  
**public class** LoginActivity **extends** AppCompatActivity {  
  
 **private** EditText **etUsernameLogin**, **etPasswdLogin**;  
 **private** Button **btnSignUpLogin**, **btnSignInLogin**;  
 String **username**, **password**;  
 **private** User **user**;  
 **private** DatabaseHandler **presenter**;  
  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_login***);  
  
 initView();  
 initData();  
 login();  
 register();  
 }  
  
 **private void** initView() {  
 **etUsernameLogin** = findViewById(R.id.***etUsernameLogin***);  
 **etPasswdLogin** = findViewById(R.id.***etPasswdLogin***);  
 **btnSignInLogin** = findViewById(R.id.***btnSignInLogin***);  
 **btnSignUpLogin** = findViewById(R.id.***btnSignUpLogin***);  
 }  
  
 **private void** register() {  
 **btnSignUpLogin**.setOnClickListener(V -> {  
 Intent reg = **new** Intent(**this**, RegisterActivity.**class**);  
 startActivity(reg);  
 finish();}  
 );  
 }  
  
 **private void** initData() {  
 **user** = **new** User();  
 **presenter** = **new** DatabaseHandler(**this**);  
 }  
  
 **private** Boolean validation() {  
 **username** = **etUsernameLogin**.getText().toString();  
 **password** = **etPasswdLogin**.getText().toString();  
  
 **if**(**username**.isEmpty()){  
 Toast.*makeText*(**this**, **"Isikan username"**, Toast.***LENGTH\_SHORT***).show();  
 Log.*e*(**"Validation"**,**"false"**);  
 **return false**;  
 }  
  
 **if**(**password**.isEmpty()){  
 Toast.*makeText*(**this**, **"Isikan Password"**, Toast.***LENGTH\_SHORT***).show();  
 Log.*e*(**"Validation"**,**"false"**);  
 **return false**;  
 }  
  
 Log.*e*(**"Validation"**,**"true"**);  
 **return true**;  
 }  
  
 **private void** login() {  
 **btnSignInLogin**.setOnClickListener(v -> actLogin());  
 }  
  
 **private void** actLogin(){  
 **if**(validation()){  
 **if**(loginData()){  
 Log.*e*(**"actLogin"**,**"true"**);  
 initPreference();  
 Intent main = **new** Intent(**this**, MainActivity.**class**);  
 startActivity(main);  
 finish();  
 } **else** {  
 Log.*e*(**"actLogin"**,**"false"**);  
 Toast.*makeText*(**this**, **"Login gagal"**, Toast.***LENGTH\_LONG***).show();  
 }  
 }  
 }  
  
 **private** Boolean loginData(){  
 **user** = **presenter**.getLogin(**username**);  
  
 **if** (**password**.equals(**user**.getPassword())){  
 Log.*e*(**"loginData"**, **"true"**);  
 **return true**;  
 } **else** {  
 Log.*e*(**"loginData"**, **"false"**);  
 **return false**;  
 }  
 }  
  
 **private void** initPreference() {  
 SharedPreferences preferences = getSharedPreferences(**"LoginPreference"**, Context.***MODE\_PRIVATE***);  
 SharedPreferences.Editor editor = preferences.edit();  
 editor.putString(**"password"**, **password**);  
 editor.putString(**"username"**, **username**);  
 editor.commit();  
 editor.apply();  
 }  
}

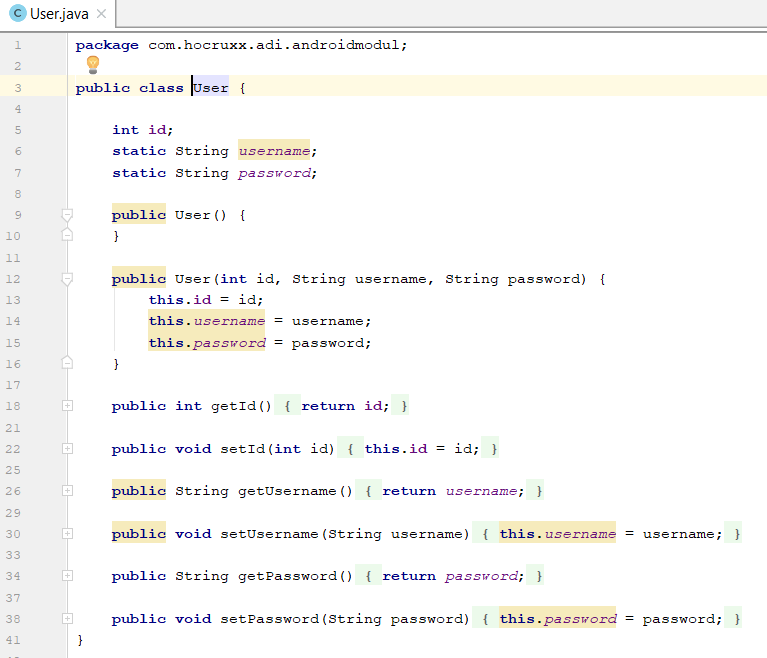
1. Kemudian kita akan membuat sebuah class baru yang berfungsi untuk menampung segala kerperluan seorang user yang bersikan username dan password.



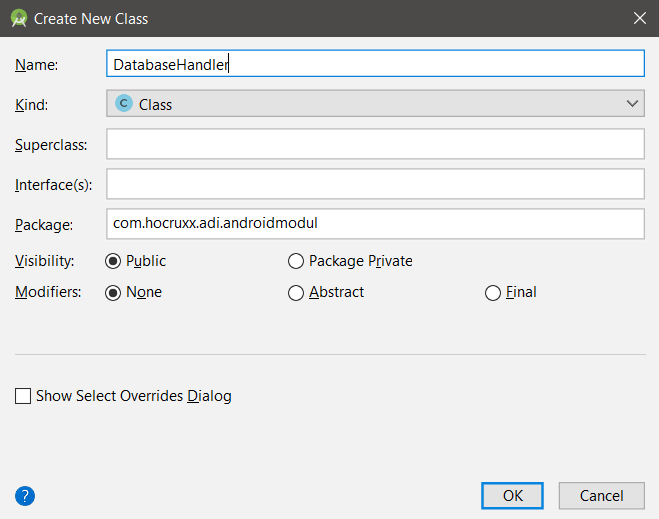
1. Kita namani class nya “User” (Di pemograman Bahasa java penulisan Casesensitive).



1. Setelah membuat class nya kita masukkan *sourcecode* nya



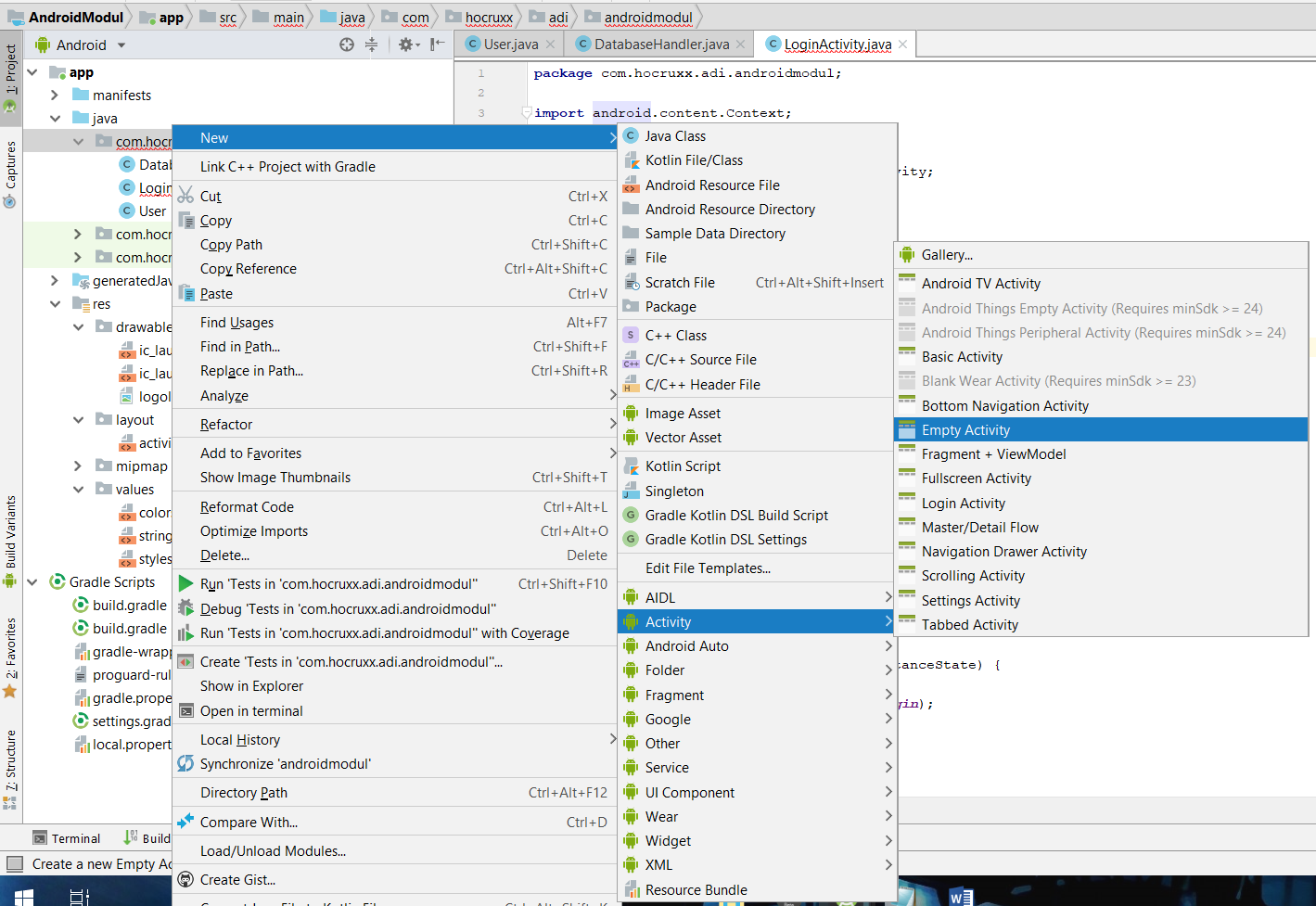
1. Kita membuat class baru lagi yang bernama “DatabaseHandler” yang berguna untuk operasi database create, update, update, dan delete



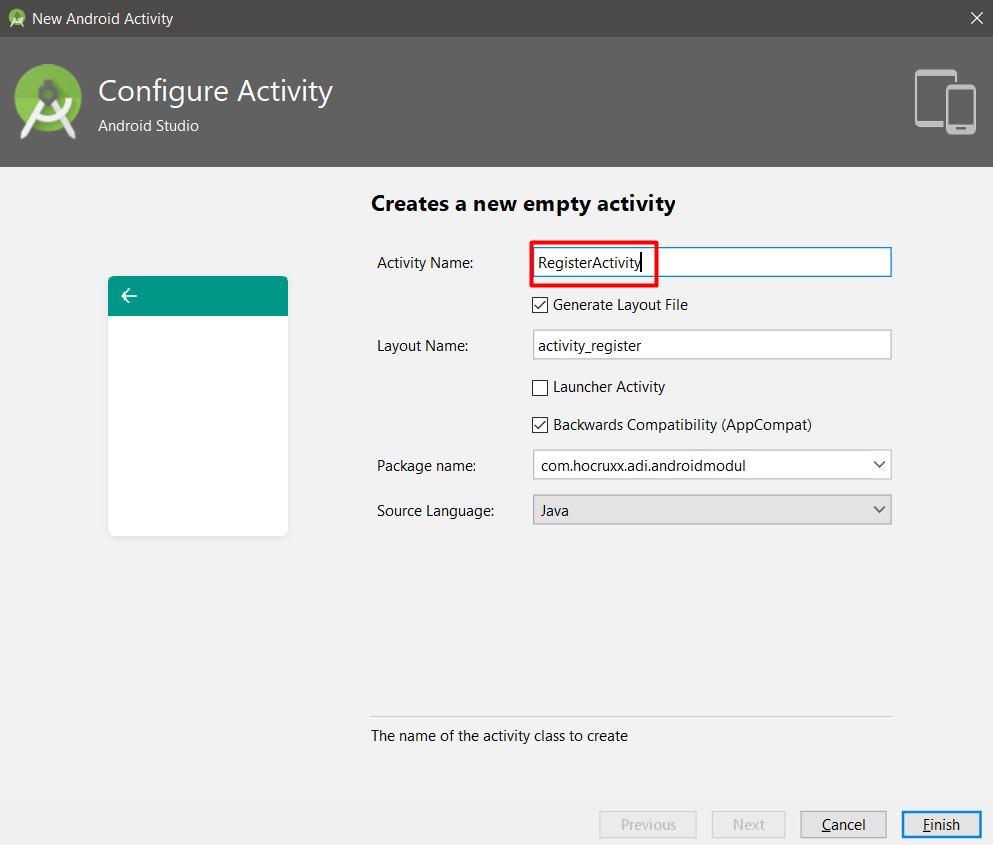
1. Kemudian, masukkan *sourcecode* DatabaseHandler nya

**import** android.content.ContentValues;  
**import** android.content.Context;  
**import** android.database.Cursor;  
**import** android.database.sqlite.SQLiteDatabase;  
**import** android.database.sqlite.SQLiteOpenHelper;  
**import** android.util.Log;  
  
**import** java.util.ArrayList;  
**import** java.util.List;  
  
**public class** DatabaseHandler **extends** SQLiteOpenHelper {  
  
 *// static variable* **private static final int *DATABASE\_VERSION*** = 1;  
  
 *// Database name* **private static final** String ***DATABASE\_NAME*** = **"modulandroid"**;  
  
 *// table name* **private static final** String ***TABLE\_MAHASISWA*** = **"mahasiswa"**;  
  
 *// column tables* **private static final** String ***KEY\_ID*** = **"id"**;  
 **private static final** String ***KEY\_NAME*** = **"username"**;  
 **private static final** String ***KEY\_TALL*** = **"password"**;  
  
 **public** DatabaseHandler(Context context){  
 **super**(context, ***DATABASE\_NAME***, **null**, ***DATABASE\_VERSION***);  
 }  
  
 *//Create table* @Override  
 **public void** onCreate(SQLiteDatabase db) {  
 String CREATE\_MAHASISWA\_TABLE = **"CREATE TABLE "** + ***TABLE\_MAHASISWA*** + **"("** + ***KEY\_ID*** + **" INTEGER PRIMARY KEY,"** + ***KEY\_NAME*** + **" TEXT,"** + ***KEY\_TALL*** + **" TEXT"** + **")"**;  
 db.execSQL(CREATE\_MAHASISWA\_TABLE);  
 }  
  
 *// on Upgrade database* @Override  
 **public void** onUpgrade(SQLiteDatabase db, **int** oldVersion, **int** newVersion) {  
 db.execSQL(**"DROP TABLE IF EXISTS "** + ***TABLE\_MAHASISWA***);  
 onCreate(db);  
 }  
  
 **public void** addUser(User usermodel){  
 SQLiteDatabase db = getWritableDatabase();  
 ContentValues values = **new** ContentValues();  
 values.put(***KEY\_NAME***, usermodel.getUsername());  
 values.put(***KEY\_TALL***, usermodel.getPassword());  
 db.insert(***TABLE\_MAHASISWA***, **null**, values);  
 db.close();  
 }  
  
 **public** User getMahasiswa(**int** id) {  
 SQLiteDatabase db = **this**.getReadableDatabase();  
  
 Cursor cursor = db.query(***TABLE\_MAHASISWA***, **new** String[] { ***KEY\_ID***,  
 ***KEY\_NAME***, ***KEY\_TALL*** }, ***KEY\_ID*** + **"=?"**,  
 **new** String[] { String.*valueOf*(id) }, **null**, **null**, **null**, **null**);  
 **if** (cursor != **null**)  
 cursor.moveToFirst();  
  
 User mahasiswa = **new** User(Integer.*parseInt*(cursor.getString(0)),  
 cursor.getString(1), cursor.getString(2));  
 *// return mahasiswa* **return** mahasiswa;  
 }  
  
 **public** User getLogin(String username) {  
 User mahasiswa;  
 SQLiteDatabase db = **this**.getReadableDatabase();  
  
 Cursor cursor = db.query(***TABLE\_MAHASISWA***, **new** String[] { ***KEY\_ID***,  
 ***KEY\_NAME***, ***KEY\_TALL*** }, ***KEY\_NAME*** + **"=?"**,  
 **new** String[] { username }, **null**, **null**, **null**, **null**);  
  
 **if** (cursor != **null**) {  
 cursor.moveToFirst();  
 }  
  
 **int** nomor = cursor.getCount();  
  
 **if**(nomor>0){  
 mahasiswa = **new** User(Integer.*parseInt*(cursor.getString(0)),  
 cursor.getString(1), cursor.getString(2));  
 } **else** {  
 mahasiswa = **new** User(0,  
 **"failed"**, **"failed"**);  
 }  
  
 *// return mahasiswa* **return** mahasiswa;  
 }  
  
 *// get All Record* **public** List<User> getAllRecord() {  
 List<User> mahasiswaList = **new** ArrayList<User>();  
 *// Select All Query* String selectQuery = **"SELECT \* FROM "** + ***TABLE\_MAHASISWA***;  
  
 SQLiteDatabase db = **this**.getWritableDatabase();  
 Cursor cursor = db.rawQuery(selectQuery, **null**);  
  
 **if** (cursor.moveToFirst()) {  
 **do** {  
 User userModels = **new** User();  
 userModels.setId(Integer.*parseInt*(cursor.getString(0)));  
 userModels.setUsername(cursor.getString(1));  
 userModels.setPassword(cursor.getString(2));  
  
 mahasiswaList.add(userModels);  
 } **while** (cursor.moveToNext());  
 }  
  
 *// return mahasiwa list* **return** mahasiswaList;  
 }  
  
 **public int** updateMahasiswa(User mahasiswa) {  
 SQLiteDatabase db = **this**.getWritableDatabase();  
  
 ContentValues values = **new** ContentValues();  
 values.put(***KEY\_NAME***, mahasiswa.getUsername());  
 values.put(***KEY\_TALL***, mahasiswa.getPassword());  
  
 *// updating row* **return** db.update(***TABLE\_MAHASISWA***, values, ***KEY\_ID*** + **" = ?"**,  
 **new** String[] { String.*valueOf*(mahasiswa.getId()) });  
 }  
  
 **public void** deleteModel(User mahasiswa) {  
 SQLiteDatabase db = **this**.getWritableDatabase();  
 db.delete(***TABLE\_MAHASISWA***, ***KEY\_ID*** + **" = ?"**,  
 **new** String[] { String.*valueOf*(mahasiswa.getId()) });  
 db.close();  
 }  
  
}

1. Untuk menambahkan seorang user dalam aplikasi, maka dibutuhkan suatu Register, kita akan membuat activity baru yang bernama Register Activity



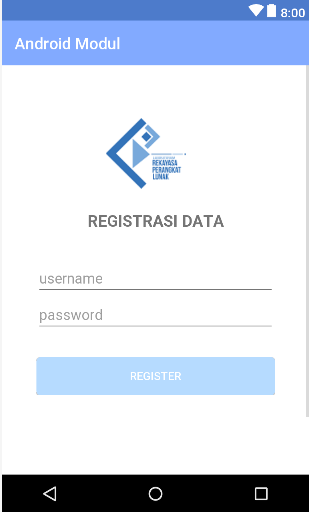
1. Activity Name nya kita beri nama “RegisterActivity” dan Layout Name nya kita beri nama “activity\_register” . Kemudian kalau sudah benar konfigurasinya maka klik tombol finish untuk menyelesaikannya



1. Setelah Activity nya terbuat, kita bikin design nya terlebih dahulu, melalu folder res 🡪 layout, berikut source code nya :

*<?***xml version="1.0" encoding="utf-8"***?>*<**ScrollView 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="@color/white"  
 tools:context=".LoginActivity"**>  
  
 <**RelativeLayout  
 android:id="@+id/rlLogin"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="60dp"**>  
  
 <**ImageView  
 android:id="@+id/ivLogin"  
 android:layout\_width="100dp"  
 android:layout\_height="100dp"  
 android:layout\_gravity="center\_horizontal"  
 android:layout\_marginLeft="130dp"  
 android:src="@drawable/logolab"  
 android:contentDescription="LOGO"** />  
  
 <**android.support.v7.widget.AppCompatTextView  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:text="@string/regtittle"  
 android:textAlignment="center"  
 android:textSize="20sp"  
 android:textStyle="bold"  
 android:layout\_marginTop="120dp"** />  
  
  
 <**LinearLayout  
 android:id="@+id/llLogin"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerInParent="true"  
 android:layout\_marginTop="330dp"  
 android:layout\_marginEnd="50dp"  
 android:orientation="vertical"  
 android:padding="18dp"**>  
  
 <**EditText  
 android:id="@+id/etUsernameRegister"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="@string/user"  
 android:inputType="textEmailAddress"  
 android:maxLines="1"  
 android:singleLine="true"** />  
  
 <**EditText  
 android:id="@+id/etPasswordRegister"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="30dp"  
 android:hint="@string/password"  
 android:imeActionId="6"  
 android:imeActionLabel="Sign In"  
 android:imeOptions="actionUnspecified"  
 android:inputType="textPassword"  
 android:maxLines="1"  
 android:singleLine="true"** />  
  
 <**LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
  
 <**Button  
 android:id="@+id/btRegister"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1" android:background="@drawable/roundedbutton"  
 android:text="@string/register"  
 android:textColor="@color/white"** />  
  
 </**LinearLayout**>  
 </**LinearLayout**>  
 </**RelativeLayout**>  
</**ScrollView**>

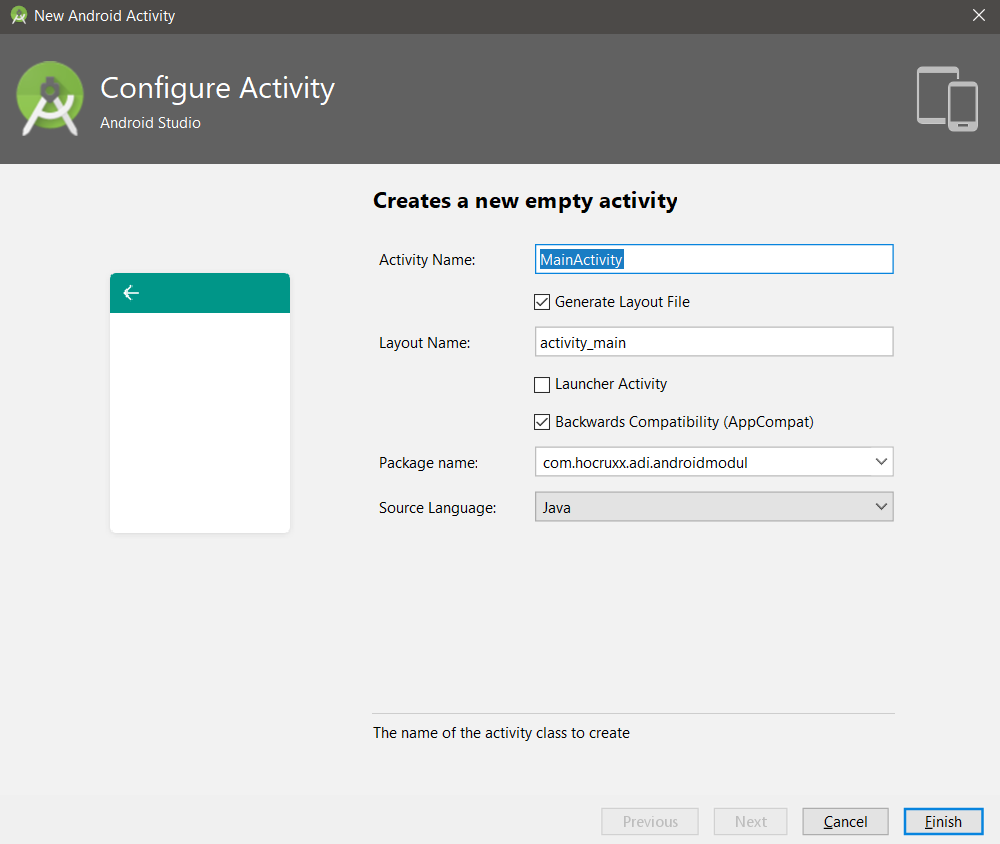
1. Tampilannya akan muncul seperti gambar di bawah ini



1. Selanjutnya kita masukkan *sourcecode* Regsiter Activity nya

**import** android.content.Intent;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.util.Log;  
**import** android.widget.Button;  
**import** android.widget.EditText;  
  
**public class** RegisterActivity **extends** AppCompatActivity {  
  
 **private** DatabaseHandler **databaseHandler**;  
 **private** User **usermodel**;  
 **private** EditText **etUsernameRegister**;  
 **private** EditText **etPasswordRegister**;  
 **private** Button **btRegister**;  
 **private** String **username**,**password**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_register***);  
  
 initView();  
 register();  
 }  
  
 **private void** register() {  
 **btRegister**.setOnClickListener(V ->  
 initDataHandler()  
 );  
 }  
  
 **private void** initView() {  
 **etUsernameRegister** = findViewById(R.id.***etUsernameRegister***);  
 **etPasswordRegister** = findViewById(R.id.***etPasswordRegister***);  
 **btRegister** = findViewById(R.id.***btRegister***);  
 }  
  
 **private void** initUser() {  
 **username** = **etUsernameRegister**.getText().toString();  
 **password** = **etPasswordRegister**.getText().toString();  
  
 **usermodel** = **new** User();  
 **usermodel**.setUsername(**username**);  
 **usermodel**.setPassword(**password**);  
 }  
  
 **private void** initDataHandler() {  
 initUser();  
  
 **databaseHandler** = **new** DatabaseHandler(**this**);  
 **databaseHandler**.addUser(**usermodel**);  
 User model = **databaseHandler**.getMahasiswa(1);  
 Log.*e*(**"record"**, model.getUsername().toString());  
 Intent admin = **new** Intent(RegisterActivity.**this**, LoginActivity.**class**);  
 startActivity(admin);  
 finish();  
 }  
}

1. Activity login dan register sudah kita buat, maka kita akan membuat Activity baru lagi, tahap membuat activity baru seperti langkah sebelumnya. Activity Name nya kita beri nama “MainActivity” dan Layout Name nya kita beri nama “activity\_main” harap diperhatikan besar kecilnya penulisan nama.



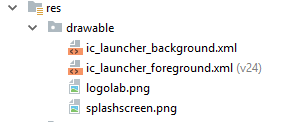
1. Masuk ke layout activity\_main.xml, kemudian masukkan source code nya :

*<?***xml version="1.0" encoding="utf-8"***?>*<**android.support.constraint.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity"**>  
  
 <**android.support.v7.widget.CardView xmlns:android="http://schemas.android.com/apk/res/android"  
 android:id="@+id/cardView"  
 style="@style/CardView.Light"  
 android:layout\_width="320dp"  
 android:layout\_height="400dp"  
 android:layout\_marginBottom="8dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.100000024"**>  
  
 <**android.support.constraint.ConstraintLayout xmlns:app="http://schemas.android.com/apk/res-auto"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"**>  
  
 <**ImageView  
 android:id="@+id/media\_image"  
 android:layout\_width="0dp"  
 android:layout\_height="0dp"  
 android:scaleType="fitCenter"  
 android:src="@drawable/wall1"  
 app:layout\_constraintBottom\_toTopOf="@+id/primary\_text"  
 app:layout\_constraintDimensionRatio="H,16:9"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_chainStyle="packed"** />  
  
 <**TextView  
 android:id="@+id/primary\_text"  
 android:layout\_width="0dp"  
 android:layout\_height="61dp"  
 android:layout\_marginStart="16dp"  
 android:layout\_marginTop="24dp"  
 android:layout\_marginEnd="16dp"  
 android:text="@string/Header"  
 android:textAppearance="@style/TextAppearance.AppCompat.Headline"  
 android:textColor="#DE000000"  
 app:layout\_constraintBottom\_toTopOf="@+id/tvNameMain"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/media\_image"** />  
  
 <**TextView  
 android:id="@+id/tvNameMain"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="16dp"  
 android:layout\_marginTop="0dp"  
 android:layout\_marginEnd="16dp"  
 android:text="@string/name"  
 android:textAppearance="@style/TextAppearance.AppCompat.Body1"  
 android:textColor="#8A000000"  
 android:textSize="15sp"  
 app:layout\_constraintBottom\_toTopOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/primary\_text"** />  
  
  
 <**TextView  
 android:id="@+id/tvdescription"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="32dp"  
 android:layout\_marginBottom="8dp"  
 android:text="@string/description"  
 android:textAppearance="@style/TextAppearance.AppCompat.Body1"  
 android:textColor="#8A000000"  
 android:textSize="20sp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="@+id/tvNameMain"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="@+id/tvNameMain"  
 app:layout\_constraintTop\_toTopOf="@+id/tvNameMain"  
 app:layout\_constraintVertical\_bias="1.0"** />  
  
 </**android.support.constraint.ConstraintLayout**>  
  
 </**android.support.v7.widget.CardView**>  
  
  
 <**Button  
 android:id="@+id/btnLogoutMain"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="8dp"  
 android:layout\_marginTop="8dp"  
 android:layout\_marginEnd="6dp"  
 android:layout\_marginBottom="8dp"  
 android:background="@drawable/roundedbutton"  
 android:minWidth="0dp"  
 android:paddingStart="8dp"  
 android:paddingEnd="8dp"  
 android:text="@string/logout"  
 android:textColor="#de101010"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.0"  
 app:layout\_constraintStart\_toStartOf="@+id/cardView"  
 app:layout\_constraintTop\_toBottomOf="@+id/cardView"  
 app:layout\_constraintVertical\_bias="0.753"** />  
  
</**android.support.constraint.ConstraintLayout**>

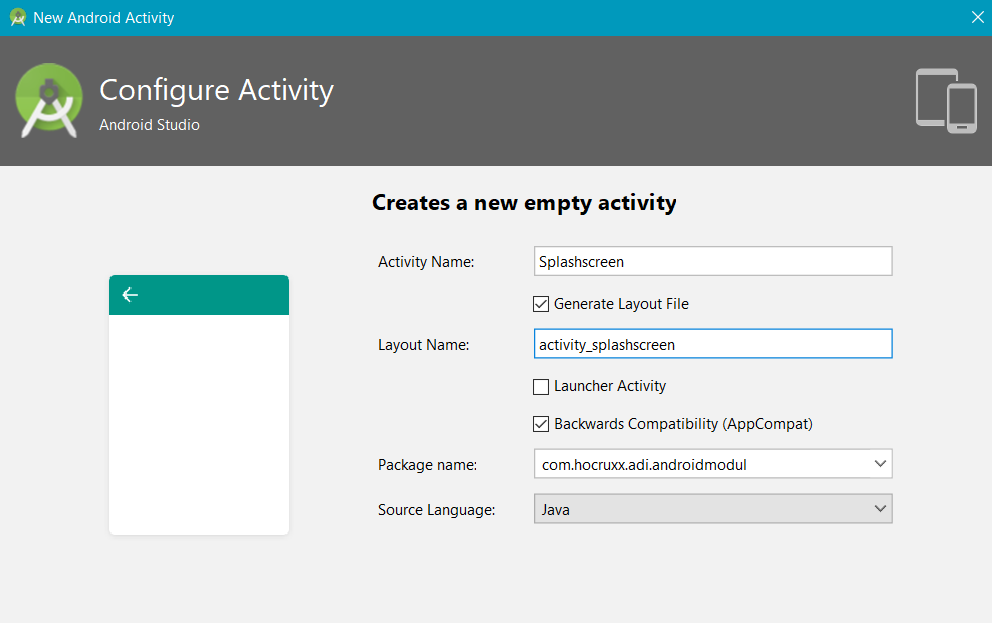
1. Setelah layout sudah kita buat, maka kita masuk ke class MainActivity.java, dan masukkan source code nya :

**import** android.content.Context;  
**import** android.content.DialogInterface;  
**import** android.content.Intent;  
**import** android.content.SharedPreferences;  
**import** android.support.v7.app.AlertDialog;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.os.Bundle;  
**import** android.util.Log;  
**import** android.view.View;  
**import** android.widget.Button;  
**import** android.widget.TextView;  
  
**public class** MainActivity **extends** AppCompatActivity {  
  
 **private** DatabaseHandler **databaseHandler**;  
 **private** User **usermodel**;  
 **private** TextView **tvNameMain**;  
 **private** Button **btnLogoutMain**;  
 **private** Button **exit**;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 setContentView(R.layout.***activity\_main***);  
  
 initView();  
 initPreference();  
 logout();  
 }  
  
 @Override  
 **public void** onBackPressed() {  
 showAlertDialog();  
 }  
  
 **private void** initView() {  
 **tvNameMain** = findViewById(R.id.***tvNameMain***);  
 **btnLogoutMain** = findViewById(R.id.***btnLogoutMain***);  
 **exit** = findViewById(R.id.***btnLogoutMain***);  
 }  
  
  
 **private void** initPreference() {  
 SharedPreferences preferences = getSharedPreferences(**"LoginPreference"**, ***MODE\_PRIVATE***);  
 String username = preferences.getString(**"username"**, **""**);  
  
 **tvNameMain**.setText(username);  
 }  
  
 **private void** deletePreference(){  
 SharedPreferences preferences = getSharedPreferences(**"LoginPreference"**, ***MODE\_PRIVATE***);  
 preferences.edit().remove(**"username"**).commit();  
 preferences.edit().remove(**"password"**).commit();  
 }  
  
 **private void** logout() {  
 **exit**.setOnClickListener(view -> showAlertDialog());  
 }  
  
 **public void** showAlertDialog() {  
 **new** AlertDialog.Builder(**this**)  
 .setMessage(**"Apa kalian ingin Logout?"**)  
 .setCancelable(**false**)  
 .setPositiveButton(**"Yes"**, **new** DialogInterface.OnClickListener() {  
 **public void** onClick(DialogInterface dialog, **int** id) {  
 deletePreference();  
 Intent login = **new** Intent(MainActivity.**this**, LoginActivity.**class**);  
 startActivity(login);  
 finish();  
 }  
 })  
 .setNegativeButton(**"No"**, **null**)  
 .show();  
 }  
  
}

1. Kita akan membuat Splash screen pada tampilan awal aplikasi saat dijalankan, masukkan gambar / icon yang akan digunakan dan menggunakan type .png agar dapat dibaca oleh system. Lakukan drag and drop pada folder berikut res🡪 drawable.



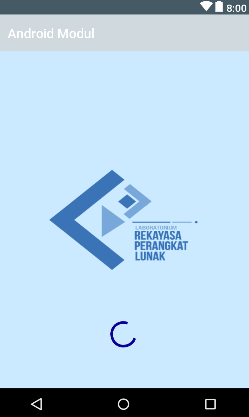
1. Langkah selanjutnya, membuat activity baru seperti langkah sebelumnya yang sudah kita buat, dengan nama Activity nya “SplashScreen” dan nama Layout nya “activity\_splashscreen”



1. Masuk ke bagian layout activiy\_splashcreen.xml, dan kemudian masukkan source code nya :

*<?***xml version="1.0" encoding="utf-8"***?>*<**RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"** >  
  
 <**ImageView  
 android:src="@drawable/splashscreen"  
 android:layout\_width="fill\_parent"  
 android:layout\_height="fill\_parent"  
 android:scaleType="fitXY"**/>  
  
 <**ProgressBar  
 android:id="@+id/progressBar1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentBottom="true"  
 android:layout\_centerHorizontal="true"  
 android:layout\_marginBottom="58dp"** />  
  
</**RelativeLayout**>

1. Maka akan muncul tampilannya kurang lebih seperti ini



1. Kemudian masukkan source code Splashscreen.java nya

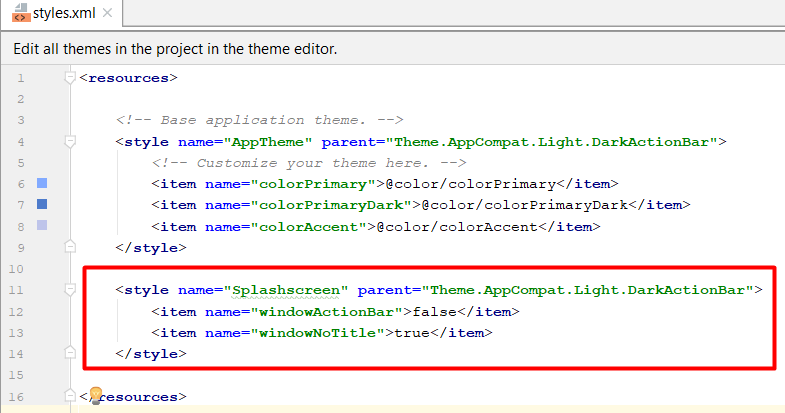
**import** android.content.SharedPreferences;  
**import** android.support.v7.app.AppCompatActivity;  
**import** android.content.Intent;  
**import** android.os.Bundle;  
**import** android.os.Handler;  
**import** android.view.Window;  
**import** android.view.WindowManager;  
  
**public class** Splashscreen **extends** AppCompatActivity {  
  
 **private static int** *splashInterval* = 2000;  
  
 @Override  
 **protected void** onCreate(Bundle savedInstanceState) {  
 **super**.onCreate(savedInstanceState);  
 **this**.requestWindowFeature(Window.***FEATURE\_NO\_TITLE***);  
 getWindow().setFlags(WindowManager.LayoutParams.***FLAG\_FULLSCREEN***,  
 WindowManager.LayoutParams.***FLAG\_FULLSCREEN***);  
  
 setContentView(R.layout.***activity\_splashscreen***);  
  
 handler();  
 }  
  
 **private void** handler(){  
 **new** Handler().postDelayed(() -> {  
 **if** (initPreference()){  
 Intent main = **new** Intent(Splashscreen.**this**, MainActivity.**class**);  
 startActivity(main);  
 finish();  
 } **else** {  
 Intent i = **new** Intent(Splashscreen.**this**, LoginActivity.**class**);  
 startActivity(i);  
 finish();  
 }  
 }, *splashInterval*);  
 }  
  
 **private** Boolean initPreference() {  
 SharedPreferences preferences = getSharedPreferences(**"LoginPreference"**, ***MODE\_PRIVATE***);  
 String username = preferences.getString(**"username"**, **""**);  
  
 **if** (username.isEmpty()){  
 **return false**;  
 }  
  
 **return true**;  
 }  
}

1. Setelah itu, kita masuk ke dalam folder Manifest nya, melalui app🡪Manifest🡪AndroidManifest.xml. Kemudian masukkan *sourcecode* nya :

*<?***xml version="1.0" encoding="utf-8"***?>*<**manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.hocruxx.adi.androidmodul"**>  
  
 <**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/AppTheme"**>  
  
 *<!-- Splash screen -->* <**activity  
 android:name="com.hocruxx.adi.androidmodul.Splashscreen"  
 android:label="@string/app\_name"  
 android:theme="@style/Splashscreen"  
 android:screenOrientation="portrait"  
 android:noHistory="true"**>  
 <**intent-filter**>  
 <**action android:name="android.intent.action.MAIN"** />  
  
 <**category android:name="android.intent.category.LAUNCHER"** />  
 </**intent-filter**>  
 </**activity**>  
 <**activity android:name=".RegisterActivity"** />  
 <**activity android:name=".MainActivity"** />  
 <**activity android:name=".LoginActivity"**></**activity**>  
 </**application**>  
  
</**manifest**>

Ingat untuk mengganti nama package “com.xxx.xxx” menjadi nama package masing projek.

1. Tambahkan beberaepa *sourcecode* yang ada di styles.xml untuk mengatur tampilan Splash Screen nya



1. Selesai 😊