

FIRDAN ARDIANSYAH



Pengenalan Dasar

ANDROID

Programming

Buku ini kupersembahkan untuk :

Bundaku Tercinta.

Ria Amelia.

*Teman-teman satu perjuangan
di kampus Gunadarma.*

Para Mujahidin Palestina

Pengantar Penulis.

Telah banyak juga buku-buku yang membahas mengenai Android programming, mulai yang dasar hingga yang tingkat mahir, namun pada buku ini penulis hanya membahas mengenai yang dasar-dasar saja karena penulis juga masih harus banyak belajar.

Anda bebas mendistribusikan dan membagikan buku elektronik ini kepada siapa saja ataupun ingin menggunakan buku ini sebagai bahan pembelajaran ataupun pelatihan bagi suatu institusi pendidikan dengan menyertakan sumber dan nama penulis.

Penulis tidak memungut uang sepeserpun atas waktu dan tenaga yang penulis gunakan untuk menyelesaikan buku ini, namun penulis hanya mengharap bagi siapa saja yang mendapatkan buku ini dan mendapat manfaat setelah menggunakannya untuk bersedekah kepada orang-orang yang kurang mampu sejumlah Rp 1000 rupiah, namun jika ada yang ingin memberikan lebih silahkan saja :).

Buku ini terdiri dari 5 bab yaitu sebagai berikut :

Bab 1 : Berisi sejarah Android.

Bab 2 : Berisi cara instalasi Android SDK dan Eclipse.

Bab 3 : Membuat aplikasi hello world di Android.

Bab 4 : Merancangan tampilan program dan juga beberapa cara untuk memanipulasi widget.

Bab 5 : Membahas cara membuat aplikasi pemesanan fast food dan juga membuat aplikasi penghitung luas bangun datar.

Akhir kata penulis ucapkan banyak terima kasih dan semoga buku ini bermanfaat bagi yang menggunakannya.

Depok, September 2011.

Bab 1

Basi Basi Android

Disini saya cuma sekedar pingin berbasa-basi mengenai apa itu Android, jadi kalo anda ingin melewati bab 1 ini langsung aja dilewatkan ya, tapi bab 1 ini penting juga sih buat yang mau tau sejarah mengenai Android (weleh, gak konsisten banget nih orang).

Android adalah sistem operasi untuk telepon seluler yang berbasis Linux. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka sendiri sehingga dapat digunakan oleh bermacam peranti penggerak. Awalnya Google Inc. membeli Android Inc. pendatang baru yang membuat software (perangkat lunak) untuk telepon genggam. Kemudian untuk mengembangkan Android di bentuklah Open Handset Alliance yang merupakan gabungan dari 34 perusahaan peranti keras, peranti lunak dan telekomunikasi termasuk Google, HTC, Intel, Motorola, Qualcomm, T-Mobile, dan NVidia.

Pada saat perilis perdana Android pada tanggal 5 november 2007, Android bersama Open Handset Alliance menyatakan mendukung pengembangan standar terbuka

Firdan Ardiansyah | Pengenalan Dasar Android Programming.

pada perangkat seluler. Di lain pihak, Google merilis kode-kode Android dibawah lisensi Apache, sebuah lisensi perangkat lunak dan standar terbuka perangkat seluler. Terdapat dua jenis distributor sistem operasi Android. Pertama yang dapat dukungan penuh dari Google atau Google Mail Service (GMS) dan kedua adalah yang benar-benar bebas distribusinya tanpa dukungan langsung dari Google atau dikenal sebagai Open Handset Distribution (OHD).

Berikut ini adalah sejarah perjalanan Android :

1. Kerjasama Android Inc. dengan Google Inc.
2. 2007 - 2008 Produk awal Android
3. Android versi 1.1
4. Android Versi 1.5 (Cupcake)
5. Android Versi 1.6 (Donut)
6. Android Versi 2.0/2.1 (Eclair)
7. Android Versi 2.2 (Froyo / Frozen Yoghurt)
8. Android Versi 2.3 (Gingerbread)
9. Android Versi 3.0 (Honeycomb)

Bab 2

Welcome To The “Android” Jungle!!!

Yang Anda butuhkan untuk memulai pemrograman Android adalah Android SDK (Software Development Kit) dan juga Java Development Kit. Setelah itu Anda membutuhkan Java IDE yang digunakan untuk menuliskan coding Android, disini saya menggunakan Eclipse Helios karena Eclipse sangat mensupport proses penulisan code Android dan menjadikan proses development Android menjadi lebih mudah.

Versi Android SDK, Java, dan Eclipse tersedia bagi berbagai macam sistem operasi seperti, Windows, Mac OS, dan Linux. Jadi Anda tinggal memilih Android SDK, Java, dan Eclipse tersebut sesuai dengan sistem operasi yang Anda gunakan.

Code Android ditulis menggunakan sintaks Java, dan inti dari libraries Android merupakan fitur kebanyakan yang terdapat pada inti Java APIs. Sebelum project yang kita buat berjalan, project tersebut pertama kali akan di terjemahkan kedalam Dalvik byte code.

Android SDK yang nanti akan kita Download didalamnya sudah terdapat libarari

Android, dokumentasi yang lengkap, dan juga beberapa contoh aplikasi. Didalam Android SDK tersebut juga terdapat tools untuk membantu Anda menulis dan me-debug aplikasi yang kita buat, seperti misalnya Emulator Android untuk menjalankan projek Android yang anda buat dan juga Dalvik Debug Monitoring Service (DDMS) untuk membantu Anda dalam memperbaiki aplikasi yang Anda buat jika terdapat kesalahan.

Memulai Android

1. Install Java SDK 1.6

Install java SDK (Software Development Kit) 1.6 pada komputer anda, jika anda tidak memiliki java SDK, maka anda dapat mendownloadnya di

<http://www.oracle.com/technetwork/java/javase/downloads/index.html>



2. Install Eclipse

Install eclipse pada komputer anda, sebaiknya gunakan eclipse 3.5 (Galileo) atau eclipse 3.6 (Helios). Anda dapat mendownload eclipse dimana saja karena eclipse ini bersifat open source, atau anda dapat juga mendownload langsung dari situs resmi eclipse

<http://www.eclipse.org/downloads/packages/eclipse-ide-java-developers/heliosr2>

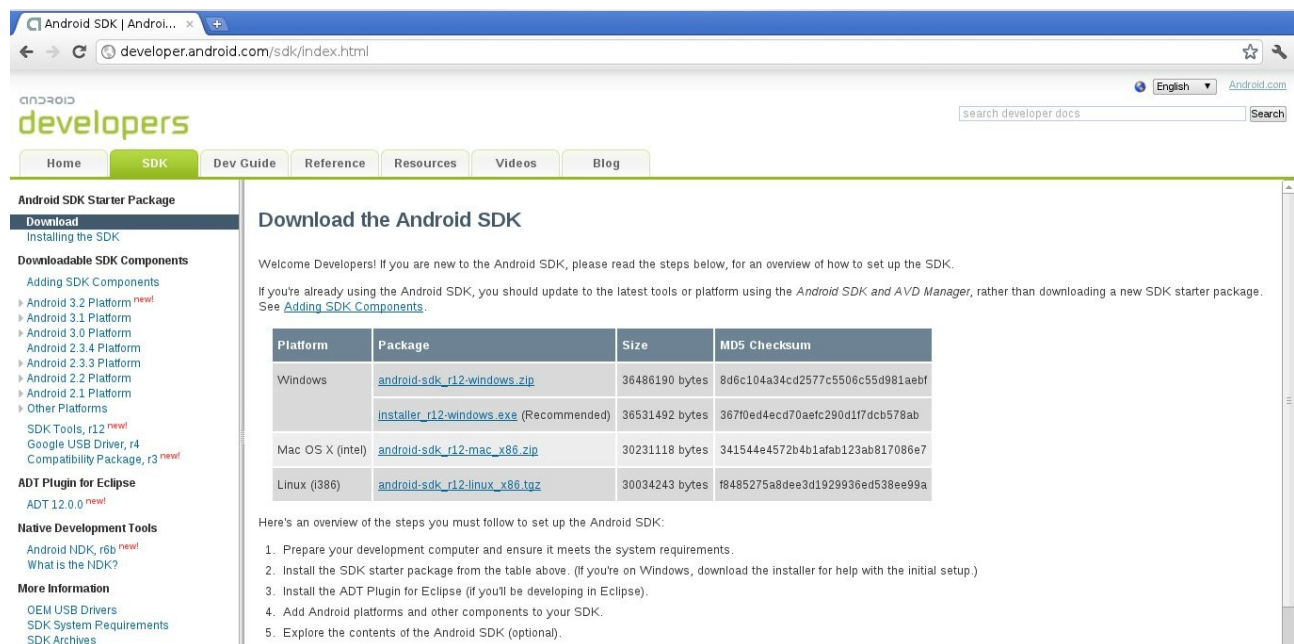


Setelah Eclipse berhasil didownload, selanjutnya ekstrak file tersebut, Anda bebas untuk memilih di folder mana Anda ingin mengekstraknya.

3. Install Android SDK

Install Android SDK pada komputer anda, jika anda belum memiliki android SDK, anda dapat mendownloadnya di

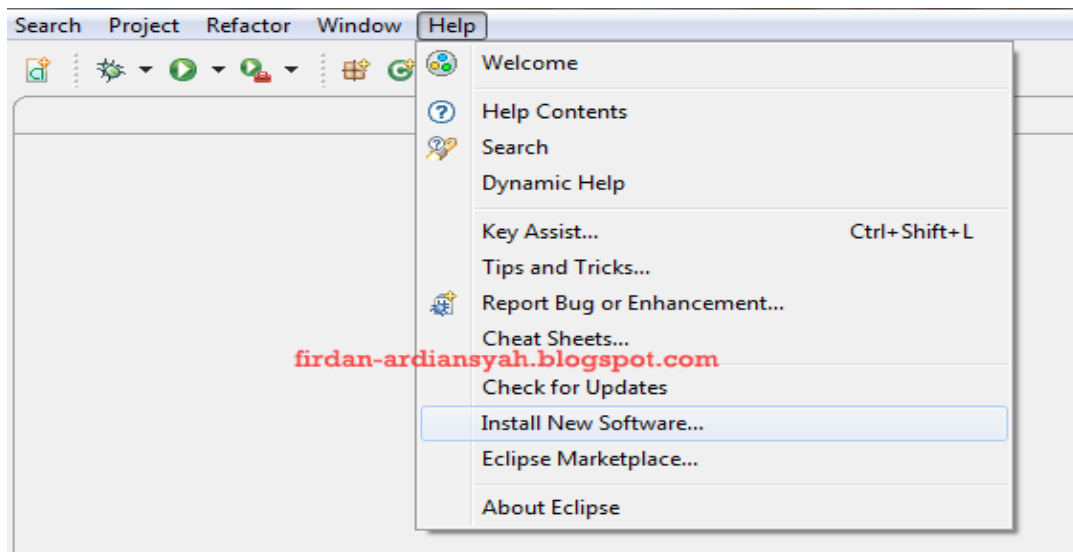
<http://developer.android.com/sdk/index.html>.



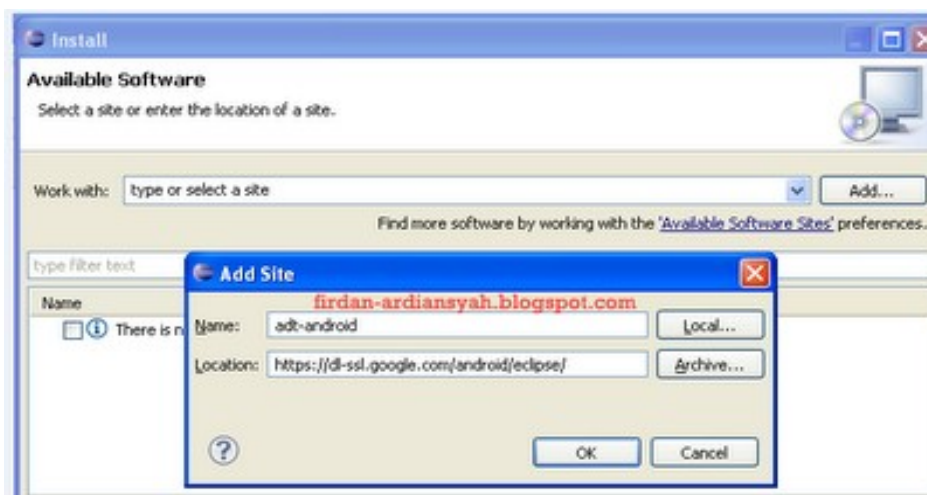
Setelah proses download selesai, ekstrak file Android SDK tersebut di folder mana saja yang anda sukai, atau agar lebih mudah ekstrak file Android SDK tersebut satu folder dengan folder dimana Anda mengekstrak file eclipse.

4. Install Eclipse ADT Plugin

Buka eclipse yang telah anda download, lalu pilih menu Help – Install New Software

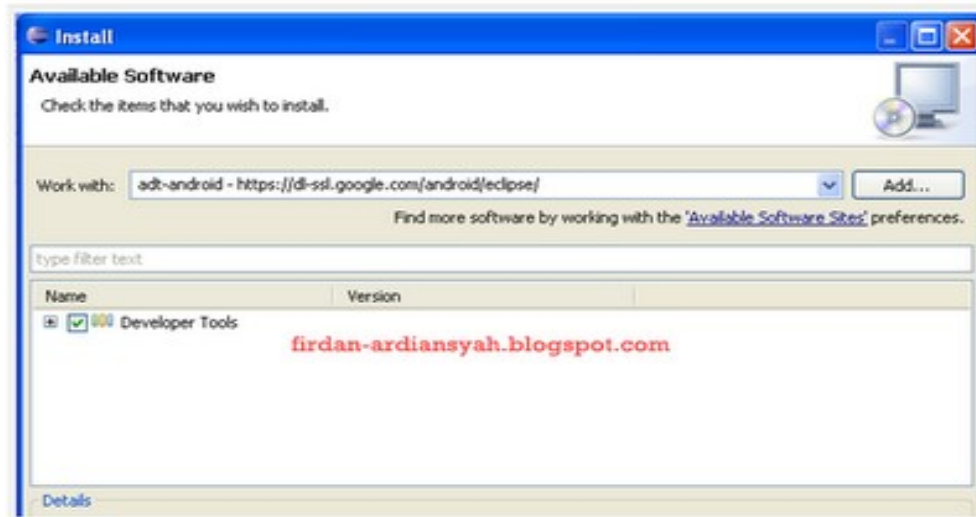


Klik button add, lalu pada bagian name ketika adt -android dan pada bagian location ketikkan <https://dl-ssl.google.com/android/eclipse/> seperti gambar dibawah ini



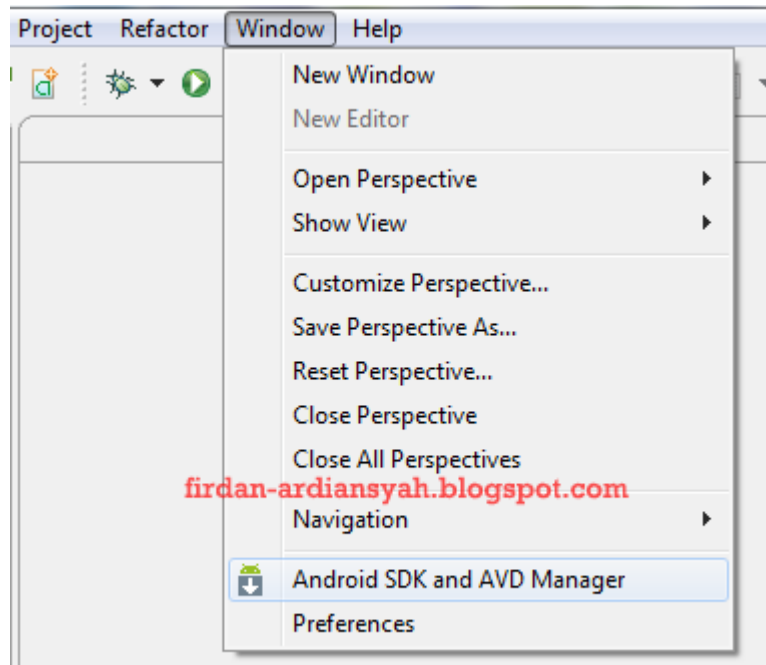
Setelah selesai, maka akan keluar tampilan "Developer Tools", jangan lupa centang

bagian Developer tools lalu klik next.



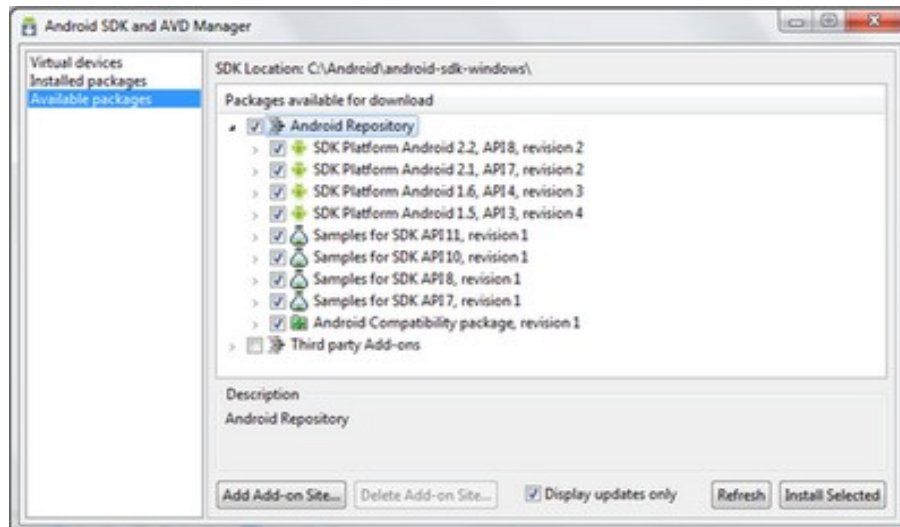
5. Download Library-Library

Selanjutnya download Library-library android dengan cara pilih menu Windows – Android SDK and AVD Manager.



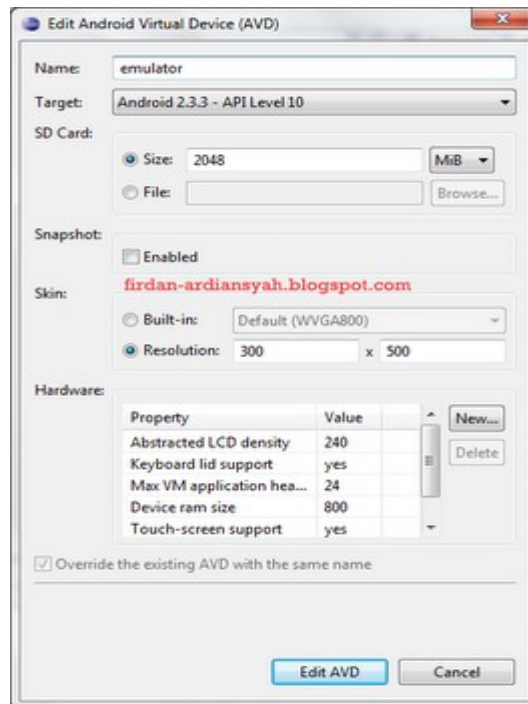
Lalu pilih semua package yang tersedia, klik install selected dan tunggu sampai proses download selesai. Proses download ini memang lama karena 1 file kira-kira besarnya

90mb, jadi saya saranin download aja yang SDK Platform Android 2.2, API 8, Revision 2. Kalo ada waktu luang, baru download versi platform yang lainnya.



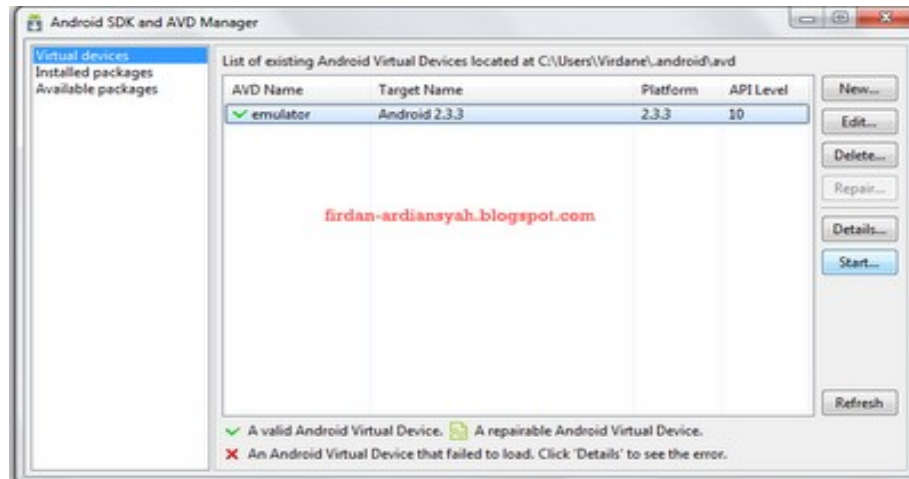
6. Membuat Emulator

Setelah proses download library android diatas selesai, maka langkah selanjutnya adalah membuat emulator android. Pilih menu windows – Android SDK and AVD Manager. Setelah keluar windows Android SDK and AVD manager pilih New, lalu isikan data seperti gambar dibawah ini.



Firdan Ardiansyah | Pengenalan Dasar Android Programming.

Setelah emulator android berhasil dibuat, maka langkah selanjutnya adalah mengetes emulator android tersebut apakah berjalan dengan baik atau tidak dengan cara, pilih menu windows - – Android SDK and AVD Manager. Pilih Virtual Devices. Lalu pilih emulator android yang kita buat tadi, lalu klik Star.



Tunggu beberapa saat sampai emulator android menampilkan gambar seperti di bawah ini.



Sampai tahap ini Anda telah berhasil mengkonfigurasi Eclipse IDE agar dapat digunakan untuk memulai development program Android.

Bab 3

Aplikasi Pertama

Hello World.

Kalo Anda sering baca-baca buku tentang programming, entah itu buku PHP, java, Visual Basic, C++ atau apa sajalah, pasti program pertama yang dibuat adalah program hello world. Gak tau kenapa harus hello world yang dibuat, mungkin itu sudah menjadi konspirasi bagi para penulis buku mengenai programming. Jadi biar gak kena sial karena gak ngikutin aturan main para penulis buku yang lebih senior, maka sekarang kita akan mencoba membuat aplikasi pertama kita, yaitu aplikasi hello world.

Membuat Aplikasi Hello World.

Untuk membuat sebuah projek Android baru, buka eclipse anda, lalu pilih file-new-other lalu pilih Android-Android Project tekan next. Selanjutnya akan keluar jendela New Android Project. Pada bagian Project Name isikan dengan hello world, build target pilih android 2.2. Selanjutnya pada bagian properties isikan seperti data dibawah ini :

Application Name : Hello World

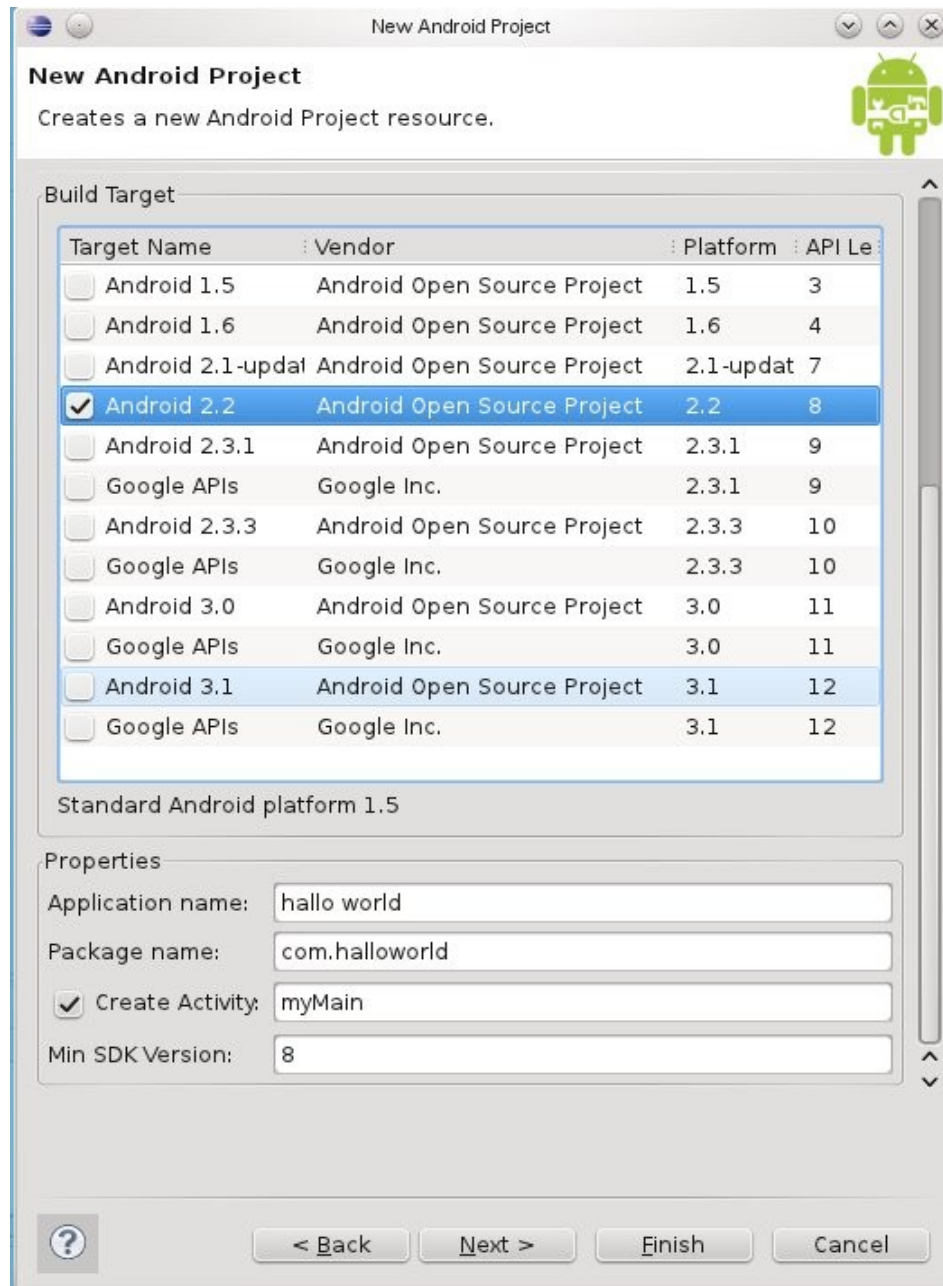
Package Name : com.helloworld

Create Activity : myMain

Min SDK Version : 8

Firdan Ardiansyah | Pengenalan Dasar Android Programming.

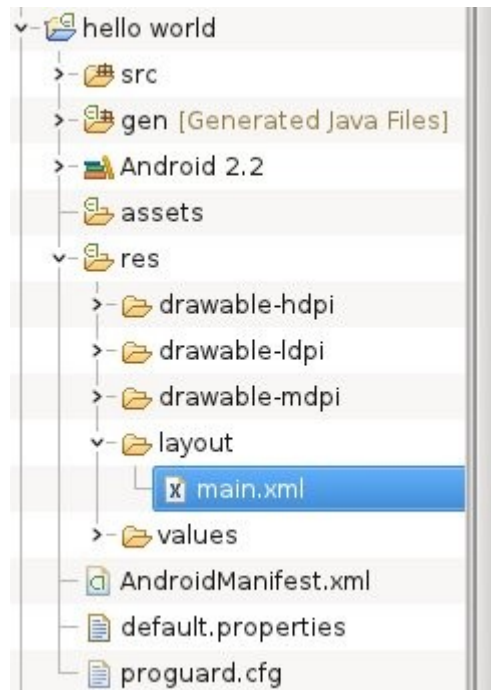
atau untuk lebih jelasnya silahkan lihat gambar dibawah ini :



Selanjutnya klik Finish.

Firdan Ardiansyah | Pengenalan Dasar Android Programming.

Setelah project hello world berhasil dibuat, selanjutnya buka file main.xml yang terdapat pada folder res-layout



Edit file main.xml tersebut menjadi seperti coding xml dibawah ini :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
<TextView
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:text="Hallo World!!!. ini program android pertama saya"
    />
</LinearLayout>
```

Selanjutnya coba anda jalankan program android hallo world tersebut melalui emulator yang sebelumnya telah anda buat dengan cara klik kanan project hello world, lalu pilih Run As – Android Application.

Firdan Ardiansyah | Pengenalan Dasar Android Programming.

Tunggu sampai emulator muncul, setelah muncul maka akan ditampilkan program Android Hello World yang tadi Anda buat seperti gambar dibawah ini :



Pada program diatas anda hanya membuat sebuah TextView yang berisikan teks “Hallo World!!!. Ini program android pertama saya” pada main.xml tanpa menambahkan variabel string apapun. main.xml tersebut merupakan file untuk mengatur user interface atau tampilan dari program yang anda buat pada Android.

Hallo World Dengan Gambar.

Sekarang kita akan mencoba merubah teks hello world yang sebelumnya ditampilkan dalam bentuk teks menjadi sebuah gambar, anda bisa membuat gambarnya terlebih dahulu dengan menggunakan gimp yang terdapat pada linux ataupun paint yang terdapat pada windows. Atau jika anda malas membuatnya, gunakan saja gambar

dibawah ini :



setelah anda mendapatkan atau membuat gambar hello world, selanjutnya copy file gambar tersebut ke folder res-drawable-hdpi, perlu diperhatikan juga untuk penulisan nama file gambarnya harus dengan huruf kecil semua dan tidak boleh ada spasi, contoh disini saya menggunakan nama imageshelloworld. Buka kembali file main.xml lalu lakukan perubahan seperti coding xml dibawah ini :

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

<ImageView
    android:layout_width = "wrap_content"
    android:layout_height= "wrap_content"
    android:layout_gravity = "center"
    android:src = "@drawable/imageshelloworld"
    />

</LinearLayout>
```

Jika pada file xml sebelumnya Anda menggunakan TextView untuk menampilkan teks hello world, maka disini untuk menampilkan gambar menggunakan xml ImageView.

Jalankan project hello world tersebut melalui emulator android, maka output tampilannya akan seperti gambar dibawah ini :



Pada bab 3 ini kita telah mempelajari mengenai menampilkan teks dan gambar melalui main.xml. Masih banyak lagi yang akan kita bahas pada bab-bab selanjutnya, jadi jangan lupa bikin kopi dan bakar rokoknya biar gak ngantuk :). Keep rockin dude.

Bab 4

Merancang Tampilan Program

Pada bab ke 4 ini Anda akan mempelajari mengenai bagaimana membuat sebuah user interface pada program Android.

Android Widget Toolbox

Android memiliki standard toolbox Views untuk membantu anda merancang sebuah tampilan sederhana.

Beberapa toolbox yang sering digunakan dalam merancang sebuah tampilan program di Android seperti berikut ini.

- **TextView** = TextView merupakan teks yang hanya bisa dibaca (read only) atau jika kita terbiasa membuat sebuah aplikasi desktop menggunakan Java atau VB, TextView dapat disamakan dengan Label.
- **EditText** = Digunakan untuk memasukan data, sehingga user dapat menuliskan suatu nilai kedalam EditText tersebut.
- **ListView** = Digunakan untuk menampilkan item didalam list.
- **Spinner** = Widget yang menampilkan TextView dan berhubungan dengan ListView, sehingga memungkinkan user untuk memilih salah satu menu yang terdapat didalam Spinner tersebut.
- **Button** = Tombol yang biasanya digunakan untuk eksekusi perintah.

- **CheckBox** = Tombol yang merepresentasikan apakah terpilih (checked) atau tidak (unchecked).
- **RadioButton** = Tombol-tombol yang ditampilkan secara group, biasanya digunakan untuk user memilih salah satu dari tombol yang ada pada RadioButton tersebut.

Nanti kita akan mencoba membuat masing-masing dari widget tersebut, namun sebenarnya itu hanya beberapa widget yang dimiliki oleh Android. Untuk daftar lengkapnya mengenai widget, Anda bisa mencarinya di situs ini :

<http://code.google.com/android/reference/view-gallery.html>.

Layouts

Layout Managers (Biasa disebut dengan Layouts) digunakan untuk mengontrol posisi utama dari layar. Layouts dapat di lekatkan (embed) dengan layout lainnya, jadi dengan kata lain Anda dapat memasukan lebih dari satu buah layout pada tampilan yang nantinya akan dibuat.

Android SDK juga menyediakan beberapa layouts sederhana untuk membantu anda membangun tampilan program. Jadi Anda tinggal memilih ingin menggunakan layout yang mana saja untuk membuat tampilan program yang nantinya akan dibuat menjadi mudah dimengerti dan digunakan.

Beberapa layouts yang terdapat pada Android :

- **FrameLayout** = Layouts yang paling sederhana, FrameLayout menampilkan setiap view di kiri atas.
- **LinearLayout** = LinearLayout menambahkan setiap child View secara datar, artinya LinearLayout hanya menampilkan satu buah child View perbaris.
- **RelativeLayout** = Dengan menggunakan RelativeLayout, Anda mendefinisikan posisi dari masing-masing child View menjadi relatif.
- **TableLayout** = TableLayout memungkinkan layout ditampilkan secara garis dan kolom atau mirip tabel (ya namanya juga tablelayout).
- **AbsoluteLayout** = Setiap child View di definisikan dalam kordinat.

Nanti Anda akan mempraktikan secara langsung salah satu layouts tersebut, atau untuk lebih

memahami definisi dari masing-masing layout tersebut Anda dapat berkunjung ke situs :

<http://code.google.com/android/devel/ui/layout.html>

Implementasi widget

Kita akan mencoba membuat masing-masing widget yang telah dijelaskan sebelumnya. Kali ini kita akan mencoba mengimplementasikan widget TextView, EditText dan sebuah Button.

Langsung saja buka eclipse, buat sebuah project Android baru lalu isikan datanya seperti dibawah ini :

Project Name = implementasiwidget1

Build Target = Android 2.2

Application Name = Implementasi widget 1

Package Name = com.implement1

Create Activity = myMain

Min SDK Version = 8

Buka file main.xml, kita akan memasukan satu buah TextView, 1 buah EditText dan juga 1 buah Button. Masukan coding xml seperti dibawah ini :

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

<LinearLayout
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >

<TextView
    android:layout_width = "wrap_content"
    android:layout_height= "wrap_content"
    android:layout_gravity = "center"
    android:text = "Masukan Nama Anda"
/>
```

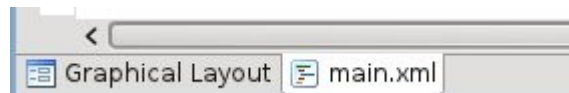
```
<EditText
android:id = "@+id/nama"
android:layout_width = "150dp"
android:layout_height="wrap_content"
android:layout_gravity = "center"
/>

<Button
android:id = "@+id/ok"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:text = "OK"
android:onClick = "hasil"
/>

<TextView
android:id = "@+id/output"
android:layout_width = "wrap_content"
android:layout_height = "wrap_content"
android:layout_gravity = "center"
android:text = ""
/>

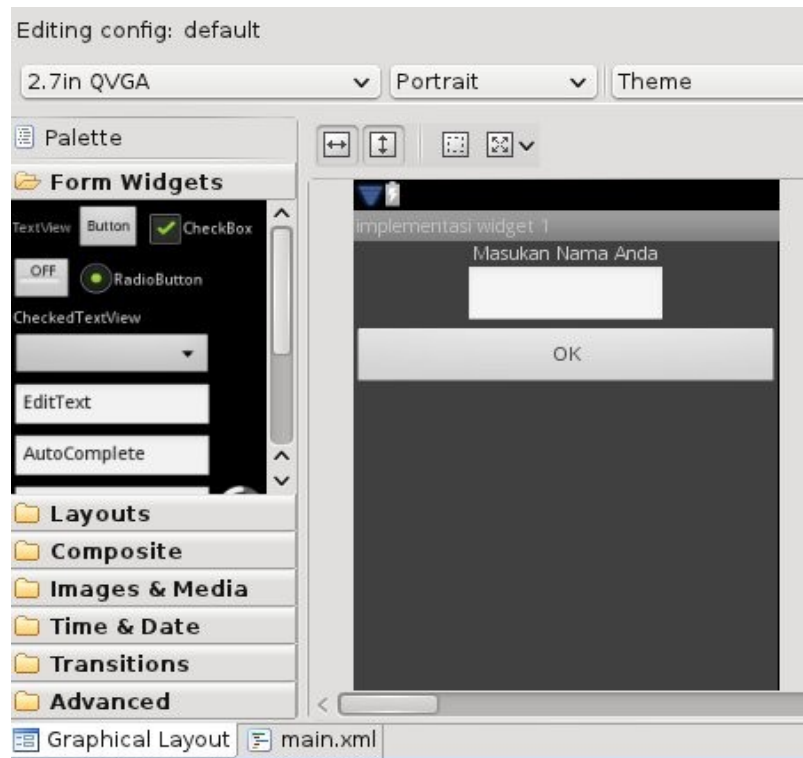
</LinearLayout>
</RelativeLayout>
```

Pada masing-masing EditText dan juga Button, telah disertakan sebuah Id (android:id), Id disini digunakan sebagai media yang nantinya digunakan untuk membuat sebuah variabel pada masing-masing widget tersebut, sekarang coba Anda lihat hasil dari pembuatan halaman tersebut dengan cara klik graphical layout yang ada pada kanan bawah main.xml.



Firdan Ardiansyah | Pengenalan Dasar Android Programming.

Maka akan didapat output tampilan seperti dibawah ini :



Ok selanjutnya kita beralih ke java, biasanya disininya ini yang bikin kepala sedikit agak mumet, tapi slow lah, kalo gak pusing bukan belajar namanya :). Langsung aja dibuka file myMain.java yang ada di folder src-com.implement1-myMain.java



Firdan Ardiansyah | Pengenalan Dasar Android Programming.

Selanjutnya bila file myMain.java telah terbuka isikan dengan coding java seperti dibawah ini :

Pertama kita import terlebih dahulu file-file yang dibutuhkan dalam pengembangan aplikasi ini :

```
package com.implement1;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
```

Selanjutnya deklarasikan class utama yang meng extends Activity, juga deklarasikan variabel untuk masing-masing widget yang sebelumnya telah dibuat pada main.xml.

```
public class myMain extends Activity {
    /** Called when the activity is first created. */

    EditText nama;
    Button ok;
    TextView output;
```

Lalu Anda harus meng Override public void onCreate, maksudnya adalah semua variabel ataupun inner class yang terdapat didalam kelas onCreate akan dieksekusi pada waktu pertama kali program dijalankan. Lalu mensetting layout yang akan ditampilkan, dalam hal ini layout main.xml pada bagian setContentView. Dan juga pendeklarasian dari masing-masing variabel ok dan output.

```
@Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        ok = (Button) findViewById (R.id.ok);
        output = (TextView) findViewById (R.id.output);
    }
```

Selanjutnya buat sebuah class baru dengan nama hasil yang memiliki parameter View, class ini adalah class yang akan tereksekusi apabila Button ok yang sebelumnya telah dibuat di klik oleh user.


```
public void hasil (View v){
    nama = (EditText) findViewById (R.id.nama);
    nama.getText().toString();
    String name = String.valueOf(nama.getText().toString());
    output.setText("Nama Anda Adalah "+name);
}
```

Listing lengkap dari class myMain.java :

```
package com.implement1;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class myMain extends Activity {
    /** Called when the activity is first created. */

    EditText nama;
    Button ok;
    TextView output;

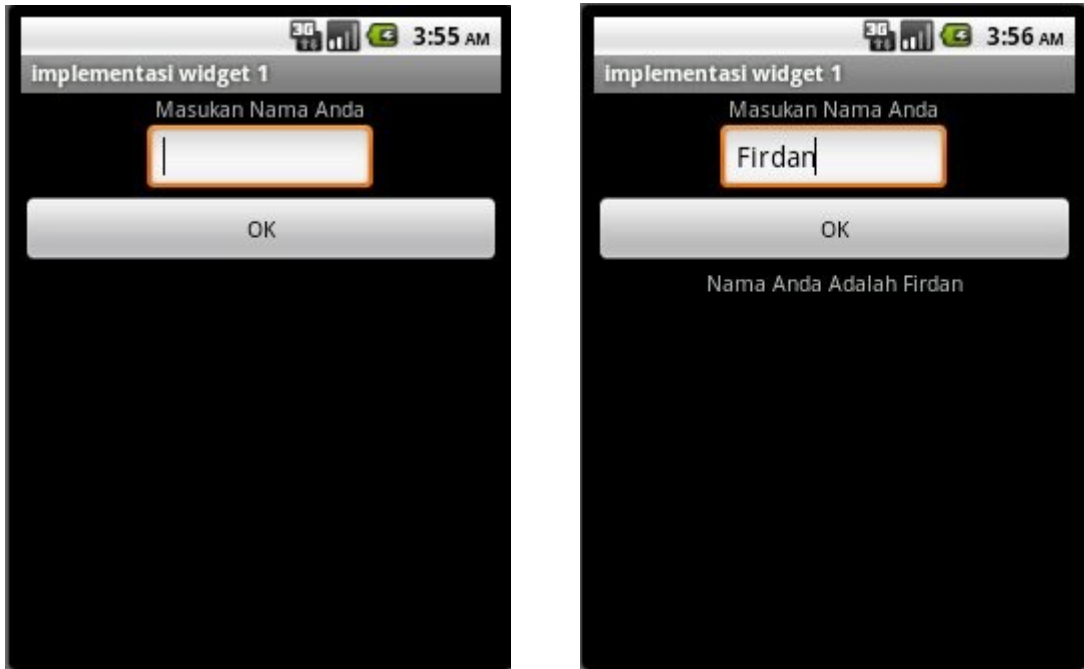
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        ok = (Button) findViewById (R.id.ok);
        output = (TextView) findViewById (R.id.output);
    }

    public void hasil (View v){
        nama = (EditText) findViewById (R.id.nama);
        nama.getText().toString();
        String name = String.valueOf(nama.getText().toString());
        output.setText("Nama Anda Adalah "+name);
    }
}
```

Firdan Ardiansyah | Pengenalan Dasar Android Programming.

Langkah terakhir tinggal kita lihat hasilnya melalui emulator seperti gambar dibawah ini :



Bagaimana? Apa Anda puas dengan project Android yang telah anda buat? Sebaiknya jangan puas dulu, karena masih banyak yang akan kita bahas. Kalo kopinya abis, seduh lagi kopinya, kalo rokoknya abis, beli lagi di warung ya :).

Implementas Widget 2.

Sekarang kita akan mempelajari bagaimana cara membuat widget RadioButton dan juga CheckBox yang definisinya sudah dijelaskan diatas beserta beberapa properties pendukung seperti ScrollView (nanti Anda akan tahu sendiri buat apa ScrollView itu, makanya baca terus ok).

Kali ini kita akan mencoba membuat sebuah aplikasi mengenai penghitungan gaji berdasarkan golongan dan juga status, jika golongan 1 maka gaji yang didapat adalah Rp. 1.000.000, jika golongan 2 gaji yang didapat adalah Rp 2.000.000 dan jika sudah menikah maka mendapatkan bonus sebesar Rp. 500.000.

Langsung aja buat sebuah projek Android baru, lalu isikan datanya seperti dibawah ini :

Project Name = Perhitungan Gaji
Build Target = Android 2.2
Application Name = Perhitungan Gaji
Package Name = com.perhitungangaji
Create Activity = myMain
Min SDK Version = 8
Lalu tekan Finish.

Ok langkah pertama kita buka dulu file main.xml yang biasa digunakan untuk mengatur user interface dari program yang akan kita buat, lalu masukan coding xmlnya seperti coding dibawah ini :

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

    <ScrollView
    android:layout_width = "fill_parent"
    android:layout_height= "wrap_content"
```

```
>
<LinearLayout
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
>

<TextView
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:text = "Masukan Nama Anda : "
/>

<EditText
android:id = "@+id/nama"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
/>

<TextView
android:layout_marginTop = "15dp"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:text = "Jenis Kelamin"
/>

<CheckBox
android:id = "@+id/menikah"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:text = "Menikah"
/>

<CheckBox
android:id = "@+id/lajang"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:text = "lajang"
/>

<TextView
android:layout_marginTop = "15dp"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:text = "Golongan"
/>

<RadioGroup
android:id = "@+id/rgGolongan"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content">
```

```
<RadioButton
android:id = "@+id/rbGolongan1"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:text = "Golongan 1"
/>

<RadioButton
android:id = "@+id/rbGolongan2"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:text = "Golongan 2"
/>
</RadioGroup>

<Button
android:layout_marginTop = "15dp"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:text = "Hitung Gaji"
android:onClick = "hitung"
/>

<TextView
android:id = "@+id/outputNama"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:gravity = "center"
android:textSize = "20dp"
android:text = ""
/>

<TextView
android:id = "@+id/outputGaji"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:gravity = "center"
android:textSize = "35dp"
android:text = ""
/>

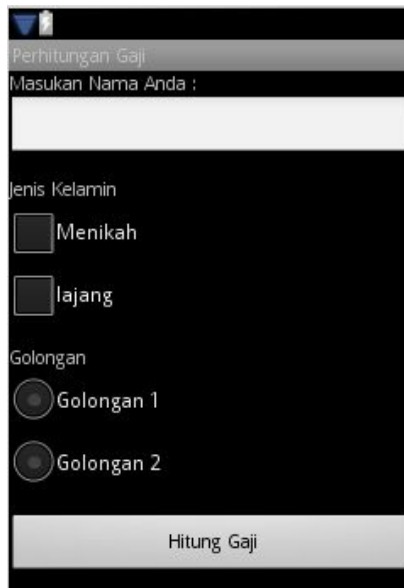
</LinearLayout>
</ScrollView>
</LinearLayout>
```

Mungkin Anda melihat sebuah properties baru diatas, yaitu `<ScrollView`, dengan menggunakan `ScrollView` memungkinkan kita untuk menggeser menu secara vertical jika layar handphone yang digunakan tidak cukup untuk menampilkan menu secara keseluruhan.

Firdan Ardiansyah | Pengenalan Dasar Android Programming.

Pada bagian TextView juga terdapat sebuah properties baru yang saya masukan, yaitu `android:layout_marginTop = "15dp"`, maksud dari properties itu adalah membuat jarak atas dari TextView tersebut sehingga terdapat jarak dengan widget yang ada di atasnya. Anda juga dapat membuat jarak kanan, kiri dan bawah dengan menggunakan properties `android:layout_marginRight`, `android:layout_marginLeft`, `android:layout_marginBottom`.

Coba kita lihat tampilan tampilan programnya pada panel Graphical Layout, maka tampilannya akan seperti pada gambar dibawah ini :



Sekarang kita akan mengotak atik bagian kelas dari program penghitungan gaji tersebut, langsung saja buka file myMain.java lalu ketikkan koding javanya seperti dibawah ini :

Terlebih dahulu import file-file yang dibutuhkan dalam pengembangan program :

```
package com.perhitungangaji;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
```

```
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;
```

Lalu buat sebuah class public dan masukan juga variabel variabel sesuai dengan file main.xml yang dibuat :

```
public class myMain extends Activity {
    /** Called when the activity is first created. */

    int gajiStatus, gajiGolongan;
    RadioButton goll, gol2;
    RadioGroup golongan;
    Button hitung;
    CheckBox status;
```

Override sebuah class onCreate, class ini merupakan class pertama yang akan dieksekusi pada saat program pertama kali dijalankan.

```
@Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

    }
```

Sekarang buat sebuah class baru dengan nama hitung, kelas ini merupakan class yang akan dieksekusi pada saat user mengklik tombol Hitung Gaji, nama class nya pun didasarkan pada nama properties dari widget button Hitung Gaji yaitu `android:onClick = "hitung"` seperti dibawah ini :

```
public void hitung (View v){
```

Lalu terakhir masukan variabel dan juga proses perhitungan untuk melakukan perhitungan Total Gaji seperti di bawah ini :

```
EditText nama = (EditText) findViewById (R.id.nama);
TextView outputNama = (TextView) findViewById (R.id.outputNama);
String name = nama.getText().toString();
outputNama.setText("Total Gaji "+name);
```

```
        status = (CheckBox) findViewById (R.id.menikah);
        if (status.isChecked()){
            gajiStatus = 500000;
        };

        golongan = (RadioGroup) findViewById (R.id.rgGolongan);
        int gol = golongan.getCheckedRadioButtonId();
        if (gol == R.id.rbGolongan1){
            gajiGolongan = 1000000;
        }else
        if (gol == R.id.rbGolongan2){
            gajiGolongan = 2000000;
        };

        int totalGaji = gajiStatus+gajiGolongan;
        TextView total = (TextView) findViewById (R.id.outputGaji);
        total.setText(String.valueOf(totalGaji));
    }
};
```

Listing lengkap Class myMain.java :

```
package com.perhitungangaji;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.CheckBox;
import android.widget.EditText;
import android.widget.RadioButton;
import android.widget.RadioGroup;
import android.widget.TextView;

public class myMain extends Activity {
    /** Called when the activity is first created. */

    int gajiStatus, gajiGolongan;
    RadioButton gol1, gol2;
    RadioGroup golongan;
    Button hitung;
    CheckBox status;

    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```



```
public void hitung (View v){  
    EditText nama = (EditText) findViewById (R.id.nama);  
    TextView outputNama = (TextView) findViewById (R.id.outputNama);  
    String name = nama.getText().toString();  
    outputNama.setText("Total Gaji "+name);  
  
    status = (CheckBox) findViewById (R.id.menikah);  
    if (status.isChecked()){  
        gajiStatus = 500000;  
    };  
  
    golongan = (RadioGroup) findViewById (R.id.rgGolongan);  
    int gol = golongan.getCheckedRadioButtonId();  
    if (gol == R.id.rbGolongan1){  
        gajiGolongan = 1000000;  
    }else  
    if (gol == R.id.rbGolongan2){  
        gajiGolongan = 2000000;  
    };  
  
    int totalGaji = gajiStatus+gajiGolongan;  
    TextView total = (TextView) findViewById (R.id.outputGaji);  
    total.setText(String.valueOf(totalGaji));  
}  
};
```

Sekarang coba Anda jalankan aplikasi Penghitungan Gaji tersebut melalui Emulator, maka akan tampak seperti gambar dibawah ini :



Selanjutnya Masukan Nama, status dan juga golongan seperti dibawah ini :



Terus tekan tombol hitung gaji, lalu scroll kebawah untuk melihat total gaji yang akan ditampilkan (properties ScrollView yang sebelumnya dibuatlah yang bikin tampilan aplikasi ini bisa di Scroll secara vertical) :



Bagaimana? Mudah bukan membuat aplikasi perhitungan gaji dengan menggunakan widget RadioGroup dan juga CheckBox? Praktekin ya, jangan cuma dibaca doank :).

Mempercantik Tampilan Background.

Mungkin Anda bosan melihat warna background dari program yang dibuat selalu berwarna hitam, sekarang kita akan merubah warna hitam tersebut menjadi warna apa saja terserah Anda.

Untuk mempermudah dan mempercepat pembahasan, buka kembali program Perhitungan Gaji yang sebelumnya sudah dibuat, lalu buka file main.xml nya lalu lakukan penambahan properties pada bagian Layout pertama, seperti dibawah ini :

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:background = "#070898"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    >
```

Diatas telah ditambahkan sebuah properties baru, yaitu `android:background = "#070898"` dan maksud dari `"#070898"` adalah merupakan kode warna html yang dapat digunakan untuk merubah warna background pada Android. Untuk kode-kode warna html anda bisa tanya ke mbah google ya. Sekarang kita lihat hasilnya dibawah ini :



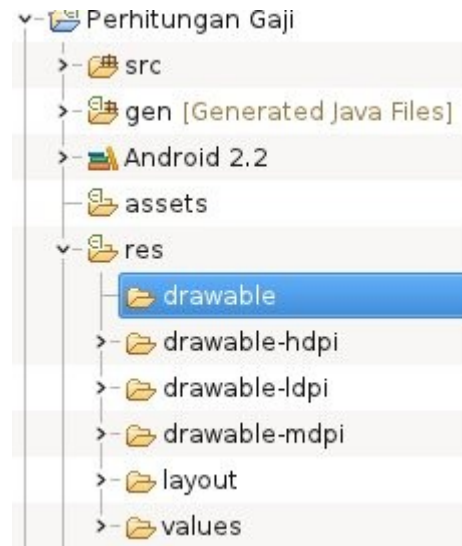
karena kode warna `"#070898"` merupakan kode warna html untuk warna biru, maka background program kita sekarang sudah berubah menjadi warna biru. Mudah bukan untuk merubah warna background.

Manipulasi Class Untuk Mempertahankan Background.

Sebelumnya kita hanya merubah sebuah warna background program dari hitam menjadi biru, sekarang kita akan membahas bagaimana membuat sebuah background namun memiliki ujung yang tumpul, yang nantinya background ini akan kita pakai pada layout kedua dari program Perhitungan Gaji.

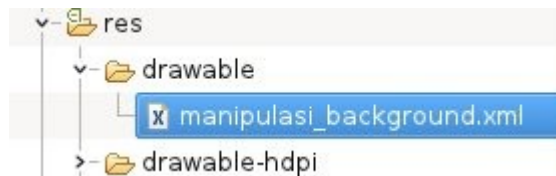
Buka kembali program Perhitungan Gaji, lalu pada folder res, buat sebuah folder baru dengan cara klik kanan folder res-new-folder. Beri nama folder baru tersebut dengan nama drawable. Maka kita akan mempunyai sebuah folder baru

dengan nama drawable seperti gambar dibawah ini :



Setelah folder dibuat, selanjutnya adalah membuat sebuah file xml baru yang nantinya digunakan untuk mempercantik tampilan background program. klik kanan folder drawable tersebut lalu pilih new-other, lalu akan terbuka sebuah jendela baru, pilih folder Android-Android XML File lalu klik next. Pada file, isikan dengan nama manipulasi_background (tidak boleh ada spasi), lalu pada klik layout pada type resourcenya lalu klik finish.

Biasanya file xml yang baru dibuat tersebut tidak langsung berada di dalam folder drawable, biasanya file tersebut ada didalam folder layout, jadi Anda tinggal pindahkan saja file xml tersebut ke dalam folder drawable dengan cara klik kanan file manipulasi_background.xml pilih copy, lalu paste kedalam folder drawable, file manipulasi_background yang ada didalam folder layout Anda hapus saja, maka sekarang file manipulasi_background tersebut sudah ada didalam folder drawable.



Sekarang buka file `manipulasi_background` tersebut lalu isikan dengan coding seperti dibawah ini :

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

  <item>
    <shape>
      <solid
        android:color="#51acf9"
      />
      <stroke
        android:width="2dip"
        android:color="#51acf9"
      />
      <corners
        android:radius="10dip"
      />
      <padding
        android:left="15dip"
        android:right="10dip"
        android:top="10dip"
        android:bottom="10dip"
      />
    </shape>
  </item>
</selector>
```

Keterangan :

```
<solid
  android:color="#51acf9"
/>
```

solid digunakan untuk warna dalam layout.

```
<stroke
  android:width="2dip"
  android:color="#51acf9"
/>
```

stroke digunakan untuk membuat tepian pada layout, `android:width` merupakan lebar tepi dan `android:color` untuk warna tepiannya, disini saya samakan saja warnanya dengan warna intinya.

```
<padding
  android:left="15dip"
  android:right="10dip"
  android:top="10dip"
  android:bottom="10dip"
/>
```

padding digunakan untuk membatasi jarak atas, bawah, kiri dan kanan dengan layout pertama.

Setelah selesai, selanjutnya buka file main.xml pada program Perhitungan Gaji, lalu lakukan perubahan koding pada LinearLayout kedua, ingat ok LinearLayout yang kedua bukan yang pertama, karena yang pertama warna backgroundnya sudah kita rubah menjadi warna biru tua. Untuk lebih jelasnya lihat coding xml pada LinearLayout kedua dibawah ini :

```
<LinearLayout
    android:orientation="vertical"
    android:background = "@drawable/manipulasi_background"
    android:layout_marginLeft = "5dp"
    android:layout_marginRight= "5dp"
    android:layout_marginTop = "5dp"
    android:layout_marginBottom = "5dp"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
>
```

Lakukan juga perubahan warna teks menjadi hitam pada masing-masing properties TextView agar tidak bentrok dengan warna yang digunakan pada background kedua dengan perintah `android:textColor = "#000000"`

Untuk lebih jelasnya, kita lihat saja listing kode main.xml yang baru saja kita manipulasi :

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:background = "#070898"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
>

<ScrollView
    android:layout_width = "fill_parent"
    android:layout_height= "wrap_content"
```

```
>
<LinearLayout
    android:orientation="vertical"
    android:background = "@drawable/manipulasi_background"
    android:layout_marginLeft = "5dp"
    android:layout_marginRight=" 5dp"
    android:layout_marginTop = "5dp"
    android:layout_marginBottom = "5dp"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
>

<TextView
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:text = "Masukan Nama Anda : "
android:textColor = "#000000"
/>

<EditText
android:id = "@+id/nama"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
/>

<TextView
android:layout_marginTop = "15dp"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:textColor = "#000000"
android:text = "Status"
/>

<CheckBox
android:id = "@+id/menikah"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:textColor = "#000000"
android:text = "Menikah"
/>

<TextView
android:layout_marginTop = "15dp"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:textColor = "#000000"
android:text = "Golongan"
/>

<RadioGroup
android:id = "@+id/rgGolongan"
```



```
android:layout_width = "fill_parent"
android:layout_height= "wrap_content">

<RadioButton
android:id = "@+id/rbGolongan1"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:textColor = "#000000"
android:text = "Golongan 1"
/>

<RadioButton
android:id = "@+id/rbGolongan2"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:textColor = "#000000"
android:text = "Golongan 2"
/>
</RadioGroup>

<Button
android:id = "@+id/hitung"
android:layout_marginTop = "15dp"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:text = "Hitung Gaji"
android:onClick = "hitung"
/>

<TextView
android:id = "@+id/outputNama"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:gravity = "center"
android:textSize = "20dp"
android:textColor = "#000000"
android:text = ""
/>

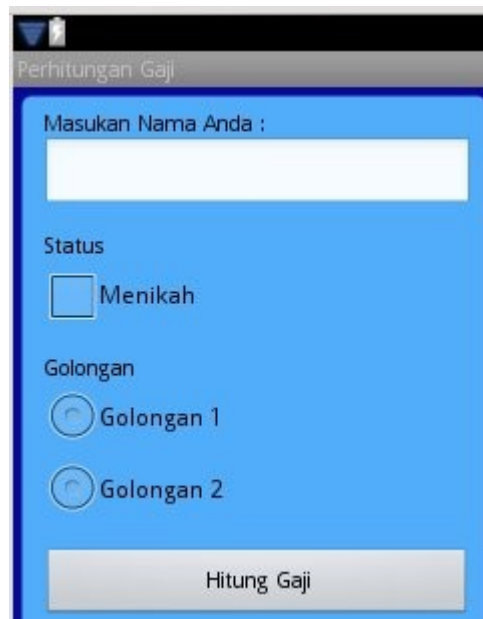
<TextView
android:id = "@+id/outputStatus"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:gravity = "center"
android:textSize = "20dp"
android:textColor = "#000000"
android:text = ""
/>

<TextView
android:id = "@+id/outputGaji"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:gravity = "center"
```

Firdan Ardiansyah | Pengenalan Dasar Android Programming.

```
android:textSize = "35dp"  
android:textColor = "#000000"  
android:text = ""  
</>  
</LinearLayout>  
</ScrollView>  
</LinearLayout>
```

Sekarang kita lihat hasilnya pada panel Graphical Layout



Sudah agak enak dilihat bukan dibandingkan dengan yang sebelumnya :).
sekarang kita jaja jalankan Aplikasi tersebut di emulator.



Lebih enak dilihat bukan :). sekarang aplikasi kita sudah memiliki 2 buah warna, yaitu biru tua untuk background pertama dan juga biru muda untuk background kedua.

Manipulasi Button.

Sekarang kita akan membahas mengenai manipulasi Button, jadi kita akan merubah Button standar yang disediakan oleh Android menjadi button kreasi kita sendiri. Untuk memanipulasi Button, kita membutuhkan dua buah gambar, yaitu gambar sebelum tombol ditekan dan gambar pada saat tombol ditekan, jadi biar ada transisi warna saat tombol dieksekusi.

Misalnya saja saya menggunakan tombol seperti dibawah ini :



Buka kembali file Perhitungan gaji yang telah dibuat lalu copy kedua gambar dengan nama ok1 dan ok2 tersebut kedalam folder drawable yang sebelumnya telah dibuat. Lalu buat sebuah file xml baru dengan nama tombol_ok.xml dan copy file xml tersebut kedalam folder drawable. Contohnya seperti gambar dibawah ini :



Sekarang kita lakukan proses penulisan coding xml pada file tombol_ok.xml. Buka filenya lalu tuliskan seperti coding dibawah ini :

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

  <item
    android:state_focused = "true"
    android:state_pressed = "false"
    android:drawable = "@drawable/ok2"/>

  <item
    android:state_focused = "false"
    android:state_pressed = "false"
    android:drawable="@drawable/ok1" />

  <item
    android:drawable = "@drawable/ok2"/>

</selector>
```

Setelah itu, buka kembali file main.xml karena kita akan melakukan perubahan properties yang dimiliki oleh button hitung menjadi seperti dibawah ini :

```
<Button
  android:id = "@+id/hitung"
  android:layout_gravity = "center"
  android:layout_marginTop = "15dp"
  android:layout_width = "wrap_content"
  android:layout_height = "wrap_content"
  android:background = "@drawable/tombol_ok"
  android:onClick = "hitung"
/>
```

Sekarang coba kita jalankan programnya melalui emulator.

Tombol Perhitungan yang sebelumnya tampilannya standar yang diberikan oleh Android sekarang sudah berhasil kita ubah menjadi gambar, sebelum diklik warna tombolnya hitam, pada saat diklik akan berubah menjadi merah dan setelah diklik akan kembali menjadi hitam.



Sebelum diklik

Saat diklik

Setelah diklik

Ok mungkin sampai disini saja pembahasan kita di bab 4 ini mengenai user interface, selanjutnya pada bab 5 kita akan membuat sebuah aplikasi-aplikasi Android yang lebih menarik lagi.

Oia, saya yakin rokok dan kopinya sudah abis lagi ya :). Beli dan bikin lagi ok, hehehehe...

Bab 5

Rage Against Android

Application

Pada bab 5 ini Anda akan mempelajari cara membuat beberapa aplikasi Android, seperti membuat aplikasi penghitung luas bangun datar dan juga aplikasi untuk menelpon perusahaan makanan cepat saji seperti pizza hut atau KFC biar gak capek-capek ngetik nomer teleponnya.

Membuat Aplikasi Pemesanan Fast Food.

Aplikasi ini sederhana banget, kenapa sederhana? Ya emang sederhana, karena kita hanya akan membuat beberapa button yang masing-masing button tersebut akan menuju web dan juga dapat langsung menelpon nomor telepon dari beberapa perusahaan makanan cepat saji tersebut.

Daripada ngomongin makanannya, mendingan langsung aja kita praktekin ok, buat sebuah project Android baru lalu isikan datanya seperti dibawah ini :

Project Name = Fast Food

Build Target = Android 2.2

Package Name = com.FastFood

Application Name = Fast Food

Create Activity = myMain

Min SDK Version = 8

Selanjutnya buka file main.xml lalu isikan coding xml nya seperti dibawah ini :

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

<Button
    android:id = "@+id/skfc"
    android:layout_width = "fill_parent"
    android:layout_height= "wrap_content"
    android:text = "Situs KFC"
/>

<Button
    android:id = "@+id/pkfc"
    android:layout_width = "fill_parent"
    android:layout_height= "wrap_content"
    android:text = "Pesan KFC"
/>

<Button
    android:id = "@+id/smcd"
    android:layout_width = "fill_parent"
    android:layout_height= "wrap_content"
    android:text = "Situs McDonałd's"
/>

<Button
    android:id = "@+id/pmcd"
    android:layout_width = "fill_parent"
    android:layout_height= "wrap_content"
    android:text = "Pesan McDonałds"
/>

</LinearLayout>
```

Tampilannya seperti pada gambar dibawah ini :



Selanjutnya buka file myMain.java. Lalu isikan seperti coding dibawah ini :

Import semua file yang dibutuhkan dalam pengembangan aplikasi ini :

```
package com.FastFood;

import android.app.Activity;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;
```

Pemberian method pada masing-masing button dan juga pendefinisian masing-masing variabel.

```
Button webkfc = (Button) findViewById (R.id.skfc);
Button phonekfc = (Button) findViewById (R.id.pmcd);
Button webmcd = (Button) findViewById (R.id.smcd);
Button phonemcd = (Button) findViewById (R.id.pmcd);

webkfc.setOnClickListener(new Button.OnClickListener(){
```



```
        public void onClick(View v){
            callintent(v);
        }
    });

    phonekfc.setOnClickListener(new Button.OnClickListener(){
        public void onClick (View v){
            callintent(v);
        }
    });

    webmcd.setOnClickListener(new Button.OnClickListener(){
        public void onClick (View v){
            callintent(v);
        }
    });

    phonemcd.setOnClickListener(new Button.OnClickListener(){
        public void onClick (View v){
            callintent(v);
        }
    });
});
```

Buat kelas baru dengan nama callintent dimana kelas ini yang nantinya akan dipanggil oleh setiap button untuk melakukan eksekusinya. Dikelas ini juga terdapat kondisional switch case agar method untuk mengatur variabel apa yang dieksekusi.

```
public void callintent(View view) {
    Intent intent = null;
    switch (view.getId()) {
        case R.id.pkfc:
            intent = new Intent(Intent.ACTION_CALL, Uri.parse("tel:14022"));
            startActivity(intent);
            break;
        case R.id.pgcd:
            intent = new Intent(Intent.ACTION_CALL, Uri.parse("tel:14045"));
            startActivity(intent);
            break;
        case R.id.skfc:
            intent = new Intent(Intent.ACTION_VIEW,
                Uri.parse("http://kfcgaul.com"));
            startActivity(intent);
            break;
        case R.id.smcd:
            intent = new Intent(Intent.ACTION_VIEW,
                Uri.parse("http://www.mcdelivery24hrs.com"));
            startActivity(intent);
    }
}
```

```
        break;
        default:
        break;
    }
}
```

Listing lengkap class myMain.java :

```
package com.FastFood;

import android.app.Activity;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.Toast;

public class myMain extends Activity {
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        Button webkfc = (Button) findViewById (R.id.skfc);
        Button phonekfc = (Button) findViewById (R.id.pmcd);
        Button webmcd = (Button) findViewById (R.id.smcd);
        Button phonemcd = (Button) findViewById (R.id.pmcd);

        webkfc.setOnClickListener(new Button.OnClickListener(){
            public void onClick(View v){
                callintent(v);
            }
        });

        phonekfc.setOnClickListener(new Button.OnClickListener(){
            public void onClick (View v){
                callintent(v);
            }
        });

        webmcd.setOnClickListener(new Button.OnClickListener(){
            public void onClick (View v){
                callintent(v);
            }
        });

        phonemcd.setOnClickListener(new Button.OnClickListener(){
            public void onClick (View v){
                callintent(v);
            }
        });
    }
}
```

```
}

    public void callintent(View view) {
        Intent intent = null;
        switch (view.getId()) {
            case R.id.pkfc:
                intent = new Intent(Intent.ACTION_CALL, Uri.parse("tel:14022"));
                startActivity(intent);
                break;
            case R.id.pgcd:
                intent = new Intent(Intent.ACTION_CALL, Uri.parse("tel:14045"));
                startActivity(intent);
                break;
            case R.id.skfc:
                intent = new Intent(Intent.ACTION_VIEW,
Uri.parse("http://kfcgaul.com"));
                startActivity(intent);
                break;
            case R.id.smcd:
                intent = new Intent(Intent.ACTION_VIEW,
Uri.parse("http://www.mcdelivery24hrs.com"));
                startActivity(intent);
                break;
            default:
                break;
        }
    }
}

@Override
protected void onActivityResult(int requestCode, int resultCode,
    Intent data) {
    if (resultCode == Activity.RESULT_OK && requestCode == 0) {
        String result = data.toURI();
        Toast.makeText(this, result, Toast.LENGTH_LONG);
    }
}
}
```

Agar aplikasi ini dapat berjalan, Anda harus menambahkan uses-permission pada AndroidManifest.xml. Buka file AndroidManifest.xml yang terdapat di dalam project Fast Food ini.



Anda tidak harus merubah keseluruhan coding xml yang terdapat pada AndroidManifest tersebut, cukup dengan menambahkan perintah xml seperti dibawah ini :

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

Untuk lebih jelasnya kita lihat perintah xml di AndroidManifest.xml berikut ini :

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.FastFood"
    android:versionCode="1"
    android:versionName="1.0">
    <uses-sdk android:minSdkVersion="8" />

    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".myMain"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

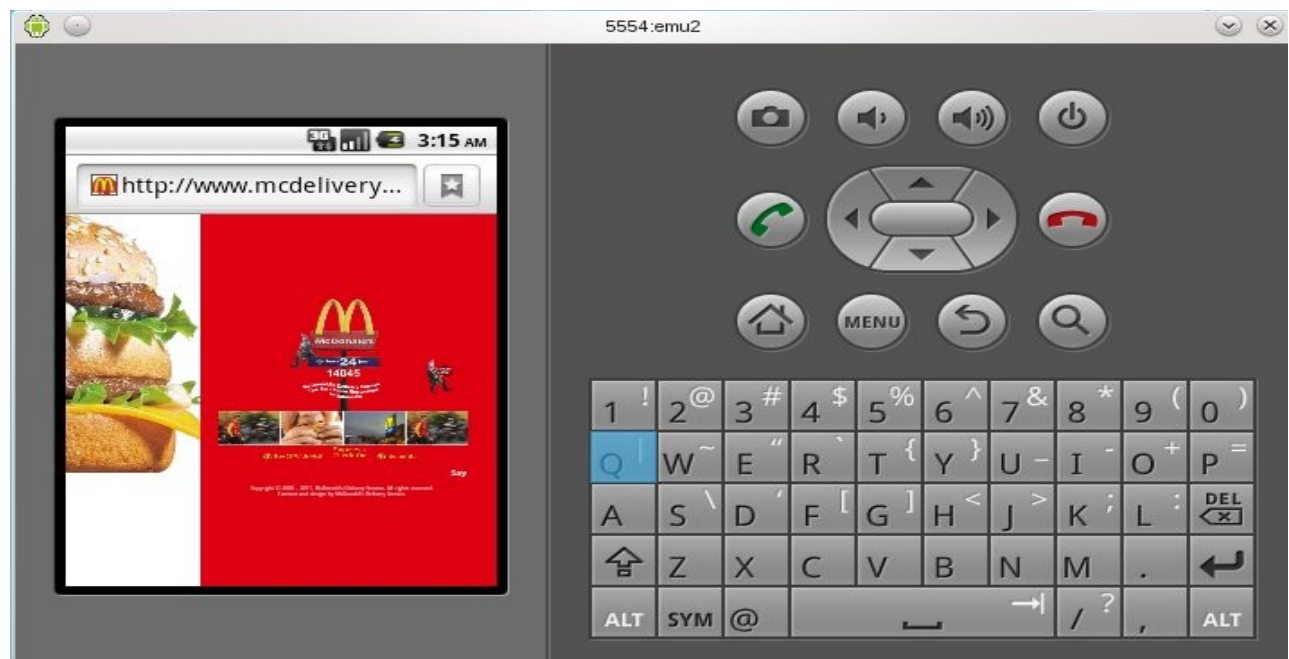
        <uses-permission android:name="android.permission.CALL_PRIVILEGED"></uses-permission>
        <uses-permission android:name="android.permission.CALL_PHONE"></uses-permission>
        <uses-permission android:name="android.permission.INTERNET"></uses-permission>

    </application>
</manifest>
```

Sekarang coba kita jalankan aplikasi tersebut melalui emulator



Kita coba untuk melihat situs Mc Donald's, langsung diklik saja tombolnya. Maka outputnya akan menjadi seperti berikut :



Ok aplikasinya berjalan dengan lancar, jadi sekarang kalo laper tinggal lihat web untuk mencari menu dari KFC atau McD terus tinggal di telepon :).

Membuat Aplikasi Penghitung Luas Bangun Datar.

Kali ini kita akan mempelajari tentang bagaimana cara membuat aplikasi penghitung luas bangun datar, seperti menghitung luas persegi panjang dan segitiga, juga kita akan mempelajari tentang bagaimana bekerja dengan tidak hanya satu form saja tapi dengan beberapa form sekaligus.

Langsung saja buat sebuah Project Android baru, lalu isikan dengan data seperti dibawah ini :

Project Name = Bangun Datar

Build Target = Android 2.2

Package Name = com.bangundatar

Application Name = Bangun Datar

Create Activity = myMain

Min SDK Version = 8

Jika sudah, klik finish.

Seperti biasa, langkah pertama yang akan kita lakukan adalah membuat tampilan utama dari program yang akan dibuat. Buka file main.xml, lalu ketikkan codingnya seperti dibawah ini :

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

<TextView
    android:layout_width = "fill_parent"
    android:layout_height= "wrap_content"
    android:text = "Pilih Menu : "
/>

<Button
```

```
android:id = "@+id/btnPersegi"  
android:layout_width = "fill_parent"  
android:layout_height= "wrap_content"  
android:text = "Luas Persegi Panjang"  
/>  
  
<Button  
android:id = "@+id/btnSegitiga"  
android:layout_width = "fill_parent"  
android:layout_height= "wrap_content"  
android:text = "Luas Segitiga"  
/>  
  
</LinearLayout>
```

Selanjutnya buat sebuah file xml baru, dengan cara klik kanan folder layout-New-Other, lalu akan keluar sebuah jendela baru pilih folder Android lalu pilih Android XML File. Beri nama file xml baru tersebut dengan nama Segitiga dan untuk type resourcenya pilih Layout.

Buka file Segitiga.xml yang sudah dibuat tersebut lalu isikan coding xml seperti dibawah ini :

```
<?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">  
  
  <TextView  
    android:layout_width = "fill_parent"  
    android:layout_height= "wrap_content"  
    android:text = "Masukan Alas"  
  />  
  
  <EditText  
    android:id = "@+id/alasSegitiga"  
    android:layout_width = "fill_parent"  
    android:layout_height= "wrap_content"  
  />  
  
  <TextView  
    android:layout_width = "fill_parent"  
    android:layout_height= "wrap_content"  
    android:text = "Masukan Tinggi"  
  />
```

```
<EditText
android:id = "@+id/tinggiSegitiga"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
/>

<Button
android:id = "@+id/hitungLuasSegitiga"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:onClick = "hitungSegitiga"
android:text = "Hitung"
/>

<TextView
android:id = "@+id/luasSegitiga"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:layout_gravity = "center"
android:textSize = "25dp"
android:text = ""
/>

</LinearLayout>
```

Buat lagi sebuah file xml baru dengan nama persegi panjang, lalu isikan coding xml seperti dibawah ini :

```
<?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">

  <TextView
  android:layout_width = "fill_parent"
  android:layout_height= "wrap_content"
  android:text = "Masukan Panjang"
  />

  <EditText
  android:id = "@+id/panjangPersegi"
  android:layout_width = "fill_parent"
  android:layout_height= "wrap_content"
  />

  <TextView
  android:layout_width = "fill_parent"
  android:layout_height= "wrap_content"
```



```
android:text = "Masukan Lebar"
/>

<EditText
android:id = "@+id/lebarPersegi"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
/>

<Button
android:id = "@+id/hitungLuasPersegi"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:onClick = "hitung"
android:text = "Hitung"
/>

<TextView
android:id = "@+id/luasPersegiPanjang"
android:layout_width = "fill_parent"
android:layout_height= "wrap_content"
android:layout_gravity = "center"
android:textSize = "25dp"
/>

</LinearLayout>
```

Anda telah selesai membuat tiga buah layout xml yang dibutuhkan untuk membuat aplikasi penghitung bangun datar, sekarang coba kita lihat output masing-masing layout tersebut dibawah ini :



Menu Utama



Menu Segitiga



Menu Persegi Panjang

Sekarang kita bermain dengan javanya, buka file myMain.java lalu isikan coding java seperti berikut ini :

Pertama import semua file yang dibutuhkan untuk membuat aplikasi penghitung bangun datar ini :

```
package com.bangundatar;

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

Selanjutnya buat sebuah public class yang meng-extends Activity dan juga sebuah class onCreate seperti dibawah ini :

```
public class myMain extends Activity {
    Button persegi, segitiga;
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);
    }
}
```

Lalu pendefinisian variabel kepada masing-masing widget yang ada pada layout main.xml juga memberikan perintah Intent pada masing-masing widget button, sehingga button-button tersebut dapat digunakan untuk memanggil form-form yang dibutuhkan.

```
persegi = (Button) findViewById (R.id.btnPersegi);
persegi.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View arg0) {
        // TODO Auto-generated method stub
        startActivity (new Intent("com.bangundatar.persegi"));
    }
});

segitiga = (Button) findViewById (R.id.btnSegitiga);
segitiga.setOnClickListener(new View.OnClickListener() {
```

```
        @Override
        public void onClick(View v) {
            // TODO Auto-generated method stub
            startActivity (new Intent("com.bangundatar.segitiga"));
        }
    });
```

Listing lengkap myMain.java :

```
package com.bangundatar;

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

public class myMain extends Activity {
    Button persegi, segitiga;
    /** Called when the activity is first created. */
    @Override
    public void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.main);

        persegi = (Button) findViewById (R.id.btnPersegi);
        persegi.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View arg0) {
                // TODO Auto-generated method stub
                startActivity (new Intent("com.bangundatar.persegi"));
            }
        });

        segitiga = (Button) findViewById (R.id.btnSegitiga);
        segitiga.setOnClickListener(new View.OnClickListener() {

            @Override
            public void onClick(View v) {
                // TODO Auto-generated method stub
                startActivity (new Intent("com.bangundatar.segitiga"));
            }
        });
    }
}
```

Selanjutnya buat sebuah class baru dengan cara, klik kanan folder project Bangun Datar lalu pilih new-class. Akan keluar sebuah jendela baru lalu berinama class baru tersebut dengan nama PersegiPanjang.

Buka class PersegiPanjang tersebut lalu isikan dengan coding java seperti dibawah ini :

Import semua file yang dibutuhkan :

```
package com.bangundatar;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
```

Buat sebuah public class dengan nama PersegiPanjang yang meng-extends Activity juga berikan nama pada masing-masing variabel widget;

```
public class PersegiPanjang extends Activity{

    EditText panjang, lebar;
    TextView luas;
    Button hitung;
```

Buat sebuah class baru dengan nama onCreate lalu definisikan masing-masing variabel widget yang akan digunakan :

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    // TODO Auto-generated method stub
    super.onCreate(savedInstanceState);
    setContentView(R.layout.persegipanjang);

    panjang = (EditText) findViewById (R.id.panjangPersegi);
    lebar = (EditText) findViewById (R.id.lebarPersegi);
    luas = (TextView) findViewById (R.id.luasPersegiPanjang);
    hitung = (Button) findViewById (R.id.hitungLuasPersegi);
}
```

Buat sebuah class baru dengan nama hitung dimana class inilah yang akan melakukan proses perhitungan luas persegi panjang:

```
public void hitung(View v){
    int p = Integer.parseInt(panjang.getText().toString());
    int l = Integer.parseInt(lebar.getText().toString());
    int luasPP = p*l;

    luas.setText(String.valueOf(luasPP));
}
```

Listing lengkap file PersegiPanjang.java :

```
package com.bangundatar;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class PersegiPanjang extends Activity{

    EditText panjang, lebar;
    TextView luas;
    Button hitung;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        // TODO Auto-generated method stub
        super.onCreate(savedInstanceState);
        setContentView(R.layout.persegipanjang);

        panjang = (EditText) findViewById (R.id.panjangPersegi);
        lebar = (EditText) findViewById (R.id.lebarPersegi);
        luas = (TextView) findViewById (R.id.luasPersegiPanjang);
        hitung = (Button) findViewById (R.id.hitungLuasPersegi);
    }

    public void hitung(View v){
        int p = Integer.parseInt(panjang.getText().toString());
        int l = Integer.parseInt(lebar.getText().toString());
        int luasPP = p*l;

        luas.setText(String.valueOf(luasPP));
    }
}
```

Buat sebuah file class baru, lalu beri nama Segitiga, lalu buka file Segitiga.java tersebut dan isikan dengan coding seperti dibawah ini :

```
package com.bangundatar;  
  
import android.app.Activity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;
```

Buat sebuah public class baru dengan nama Segitiga yang meng-extends Activity dan juga berikan nama pada masing-masing variabel widget seperti berikut :

```
public class Segitiga extends Activity{  
  
    EditText alasSegitiga, tinggiSegitiga;  
    TextView luasSegitiga;  
    Button hitung;
```

Buat sebuah class baru dengan nama onCreate, dimana pada class ini kita juga mendefinisikan variabel dari masing-masing widget yang dibutuhkan :

```
@Override  
    protected void onCreate(Bundle savedInstanceState) {  
        // TODO Auto-generated method stub  
        super.onCreate(savedInstanceState);  
        setContentView (R.layout.segitiga);  
  
        alasSegitiga = (EditText) findViewById (R.id.alasSegitiga);  
        tinggiSegitiga = (EditText) findViewById (R.id.tinggiSegitiga);  
        hitung = (Button) findViewById (R.id.hitungLuasSegitiga);  
        luasSegitiga = (TextView) findViewById (R.id.luasSegitiga);  
  
    }
```

Buat juga sebuah class dengan nama `hitungSegitiga`, dimana class ini yang akan melakukan perhitungan untuk mengetahui luas segitiga :

```
public void hitungSegitiga(View v){
    int alas = Integer.parseInt(alasSegitiga.getText().toString());
    int tinggi = Integer.parseInt(tinggiSegitiga.getText().toString());
    int luas = alas*tinggi;

    luasSegitiga.setText(String.valueOf(luas));
}
```

Listing lengkap `Segitiga.java` :

```
package com.bangundatar;

import android.app.Activity;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;

public class Segitiga extends Activity{

    EditText alasSegitiga, tinggiSegitiga;
    TextView luasSegitiga;
    Button hitung;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        // TODO Auto-generated method stub
        super.onCreate(savedInstanceState);
        setContentView (R.layout.segitiga);

        alasSegitiga = (EditText) findViewById (R.id.alasSegitiga);
        tinggiSegitiga = (EditText) findViewById (R.id.tinggiSegitiga);
        hitung = (Button) findViewById (R.id.hitungLuasSegitiga);
        luasSegitiga = (TextView) findViewById (R.id.luasSegitiga);
    }

    public void hitungSegitiga(View v){
        int alas = Integer.parseInt(alasSegitiga.getText().toString());
        int tinggi = Integer.parseInt(tinggiSegitiga.getText().toString());
        int luas = alas*tinggi;

        luasSegitiga.setText(String.valueOf(luas));
    }
}
```

Terakhir buka file AndroidManifest.xml, lalu tambahkan perintah seperti berikut ini agar widget Button yang ada pada menu utama dapat digunakan untuk memanggil form/layout yang sesuai :

```
<activity android:name=".PersegiPanjang"
    android:label="@string/app_name">
    <intent-filter>
        <action android:name="com.bangundatar.persegi" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>

<activity android:name=".Segitiga"
    android:label="@string/app_name">
    <intent-filter>
        <action android:name="com.bangundatar.segitiga" />
        <category android:name="android.intent.category.DEFAULT" />
    </intent-filter>
</activity>
```

Listing lengkap AndroidManifest.xml :

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.bangundatar"
    android:versionCode="1"
    android:versionName="1.0">
    <uses-sdk android:minSdkVersion="8" />

    <application android:icon="@drawable/icon" android:label="@string/app_name">
        <activity android:name=".myMain"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>

        <activity android:name=".PersegiPanjang"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="com.bangundatar.persegi" />
                <category android:name="android.intent.category.DEFAULT" />
            </intent-filter>
        </activity>

        <activity android:name=".Segitiga"
            android:label="@string/app_name">
            <intent-filter>
                <action android:name="com.bangundatar.segitiga" />
            </intent-filter>
        </activity>
```



```
<category android:name="android.intent.category.DEFAULT" />
</intent-filter>
</activity>

</application>
</manifest>
```

Sekarang semuanya sudah selesai, tinggal coba Anda jalankan aplikasi penghitung luas bangun datar tersebut melalui emulator, hasilnya seperti gambar dibawah ini :

Output Menu Utama :



Di menu utama ini terdapat dua buah tombol yang digunakan untuk memilih menu, menu pertama adalah untuk melakukan perhitungan luas persegi panjang dan menu kedua adalah menu untuk melakukan perhitungan luas segitiga.

Output form Persegi Panjang :



Pada menu Persegi panjang, Anda dapat memasukkan nilai panjang dan lebar dari Persegi panjang yang anda ingin ketahui nilainya, selanjutnya setelah menekan tombol hitung, aplikasi ini akan melakukan perkalian nilai panjang dan lebar dan menampilkan hasil perhitungannya.

Output Menu Segitiga :



Sama seperti pada menu persegi panjang, menu menghitung luas segitiga ini juga akan melakukan perkalian nilai alas dan tinggi yang diinputkan oleh Anda, lalu setelah menekan tombol hitung akan keluar hasil perkaliannya.

Tentang Penulis :



Penulis bernama Firdan Ardiansyah, lahir di Jakarta pada tanggal 03 mei 1990. Penulis memulai pendidikannya di SD MCT XV Rangkasbitung lalu melanjutnya sekolahnya ke SMP Negeri 1 Rangkasbitung dan melanjutkan pendidikan SMA di SMK Negeri 1 Rangkasbitung jurusan Teknik Komputer dan Jaringan.

Saat menulis buku ini, penulis masih terdaftar sebagai mahasiswa semester akhir di Universitas Gunadarma jurusan Sistem Informasi. Penulis juga aktif mengikuti forum–forum yang membahas Java dan Android Programming.

Penulis dapat dihubungi di :

bimbim_rocknroll@yahoo.com

virdane.ardiansyah@gmail.com