



KEMENTERIAN KOMUNIKASI DAN INFORMATIKA
REPUBLIK INDONESIA
DIREKTORAT JENDERAL APLIKASI INFORMATIKA



SMK CODING

android

Development

Membangun Aplikasi berbasis Android
menggunakan Android Studio dan
bahasa Pemrograman Java



DAFTAR ISI

DAFTAR ISI	1
Dasar-Dasar Android	2
Android Studio	4
Fitur baru diperbarui di Android Studio 3.x	5
Develop.....	5
Build	6
Persyaratan System.....	7
Introduction to Android Studio.....	8
Project Android Pertama.....	8
Project Wizard.....	9
Antarmuka Android Studio.....	12
Activity	12
Mengenal Konsep Activity dan Siklus Activity	12
Siklus Activity	13
Views Dan ViewGroup.....	17
MATERI 1	20
LinearLayout	20
MATERI 2.....	24
RelativeLayout.....	24
MATERI 3.....	27
Intent	27
MATERI 4.....	34
Intent Result	34
MATERI 5.....	42
Fragment.....	42
MATERI 6.....	48
ViewPager	48
MATERI 7	54
Working With REST API (Retrofit dan Okhttp).....	54
MATERI 8.....	71
Membuat CRUD Data Siswa menggunakan	71
Room Persistence.....	71
DAFTAR PUSTAKA.....	97

Dasar-Dasar Android

Android adalah sistem operasi untuk telepon seluler berbasis linux sebagai kernelnya. Saat ini Android dapat disebut raja dari smartphone. Android mengalami perkembangan yang sangat pesat karena Android menyediakan platform terbuka (Open Source) bagi para pengembang untuk menciptakan aplikasi mereka sendiri. Awalnya, perusahaan search engine terbesar saat ini, yaitu Google Inc. membeli Android Inc., pendatang baru yang membuat peranti lunak untuk ponsel. Android, Inc. didirikan oleh Andy Rubin, Rich Milner, Nick Sears dan Chris White pada tahun 2003. Pada agustus 2005 Google membeli Android Inc. Kemudian untuk mengembangkan Android dibentuklah Open Handset Alliance konsorsium dari 34 perusahaan hardware, software, dan telekomunikasi, termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile dan Nvidia.

Pada saat perilisan perdana Android, 5 November 2007, Android bersama Handset Alliance menyatakan dukungannya terhadap pengembangan standard terbuka pada perangkat seluler. Di lain pihak, Google juga merilis kode-kode Android dibawah lisensi Apache, sebuah lisensi perangkat lunak dan standard terbuka perangkat seluler. Sekarang terdapat dua jenis distributor sistem operasi Android. Pertama yang mendapat dukungan penuh dari Google atau Google Mail Service (GMS), sedangkan yang kedua adalah yang didistribusikan bebas tanpa dukungan langsung dari Google, dan dikenal sebagai Open Handset Distribution (OHD).



Sumber gambar : http://pngimg.com/uploads/android_logo/android_logo_PNG35.png

Awal sejarah perkembangan Android dimulai ketika Google membeli perusahaan Android Inc., sebuah perusahaan yang berada di Palo Alto, California Amerika Serikat, pada bulan juli tahun 2000. Para pendiri Android Inc. yang akhirnya bekerja pada Google diantaranya Andy Rubin, Rich Miner, Nick Sears, dan Chris White. Saat itu banyak yang menganggap fungsi Android Inc. hanyalah sebagai perangkat lunak pada ponsel. Sejak saat itu muncul rumor bahwa Google hendak memasuki pasar telepon seluler. Google sendiri mempunyai sebuah tim yang dipimpin oleh rubin yang bertugas mengembangkan program perangkat seluler yang didukung oleh kernel Linux. Hal ini menunjukkan indikasi bahwa Google sedang bersiap menghadapi persaingan pasar telepon seluler. Gambar 2. Sang pengembang Google Android, Andy Rubin. Pada 9 Desember 2008, diumumkan anggota baru yang bergabung dalam program kerja Android ARM Holdings, Atheros, Communications, diproduksi oleh Asustek 3 Computer Inc, Garmin Ltd, Softbank, Sony Ericson, Toshiba Corp, dan Vodafone Group Plc. Google menyediakan banyak kemudahan untuk pengembang software pihak ketiga, seperti plugin Android Development Tool (ADT) untuk Eclipse (dan juga tool standalone) termasuk di dalamnya kemampuan untuk logging secara realtime, emulator realistis yang menjalankan kode ARM native dan pelaporan error.



Sumber gambar : <https://rsinewsupdate.files.wordpress.com/2017/04/andy-rubin-androidguys-200417.jpg>

Pada 9 Desember 2008, diumumkan anggota baru yang bergabung dalam program kerja Android ARM Holdings, Atheros, Communications, diproduksi oleh Asustek 3 Computer Inc, Garmin Ltd, Softbank, Sony Ericson, Toshiba Corp, dan Vodafone Group Plc. Google menyediakan banyak kemudahan untuk pengembang software pihak ketiga, seperti plugin Android Development Tool (ADT) untuk Eclipse (dan juga tool standalone) termasuk di dalamnya kemampuan untuk logging secara realtime, emulator realistis yang menjalankan kode ARM native dan pelaporan error.

Android Studio



Android studio merupakan sebuah Integrated Development Environment (IDE) untuk platform Android. Android Studio ini diumumkan pada tanggal 16 Mei 2013 pada Konferensi Google I/O oleh Produk Manajer Google, Ellie Powers. Android studio bersifat free dibawah Apache License 2.0. Android Studio awalnya dimulai dengan versi 0.1 pada bulan mei 2013, Kemudian dibuat versi beta 0.8 yang dirilis pada bulan juni 2014. Yang paling stabil dirilis pada bulan Desember 2014, dimulai dari versi 1.0. Berbasiskan JetBrains' IntelliJ IDEA, Studio di desain khusus untuk Android Development.

Ini sudah bisa di download untuk Windows, Mac OS X, dan Linux. Dan versi sekatang sudah masuk di versi 3.

Fitur baru diperbarui di Android Studio 3.x

Google mengumumkan di Google I / O 2017, Android Studio 3.x adalah pembaruan besar yang difokuskan pada percepatan pengembangan aplikasi Anda di Android. Akhirnya versi stabil Android Studio telah dirilis .Rilis Android Studio ini dikemas dengan banyak pembaruan baru, tetapi ada tiga area fitur utama yang tidak ingin Anda lewatkan,

termasuk: seperangkat alat profil aplikasi baru untuk mendiagnosis masalah kinerja dengan cepat, dukungan untuk bahasa pemrograman Kotlin, dan seperangkat alat dan penyihir baru untuk mempercepat pengembangan Anda di Android Oreo API terbaru.

Google juga menginvestasikan waktu untuk meningkatkan stabilitas dan kinerja di banyak area Android Studio. Jika Anda mencari stabilitas tinggi, ingin membuat aplikasi berkualitas tinggi untuk Android Oreo, kembangkan dengan bahasa Kotlin, atau gunakan tools aplikasi Android terbaru, maka Anda harus mengunduh Android Studio versi terbaru.

Develop

Kotlin Programming language

Seperti yang diumumkan di Google I / O 2017, bahasa pemrograman Kotlin sekarang secara resmi didukung untuk pengembangan Android. Kotlin adalah bahasa ekspresif dan ringkas yang dapat diinteroperasikan dengan bahasa java Android yang sudah ada, yang berarti Anda dapat menggunakan sedikit atau sebanyak apa pun bahasa di

aplikasi Anda seperti yang Anda inginkan. Kotlin adalah bahasa siap produksi yang digunakan oleh banyak aplikasi Android populer di Google Play hari ini.

Java 8 Language features

Di Android Studio 3.0, Google terus meningkatkan dukungan untuk fitur bahasa Java 8. Dengan migrasi ke toolchain berbasis javac, menggunakan fitur bahasa Java 8 dalam proyek Anda bahkan lebih mudah

Layout Editor

komponen dalam Editor Layout memiliki insert tampilan drag-and-drop yang lebih baik, dan panel ketika ada error.

Android Things Support

Android Studio 3.x menyertakan satu set template baru di wizard Proyek Baru dan wizard Modul Baru yang dikembangkan untuk platform Android Things

Build

Instant App Support

Dengan rilis Android Studio ini, Anda dapat menambahkan fitur Aplikasi Instant ke proyek Anda. Aplikasi Instant adalah aplikasi Android ringan yang dapat langsung dijalankan oleh pengguna tanpa instalasi.

Persyaratan System

Berikut ini beberapa persyaratan system untuk menggunakan Android Studio :

Windows

- Microsoft Windows 8/7/Vista/2003 (32 atau 64 bit)
- RAM minimum 4 GB, Rekomendasi 8 GB RAM - Disk kosong 500 MB
- Sedikitnya 16 GB untuk Android SDK, Emulator system images, and caches.
- Java Development Kit (JDK) 7 atau yang lebih tinggi
- Resolusi minimum 1280 x 800

OS X

- Mac OS X 10.8.5 atau yang lebih tinggi, up to 10.10 to up 10.10.1 up 10.10.2 (Yosemite)
- RAM minimum 4 GB, Rekomendasi RAM 8 GB
- Disk kosong 500 MB
- Sedikitnya 16 GB untuk Android SDK, emulator system images, dan caches
- Java Development Kit (JDK) 7 atau yang lebih tinggi
- Resolusi minimum 1280x 800

GNU/Linux

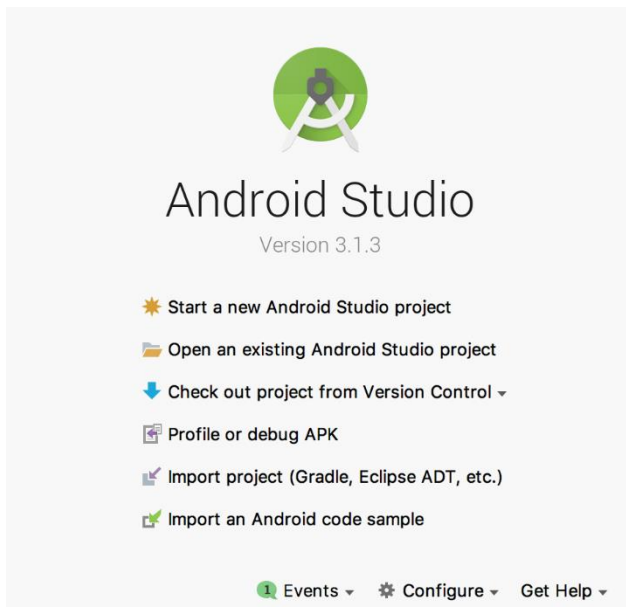
- GNOME atau KDE atau Unity Desktop pada Ubuntu atau Fedora
- Minimum RAM 4 GB, Rekomendasi RAM 8 GB
- Disk kosong 500 MB 16
- Sedikitnya 16 GB untuk Android SDK, emulator system images, dan caches
- Java Development Kit (JDK) 7 atau yang lebih tinggi.
- Resolusi minimum 1280 x 800

Introduction to Android Studio

Supaya lancar menggunakan Android Studio, sebaiknya kita tahu bagaimana strukturnya terlebih dahulu.

Project Android Pertama

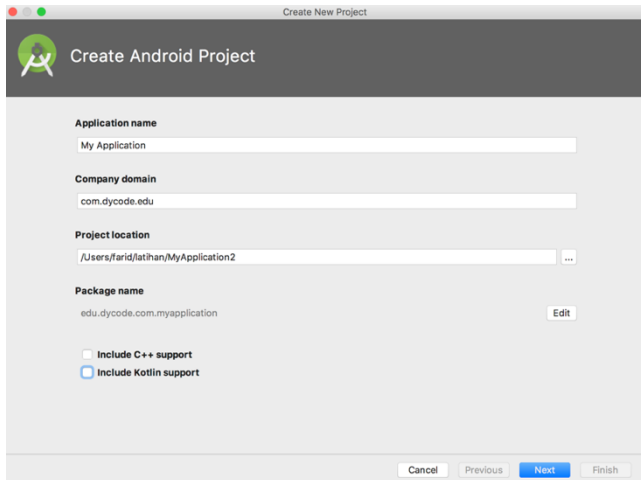
Pertama kali menjalankan Android Studio, kita akan melihat tampilan seperti berikut ini.



Untuk mulai proyek baru pilihlah “Start a new Android Project”.

Project Wizard

1. Dalam dialog ini, kita bisa memberi nama aplikasi yang akan dibuat, dan company domain. Company domain akan digunakan sebagai alat identifikasi ketika aplikasi akan dipublikasikan. Kita juga dapat mengganti lokasi di mana proyek akan disimpan.



2. Dialog berikut ini adalah *target devices*, kita bisa memilih peranti target dari aplikasi yang akan kita buat. Kita juga bisa menentukan nilai minimum SDK yang akan didukung oleh aplikasi.

Create New Project

Target Android Devices

Select the form factors and minimum SDK
Some devices require additional SDKs. Low API levels target more devices, but offer fewer API features.

☒ **Phone and Tablet**
API 19: Android 4.4 (KitKat) [dropdown]
By targeting **API 19 and later**, your app will run on approximately **90.1%** of devices. [Help me choose](#)
☐ Include Android Instant App support

☐ **Wear**
API 21: Android 5.0 (Lollipop) [dropdown]

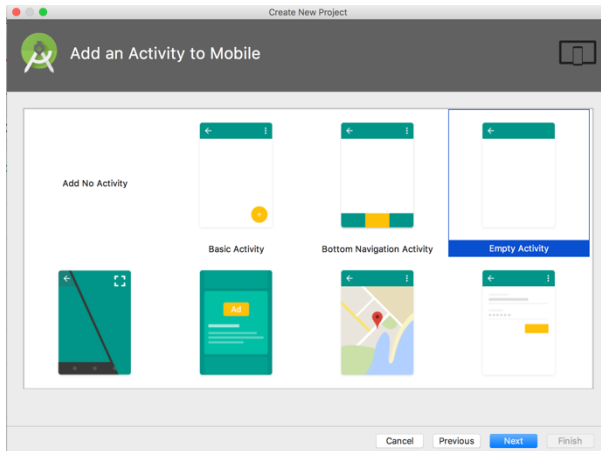
☐ **TV**
API 21: Android 5.0 (Lollipop) [dropdown]

☐ **Android Auto**

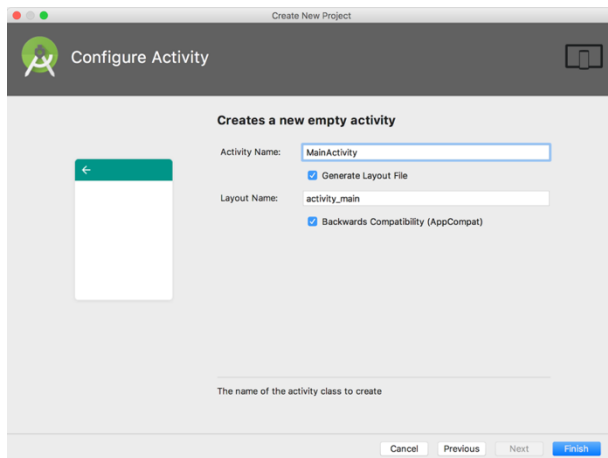
☐ **Android Things**
API 24: Android 7.0 (Nougat) [dropdown]

Cancel Previous **Next** Finish

3. Dialog di bawah ini adalah *default template*. Di dalamnya terdapat beberapa *template* yang bisa kita gunakan seperti Empty Activity, Login Activity, Navigation Drawer Activity dan lain-lain.

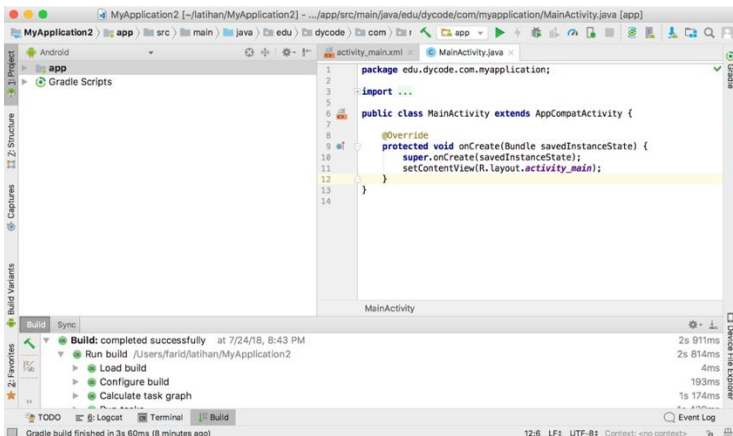


4. Dialog berikut ini adalah penamaan activity yang pertama kali kita buat. Usai memberi nama, tekan Finish.



Antarmuka Android Studio

OK, membuat proyek pertama kali di Project Wizard, *done!* Kali ini kita akan menemui tampilan penuh Android Studio. Untuk meningkatkan produktivitas, mari kita bahas lebih jauh tentang antarmuka (*interface*) dari Android Studio ini.



Activity

Mengenal Konsep Activity dan Siklus Activity

Activity merupakan public class dalam aplikasi android. Setiap Activity merupakan sesuatu yang unik atau single, yang ditujukan untuk handle macam-macam hal yang bisa dilakukan oleh user. Umumnya, activity berhubungan dengan user dimana activity menciptakan windows atau UI yang mana ditampilkan dengan konsep setContentView(View). Ada dua metode yang pasti dimiliki oleh satu activity yaitu :

1. onCreate Untuk menginisiasi suatu activity, biasanya dipanggil dengan perintah setContentView(int) untuk resource yang didefinisikan di layout UI, dengan perintah findViewById(int) untuk

memanggil widget yang dibutuhkan UI untuk berinteraksi dengan aplikasi.

2. `onPause` Untuk menyatakan ketika user meninggalkan suatu activity. Untuk penggunaan dengan `Context.startActivity()`, semua kelas activity harus sesuai dengan yang dideklarasikan dalam suatu paket di `AndroidManifest.xml`. Activity adalah bagian penting dari model aplikasi.

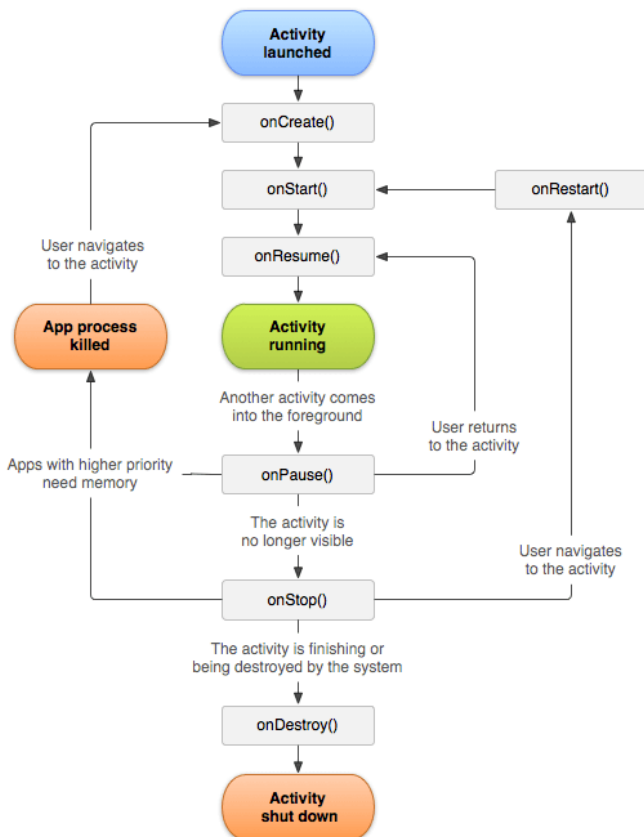
Siklus Activity

Activity aplikasi android dikelola dengan system yang dikenal dengan nama “activity stack”. Ketika suatu activity start, activity diletakkan pada stack paling atas dan activity-activity yang sudah berjalan ada dibawahnya dan akan terus berada pada posisi atas stack sampai muncul activity yang baru. Pada dasarnya Activity memiliki 4 keadaan :

1. `active/running`, jika activity berada posisi atas stack
2. `pause`, jika activity tidak dipakai atau dibutuhkan pada suatu saat tertentu, tetapi activity itu masih ada atau `visible`, ketika activity baru yang di tangani oleh system activity yang lama disebut `pause` dan masih berada di `memory`, bisa jadi suatu activity yang sudah keadaan `pause` tidak ada di `memory` yang kemungkinan karena kekurangan `memory`.
3. `Stopped`, jika activity sudah tidak dipakai lagi atau digantikan oleh activity lain, activity yang sudah `stopped` tidak pernah dipanggil lagi, dan secara permanen `memory` pun tidak menyimpan info mengenai activity ini.
4. `Restart`, jika activity `pause` atau `stopped`, system dapat `menge-drop` activity ini dari `system memory`, dan ketika user membutuhkan

activity tersebut, activity akan kembali keadaan awal, artinya activity mengalami proses restart.

Berikut ini adalah gambaran mengenai Activity Lifecycle atau siklus dari activity dalam system android :



Sumber gambar : <https://www.javatpoint.com/images/androidimages/Android-Activity-Lifecycle.png>

Berikut ini merupakan table lifecycle Activity Android :

Method	Description	Killable	Next
onCreate()	Ketika sebuah activity dibuat, pada method inilah kita melakukan inisialisasi seperti create view, list data, dan lain-lain. Method onCreate() selalu diikuti oleh onStart()	N	onStart()
onRestart()	Ketika sebuah activity dihentikan, dan merupakan prioritas untuk memanggil activity itu kembali. Method ini selalu diikuti oleh onStart()	N	onStart()
onStart()	Ketika sebuah activity di panggil sebelum diperlihatkan ke user, method ini diikuti oleh onResume() atau onStop()	N	onResume() or onStop()
onResume()	Ketika sebuah activity start/mulai melakukan interaksi dengan user, pada saat ini berada pada posisi teratas dari activity stack yang mana user	N	onPause()

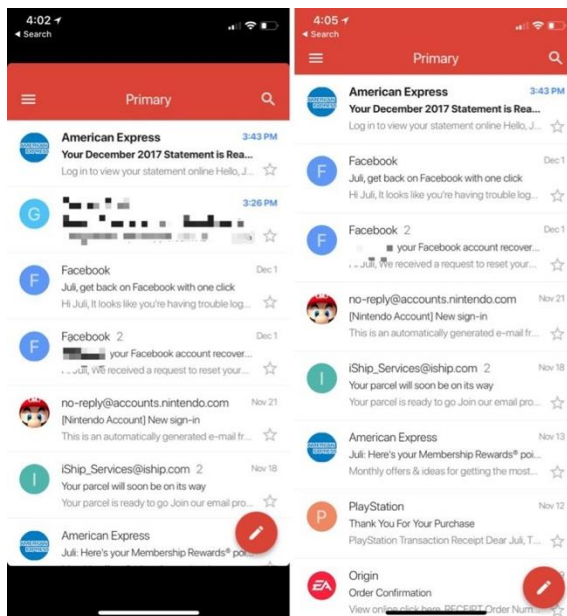
Method	Description	Killable	Next
	akan melakukan input. Selalu diikuti oleh onPause()		
onPause()	Ketika sebuah activity lainnya dipanggil atau dimulai, method ini digunakan ketika data tidak harus disimpan kedalam system secara permanen, method ini diikuti onResume() atau onStop()	Y	onResume() or onStop()
onStop()	Ketika sebuah activity tidak lagi dibutuhkan atau tidak terlihat lagi oleh user, method ini diikuti oleh onRestart() atau onDestroy()	Y	onRestart() or onDestroy()
onDestroy()	Ketika sebuah activity secara permanent tidak lagi dibutuhkan (activity dihancurkan), dapat juga dilakukan dengan fungsi finish() atau isFinishing()	Y	-

Maksud dari killable disini adalah apakah memungkinkan bagi system untuk memastikan proses host activity atau tidak. Sehingga method yang ditandai dengan huruf N berarti melindungi host activity (tidak dimatikan) Activity yang secara teknis adalah N

bisa saja dimatikan secara otomatis oleh system apabila terjadi low memori atau tidak cukupnya resource untuk menjaga activity itu tetap hidup. Kejadian tidak cukupnya resource bisa jadi terjadi karena banyak proses yang sedang dieksekusi dan memerlukan kapasitas memori yang besar, sehingga terjadi low memori yang menyebabkan beberapa activity otomatis hilang dari memori.

Views Dan ViewGroup

Pada modul ini, Anda akan mempelajari komponen view dan viewgroup. Kedua komponen ini dapat berkolaborasi sehingga membentuk antar muka dengan contoh seperti pada gambar di bawah ini:



Pada dasarnya semua elemen antar pengguna di aplikasi Android dibangun menggunakan dua buah komponen inti, yaitu view dan viewgroup.

Sebuah view adalah obyek yang menggambar komponen tampilan ke layar yang mana pengguna dapat melihat dan berinteraksi langsung.

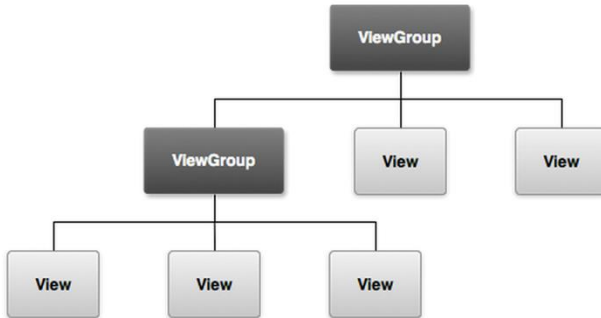
Contoh komponen turunan dari view seperti :

- **TextView**, komponen yang berguna untuk menampilkan teks ke layar.
- **Button**, komponen yang membuat pengguna dapat berinteraksi dengan cara ditekan untuk melakukan sesuatu.
- **ImageView**, Komponen untuk menampilkan gambar.

Sedangkan viewgroup adalah sebuah obyek yang mewadahi obyek-obyek view dan viewgroup itu sendiri sehingga membentuk satu kesatuan tampilan aplikasi yang utuh. Contoh komponen viewgroup adalah:

- **LinearLayout**
- **FrameLayout**
- **RelativeLayout**
- **ConstraintLayout**

Hierarki komponen view dan viewgroup dapat digambarkan dengan diagram berikut:

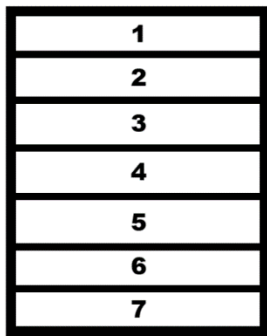


Sumber gambar : <https://i.stack.imgur.com/KSCNf.png>

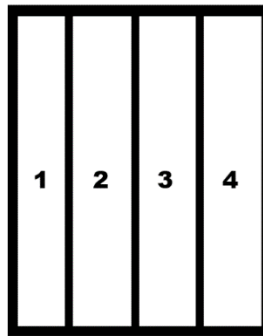
MATERI 1

LinearLayout

LinearLayout adalah susunan tata letak yang paling simple dapat digunakan para developer android. Karena layout ini hanya memberikan susunan tata letak komponen secara garis lurus. Bisa secara horizontal maupun vertical.



Vertical Linear Layout



Horizontal Linear Layout

Berikut contoh penggunaan LinearLayout

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <Button
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="button1" />
```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="button2" />

    <Button
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="button3" />

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <Button
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="button4" />

    <Button
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:text="button5" />

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:gravity="end|right"
    android:orientation="horizontal">

```

```

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="button6" />

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="button7" />

</LinearLayout>

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">

    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:src="@mipmap/ic_launcher" />

    <LinearLayout
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:orientation="vertical">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="text_1" />

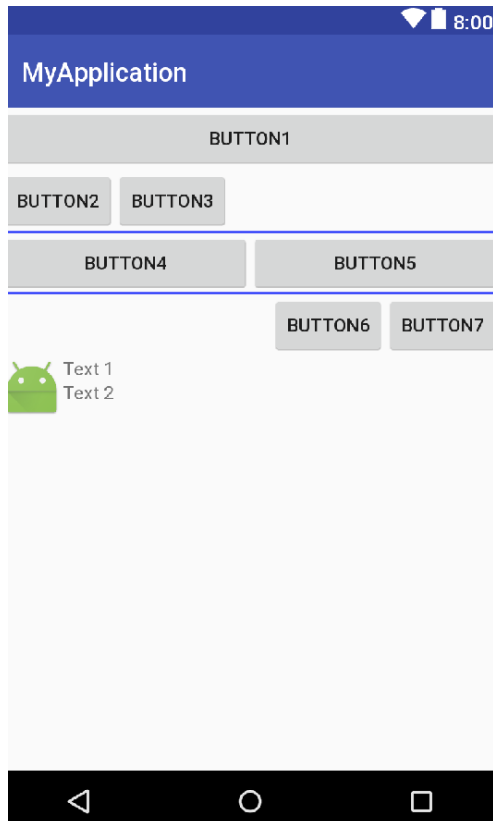
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="text_2" />

    </LinearLayout>
</LinearLayout>

</LinearLayout>

```

Hasil Tampilannya seperti ini



MATERI 2

RelativeLayout

Untuk relativitylayout kita bisa mengaturnya mengikuti komponen lainnya, semisal ada komponen A, B, dan C maka kita bisa mengaturnya seperti , A berada dibawah B, dan C berada di samping kanan/kiri A dan sebagainya. Intinya adalah komponen satu dan lainnya saling berkaitan.

Contoh penggunaan RelativeLayout

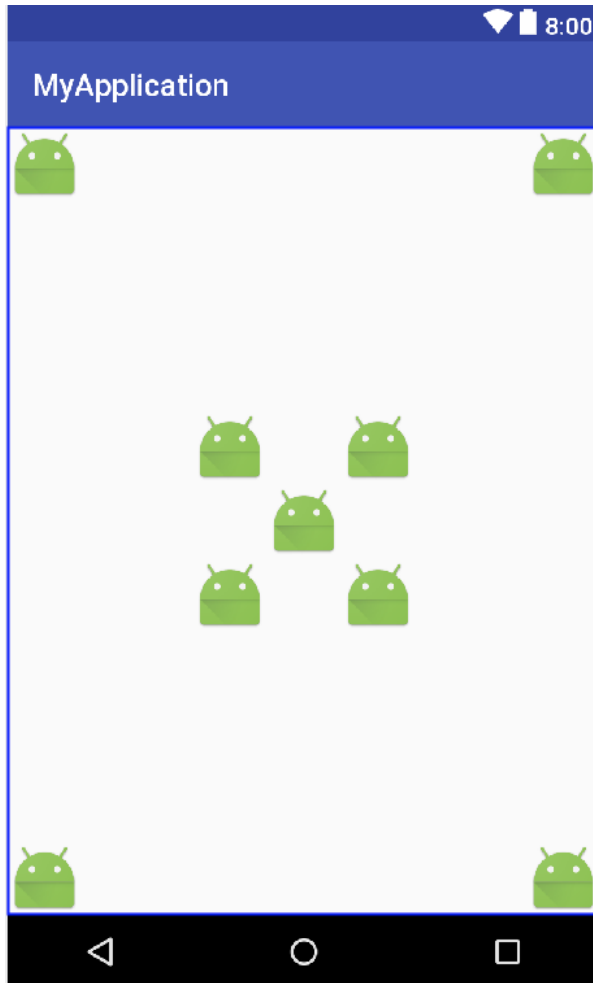
```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">
    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:layout_alignParentTop="true"
        android:src="@mipmap/ic_launcher" />
    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentEnd="true"
        android:layout_alignParentRight="true"
        android:layout_alignParentTop="true"
        android:src="@mipmap/ic_launcher" />
    <ImageView
        android:id="@+id/img_center"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:src="@mipmap/ic_launcher" />
    <ImageView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
```

```

        android:layout_alignParentBottom="true"
        android:layout_alignParentLeft="true"
        android:layout_alignParentStart="true"
        android:src="@mipmap/ic_launcher" />
<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_alignParentEnd="true"
    android:layout_alignParentRight="true"
    android:src="@mipmap/ic_launcher" />
<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@mipmap/ic_launcher"
    android:layout_toLeftOf="@id/img_center"
    android:layout_above="@id/img_center"
    android:layout_toStartOf="@id/img_center"/>
<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@mipmap/ic_launcher"
    android:layout_toRightOf="@id/img_center"
    android:layout_above="@id/img_center"
    android:layout_toEndOf="@id/img_center"/>
<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@mipmap/ic_launcher"
    android:layout_toLeftOf="@id/img_center"
    android:layout_below="@id/img_center"
    android:layout_toStartOf="@id/img_center"/>
<ImageView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:src="@mipmap/ic_launcher"
    android:layout_toRightOf="@id/img_center"
    android:layout_below="@id/img_center"
    android:layout_toEndOf="@id/img_center"/>
</RelativeLayout>

```

Hasil tampilannya seperti ini



MATERI 3

Intent

Intent adalah sebuah Object pesan dalam programming Android yang berfungsi untuk perpindahan Halaman. Intent juga sebuah object yang bisa berkomunikasi dengan activity lain, baik itu activity pada fungsi internal, maksudnya seperti camera, telephone dll pada ponsel atau memanggil activity yang masih berada di dalam satu package atau di beda package yang penting masih dalam satu project (project yang sama). Ada tiga penggunaan atau fungsi umum intent dalam aplikasi Android yaitu:

- **Untuk pindah halaman dari satu Activity ke activity yang lain**, Contohnya contohnya ada 3 buah activity di activity 1 berisi tombol Login dan di activity 2 adalah tampilan untuk Loginnya. Ketika button Login di klik dia akan menampilkan isi dari Activity 2.
- **Untuk transfer data dari satu Activity ke Activity lainnya**, Contohnya pada Activity 2 tadi ada button untuk masuk dan saat Button Masuk di klik Akan tampil informasi user pada Activity 3.
- **Untuk memanggil Activity pada .internal device android**, Contohnya di Activity 3 ada sebuah button untuk dengan tulisan Camera saat di klik itu akan mengarahkan pada camera di android

Langkah-langkah pembuatan aplikasi intent adalah sebagai berikut :

1. Buat sebuah project baru
2. Isilah dengan parameter sebagai berikut

Project Name	Intent
Package Name	com.dycode.edu.intent
Min SDK	19

3. Pada layout activity_main.xml buatlah code seperti berikut

```
<?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:layout_width="match_parent"
    android:orientation="vertical"
    android:layout_height="match_parent">

    <EditText
        android:id="@+id/editText"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:ems="10"
        android:hint="masukan nama"
        android:inputType="textPersonName"
        android:gravity="center" />

    <Button
        android:id="@+id/button"
        android:layout_width="wrap_content"
        android:layout_gravity="center"
        android:layout_height="wrap_content"
        android:layout_marginTop="8dp"
        android:text="Send Data" />
</LinearLayout>
```

4. Kemudian pada MainActivity tambahkan code berikut

```
public class MainActivity extends AppCompatActivity {

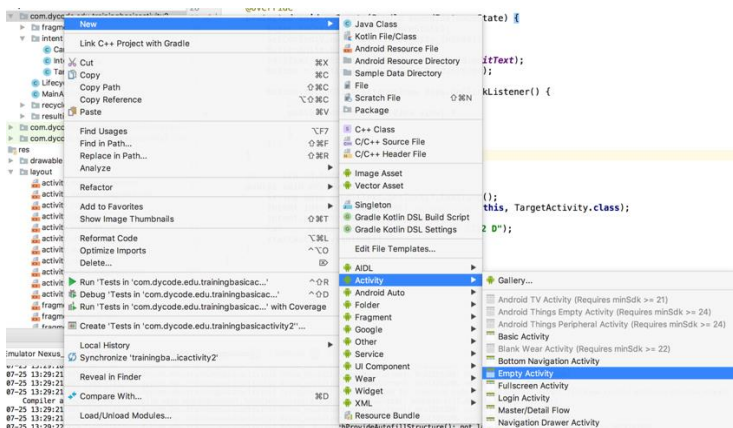
    public static final String EXTRA_DATA = "EXTRA_DATA";

    private EditText editText;
    private Button btnSenData;
```

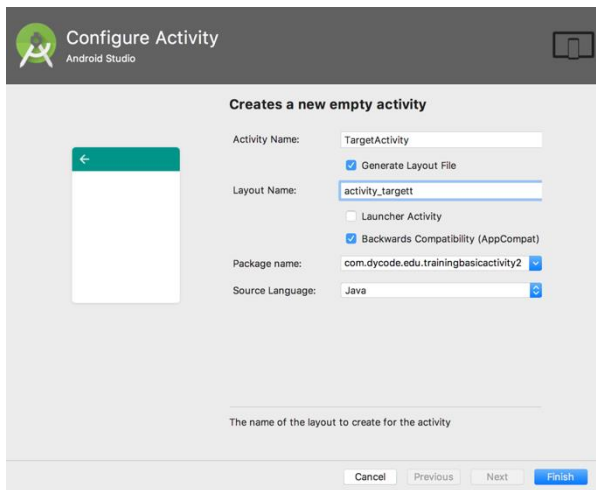
@Override

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_intent);  
    editText = (EditText)findViewById(R.id.editText);  
    btnSendData = (Button)findViewById(R.id.button);  
  
    button.setOnClickListener(new View.OnClickListener() {  
        @Override  
        public void onClick(View view) {  
            String data = editText.getText().toString();  
            Intent intent = new Intent(this, TargetActivity.class);  
            intent.putExtra(EXTRA_DATA, data);  
            startActivity(intent);  
        }  
    });  
}
```

5. Buat activity baru dengan cara **klik kanan di package project kita, -> New -> Activity -> Empty Activity**



6. Ubah activity name nya menjadi **TargetActivity**



7. Buka **activity_target.xml** dan ubah code nya menjadi seperti ini :

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="16dp"
        android:textSize="36sp"
        android:text="TextView"
    />
</RelativeLayout>
```

8. Kemudian pada **TargetActivity.java** ubah menjadi code berikut

```
public class TargetActivity extends AppCompatActivity {

    private TextView txtResult;

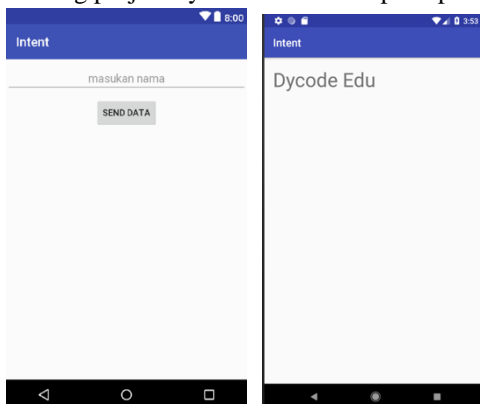
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_target);
        txtResult = (TextView)findViewById(R.id.textview);

        //get data
        if(getIntent().getExtras() != null ){
            String data =
            getIntent().getExtras().getString(IntentActivity.EXTRA_DATA);
            txtResult.setText(data);

        }

    }
}
```

9. Running project nya maka akan tampil seperti ini



Plugin Parcelable And ButterKnife

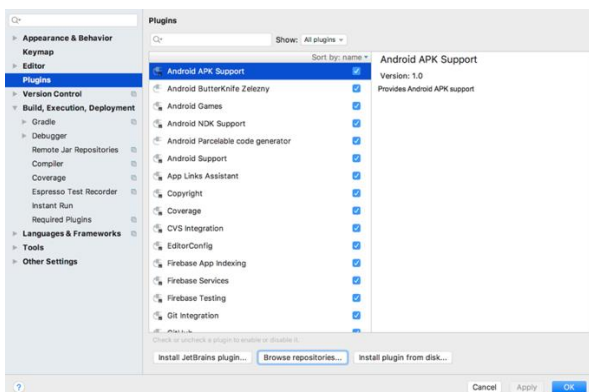
Oke untuk pembuatan project kedepannya kita akan menggunakan plugin ButterKnife dan parcelable

- **ButterKnife** adalah library Android yang dikembangkan oleh Jake Wharton. Library ini sangat membantu meringankan tugas developer android karena mampu menyederhanakan penulisan komponen view di Android, atau istilah singkatnya kita bisa **menghilangkan 'findViewById'**

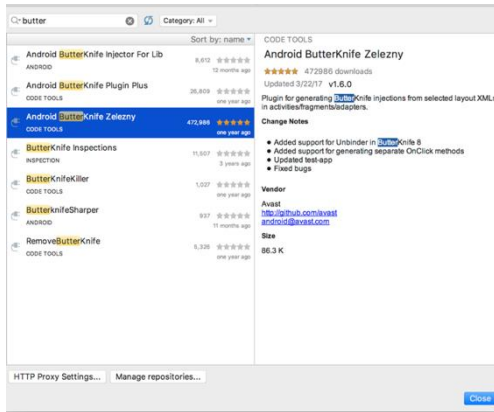
- **Parcelable** adalah suatu interface pada pemrograman Android, yang memungkinkan suatu instansi dari kelas/objek untuk bisa disimpan dan diambil kembali dari sebuah Parcel. Sedangkan Parcel sendiri merupakan suatu kontainer untuk menampung kelas tersebut

Untuk bisa menggunakan plugin tersebut , kita harus install plugin nya di android studio :

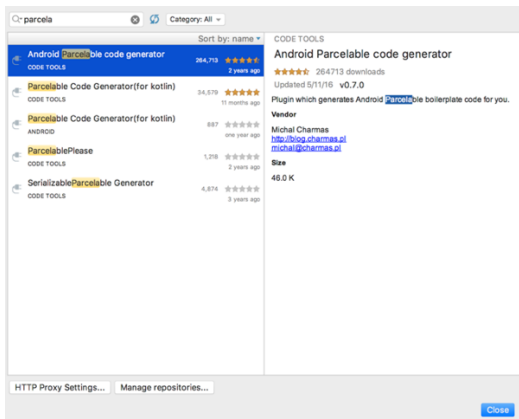
1. Buka File -> Settings -> Plugins -> Browse repositories



2. Lalu ketik “**Android Butterknife Zelezny**”
 3. Install
 4. Kemudian install lagi plugin Parcelable nya
 5. Ketik “**Android Parcelable Code Generator**”
- Gambar (Penambahan Plugin ButterKnife)



Gambar (Penambahan Plugin Parcelable)



6. Jika sudah di download jangan lupa untuk me **restart** android studio nya

MATERI 4

Intent Result

Langkah-langkah pembuatan aplikasi intent result adalah sebagai berikut :

1. Buat sebuah project baru
2. Isilah dengan parameter sebagai berikut

Project Name	IntentResult
Package Name	com.dycode.edu.intent
Min SDK	19

3. Pada layout activity_main.xml buatlah code seperti berikut

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    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">


    <TextView
        android:id="@+id/text_view_result"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:gravity="center"
        android:layout_marginTop="32dp"
        android:text="Name" />


    <Button
        android:id="@+id/btn_edit"
        android:layout_marginTop="12dp"
        android:layout_centerHorizontal="true"
        android:layout_below="@id/text_view_result"
```


```
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Edit Name" />
    </RelativeLayout>
```

4. Untuk materi sekarang dan kedepannya, kita menggunakan butterknife, tambahkan library butterknife di gradle,
5. Buka Gradle nya build.gradle (ModuleApp)


Gradle Scripts


 build.gradle (Project: TrainingBasicActivity2)


 build.gradle (Module: app)

 gradle-wrapper.properties (Gradle Version)

 proguard-rules.pro (ProGuard Rules for app)

 gradle.properties (Project Properties)

 settings.gradle (Project Settings)

 local.properties (SDK Location)

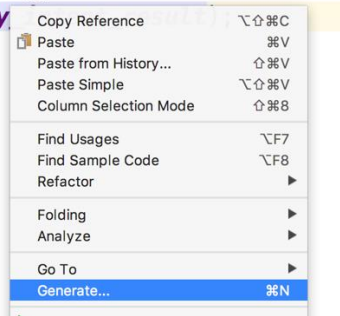
6. Lalu tambahkan code berikut didalam dependencies untuk menginstall butterknife dan pastikan pc/laptop anda terhubung dengan internet

```
implementation 'com.jakewharton:butterknife:8.8.1'

annotationProcessor 'com.jakewharton:butterknife-compiler
:8.8.1'
```

7. Dan jangan lupa untuk **sync now** project nya di sebelah kanan atas
8. Mendeklarasikan komponen view dengan ButterKnife Arahkan pointer pada xml nya, lalu klik kanan pilih generate , dan pilih semua view yang mau di inject, dan Confirm

```
ate(Bundle savedInstanceState) {
    savedInstanceState);
    .layout.activity
```



<input checked="" type="checkbox"/>	Element	ID	OnClick	Variable Name
<input checked="" type="checkbox"/>	TextView	text_view_result	<input type="checkbox"/>	textViewResult
<input checked="" type="checkbox"/>	Button	btn_edit	<input type="checkbox"/>	btnEdit

☐ Create ViewHolder
☐ Split OnClick methods

9. Setelah Confirm maka tampilannya akan seperti ini

```
public class MainActivity extends AppCompatActivity {

    @BindView(R.id.text_view_result)
    TextView textViewResult;
    @BindView(R.id.btn_edit)
    Button btnEdit;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        ButterKnife.bind(target: this);
    }
}
```

10. Kemudian tambahkan code nya menjadi seperti berikut

```

public class MainActivity extends AppCompatActivity {

    private static final int RC_EDIT_NAME = 112;
    @BindView(R.id.text_view_result)
    TextView textViewResult;
    @BindView(R.id.btn_edit)
    Button btnEdit;

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

    @OnClick(R.id.btn_edit)
    public void onViewClicked() {
        Intent intent = new Intent(this, EditNameActivity.class);
        startActivityForResult(intent, RC_EDIT_NAME);
    }

    @Override
    protected void onActivityResult(int requestCode, int resultCode,
    Intent data) {
        if(requestCode == RC_EDIT_NAME && resultCode ==
    RESULT_OK){
            String name =
    data.getExtras().getString(EditNameActivity.EXTRA_NAME);
            textViewResult.setText(name);
        }
    }
}

```

11. Buat activity baru dengan nama **EditNameActivity**
12. Buka activity_edit_name.xml dan tambahkan code berikut

```

<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
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">

    <EditText
        android:id="@+id/editText2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_marginLeft="8dp"
        android:layout_marginRight="8dp"
        android:layout_marginTop="8dp"
        android:ems="10"

        android:inputType="textPersonName" />

    <Button
        android:id="@+id/btn_save"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="8dp"
        android:layout_below="@id/editText2"
        android:layout_centerHorizontal="true"
        android:layout_marginRight="8dp"
        android:layout_marginTop="8dp"
        android:text="Save Name"
    />
</RelativeLayout>

```


13. Dan terakhir pada EditNameActivity tambahkan code berikut

```
public class EditNameActivity extends
AppCompatActivity {

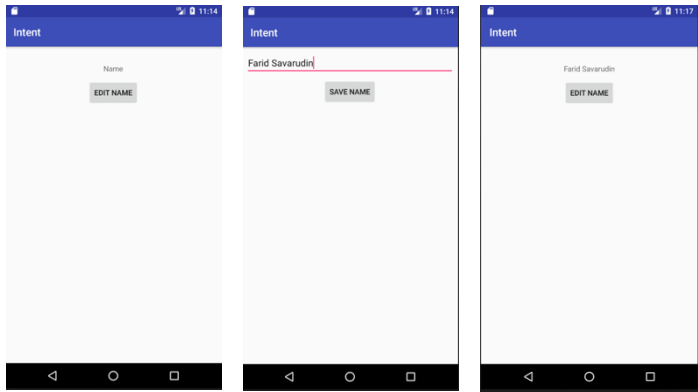
    public static final String EXTRA_NAME =
"EXTRA_NAME";

    @BindView(R.id.editText2)
    EditText editText2;
    @BindView(R.id.btn_save)
    Button btnSave;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_edit_name);
        ButterKnife.bind(this);
    }

    @OnClick(R.id.btn_save)
    public void onViewClicked() {
        String name = editText2.getText().toString();
        Intent intent = new Intent();
        intent.putExtra(EXTRA_NAME, name);
        setResult(RESULT_OK, intent);
        finish();
    }
}
```

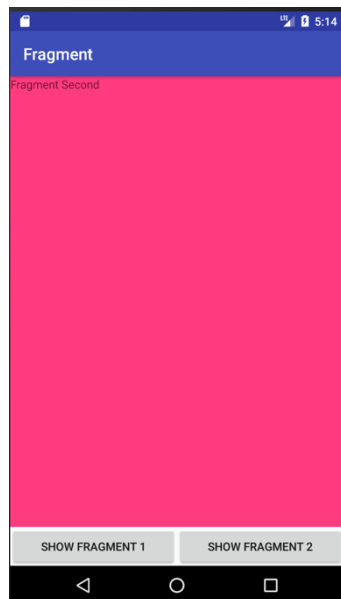
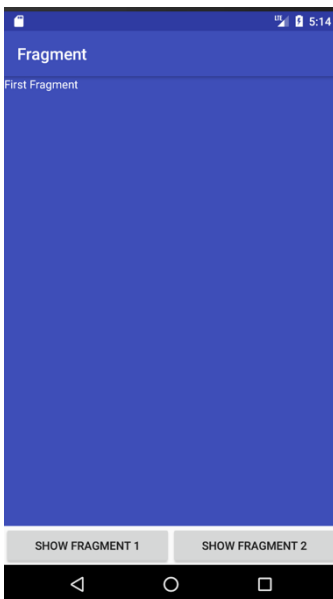
14. Coba running aplikasi nya kalau berhasil maka tampilannya akan seperti ini



MATERI 5

Fragment

fragment adalah komponen yang memiliki fungsi untuk menampilkan antarmuka ke pengguna melalui activity dengan memiliki layout xml sendiri, Fragment digunakan agar komponen tampilan aplikasi menjadi fleksibel dan dapat digunakan kembali (*reusable*). Fragment juga bisa disebut sub nya activity, satu activity bisa memiliki lebih dari satu fragment. Satu kelas Java dinyatakan sebagai sebuah fragment ketika kelas tersebut meng-*extends (inherit)* kelas Fragment. Untuk gambaran aplikasi nya kita akan buat seperti ini



1. Buat sebuah project baru
2. Isilah dengan parameter sebagai berikut

Project Name	Fragment
Package Name	com.dycode.edu.fragment
Min SDK	19

3. Di project ini kita menggunakan library butterknife
4. Buka Build,gradle(Module:app) dan tambahkan library untuk library butterknife di dependencies

```
implementation 'com.jakewharton:butterknife:8.8.0'
annotationProcessor 'com.jakewharton:butterknife-compiler:8.8.0'
```

5. Lalu sync now
6. Buka activity_main.xml dan ubah code nya menjadi seperti ini

```
<?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:layout_width="match_parent"
    android:orientation="vertical"
    android:layout_height="match_parent">
    <FrameLayout
        android:id="@+id/frame_container"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"/>
    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

```

        android:orientation="horizontal">
        <Button
            android:id="@+id/btn_fragment1"
            android:layout_width="0dp"
            android:layout_weight="1"
            android:text="@string/show_fragment_1"
            android:layout_height="wrap_content" />
        <Button
            android:id="@+id/btn_fragment2"
            android:layout_width="0dp"
            android:layout_weight="1"
            android:text="@string/show_fragment_2"
            android:layout_height="wrap_content" />

    </LinearLayout>
</LinearLayout>

```

7. Buka MainActivity , kemudian generate view nya menggunakan butterknife dan tambahkan code nya sehingga menjadi seperti berikut

```

public class MainActivity extends AppCompatActivity {

    @BindView(R.id.frame_container)
    FrameLayout frameContainer;
    @BindView(R.id.btn_fragment1)
    Button btnFragment1;
    @BindView(R.id.btn_fragment2)
    Button btnFragment2;

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

        FragmentManager fm = getSupportFragmentManager();
        FragmentTransaction fragmentTransaction =
        fm.beginTransaction();
    }
}

```

```

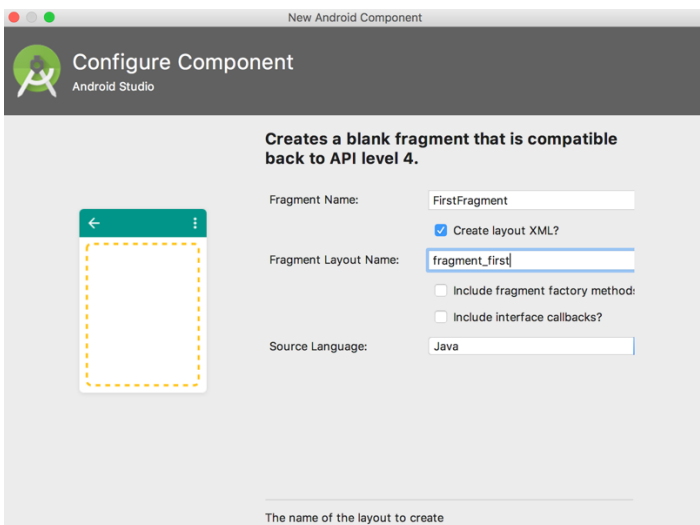
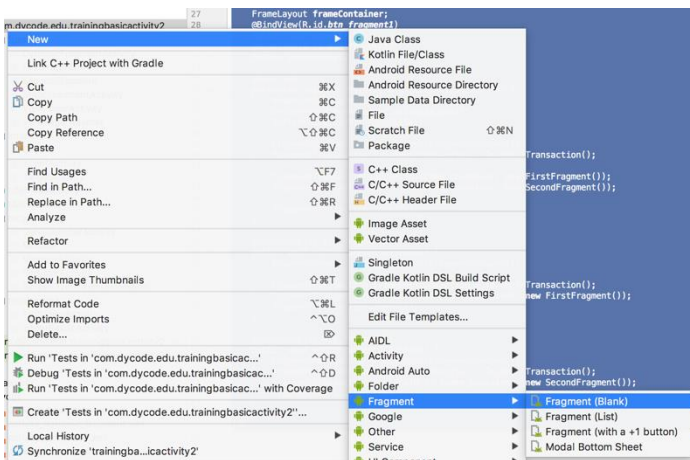
        fragmentTransaction.add(R.id.frame_container, new
FirstFragment());
        fragmentTransaction.add(R.id.frame_container, new
SecondFragment());
        fragmentTransaction.commit();
    }

    @OnClick(R.id.btn_fragment1)
    public void onBtnFragment1Clicked() {
        FragmentManager fm = getSupportFragmentManager();
        FragmentTransaction fragmentTransaction =
fm.beginTransaction();
        fragmentTransaction.replace(R.id.frame_container, new
FirstFragment());
        fragmentTransaction.commit();
    }

    @OnClick(R.id.btn_fragment2)
    public void onBtnFragment2Clicked() {
        FragmentManager fm = getSupportFragmentManager();
        FragmentTransaction fragmentTransaction =
fm.beginTransaction();
        fragmentTransaction.replace(R.id.frame_container, new
SecondFragment());
        fragmentTransaction.commit();
    }
}

```

Setelah selesai , sekarang kita tambahkan fragment nya, di project ini kita tambahkan 2 fragment didalam 1 Activity Buat fragment baru dengan nama **FirstFragment** dan **SecondFragment**, jangan lupa untuk men **un-check** Fragment factory dan dan interface callback nya



8. Buka fragment_first dan ubah code nya menjadi

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorPrimary">

    <TextView
        android:id="@+id/text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="First Fragment"
        android:textColor="#FFF"/>

</LinearLayout>
```

9. Dan yang terakhir di fragment_second.xml ubah code nya menjadi seperti ini

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical" android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorAccent">

    <TextView
        android:id="@+id/text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/fragment_second"/>

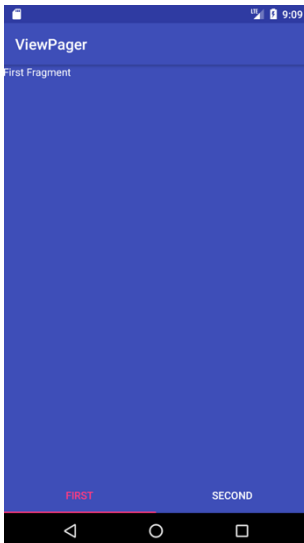
</LinearLayout>
```

10. Kalo sudah silahkan Running aplikasi nya

MATERI 6

ViewPager

ViewPager adalah komponen android yang sering dipakai untuk menampilkan data dalam format full screen dan bisa berpindah antar data dengan menggeser ke kiri atau kekanan. Contohnya adalah saat kita membuka galeri image. Kita akan buat aplikasi nya seperti ini, kurang lebih sama seperti project sebelum nya



1. Buat sebuah project baru
2. Isilah dengan parameter sebagai berikut

Project Name	ViewPager
Package Name	com.dycode.edu.viewpager
Min SDK	19

3. Buka Build.gradle(Module:app) di dependencies dan tambahkan library material design dan butterknife

```
implementation 'com.android.support.design:27.1.1'
implementation 'com.android.support:support-v4:27.1.1'

implementation 'com.jakewharton:butterknife:8.8.0'
annotationProcessor 'com.jakewharton:butterknife-compiler:8.8.0'
```

4. Pada activity_main.xml tambahkan code berikut

```
<?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:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical">

    <android.support.v4.view.ViewPager
        android:id="@+id/pager"
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1" />

    <android.support.design.widget.TabLayout
        android:id="@+id/tab"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
```

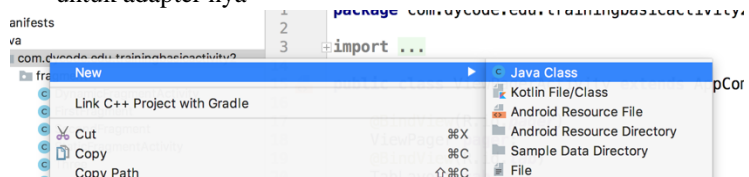
```

        android:background="@color/colorPrimary"
        app:tabMode="fixed"
        app:tabTextColor="#FFF"
        app:tabSelectedTextColor="@color/colorAccent"
    />

```

```
</LinearLayout>
```

5. Kita buat Java class baru dengan nama **ViewPagerAdapter** untuk adapter nya



```

public class ViewPagerAdapter extends FragmentStatePagerAdapter
{

    public ViewPagerAdapter(FragmentManager fm) {
        super(fm);
    }

    @Override
    public Fragment getItem(int position) {
        if (position == 0) {
            return new FirstFragment();
        } else if (position == 1) {
            return new SecondFragment();
        }
        throw new IllegalStateException("Position not valid");
    }

    @Override
    public int getCount() {
        return 2;
    }

    @Override
    public CharSequence getPageTitle(int position) {

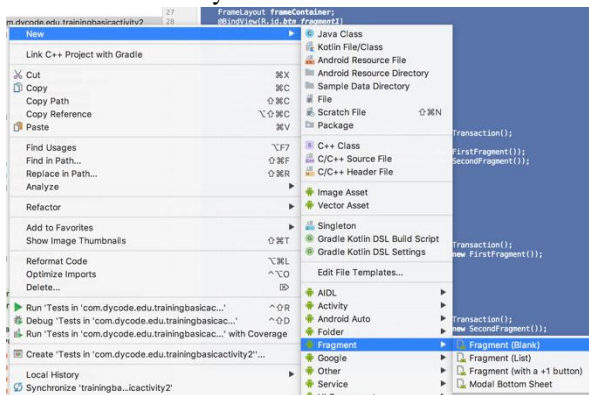
```

```

if (position == 0 || position % 2 == 0) {
    return "First";
} else if (position % 2 == 1) {
    return "Second";
}
throw new IllegalStateException("Position not valid");
}
}

```

6. Sama hal nya dengan project sebelumnya, Buat fragment baru dengan nama **FirstFragment** dan **SecondFragment**, jangan lupa untuk men **un-check** Fragment factory dan interface callback nya



7. Kemudian pada fragment_first.xml dan fragment_second ubah menjadi seperti ini

```

fragment_first.xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorPrimary">

```

```

<TextView
    android:id="@+id/text"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="First Fragment"
    android:textColor="#FFF"/>

</LinearLayout>

```

fragment_second.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/colorAccent">
    <TextView
        android:id="@+id/text"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="@string/fragment_second"/>
</LinearLayout>

```

8. Kemudian yang terakhir adalah di MainActivity kita tambahkan code nya menjadi seperti ini

```

public class MainActivity extends AppCompatActivity {

    @BindView(R.id.pager)
    ViewPager pager;
    @BindView(R.id.tab)
    TabLayout tab;

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

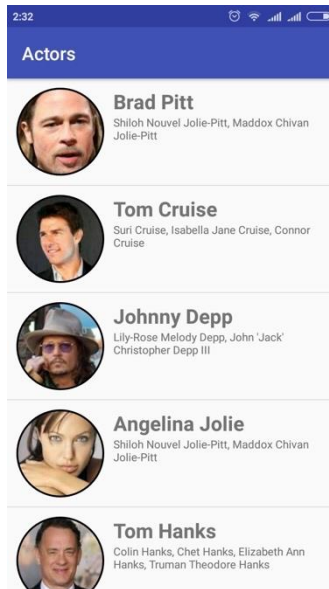
```

```
setContentView(R.layout.activity_main);  
ButterKnife.bind(this);  
  
ViewPagerAdapter viewPagerAdapter = new  
ViewPagerAdapter(getSupportFragmentManager());  
  
pager.setAdapter(viewPagerAdapter);  
tab.setupWithViewPager(pager);  
    }  
}
```

MATERI 7

Working With REST API (Retrofit dan Okhttp)

Retrofit adalah sebuah library android yang membantu pengembang untuk melakukan request ke sebuah endpoint REST API. Library ini dikembangkan oleh Square Inc (<https://github.com/square>) sebuah perusahaan yang berbasis di Amerika Serikat. Library ini menyederhanakan kode program yang digunakan untuk mengakses REST API. Tidak hanya untuk mengakses REST API dengan proses sederhana (GET, POST, PUT, DELETE) retrofit juga mendukung berbagai macam format autentikasi via http, menambahkan header pada request, menambahkan parameter serta mengirim data berupa image ke server. Untuk mengakses REST api dengan kode program bawaan android seorang programmer harus membuat banyak worker dan thread menggunakan Async Task. Hal ini bukan sebuah proses yang sederhana, apalagi jika request terhadap API tersebut menuntut adanya security, parameter khusus yang harus ditambahkan atau bahkan mengirim data selain text contoh berupa gambar ke server. Jika menggunakan Async Task dapat dipastikan programmer akan kesulitan dalam membuat program. Akan lebih baik jika semua proses yang berhubungan dengan networking tersebut ditangani oleh Retrofit. kita akan mencoba mengambil data dari <http://microblogging.wingnity.com/JSONParsingTutorial/jsonActors>, dan ini hasil dari aplikasi nya



1. Buat sebuah project baru
2. Isilah dengan parameter sebagai berikut

Project Name	Actors
Package Name	com.dycode.edu.actors
Min SDK	19

```
//recyclerview
implementation 'com.android.support:recyclerview-v7:27.1.1'

//Butterknife
annotationProcessor 'com.jakewharton:butterknife-compiler:8.8.0'
implementation 'com.jakewharton:butterknife:8.8.0'
```



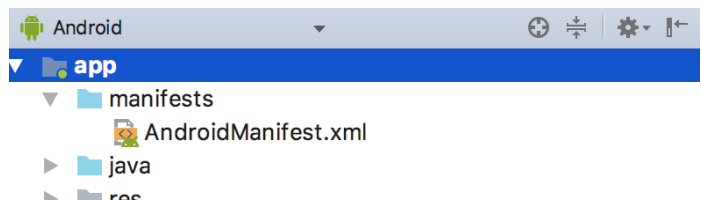
```
//for circle image
implementation 'de.hdodenhof:circleimageview:2.2.0'
//retrofit
implementation 'com.squareup.retrofit2:retrofit:2.3.0'
implementation 'com.squareup.retrofit2:converter-gson:2.3.0'

//httpClient
implementation 'com.squareup.okhttp3:okhttp:3.8.1'

//ImageLoader
implementation 'com.github.bumptech.glide:glide:3.7.0'
```

Karena aplikasi kita membutuhkan koneksi internet untuk mengakses API, maka aplikasi harus diijinkan untuk mengakses internet. Silakan buka manifest dan tambahkan line berikut ini.

```
<uses-permission
android:name="android.permission.INTERNET"/>
```



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

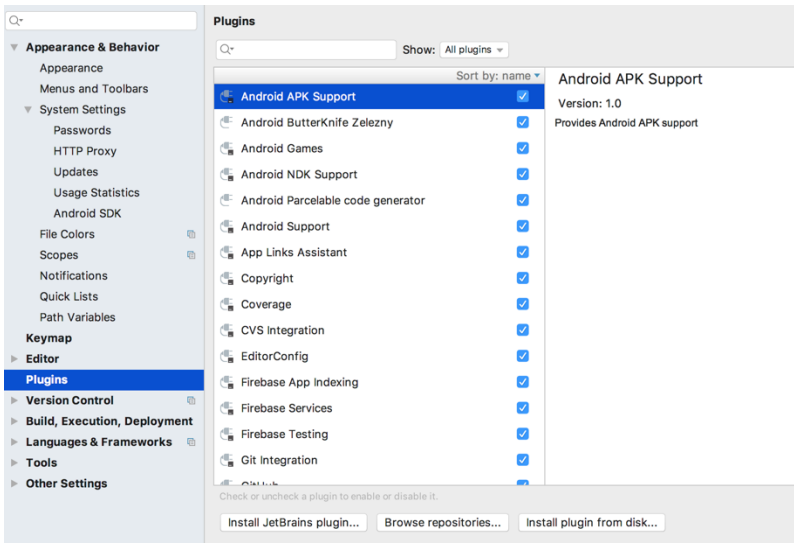
```
<uses-permission android:name="android.permission.INTERNET"/>
```

```
<application
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="Actors"
    android:roundIcon="@mipmap/ic_launcher_round"
    android:supportRtl="true"
```

3. Install Plugin RoboPOJOGenerator, buka **File->Settings->Plugins**
->Browse repositories -> lalu ketikan RoboPojGenerator ->
install,

Plugin untuk membantu kita membuat class POJO dari JSON.

Setelah selesai restart android studio nya



Category: All

RoboPOJOGenerator

33,924
★★★★★
2 months ago

CODE TOOLS

CODE TOOLS

Install

★★★★★ 33924 downloads
Updated 5/25/18 v1.9.1

Generate Java and Kotlin POJO files from JSON: GSON, FastJSON, AutoValue (GSON), Logan Square, Jackson, empty annotations template.

Supports: primitive types, multiple inner JSONArrays. key words: JsonToPojo, Json2Pojo, Kotlin, GSON, FastJSON, AutoValue, Jackson, LoganSquare, Moshi

Change Notes

Minor bug fix

Vendor

ROBOHORSE
<https://github.com/robohorse/RoboPOJOGenerator>
robohorsed@gmail.com

Size

3.3 M

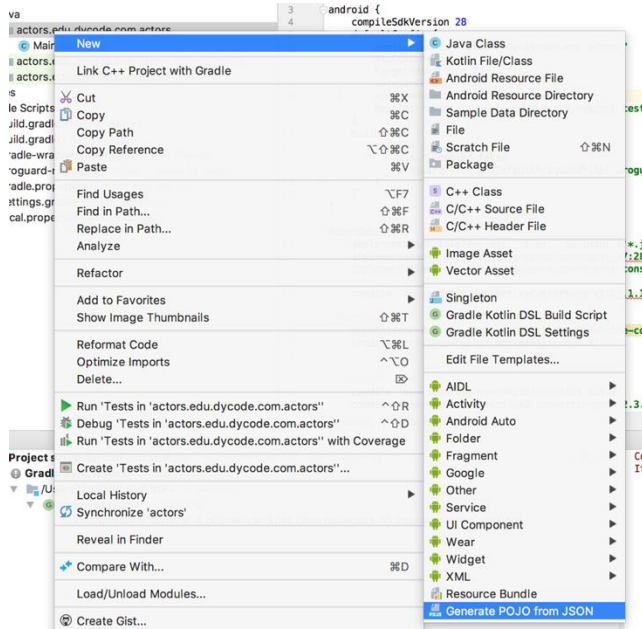
4. Kita akan mengambil data-data JSON nya dari <https://dycodeedu-artist.herokuapp.com/actors>
5. Buka url tersebut lalu copy semua data-data nya

```

- actors: [
  {
    name: "Brad Pitt",
    description: "William Bradley 'Brad' Pitt is an American actor and film producer. He has received a Golden Globe Award, a Screen Actors Guild Award, and three Academy Award nominations in acting categories.",
    dob: "December 18, 1963",
    country: "United States",
    height: "1.83 m",
    spouse: "Jennifer Aniston",
    children: "Shiloh Nouvel Pitt-Pitt, Maddox Chivan Pitt-Pitt",
    image: "https://s3.amazonaws.com/dycodeedu-artist/actors/Brad.jpg"
  },
  {
    name: "Tom Cruise",
    description: "Tom Cruise, is an American film actor and producer. He has been nominated for three Academy Awards and has won three Golden Globe Awards. He started his career at age 19 in the 1981 film 'Endless Love'. ",
    dob: "July 3, 1962",
    country: "United States",
    height: "1.78 m",
    spouse: "Raisa Winters",
    children: "Paul Cruise, Isabella Jane Cruise, Connor Cruise",
    image: "https://s3.amazonaws.com/dycodeedu-artist/actors/0188.jpg"
  },
  {
    name: "Johnny Depp",
    description: "John Christopher 'Johnny' Depp II is an American actor, film producer, and musician. He has won the Golden Globe Award and Screen Actors Guild award for Best Actor.",
    dob: "June 9, 1963",
    country: "United States",
    height: "1.78 m",
    spouse: "Carey Anne Zosky",
    children: "Lily-Rose Melody Depp, John 'Jack' Christopher Depp III",
    image: "https://s3.amazonaws.com/dycodeedu-artist/actors/Johnny.jpg"
  },
  {
    name: "Angelina Jolie",
    description: "Angelina Jolie is an American actress, film director, screenwriter, and author. She has received an Academy Award, two Screen Actors Guild Awards, and three Golden Globe Awards.",
    dob: "June 4, 1975",
    country: "United States",
    height: "1.68 m",
    spouse: "Brad Pitt",
    children: "Shiloh Nouvel Jolie-Pitt, Maddox Chivan Jolie-Pitt",
    image: "https://s3.amazonaws.com/dycodeedu-artist/actors/0186.jpg"
  },
  {
    name: "Tom Hanks",
    description: "Thomas Jeffrey 'Tom' Hanks is an American actor, producer, writer, and director. Hanks is best known for his roles in Big, A League of Their Own, Sleepless in Seattle, Forrest Gump, Apollo 13, and ...",
    dob: "July 9, 1946",
    country: "United States",
    height: "1.83 m"
  }
]

```

6. Lalu kita generate data JSON tersebut menjadi object (POJO) cara nya adalah **klik kanan di package aplikasi nya -> New -> Generate POJO from JSON**



7. Paste kan data json disini, ubah tipe nya menjadi GSON dan Root object name nya menjadi **ActorsResponse**, lalu generate

```
{
  actors: [
    {
      name: "Brad Pitt",
      description: "William Bradley 'Brad' Pitt is an American actor and film",
      dob: "December 18, 1963",
      country: "United States",
      height: "1.80 m",
      spouse: "Jennifer Aniston",
      children: "Shiloh Nouvel Jolie-Pitt, Maddox Chivan Jolie-Pitt",
      image: "http://microblogging.wingnity.com/JSONParsingTutorial/brad.jpg"
    },
    {
      name: "Tom Cruise",
      description: "Tom Cruise, is an American film actor and producer. He has",
      dob: "July 3, 1962",
      country: "United States",
      height: "1.70 m",
      spouse: "Katie Holmes".
    }
  ]
}
```

☐ none ☒ Rewrite existing classes ☐ Kotlin

☒ GSON ☒ use setters ☒ use getters ☐ use toString

☐ Jackson

☐ Logan Square

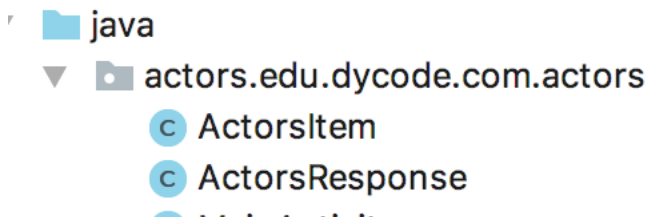
☐ Moshi

☐ AutoValue (GSON)

☐ FastJSON

Root object name

Kalo sudah maka akan terbuat 2 java class



Dan ini isi dari 2 file tersebut

```
ActorsResponse.java
public class ActorsResponse{

    @SerializedName("actors")
    private List<ActorsItem> actors;

    public void setActors(List<ActorsItem> actors){
        this.actors = actors;
    }

    public List<ActorsItem> getActors(){
        return actors;
    }
}
```

```
ActorsItem.java
public class ActorsItem{

    @SerializedName("country")
    private String country;

    @SerializedName("image")
    private String image;

    @SerializedName("children")
    private String children;

    @SerializedName("dob")
    private String dob;

    @SerializedName("name")
    private String name;

    @SerializedName("description")
    private String description;

    @SerializedName("spouse")
```

```
private String spouse;

@SerializedName("height")
private String height;

public void setCountry(String country){
    this.country = country;
}

public String getCountry(){
    return country;
}

public void setImage(String image){
    this.image = image;
}

public String getImage(){
    return image;
}

public void setChildren(String children){
    this.children = children;
}

public String getChildren(){
    return children;
}

public void setDob(String dob){
    this.dob = dob;
}

public String getDob(){
    return dob;
}

public void setName(String name){
    this.name = name;
}

public String getName(){
```

```

        return name;
    }

    public void setDescription(String description){
        this.description = description;
    }

    public String getDescription(){
        return description;
    }

    public void setSpouse(String spouse){
        this.spouse = spouse;
    }

    public String getSpouse(){
        return spouse;
    }

    public void setHeight(String height){
        this.height = height;
    }

    public String getHeight(){
        return height;
    }
}

```

8. Sekarang kita buka **activity_main.xml**, dan ubah menjadi seperti ini

```

<?xml version="1.0" encoding="utf-8"?>
<android.support.v7.widget.RecyclerView
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/list_actors"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

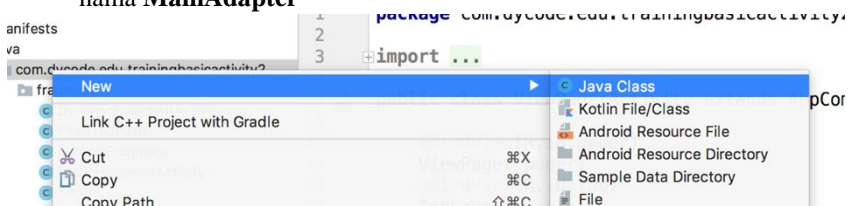
</android.support.v7.widget.RecyclerView>

```


9. Buat layout baru dengan nama **item_actors.xml**, dan tambahkan code berikut

```
<?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="wrap_content"
    android:padding="10dp">
    <de.hdodenhof.circleimageview.CircleImageView
        xmlns:app="http://schemas.android.com/apk/res-auto"
        android:id="@+id/img_profile"
        android:layout_width="96dp"
        android:layout_height="96dp"
        android:src="@mipmap/ic_launcher"
        app:civ_border_width="2dp" />
    <TextView
        android:id="@+id/txt_name"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginLeft="10dp"
        android:layout_toRightOf="@id/img_profile"
        android:text="Name"
        android:textSize="22sp"
        android:textStyle="bold" />
    <TextView
        android:id="@+id/txt_children"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@id/txt_name"
        android:layout_marginLeft="10dp"
        android:layout_toRightOf="@id/img_profile"
        android:text="Children"
        android:textSize="12sp" />
</RelativeLayout>
```

10. Selanjutnya kita buat class baru untuk adapternya, dengan nama **MainAdapter**



```
public class MainAdapter extends
RecyclerView.Adapter<MainAdapter.MainViewHolder> {

    public ArrayList<ActorsItem> listActors = new
    ArrayList<>();

    @Override
    public MainViewHolder onCreateViewHolder(ViewGroup
    parent, int viewType) {
        LayoutInflater layoutInflater =
        LayoutInflater.from(parent.getContext());
        View view = layoutInflater.inflate(R.layout.item_actors,
        parent, false);
        return new MainViewHolder(view);
    }

    @Override
    public void onBindViewHolder(final MainViewHolder
    holder, final int position) {

        holder.textName.setText(listActors.get(position).getChildren());

        holder.textChildren.setText(listActors.get(position).getName());

        Glide.with(holder.itemView.getContext())
            .load(listActors.get(position).getImage())
            .into(holder.imgProfile);
    }
}
```

```

@Override
public int getItemCount() {
    return listActors.size();
}

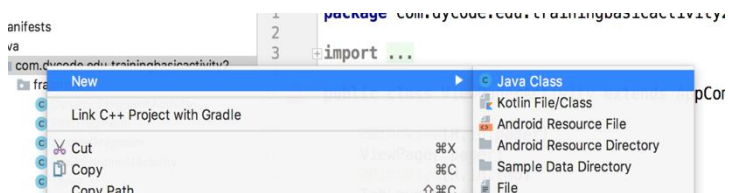
class MainViewHolder extends RecyclerView.ViewHolder {
    @BindView(R.id.txt_name)
    TextView txtName;
    @BindView(R.id.img_profile)
    CircleImageView imgProfile;
    @BindView(R.id.txt_children)
    TextView txtChildren;

    public MainViewHolder(View itemView) {
        super(itemView);
        ButterKnife.bind(this, itemView);
    }
}
}

```

11. ActorsService.java

Langkah selanjutnya adalah membuat sebuah interface. Di dalam interface ini berisi kumpulan service yang dibutuhkan oleh client (Android) dalam berkomunikasi dengan server

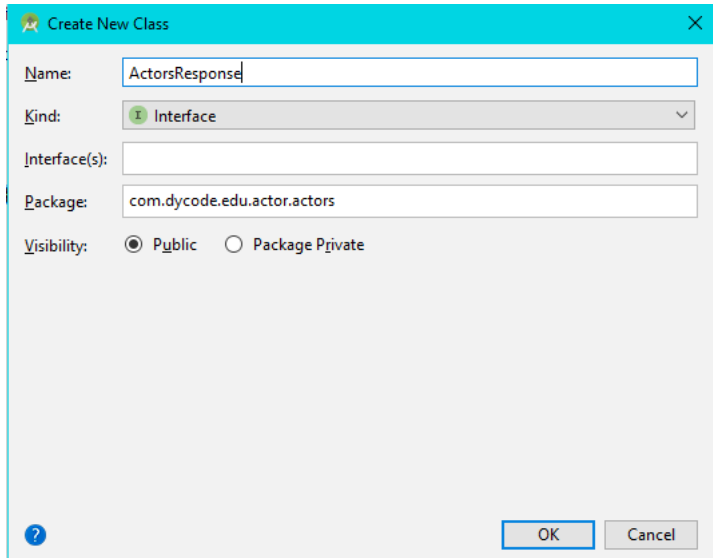


```

public interface ActorsService {

    @GET("/actors")
    Call<ActorsResponse> getActors();
}

```



Retrofit ini menggunakan library OkHttp sebagai core networkingnya. Untuk menangani request di network, harus disesuaikan request method yang ada di protokol http. Protokol itu antara lain yaitu :

GET

Method ini biasanya digunakan oleh klien untuk membaca data yang ada di server. Method ini tidak menambahkan data ke server.

POST

Method ini biasanya digunakan klient untuk merequest penambahan data baru (data yang sebelumnya tidak ada) di sisi server.

PUT

Method ini biasanya digunakan untuk mengupdate data yang sudah ada di sisi server.

DELETE

Method ini biasanya digunakan untuk menghapus data yang sudah ada di sisi server.

12. RestClient.java

Selanjutnya kita buat class baru untuk object retrofit nya, dimana objek ini yang bertanggung jawab menambahkan baseUrl dan jenis converter yang digunakan. Kenapa harus ada converter dimunculkan di sini? server akan mengirimkan JSON Object sebagai bentuk respon nya. Ketika sudah sampai ke HP Android, maka Android tidak bisa membaca JSON, Android hanya bisa membaca data dalam format Java. Oleh karena itu diperlukan sebuah converter untuk merubah JSON menjadi java

```
Public class Restclient {  
  
    private static ActorsResponse service;  
  
    public static ActorsResponse getActorsService(){  
  
        if(service == null){  
            String API_BASE_URL = "  
https://dycodeedu-artist.herokuapp.com";  
            OkHttpClient.Builder httpClient = new  
OkHttpClient.Builder();  
            Retrofit.Builder builder =  
                new Retrofit.Builder()  
                    .baseUrl(API_BASE_URL)  
                    .addConverterFactory(  
                        GsonConverterFactory.create()  
                    );  
  
            Retrofit retrofit =
```

```

        builder.client(httpClient.build()).build();

        service = retrofit.create(ActorsResponse.class);
    }
    return service;
}
}

```

13. MainActivity.java

Dan yang terakhir kita tampilkan data-data nya di mainactivity

```

public class MainActivity extends AppCompatActivity
{

    @BindView(R.id.list_actors)
    RecyclerView listActors;

    MainAdapter adapter;

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

        adapter = new MainAdapter();
        listActors.setLayoutManager(new
        LinearLayoutManager(this));
        listActors.setAdapter(adapter);
        listActors.addItemDecoration(new
        DividerItemDecoration(this,
        DividerItemDecoration.VERTICAL));

        RestClient.getActorsService().getActors().enqueue(new
        Callback<ActorsResponse>() {
            @Override
            public void onResponse(Call<ActorsResponse>
            call, Response<ActorsResponse> response) {

```

```

adapter.listActors.addAll(response.body().getActors());
    adapter.notifyDataSetChanged();
}

@Override
public void onFailure(Call<ActorsResponse>
call, Throwable t) {
    Toast.makeText(MainActivity.this,
t.getMessage(), Toast.LENGTH_SHORT).show();
}
});
}

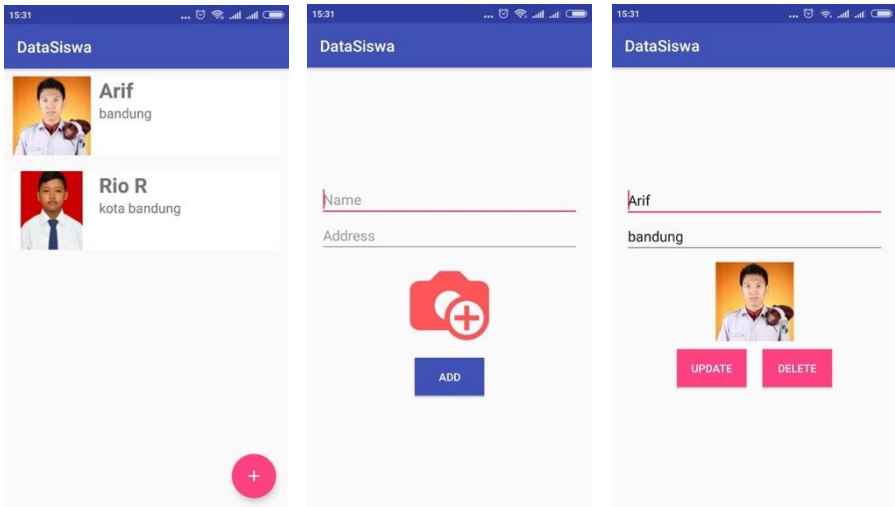
}

```

14. Jika sudah silahkan running aplikasi nya

MATERI 8

Membuat CRUD Data Siswa menggunakan Room Persistence



Kita akan membuat aplikasi data siswa , dan akan menggunakan beberapa library :

1. **Room Persistence** = untuk menyimpan data – data

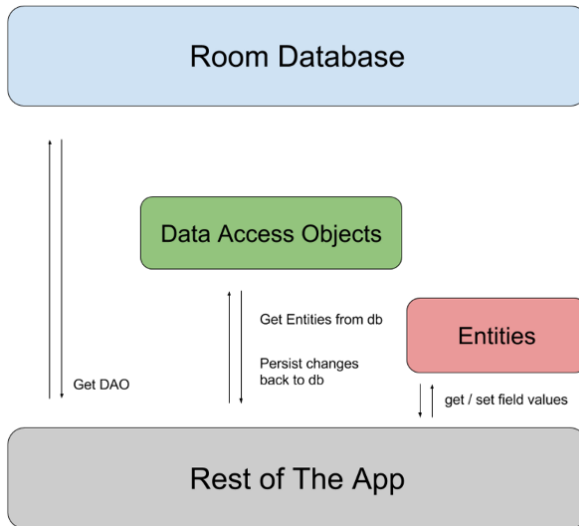
Room atau Room Persistence Library adalah sebuah library yang berfungsi sebagai abstraction layer di atas SQLite database yang memungkinkan kita menggunakan SQLite dengan lebih mudah, efektif dan efisien

Perbedaan mendasar antara Room dan SQLite sebelumnya adalah

1. **sqlite**—Menggunakan SQLiteOpenHelper dan interface SQLite standar (DBHelper, dsb).
2. **room**—Full abstraction menggunakan Room library dan akses yang lebih efisien.

Pada prinsipnya Room dibagi menjadi tiga buah komponen utama, yaitu Database, Entity, dan DAO :

3. **Entity** merepresentasikan data pada sebuah tabel seperti di database pada umumnya, dibuat menggunakan annotation pada java data object. Setiap entity mempunyai satu tabel sendiri.
4. **DAO** (Data Access Object) menggambarkan method2 yang mengakses database, termasuk methods standar seperti CRUD (Create, Read, Update, Delete). Menggunakan annotation untuk mengikat SQL query ke suatu method.
5. **Database** adalah holder class yang menggunakan annotation untuk menampilkan daftar dari entity2 yang ada dan juga database version. Kelas ini juga berisi daftar dari DAO yang ada.



2. **Recyclerview & CardView** = Untuk menampilkan list data siswa

Recycler View adalah sebuah widget atau versi lanjutan dari listview. Recycler View adalah List View yang lebih canggih dan fleksible yang bisa menampilkan banyak data dalam jumlah besar dan efesien dan dengan elemen yang mampu berubah – ubah tergantung interaksi user (pengguna)



Gambar di atas menerangkan beberapa komponen yang harus Anda ketahui sebelum menggunakan `recyclerview`.

1. **RecyclerView dan LayoutManager** : Komponen antarmuka yang bertugas untuk menampilkan data set yang dimiliki di dalamnya. `LayoutManager` akan mengatur posisi tampilan data baik itu secara list (vertikal), grid (baris dan kolom) atau `staggeredgrid` (grid yang memiliki susunan tak seragam / tak beraturan)
2. **Adapter** : Komponen yang akan mengatur bagaimana menampilkan data set ke dalam **RecyclerView**. Di sinilah terjadi proses pengisian tampilan (**ViewInflate**) dari file layout xml untuk tiap elemen dari data yang sebelumnya terpasang (*bind*) ke dalam **RecyclerView**.
3. **Dataset** : Kumpulan data yang dimiliki dan ingin ditampilkan. Bisa berupa array, list maupun obyek map.
4. **Item Animator** : Ini yang spesial. Kita bisa pasang animasi untuk tiap item di dalamnya. Contoh animasi yang umum seperti penambahan (*add*) dan penghapusan (*removal*) item. Kita akan mempelajari hal ini pada materi terpisah.

1. Buat sebuah project baru
2. Isilah dengan parameter sebagai berikut

Project Name	DataSiswa
Package Name	com.dycode.edu.datasiswa
Min SDK	19

3. Selanjutnya yaitu kita harus menambahkan library untuk kebutuhan aplikasi kita
4. Buka build.gradle(Module: app) dan di dependencies tambahkan :

```
//1. add recyclerView
implementation 'com.android.support:recyclerview-v7:27.1.1'
//2. add room
implementation 'android.arch.persistence.room:runtime:1.0.0'
annotationProcessor "android.arch.persistence.room:compiler:1.0.0"
//3. Add material design
implementation 'com.android.support:design:27.1.1'
//4. Add CardView
implementation 'com.android.support:cardview-v7:27.1.1'
//5. Add ImagePicker
implementation 'com.github.nguyenhoanglam:ImagePicker:1.3.0'
//6. Add Glide for image
implementation 'com.github.bumptech.glide:glide:4.7.1'
annotationProcessor 'com.github.bumptech.glide:compiler:4.7.1'
//7. Add butterknife
implementation 'com.jakewharton:butterknife:8.8.1'
annotationProcessor 'com.jakewharton:butterknife-compiler:8.8.1'
```

5. Buka `build.gradle(project..)` dan tambahkan repositori nya sehingga menjadi seperti ini

```
allprojects {  
    repositories {  
        google()  
        jcenter()  
        maven { url "https://maven.google.com" }  
        maven { url "https://jitpack.io" }  
    }  
}
```

6. Jika sudah , **sync now** project nya yang berada di kanan atas
7. Kemudian buat beberapa class baru dengan nama **SiswaModel** dan tambahkan code berikut,

Class ini untuk Entity Room Database nya

```
@Entity  
public class SiswaModel implements Parcelable {  
  
    @PrimaryKey(autoGenerate = true)  
    private int id;  
  
    @ColumnInfo(name = "name")  
    String name;  
  
    @ColumnInfo(name = "address")  
    String address;  
  
    @ColumnInfo(name = "path_picture")  
    String pathPicture;  
  
    public SiswaModel() {  
    }  
}
```

```

protected SiswaModel(Parcel in) {
    id = in.readInt();
    name = in.readString();
    address = in.readString();
    pathPicture = in.readString();
}

public static final Creator<SiswaModel> CREATOR = new
Creator<SiswaModel>() {
    @Override
    public SiswaModel createFromParcel(Parcel in) {
        return new SiswaModel(in);
    }

    @Override
    public SiswaModel[] newArray(int size) {
        return new SiswaModel[size];
    }
};

public int getId() {
    return id;
}

public void setId(int id) {
    this.id = id;
}

public String getName() {
    return name;
}

public void setName(String name) {
    this.name = name;
}

public String getAddress() {
    return address;
}

public void setAddress(String address) {

```

```

        this.address = address;
    }

    public String getPathPicture() {
        return pathPicture;
    }

    public void setPathPicture(String pathPicture) {
        this.pathPicture = pathPicture;
    }

    @Override
    public int describeContents() {
        return 0;
    }

    @Override
    public void writeToParcel(Parcel parcel, int i) {
        parcel.writeInt(id);
        parcel.writeString(name);
        parcel.writeString(address);
        parcel.writeString(pathPicture);
    }
}

```

8. Kita buat class baru lagi dengan **SiswaDao**, class ini berisi perintah-perintah query ke database

```

@Dao
public interface SiswaDao {

    @Query("SELECT * FROM SiswaModel")
    List<SiswaModel> getAll();

    @Insert(onConflict = OnConflictStrategy.REPLACE)
    void insertAll(SiswaModel siswaModel);

    @Delete
    public void deleteUsers(SiswaModel siswaModel);
}

```

```

    @Update
    public void update(SiswaModel siswaModel);

}

```

9. AppDatabase.java

Class ini untuk menampilkan entity-entity dan version database nya

```

@Database(entities = {SiswaModel.class}, version = 1)
public abstract class AppDatabase extends RoomDatabase {
    public abstract SiswaDao userDao();
}

```

10. MyApp.java

Class ini untuk inialisasi database Room

```

public class MyApp extends Application {

    public static AppDatabase db;

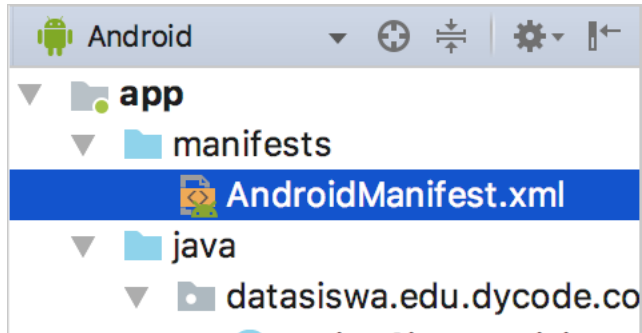
    @Override
    public void onCreate() {
        super.onCreate();
        db =
Room.databaseBuilder(getApplicationContext(),

AppDatabase.class, "siswa.db").allowMainThreadQueries().build();
    }

}

```

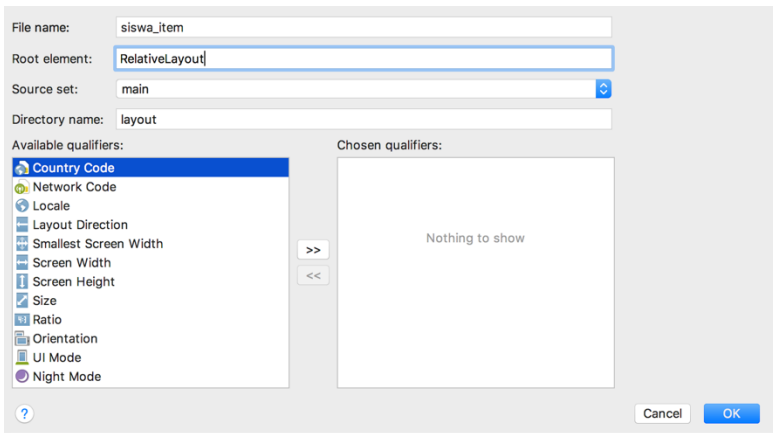
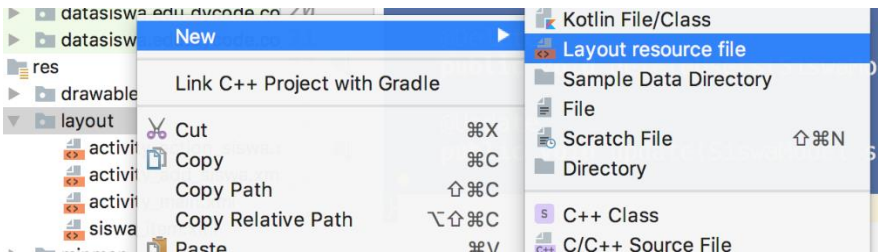

11. Setelah selesai, buka AndroidManifest, dan tambahkan properties name yang value nya mengambil dari class MyApp



```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android
package="datasiswa.edu.dycode.com.datasisv

<application
    android:name=".MyApp"
    android:allowBackup="true"
    android:icon="@mipmap/ic_launcher"
    android:label="@string/app_name"
```

12. Buat layout baru dengan nama siswa_item, layout ini berfungsi untuk menampilkan isi dari list yang ada di database, cara bikin layout baru adalah **klik kanan di layout -> New -> layout resource file**



Kemudian tambahkan code xml berikut

```
siswa_item.xml
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:padding="10dp"
    android:layout_height="wrap_content">

    <android.support.v7.widget.CardView
```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <RelativeLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content">

            <ImageView
                android:layout_width="100dp"
                android:id="@+id/img_profile"
                android:src="@mipmap/ic_launcher"
                android:layout_height="100dp" />

            <TextView
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:id="@+id/txt_name"
                android:layout_marginLeft="10dp"
                android:textSize="26sp"
                android:textStyle="bold"
                android:layout_toRightOf="@id/img_profile"
                android:text="Name"/>

            <TextView
                android:layout_width="wrap_content"
                android:text="Address"
                android:id="@+id/txt_address"
                android:layout_toRightOf="@id/img_profile"
                android:layout_height="wrap_content"
                android:layout_marginLeft="10dp"
                android:textSize="17sp"
                android:layout_below="@id/txt_name"/>

        </RelativeLayout>
    </android.support.v7.widget.CardView>
</RelativeLayout>

```

13. MainAdapter.java

Class ini untuk menampilkan data set ke RecyclerView

```

public class MainAdapter extends
RecyclerView.Adapter<MainAdapter.ViewHolder> {

    private Context mContext;
    public static final String EXTRA_SISWA =
"EXTRA_MEMBER";
    public List<SiswaModel> arrayList = new ArrayList<>();

    public MainAdapter(Context mContext, List<SiswaModel>
albumList) {
        this.mContext = mContext;
        this.arrayList = albumList;
    }

    @NonNull
    @Override
    public ViewHolder onCreateViewHolder(@NonNull
ViewGroup parent, int viewType) {
        LayoutInflater inflater =
LayoutInflater.from(parent.getContext());
        View view = inflater.inflate(R.layout.siswa_item,
parent, false);
        return new ViewHolder(view);
    }

    @Override
    public void onBindViewHolder(@NonNull final ViewHolder
holder, final int position) {
        final SiswaModel siswa = arrayList.get(position);
        holder.textName.setText(siswa.name);
        holder.textAddress.setText(siswa.address);

        //untuk menampilkan gambar menggunakan glide
        Glide.with(holder.itemView.getContext())
            .load(siswa.pathPicture)
            .into(holder.imgProfile);

        holder.itemView.setOnClickListener(new
View.OnClickListener() {

```

```

        @Override
        public void onClick(View view) {
            Intent intent = new
            Intent(holder.itemView.getContext(),
            ActionSiswaActivity.class);
            intent.putExtra(EXTRA_SISWA,
            arrayList.get(position));
            holder.itemView.getContext().startActivity(intent);
        }
    });
}

@Override
public int getItemCount() {
    return arrayList.size();
}

class ViewHolder extends RecyclerView.ViewHolder{

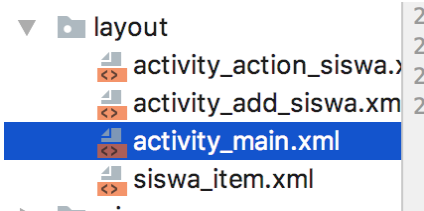
    @BindView(R.id.txt_name)
    TextView textName;
    @BindView(R.id.img_profile)
    ImageView imgProfile;
    @BindView(R.id.txt_address)
    TextView textAddress;

    public ViewHolder(View itemView) {
        super(itemView);
        ButterKnife.bind(this, itemView);
    }
}
}

```

*Note – pada ActionSiswaActivity.java akan terjadi error karena kita belum menambahkan activity tersebut

14. Buka **activity_main.xml** yang berada di folder layout dan ubah code nya menjadi seperti ini



```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
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.RecyclerView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:id="@+id/recyclerView">

    </android.support.v7.widget.RecyclerView>

    <android.support.design.widget.FloatingActionButton
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_margin="16dp"
        android:id="@+id/fab"
        android:src="@drawable/ic_add_white_24dp"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"/>
</RelativeLayout>
```

15. Kemudian buka **MainActivity.java** untuk menampilkan semua data-data mahasiswa, disini kita menggunakan butterknife untuk binding ke view, silahkan generate butterknife nya seperti cara yang pernah disampaikan di materi sebelum nya,

```
public class MainActivity extends AppCompatActivity {

    @BindView(R.id.recyclerView)
    RecyclerView recyclerView;
    @BindView(R.id.fab)
    FloatingActionButton fab;

    List<SiswaModel> lisSiswaModel = new ArrayList<>();

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

        //get data from db
        lisSiswaModel = MyApp.db.userDao().getAll();

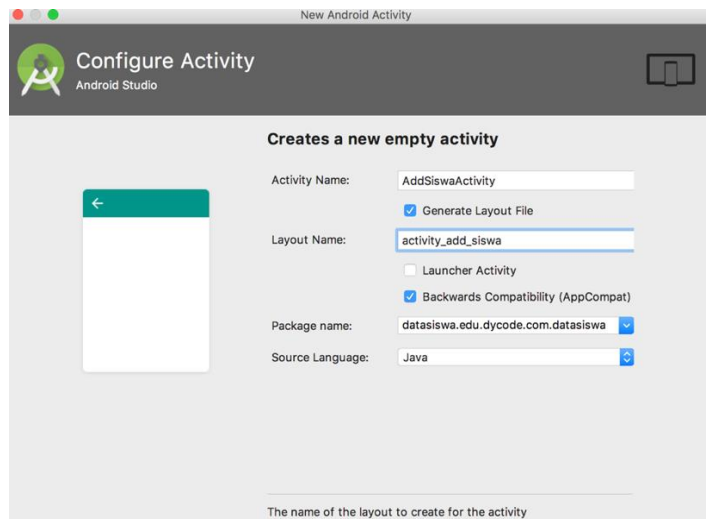
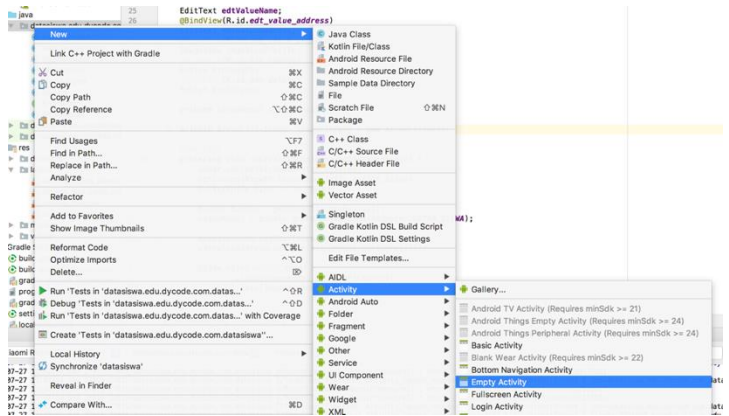
        recyclerView.setLayoutManager(new
        LinearLayoutManager(this));
        MainAdapter adapter = new MainAdapter(this,
        lisSiswaModel);
        recyclerView.setAdapter(adapter);

    }

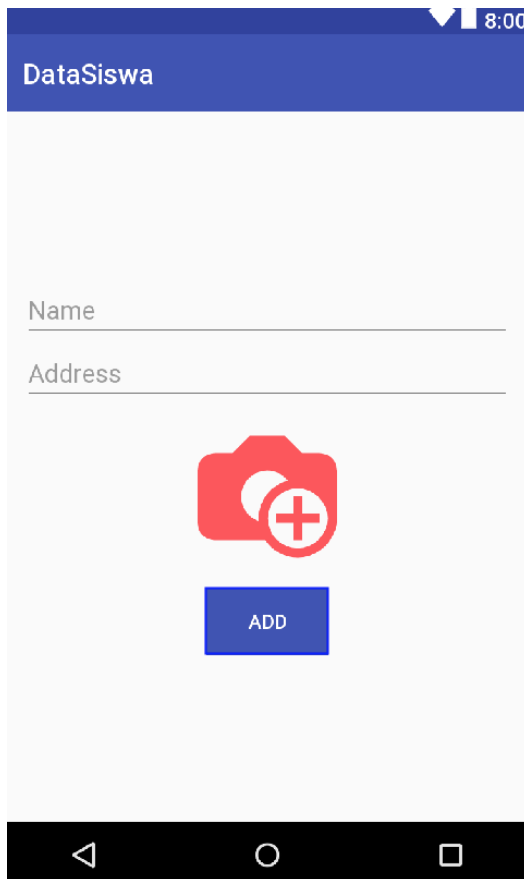
    @OnClick(R.id.fab)
    public void onViewClicked() {
        startActivity(new Intent(this, AddSiswaActivity.class));
    }
}
```

16. Add Siswa

Setelah selesai , kita buat Activity baru untuk menambahkan data siswa ke database, nama activity nya adalah AddSiswaActivity



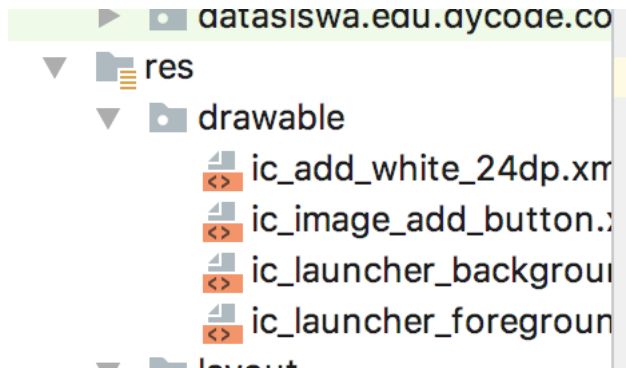
17. Buka layout **activity_add_siswa.xml** dan kita ubah layout nya menjadi seperti gambar dibawah



*Note :

- Untuk gambar camera, silahkan cari di google gambar nya dan download,

Setelah selesai copy gambar nya dan paste di folder drawable



- Untuk nama file nya **tidak boleh** ada huruf besar, ada angka dan spasi

```
<?xml version="1.0" encoding="utf-8"?>
<RelativeLayout
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:padding="16dp"
android:gravity="center"
android:layout_height="match_parent"
tools:context=".AddSiswaActivity">

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

</EditText
```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="Address"
        android:id="@+id/edt_address"
        android:layout_below="@id/edt_name"/>

```

```

<ImageView
    android:layout_width="100dp"
    android:layout_height="100dp"
    android:layout_below="@id/edt_address"
    android:layout_centerHorizontal="true"
    android:layout_marginTop="16dp"
    android:id="@+id/img_add_profile"
    android:src="@drawable/ic_image_add_button"/>

```

```

<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Add"
    android:id="@+id/btn_add"
    android:layout_marginTop="16dp"
    android:background="@color/colorPrimary"
    android:textColor="#fff"
    android:layout_below="@id/img_add_profile"
    android:layout_centerHorizontal="true"/>

```

```

</RelativeLayout>

```

18. AddSiswaActivity.java

```

public class AddSiswaActivity extends AppCompatActivity {

    @BindView(R.id.edt_name)
    EditText edtName;

    @BindView(R.id.edt_address)
    EditText edtAddress;

    @BindView(R.id.img_add_profile)
    ImageView imgAddProfile;
}

```

```

@BindView(R.id.btn_add)
Button btnAdd;
private SiswaModel mSiswaModel;

//untuk class image nya diambil dari class library
nguyenHoanglam, bukan dari android media
private ArrayList<Image> imageLibrary = new
ArrayList<>();

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_add_siswa);
    ButterKnife.bind(this);
}

@OnClick(R.id.img_add_profile)
public void onViewClickedImage() {
    ImagePicker.with(this)
        .setFolderMode(true)
        .setMaxSize(10)
        .setMultipleMode(false)
        .setCameraOnly(false)
        .setFolderTitle("Albums")
        .setSelectedImages(imageLibrary)
        .setAlwaysShowDoneButton(true)
        .setKeepScreenOn(true)
        .start();
}

@Override
protected void onActivityResult(int requestCode, int
resultCode, Intent data) {
    super.onActivityResult(requestCode, resultCode, data);
    if (data!=null){
        imageLibrary =
data.getParcelableArrayListExtra(Config.EXTRA_IMAGES);

        Glide.with(this)

```

```

        .load(imageLibrary.get(0).getPath()).into(imgAddProfile);
    }
}

@OnClick(R.id.btn_add)
public void onViewClicked() {
    if (!edtName.getText().toString().isEmpty()
        && !edtAddress.getText().toString().isEmpty()
        && !imageLibrary.isEmpty()) {

        mSiswaModel = new SiswaModel();

        mSiswaModel.setName(edtName.getText().toString());

        mSiswaModel.setAddress(edtAddress.getText().toString());

        mSiswaModel.setPathPicture(imageLibrary.get(0).getPath().
            toString());
        MyApp.db.userDao().insertAll(mSiswaModel);

        Intent intent = new Intent(new Intent(this,
            MainActivity.class));

        intent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);

        intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK);
        startActivity(intent);

    }
}
}

```

19. Update dan Delete

Langkah terakhir yaitu kita akan memanipulasi data - data nya (update & delete)

Buat activity baru dengan nama **ActionSiswaActivity**

20. Buka action_siswa_activity.xml

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:gravity="center"
    android:padding="16dp"
    android:orientation="vertical">

    <EditText
        android:layout_width="match_parent"
        android:id="@+id/edt_value_name"
        android:layout_height="wrap_content" />

    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:id="@+id/edt_value_address"/>

    <ImageView
        android:layout_width="100dp"
        android:layout_marginTop="10dp"
        android:id="@+id/img_value_profile"
        android:layout_height="100dp" />

    <LinearLayout
        android:layout_width="match_parent"
        android:gravity="center"

        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <Button
            android:layout_width="wrap_content"
```

```

        android:id="@+id/btn_update"
        android:layout_margin="10dp"
        android:textColor="#fff"
        android:background="@color/colorAccent"
        android:layout_height="wrap_content"
        android:text="update"/>

<Button
    android:layout_width="wrap_content"
    android:layout_margin="10dp"
    android:textColor="#fff"
    android:background="@color/colorAccent"
    android:id="@+id/btn_delete"
    android:layout_height="wrap_content"
    android:text="delete"/>
</LinearLayout>
</LinearLayout>

```

21. ActionSiswaActivity.java

```

public class ActionSiswaActivity extends AppCompatActivity {
    @BindView(R.id.edt_value_name)
    EditText edtValueName;
    @BindView(R.id.edt_value_address)
    EditText edtValueAddress;
    @BindView(R.id.img_value_profile)
    ImageView imgValueProfile;
    @BindView(R.id.btn_update)
    Button btnUpdate;
    @BindView(R.id.btn_delete)
    Button btnDelete;
    private SiswaModel siswaModel;
    private ArrayList<Image> imageLibrary = new
    ArrayList<>();
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_action_siswa);
        ButterKnife.bind(this);
    }
}

```

```

        Bundle bundle = getIntent().getExtras();
        siswaModel =
        bundle.getParcelable(MainAdapter.EXTRA_SISWA);

        edtValueName.setText(siswaModel.getName());
        edtValueAddress.setText(siswaModel.getAddress());

        Glide.with(this)
            .load(siswaModel.getPathPicture())
            .into(imgValueProfile);
    }

    @OnClick(R.id.img_value_profile)
    public void onImgValueProfileClicked() {
        ImagePicker.with(this)
            .setFolderMode(true)
            .setMaxSize(10)
            .setMultipleMode(false)
            .setCameraOnly(false)
            .setFolderTitle("Albums")
            .setSelectedImages(imageLibrary)
            .setAlwaysShowDoneButton(true)
            .setKeepScreenOn(true)
            .start();
    }

    @Override
    protected void onActivityResult(int requestCode, int
    resultCode, Intent data) {
        super.onActivityResult(requestCode, resultCode, data);
        if (data != null) {
            imageLibrary =
            data.getParcelableArrayListExtra(Config.EXTRA_IMAGES);

            siswaModel.setPathPicture(imageLibrary.get(0).getPath());

            Glide.with(this)

            .load(imageLibrary.get(0).getPath()).into(imgValueProfile);
        }
    }

```



```

    @OnClick(R.id.btn_update)
    public void onBtnUpdateClicked() {
        siswaModel.setName(editValueName.getText().toString());

        siswaModel.setAddress(editValueAddress.getText().toString());
        siswaModel.setPathPicture(siswaModel.getPathPicture());

        MyApp.db.userDao().update(siswaModel);

        startActivity(new Intent(this, MainActivity.class));
        Intent intent = new Intent(new Intent(this,
        MainActivity.class));
        intent.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK);
        intent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
        startActivity(intent);
    }
    @OnClick(R.id.btn_delete)
    public void onBtnDeleteClicked() {
        MyApp.db.userDao().deleteUsers(siswaModel);
        startActivity(new Intent(this, MainActivity.class));
        finish();
    }
}

```

Jika sudah, buka kembali class **MainAdapter**, di class ini ActionSiswaActivity nya belum kita import, cara import nya adalah dengan cara pindahkan cursor nya ke ActionSiswaActivity, lalu tekan **Alt+enter** untuk import .

Jika sudah silahkan running aplikasi nya

DAFTAR PUSTAKA

Codepolitan “Eliminasi findViewById dengan ButterKnife Library”

<https://www.codepolitan.com/eliminasi-findviewbyid-dengan-butterknife-library-5a44ce9421bab>

Medium Mahesa Iqbal Ridwansyah “Intent Eksplisit Dan Implisit Pada Android”

<https://medium.com/@mahesaiqbal6/intent-eksplisit-dan-implisit-pada-android-87ee48a773b4>

Modul Pelatihan Basic Android 2017

Team Politeknik Negeri Malang

Medium ajay “Building database with Room Persistence Library”

<https://medium.com/@ajaysaini.official/building-database-with-room-persistence-library-ecf7d0b8f3e9>

Rizki Syaputra, “Basic Android” Imastudio, 2016

Using the RecyclerView

<https://guides.codepath.com/android/using-the-recyclerview>

Penulis:

Farid Savarudin

farid@dycode.com

Editor:

Dani Sofyan

dani@dycode.com

Desain Sampul

Rio Rinaldi

riorin@dycode.com

Diterbitkan oleh:

Kementrian Komunikasi dan Informatika RI

Direktorat Jenderal Aplikasi Informatika

Jl. Medan Merdeka Barat No.9 Jakarta Pusat

Web: <https://aptika.kominfo.go.id/>

Phone: 021-3810305

Profil Penulis



Nama : Farid Savarudin

Tempat/Tanggal Lahir : Tasikmalaya / 24 juni 1997

Alamat : Kp. Babakan Muncang RT/RW 03/04,
Desa Cisaruni, kec.Padakembang,
Kab.Tasikmalaya

No.Hp/WA/Telegram : +6285353498964

Email : faridsavarudin@gmail.com

Website : faridsava.blogspot.co.id

Adalah seorang android developer berasal dari tasikmalaya yang sekarang bekerja di PT. Dycode Cominfotech Development sejak 2017, penulis juga aktif di berbagai komunitas developer bandung, menjadi salah satu facilitator Indonesia Android Kejar, senang sekali sharing mengenai hal – hal terbaru dari android

android

Development

Membangun Aplikasi berbasis Android
menggunakan Android Studio dan
bahasa Pemrograman Java

Buku ini berisi tentang panduan untuk
membuat aplikasi android dasar
pembuatan aplikasi sederhana dari awal hingga
menjadi sebuah aplikasi sederhana.
pada buku ini tools yang digunakan adalah
Android Studio dan menggunakan
bahasa pemrograman java.

Materi yang dibahas meliputi pembuatan
aplikasi sederhana yang sering
ada di dalam sebuah project

Pembuatan dan pembahasan aplikasi
dibuat sesederhana mungkin dengan harapan
agar pembaca dapat memahami dengan
mudah dalam mengaplikasikannya.