

24 JAM!! Pintar Pemrograman 

ANDROID

Ebook Version 2.1



24

Arif Akbarul Huda | www.omyib.com

PERHATIAN!!

Ebook ini **GRATISS!! 100%**. Silakan digunakan untuk belajar sebagai mana mestinya. Kamu bisa menggunakan Ebook ini untuk belajar, kuliah atau hal-hal bermanfaat lainnya.

Jika Kamu menyukai ebook ini, rekomendasikan ke teman-teman, publish ke internet, share ke group, forum atau silakan menyebarluaskan kemana saja.

Silakan juga Ebook ini Kamu **JUAL** jika Kamu **tidak menghargai** sama sekali hasil karya orang lain.

Demi terwujudnya ebook-ebook baru yang berkualitas, Kamu bisa berkontribusi dengan cara memberikan sejumlah DONASI.

Semoga ebook ini bermanfaat untuk developer-developer pemula khususnya, bahkan untuk Bangsa Indonesia di bidang IT

Kirim TESTIMONI, KRITIK, SARAN
Kamu di
www.omayib.com/testimoni

TESTIMONI



Saatnya kita berubah dari pengguna menjadi seorang pengembang. Indonesia masih banyak membutuhkan buku penuh semangat seperti buku ini. Terus berkarya!!!

Agung Yulianto N
(Direktur ImaigneIT Education Center)

Your Testimoni here

Your Testimoni here

Your Testimoni here

PENULIS



Pemilik website www.odayib.com ini bernama Arif Akbarul Huda. Setelah menyelesaikan studinya di S1 Elektronika dan Instrumentasi UGM, sekarang aktif di ImagineIT Center sebagai Trainer Pemrograman Android dan Adobe Flash (Actionscript2.0 dan 3.0) sambil menyelesaikan studi Pascasarjana di Magister Teknik Informasi (MTI) UGM Yogyakarta.

Penulis lahir di Wonogiri, saat ini berdomisili di Yogyakarta. Beberapa kali menjadi pemateri acara pelatihan dan juga membuat beberapa project seperti

1. Trainer Pelatihan Sistem Informasi DISPERINDAGKOP DIY
2. Trainer pelatihan sistem Informasi Pusdiklat Migas Cepu
3. Kiosk (Aplikasi touchscreen interactive untuk Pengadilan Negeri Magelang dan PN Banyuwangi) |
4. Aplikasi Android "DISKON"
5. Aplikasi Android "GUITAR TUNER"
6. (Image Processing) Algoritma deteksi ujung jari untuk Human-Computer Interaction
7. Ebook "Membuat sendiri Aplikasi Android untuk Pemula"

Silakan follow penulis di



@omAyib



facebook.com/alulalul



odayib@gmail.com

Visit

www.odayib.com

Ebook ini GRATIS!!! 100% masih jauh dari sempurna. Kami sangat menerima sekali donasi , saran, kritik, koreksi atau masukan.

DONASI

BANTU PENULIS MENYEMPURNAKAN EBOOK BERMANFAAT INI

Dengan berdonasi, Kamu mendorong penulis untuk lebih produktif menghasilkan tulisan / berbagi ilmu yang Insya Allah bermanfaat buat orang banyak. Apabila Kamu tidak ada waktu untuk berbagi ilmu, Kamu bisa berkontribusi melalui cara ini. Menarik bukan??! (Semua yang ada didunia tidak akan dibawa mati, kecuali 3 hal salah duanya ilmu yang bermanfaat dan amal jariyah).hehehe... .

Donasi dapat berupa sejumlah uang, akan Penulis gunakan untuk menunjang produktivitas menulis, membeli makan, minum dan kopi untuk menulis ebook android lebih sempurna.

Kami menerima berapapun jumlah donasi mulai dari

- ✓ Rp.25.000,00
- ✓ Rp.50.000,00
- ✓ Rp.75.000,00
- ✓ >> Rp. 100.000,00

CARA MENGIRIM DONASI

Donasi bisa dikirim transfer Bank melalui rekening berikut

	Bank BNI K.C. UGM Yogyakarta No.Rek : 0245006491 A.n : Arif Akbarul Huda
	Bank MANDIRI No. Rek : 138-00-0755778-3 A.n : Arif Akbarul Huda

Jika sudah melakukan transfer, mohon kesediaan untuk melakukan **konfirmasi pemebritahuan** ke www.odayib.com/donasi atau ke email **odayib@gmail.com**

IMAGINE IT EDUCATION CENTER



ImagineIT Education Center merupakan tempat pelatihan komputer terkemuka di Yogyakarta, dengan peserta pelatihan Mahasiswa maupun Instansi.

Saat ini ImagineIT Education Center menyediakan kelas regular, private maupun studycase untuk beberapa macam pemrograman seperti

1. Android
2. PHP, HTML, CSS, JS, JQUERY
3. Adobe Flash
4. Visual Basic
5. Jaringan Komputer
6. Image Processing , dll

ImagineIT Education Center menerima kerjasama untuk mengadakan **Workshop, seminar, atau pelatihan** seputar IT. Lebih jelasnya silakan follow FB dan Twitter Kami.

Info :



@imagin**IT**center



Facebook.com/**imagineIT**center



085747969669

Galery Imagine**IT** Education Center



DAFTAR ISI

TESTIMONI	3
PENULIS.....	4
DONASI	5
IMAGINE IT EDUCATION CENTER	6
DAFTAR ISI.....	7
JAM 1 SEPUTAR ANDROID	10
A. KONSEP UMUM.....	10
JAM 2 INSTALASI ECLIPSE.....	15
B. INSTALASI ECLIPSE.....	15
C. PROGRAM PERTAMA, ASSALAMUALAIKUM!	21
JAM 3 LAYOUT ANDROID #1	25
A. LINEAR LAYOUT	25
B. RELATIVELAYOUT SEDERHANA.....	29
C. TABLE LAYOUT	31
JAM 4 ANDROID DASAR	35
A. MENAMPILKAN GAMBAR	35
B. MEMBUAT TEKS AUTOCOMPLETE	39
JAM 5 ANDROID DASAR	43
A. MEMBUAT KOTAK DIALOG	43
B. PICKER	47
JAM 6 ANDROID WIDGET.....	52
A. MEMBUAT CHECK BOX	52
B. RADIOBUTTON	54
JAM 7 ANDROID WIDGET.....	59
A. MENGGUNAKAN LIST VIEW	59
B. MENU	62
JAM 8 ANDROID INTENT	63
A. MEMANGGIL ACTIVITY	63
JAM 8 BACKGROUND SERVICE	67
JAM 10 MULTIMEDIA AUDIO	71
A. MENGAKSES SOUND	71
JAM 11 MULTIMEDIA VIDEO	75
JAM 12 ANDROID SQLITE DATABASE	78
A. MEMBUAT DATABASE	79

JAM 13 ANDROID SQLITE DATABASE #2.....	86
A. ANDROID DATABASE #2.....	86
JAM 14 ANDROID GPS.....	97
JAM 15 GOOGLE MAP	101
JAM 17 ANDROID DAN PHP	113
A. LOGIN MENGGUNAKAN PHP	113
B. LOGIN MENGGUNAKAN DATABASE.....	120
JAM 18 ANDROID DAN MYSQL.....	123
JAM 19 COMPASS ANDROID	125
A. ANALOG COMPASS	125
JAM 20 SENSOR ANDROID	128
JAM 21 CAMERA ANDROID	129
JAM 22 BLUETOOTH ANDROID.....	130
JAM 23-24.....	131
A. APLIKASI DISKON	132
B. APLIKASI GUITAR TUNER	137

BAB 1

Pengenalan Android

Kirim **TESTIMONI** Kamu di
www.omyib.com/testimoni

JAM 1 | SEPUTAR ANDROID

Konsep Umum

Debugging Android dgn DDMS <NOT YET AVAILABLE>

Membaca ERROR<NOT YET AVAILABLE>

Mengenal Package<NOT YET AVAILABLE>

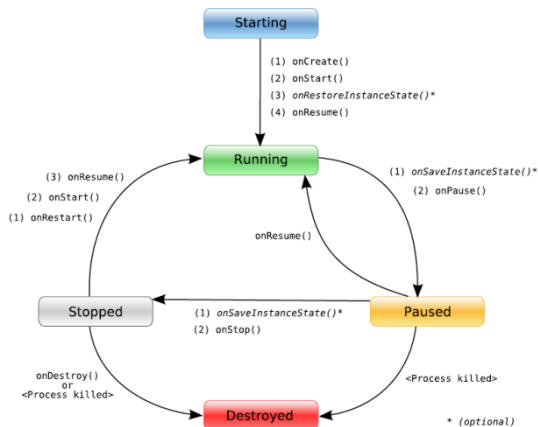
Mengenal XML<NOT YET AVAILABLE>

A. Konsep Umum

Perangkat berbasis android hanya mempunyai satu layar foreground. Normalnya saat menghidupkan android, yang pertama Kamu lihat adalah home. Kemudian bila Kamu menjalankan sebuah aplikasi catur, User Interfacenya (UI) akan menumpuk diatas layar sebelumnya (home). Kemudian bila melihat help-nya catur, maka UI help akan menimpa UI sebelumnya (catur), begitu seterusnya.

Semua proses diatas direkam di *application stack* oleh sistem Activity manager. Menekan tombol back hanya kembali ke halaman sebelumnya, analoginya mirip dengan browser dimana ketika Kamu meng-klik tombol back browser akan kembali menampilkan halaman sebelumnya.

Setiap User Interface diwakili oleh kelas Activity (Activity class). Setiap activity mempunyai siklus, dapat dilihat di gambar 1.14. Sebuah aplikasi dapat terdiri dari satu atau lebih activity yang diproses dalam Linux. Jika Kamu bingung dengan penjelasan ini, jangan terlalu dipikirkan. Tetap saja maju terus, Kamu akan paham setelah benar-benar mempraktikkan latihan-latihan dalam buku ini.



Gambar 1.14

Siklus activity

Selama siklus ini berjalan, activity bisa mempunyai lebih dari 2 status seperti yang terlihat pada gambar 1.14. Kita tidak bisa mengontrol setiap status karena semuanya sudah ditangani oleh sistem. Namun kita akan mendapat pesan saat terjadi perubahan status melalui method onXX(). Berikut penjelasan setiap status.

onCreate(Bundle)	Dipanggil saat pertama kali aplikasi dijalankan. Kita dapat menggunakan ini untuk deklarasi variabel atau membuat user interface.
onStart()	Mengindikasikan activity yang ditampilkan ke pengguna (user).
onResume()	Dipanggil saat aplikasi kita mulai berinteraksi dengan pengguna. Disini sangat cocok untuk meletakkan animasi ataupun musik.
onPause()	Dipanggil saat aplikasi yang Kita jalankan kembali ke halaman sebelumnya atau biasanya karena ada activity baru yang dijalankan. Disini cocok untuk meletakkan algoritma penyimpanan (save).
onStop()	Diapnggil saat aplikasi Kita berjalan di belakang layar dalam waktu cukup lama.
onRestart()	Activity kembali menampilkan user interface setelah status stop.
onDestroy()	Dipanggil saat aplikasi benar-benar berhenti.
onSaveInstanceState(Bundle)	Method ini mengijinkan activity untuk menyimpan setiap status intance. Misalnya dalam mengedit teks, kursor bergerak dari kiri ke kanan.
onRestoreInstanceState(Bundle)	Diapanggil saat activity kembali meng-inisialisasi dari status sebelumnya yang disimpan oleh onSaveInstanceState(Bundle).

Sebagai programmer, Kamu harus mengetahui beberapa komponen aplikasi yang sangat penting seperti activities, intents, service, dan content providers.

a. Activity

Normalnya setiap activity menampilkan satu buah user interface kepada pengguna. Misalnya sebuah activity menampilkan daftar menu minuman, kemudian pengguna dapat memilih satu jenis minuman. Contoh lainnya pada aplikasi sms, dimana satu activity digunakan untuk menulis pesan, activity berikutnya untuk menampilkan nomor kontak tujuan, atau activity lainnya digunakan untuk menampilkan pesan-pesan lama. Meskipun activity-activity diatas terdapat dalam satu aplikasi sms, namun masing-masing activity berdiri sendiri. Untuk pindah dari satu activity ke activity lainnya dapat melakukan suatu event misalnya tombol diklik atau melalui trigger tertentu.

b. Service

Service tidak memiliki user interface, namun berjalan di belakang layar. Misalnya music player, sebuah activity digunakan untuk memilih lagu kemudian di-play. Agar music player bisa berjalan dibelakang aplikasi lain maka harus menggunakan service.

c. Intents

Intens adalah mekanisme untuk menggambarkan sebuah action secara detail seperti bagaimana cara mengambil sebuah foto.

d. Content Providers

Menyediakan cara untuk mengakses data yang dibutuhkan oleh suatu activity, misalnya kita menggunakan aplikasi berbasis peta (MAP). Activity membutuhkan cara untuk mengakses data kontak untuk prosedur navigasi. Disinilah peran content providers.

Resource

Resource digunakan untuk menyimpan file-file non-coding yang diperlukan pada sebuah aplikasi misalnya file icon, file gambar, file audio, file video atau yang lain. Gambar berformat JPG atau PNG sebuah aplikasi biasanya disimpan dalam folder res/drawable, icon aplikasi disimpan dalam res/drawable-ldpi dan file audio disimpan dalam folder res/raw. File XML untuk membentuk sebuah user interface disimpan dalam folder res/layout.

Ingin mengadakan Workshop/seminar di Kampus
Kamu?? Hubungi
ImagineIT Education Center Yogyakarta

Apabila Ebook ini bermanfaat buat Kamu, Bantu
penulis menyempurnakan ebook ini dengan cara
DONASI.

BAB2

INSTALASI

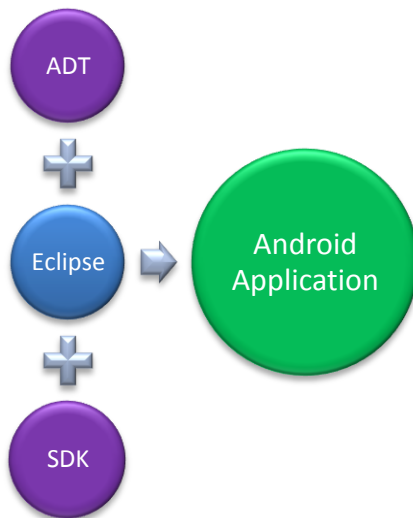
JAM 2 | INSTALASI ECLIPSE

B. Instalasi Eclipse

Kamu harus memiliki 3 buah file dibawah ini, bisa mengunduhnya langsung dari web resminya di www.eclipse.org. (download saja versi terbarunya)

1. Eclipse ([Klik link ini](#))
2. ADT 16.0.1 Plugin ([Klik link ini](#))
3. Android-SDK_r16 ([Klik link ini](#))

Eclipse merupakan sebuah editor, secara default editor ini belum bisa dipakai untuk mendevlop android. Agar bisa digunakan untuk membuat aplikasi android maka harus diinstall plugin dulu namanya ADT (Android Development Tools). Setelah terinstal, maka eclipse sudah siap digunakan, hanya saja belum dilengkapi library dan emulator. Untuk memenuhinya, maka perlu diinstal SDK Android (StKamurt Development Kit).

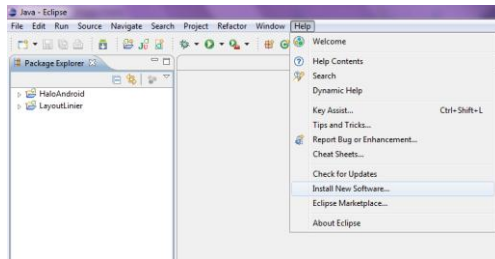


Jadi intinya, eclipse bisa digunakan setelah Kamu selesai mengkonfigurasi antara editor Eclipse, ADT dan SDK. Selanjutnya ikuti langkah demi langkah berikut. Semoga berhasil! :D

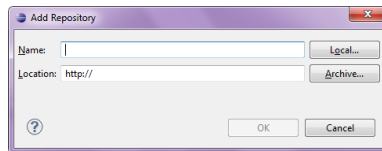
MENGINSTAL PLUGIN ADT

Cara 1. Menginstal ADT Plugin untuk eclipse **langsung dari server**. Pada cara pertama ini, Kamu harus terhubung dengan internet.

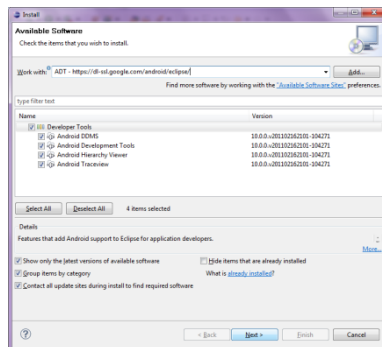
1. Jalankan Eclipse, kemudian pilih help > instal new software, Lihat Gambar 2.1.
2. Klik **add** di sebelah kanan atas.
3. Pada kotak dialog add repository, isikan **nama : ADT** dan **location :** <https://dl-ssl.google.com/android/eclipse/> (gambar 2.2).
4. Lihat kotak Available software. Centang pada item Developer Tools kemudian pilih **Next.** (Gambar 2.3)
5. Pada tampilan berikutnya, Kamu akan melihat lebih detail daftar tools yang akan di download. Pilih **Next.**



Gambar 2. 1



Gambar 2. 2

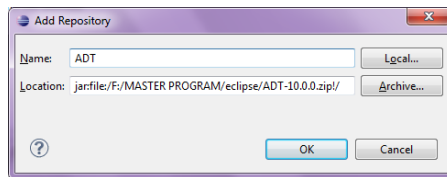


Gambar 2. 3

6. Begitu muncul license agreement. Pilih **Accept**, kemudian klik **Finish.**
7. Setelah proses instalasi selesai, restart eclipse.

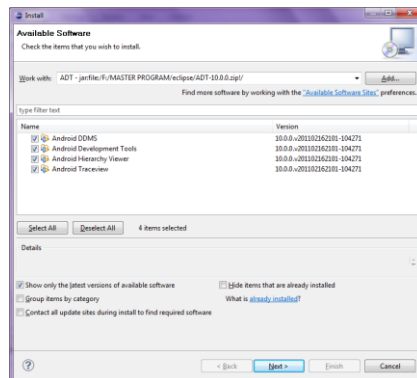
Cara 2. Apabila terkendala dengan koneksi internet, Kamu juga bisa melakukan instalasi plugin secara offline. Tapi harus mendownload terlebih dahulu ADT Plugin kemudian simpan di direktori tertentu.

1. Ikuti langkah 1 dan 2 pada cara pertama
2. Pada dialog add site, pilih **Archive**.
3. Cari dan pilih file ADT.zip yang Kamu download diawal tadi.
4. Jangan lupa memberi nama pada beris nama lihat Gambar 2. 4.



Gambar 2. 4

5. Klik ok.
6. Lihat kotak Available software (Gambar 2.5). Centang pada item Developer Tools kemudian pilih **Next**.

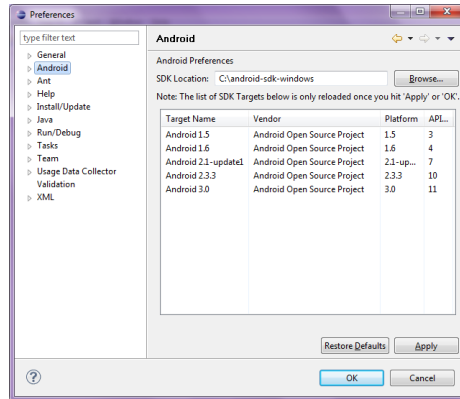


Gambar 2. 5

7. Pada tampilan berikutnya, Kamu akan melihat daftar Tools yang akan di instal. Pilih **Next**.
8. Begitu muncul license agreement. Pilih **Accept**, kemudian klik **Finish**.
9. Setelah proses instalasi selesai, restart eclipse.

KONFIGURASI SDK

1. Jalankan Eclipse, pilih **window > Preference**, muncul interface seperti Gambar 2.6.
2. Pada panel sebelah kiri, pilih **Android**



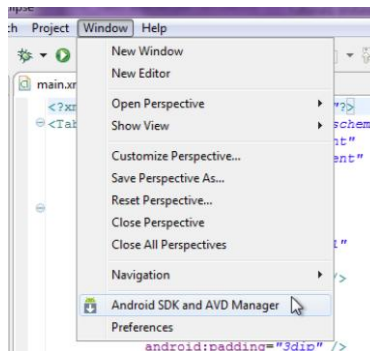
Gambar 2. 6

3. Pada SDK Location, klik **Browse**, cari kemudian pilih SDK File yang sudah Kamu miliki sebelumnya
4. Pilih salah satu platform android, kemudian klik **Apply** .
5. Klik **Ok**.

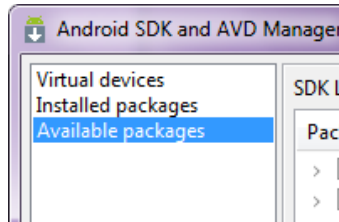
UPDATE PACKAGE SDK (OPTIONAL)

Pada tahap ini harus terhubung dengan internet untuk melakukan update repositori. Jika tidak, maka Kamu tidak punya package Android. Berikut caranya

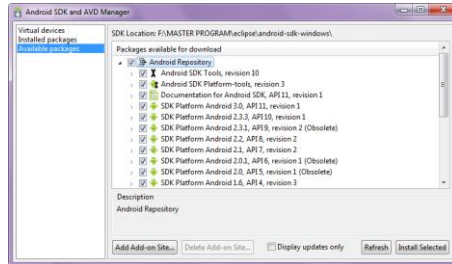
1. Pada Eclipse, pilih **Window > Android SDK and AVD manager** (Gambar 2.7)
2. Pilih **Available Package** (Gambar 2.8)
3. Centang pada **Android Repository** (Gambar 2.9)
4. Otomatis akan melakukan Update , jika tidak pilih **Refresh**



Gambar 2. 7

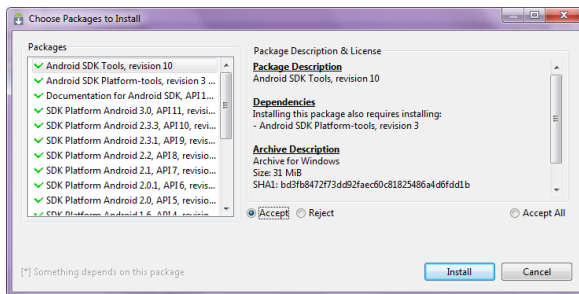


Gambar 2.8

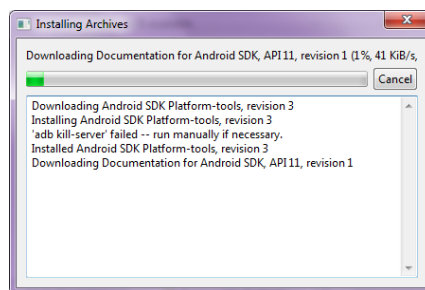


Gambar 2.9

5. Centang Android Repositori, kemudian pilih **Instal Selected**



Gambar 2.10



Gambar 2.11

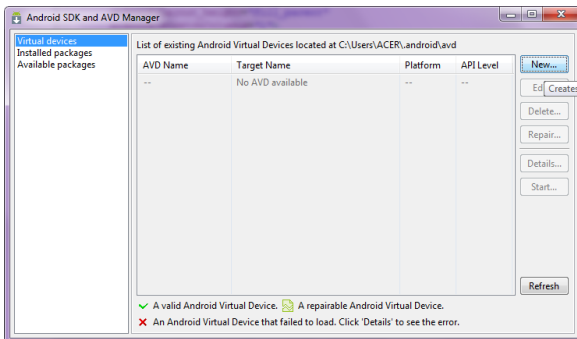
MEMBUAT ANDROID VIRTUAL DEVICE (AVD)

Nah, saatnya membuat Android Virtual Device (AVD). AVD adalah perangkat visual android yang akan menjadi simulator.

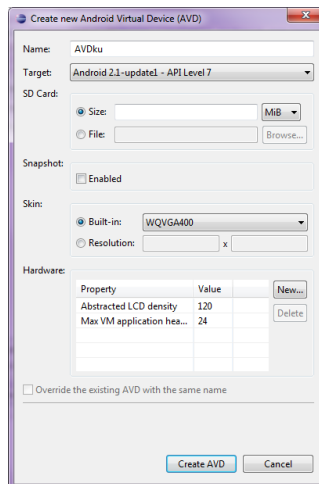


1. Pada kotak dialog Android SDK dan AVD Manager, pilih **New** (Gambar 2.12)
2. Isikan seperti berikut (Gambar 2.13)

Name	Froyo22
Target	Android 2.2
SDCard	Dibiarkan saja
Snapshot	Dibiarkan saja
Skin	WQVGA400
Hardware	Dibiarkan saja



Gambar 2.12



Gambar 2. 13

Nah, dengan demikian, sekarang Kamu sudah memiliki Emulator Android. :D Menyenangkan bukan?? Mau tahu seperti apa emulatornya?? Ini deh, saya kasih.. lihat Gambar 2.14 yaa..hehe

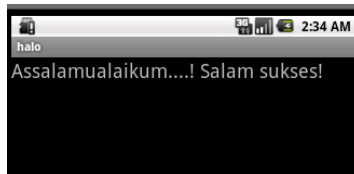


Gambar 2. 6

Selanjutnya kita bersambung dalam ebook HelloWord nya android. Disini Kamu bisa melihat emulator dengan aplikasi buatanmu sendiri! Keren kan???! :D

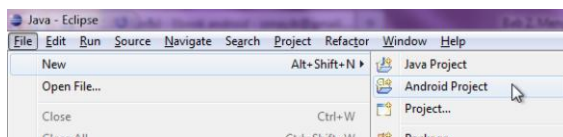
C. Program pertama, Assalamualaikum!

Sebagai pemanasan, kita membuat applikasi sederhana menampilkan tulisan Assalamualaikum di emulator.



Gambar 2.15

1. Jalankan Enclipse, **File > New > Android Project** lihat Gambar 2.16.

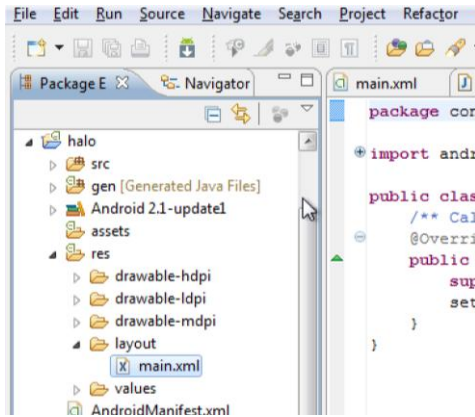


Gambar 2.16

2. Isikan kotak dialog new seperti berikut.

Project name	Halo
Build Target	Android 2.2
Application name	Halo, Assalamualaikum
Package name	contoh.halo
Create Activity	halo

3. Pada Package Explorer, buka file main.xml di folder res/layout perhatikan Gambar 2.17



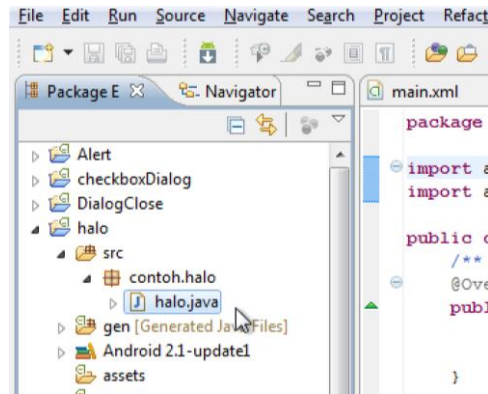
Gambar 2.17

4. Ganti kode pada main.xml menjadi seperti berikut

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
3:     android:orientation="vertical"
4:     android:layout_width="fill_parent"
5:     android:layout_height="fill_parent"
6:     >
7: <TextView
8:     android:layout_width="fill_parent"
9:     android:layout_height="wrap_content"
10:    android:text="Assalamualaikum....! Salam sukses!"
11:    />
12: </LinearLayout>
  
```

5. Pada package src/contoh.halo, buka halo.java pastikan kodenya seperti berikut ini , perhatikan Gambar 2.18



Gambar 2.18

Activity halo.java

```

1: package contoh.halo;
2:
3: import android.app.Activity;
4: import android.os.Bundle;
5:
6: public class halo extends Activity {
7:     /** Called when the activity is first created. */
8:     @Override
9:     public void onCreate(Bundle savedInstanceState) {
10:         super.onCreate(savedInstanceState);
11:         setContentView(R.layout.main);
12:     }
13: }

```

6. Jalankan run dengan shortcut **CTRL+F11** atau klik **kanan package > run as > android Project.**
7. Lihat hasilnya seperti gambar 2.15. Just it! Mudah kan?? :D

Kamu bisa mendownload langsung aplikasi ini ke perangkat android. File yang harus didownload adalah halo.apk yang terletak di folder halo/bin.

Ingin mengadakan Workshop/seminar di Kampus
Kamu?? Hubungi
ImagineIT Education Center Yogyakarta

BAB 3

DASAR ANDROID #1

Ingin mengadakan Workshop/seminar di Kampus
Kamu?? Hubungi

ImagineIT Education Center Yogyakarta

JAM 3 | LAYOUT ANDROID #1

Pada bagian ini, Kamu akan mempelajari mengenai XML

- ✓ Linear Layout
- ✓ Relative Layout
- ✓ Table Layout

A. Linear Layout

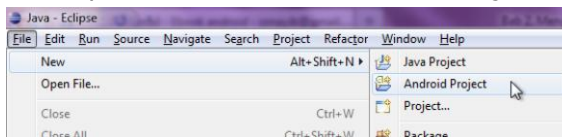
Seperti apa sih tampilan Linear Layout? Hhmm... coba lihat Gambar 3.1 berikut. Jika Kamu seorang pemula, Kamu boleh sedikit tidak mepedulikan script. Semakin banyak Kamu berlatih coding, semakin paham juga maksud programnya.



Gambar 3. 1

Yuk langsung kita mulai!:)

1. Jalankan Enclipse, File > New > Android Project.

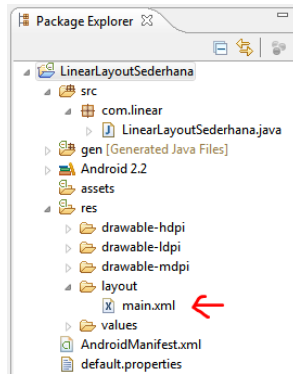


Gambar 3. 2

2. Isikan kotak dialog new seperti berikut

Project name	LinearLayoutSederhana
Build Target	Android 2.2
Application name	LinearLayout
Package name	Com.linear
Create Activity	LinearLayoutSederhana
Min SDK version	8

3. Pilih LinearLayoutSederhana /res /layout/main.xml, lihat Gambar 3.3



Gambar 3.3

4. Ketikkan kode berikut ini. Meskipun panjang, script ini memiliki kemiripan. Sementara cukup tulis satu pola saja, lakukan copy-paste-edit.

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
3:   xmlns:android="http://schemas.android.com/apk/res/android"
4:     android:orientation="vertical"
5:     android:layout_width="fill_parent"
6:     android:layout_height="fill_parent">
7:
8:   <LinearLayout
9:     android:orientation="horizontal"
10:    android:layout_width="fill_parent"
11:    android:layout_height="fill_parent"
12:    android:id="@+id/LinearLayout1"
13:    android:layout_weight="1">
14:    <TextView
15:      android:text="Merah"
16:      android:gravity="center_horizontal"
17:      android:background="#aa0000"
18:      android:layout_height="fill_parent"
19:      android:layout_weight="1"
20:      android:layout_width="wrap_content"/>
21:    <TextView
22:      android:text="Hijau"
23:      android:gravity="center_horizontal"
24:      android:background="#00aa00"
25:      android:layout_width="wrap_content"
26:      android:layout_height="fill_parent"
27:      android:layout_weight="1"/>
28:    <TextView
29:      android:text="Biru"
30:      android:gravity="center_horizontal"
31:      android:background="#0000aa"
32:      android:layout_width="wrap_content"
33:      android:layout_height="fill_parent"
34:      android:layout_weight="1"/>
35:    <TextView
36:      android:text="Kuning"

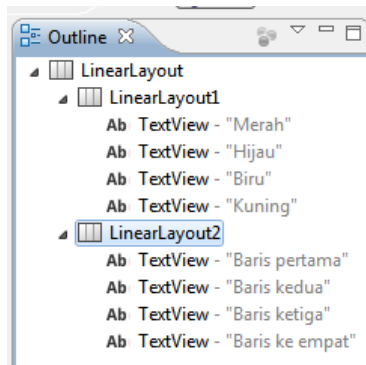
```

```

37:         android:gravity="center_horizontal"
38:         android:background="#aaaa00"
39:         android:layout_width="wrap_content"
40:         android:layout_height="fill_parent"
41:         android:layout_weight="1"/>
42:     </LinearLayout>
43:
44:     <LinearLayout
45:         android:orientation="vertical"
46:         android:layout_width="fill_parent"
47:         android:layout_height="fill_parent"
48:         android:id="@+id/LinearLayout2"
49:         android:layout_weight="1">
50:         <TextView
51:             android:text="Baris pertama"
52:             android:textSize="15pt"
53:             android:layout_width="fill_parent"
54:             android:layout_height="wrap_content"
55:             android:layout_weight="1"/>
56:         <TextView
57:             android:text="Baris kedua"
58:             android:textSize="15pt"
59:             android:layout_width="fill_parent"
60:             android:layout_height="wrap_content"
61:             android:layout_weight="1"/>
62:         <TextView
63:             android:text="Baris ketiga"
64:             android:textSize="15pt"
65:             android:layout_width="fill_parent"
66:             android:layout_height="wrap_content"
67:             android:layout_weight="1"/>
68:         <TextView
69:             android:text="Baris ke empat"
70:             android:textSize="15pt"
71:             android:layout_width="fill_parent"
72:             android:layout_height="wrap_content"
73:             android:layout_weight="1"/>
74:     </LinearLayout>
75: </LinearLayout>

```

Layout main.xml jika dilihat hierarkinya dari outline tampak seperti Gambar 3.4



Gambar 3.4. Susunan layout main.xml

5. Pastikan pada activity `LinearLayoutSederhana.java` seperti dibawah ini.

```

1: package com.linear;
2:
3: import android.app.Activity;
4: import android.os.Bundle;
5:
6: public class LinearLayoutSederhana extends Activity {
7:     /** Called when the activity is first created. */
8:     @Override
9:     public void onCreate(Bundle savedInstanceState) {
10:         super.onCreate(savedInstanceState);
11:         setContentView(R.layout.main);
12:     }
13: }
14:

```

6. Lakukan Run dengan shortcut `CTRL+F11` atau klik kanan package > run as > android Project.
7. Lihat hasilnya seperti Gambar 3.1

PENJELASAN PROGRAM

Sebuah aplikasi android dibuat dari kombinasi XML dan JAVA. Biasanya, xml digunakan untuk mengatur layout aplikasi sedangkan java berperan sebagai pusat pengendalinya. Pembahasan dibagian ini focus dulu ke xml.

Perhatikan `main.xml` baris 2, layout diawali oleh widget `LinearLayout`. Widget ini berperan sebagai parent yang orientation nya diset vertical (baris 4), artinya berapapun jumlah widget yang ada didalamnya akan tersusun secara vertical dari atas ke bawah. Sekarang perhatikan gambar 3.4, Parent `LinearLayout` memiliki 2 buah child yaitu `LinearLayout1` dan `LinearLayout2`, nantinya masing-masing child juga mempunyai child sendiri. Karena parent `LinearLayout` orientationnya vertical, maka otomatis child `LinearLayout1` dan `LinearLayout2` tersusun secara vertical atas dan bawah.

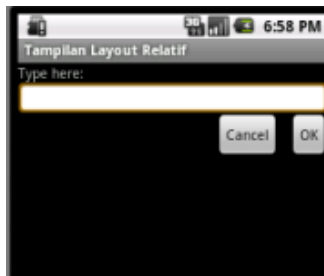
Sekarang kita perhatikan `LinearLayout1` memiliki 4 buah child `textview` berisi content "Merah","Hijau","Biru","Kuning". Orientasi `LineararLayout1` diset horizontal (baris 9), sehingga efeknya 4 `textview` susunannya berjajar dari kiri ke kanan. Masing-masing `textview` lebarnya diset `wrap_content` (baris 20) agar bisa menyesuaikan dengan panjang content didalamnya, sedangkan tingginya diset `fill_parent` sehingga efeknya tinggi masing-masing `textview` menghabiskan ruang `LinearLayout1`.

Atribut gravity pada textview (baris) mengakibatkan tulisan merah,hijau,biru dan kuning posisinya ada ditengah secara horizontal. Hal serupa juga terjadi pada LinearLayout2, hanya saja settingnya sedikit berbeda.

Nah, sekarang dibagian activity LinearLayoutSederhana, kita belum menulis kode apa-apa alias masih default. Baris 11 pada activity ini berperan memanggil layout main.xml yang baru saja kita setup.

B. RelativeLayout Sederhana

RelativeLayout adalah sebuah layout dimana posisi dari sebuah komponen (simbol,text,dsb) letaknya bisa diatur terhadap komponen lainnya. Misalnya tombol "OK" posisinya berada dibawah "EditText", kemudian tombol "cancel" posisinya berada di sebelah kiri tombol "OK" dan dibawah "EditText". Intinya, saling berkaitan antara posisi satu tombol dengan yang lain. Perhatikan Gambar 3.5.



Gambar 3.5

Sudah siap?? Yuk kita mulai!:))

1. Jalankan Enclipse, **File > New > Android Project**
2. Isikan kotak dialog new seperti berikut

Project name	RelativeLayoutSederhana
Build Target	Android 2.2
Application name	Tampilan Layout Relatif
Package name	Com.reltv
Create Activity	RelativeLayoutSederhana
Min SDK version	8

1. Pada Package Explorer, pilih **LayoutRelatif > res > layout > main.xml** kemudian tulislah script dibawah ini

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <RelativeLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
3:     android:layout_width="fill_parent"
4:     android:layout_height="fill_parent"
  android:gravity="top">
5:     <TextView
6:         android:id="@+id/label"
7:         android:layout_width="fill_parent"
8:         android:layout_height="wrap_content"
9:         android:text="Type here:"/>
10:    <EditText
11:        android:id="@+id/entry"
12:        android:layout_width="fill_parent"
13:        android:layout_height="wrap_content"
14:        android:layout_below="@id/label"/>
15:    <Button
16:        android:id="@+id/ok"
17:        android:layout_width="wrap_content"
18:        android:layout_height="wrap_content"
19:        android:layout_below="@id/entry"
20:        android:layout_alignParentRight="true"
21:        android:layout_marginLeft="10dip"
22:        android:text="OK" />
23:    <Button
24:        android:layout_alignTop="@id/ok"
25:        android:text="Cancel"
26:        android:layout_height="wrap_content"
27:        android:layout_width="wrap_content"
28:        android:layout_toLeftOf="@id/ok"
29:        android:layout_below="@id/entry"/>
30: </RelativeLayout>

```

2. Pastikan script pada **RelativeLayoutSederhana.java** seperti dibawah ini.

```

1: package contoh.layoutRelatif;
2:
3: import android.app.Activity;
4: import android.os.Bundle;
5:
6: public class RelativeLayoutSederhana extends Activity {
7:     /** Called when the activity is first created. */
8:     @Override
9:     public void onCreate(Bundle savedInstanceState) {
10:         super.onCreate(savedInstanceState);
11:         setContentView(R.layout.main);
12:     }
13: }

```

3. Lakukan Run dengan shortcut **CTRL+F11** atau klik kanan **package > run as > android Project**.
4. Lihat hasilnya seperti Gambar 3.5

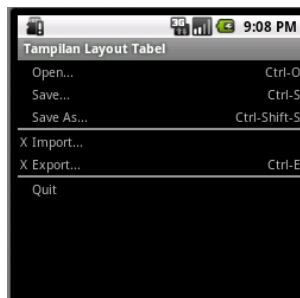
PENJELASAN PROGRAM

Kita masih focus di pembahasan layout xml, kali ini kita menggunakan widget RelativeLayout. Dengan widget ini, semua widget yang menjadi child posisinya bisa diatur secara relative terhadap komponen lainnya. Misalnya pada bagian ini, edittext ditaruh paling atas, button cancel disebelah kanan bawah edittext, sedangkan button OK dibawah editteks dan dikanan button cancel.

Masing-masing widget memiliki id yang untuk atau tidak boleh sama. Id ini sebagai acuan nama widget, mirip analoginya dengan nama manusia.

C. Table Layout

TableLayout adalah tampilan yang disusun berdasarkan baris dan kolom. Biasanya, TableLayout terdiri dari beberapa TableRow. Didalam TableRow inilah terdapat field-field yang nantinya terlihat sebagai kolom. Hasil akhir bagian ini dapat dilihat pada Gambar 3.6.



Gambar 3.6

Yuk kita mulai!:D

1. Jalankan Enclipse, **File > New > Android Project**
2. Isikan kotak dialog new seperti berikut

Project name	LayoutTabel
Build Target	Android 2.2
Application name	Tampilan Layout Tabel
Package name	contoh.layoutTabel
Create Activity	LayoutTabel
Min SDK version	8

3. Pada Package Explorer, pilih **LayoutTabel > res > layout > main.xml** lalu tuliskan xml berikut

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <TableLayout
   xmlns:android="http://schemas.android.com/apk/res/android"
3:     android:layout_width="fill_parent"
4:     android:layout_height="fill_parent"
5:     android:stretchColumns="1">
6:
7:     <TableRow>
8:         <TextView
9:             android:layout_column="1"
10:            android:text="Open..."
11:            android:padding="3dip" />
12:         <TextView
13:             android:text="Ctrl-O"
14:             android:gravity="right"
15:             android:padding="3dip" />
16:     </TableRow>
17:
18:     <TableRow>
19:         <TextView
20:             android:layout_column="1"
21:             android:text="Save..."
22:             android:padding="3dip" />
23:         <TextView
24:             android:text="Ctrl-S"
25:             android:gravity="right"
26:             android:padding="3dip" />
27:     </TableRow>
28:
29:     <TableRow>
30:         <TextView
31:             android:layout_column="1"
32:             android:text="Save As..."
33:             android:padding="3dip" />
34:         <TextView
35:             android:text="Ctrl-Shift-S"
36:             android:gravity="right"
37:             android:padding="3dip" />
38:     </TableRow>
39:
40:     <View
41:         android:layout_height="2dip"
42:         android:background="#FF909090" />
43:
44:     <TableRow>
45:         <TextView
46:             android:text="X"
47:             android:padding="3dip" />
48:         <TextView
49:             android:text="Import..."
50:             android:padding="3dip" />
51:     </TableRow>
52:

```

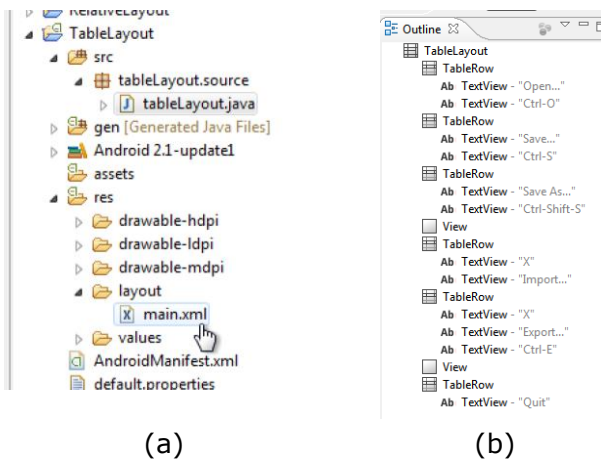


```

53:     <TableRow>
54:         <TextView
55:             android:text="X"
56:             android:padding="3dip" />
57:         <TextView
58:             android:text="Export..."
59:             android:padding="3dip" />
60:         <TextView
61:             android:text="Ctrl-E"
62:             android:gravity="right"
63:             android:padding="3dip" />
64:     </TableRow>
65:
66:     <View
67:         android:layout_height="2dip"
68:         android:background="#FF909090" />
69:
70:     <TableRow>
71:         <TextView
72:             android:layout_column="1"
73:             android:text="Quit"
74:             android:padding="3dip" />
75:     </TableRow>
76: </TableLayout>

```

Perhatikan gambar 3.7 merupakan tampilan hierarki sebuah project



Gambar 3.7 (a) Hierarki project (b) hierarki layout main.xml

4. Pastikan kode activity **LayoutTabel.java** seperti dibawah ini, lihat Gambar 3.8

Activity LayoutTabel.java

```

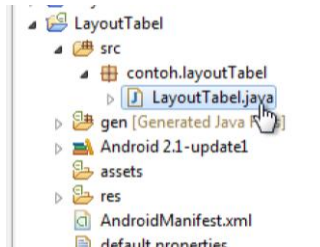
1: package contoh.layoutTabel;
2:
3: import android.app.Activity;

```

```

4: import android.os.Bundle;
5:
6: public class LayoutTabel extends Activity {
7:     /** Called when the activity is first created. */
8:     @Override
9:     public void onCreate(Bundle savedInstanceState) {
10:         super.onCreate(savedInstanceState);
11:         setContentView(R.layout.main);
12:     }
13: }

```



Gambar 3.8. activity LayoutTabel pada package

5. Lakukan Run dengan shortcut CTRL+F11 atau klik kanan package > run as > android Project.
6. Lihat hasilnya seperti Gambar 3.6.

PENJELASAN PROGRAM

Yap, saya kira sudah bisa dipahami dengan mudah mengenai layout main.xml. Sampai bagian ini kita masih bermain-main dengan xml. Layout main.xml jika dilihat dari Outlinenya tampak seperti gambar 3.7 (b).

Beberapa atribut main.xml yang perlu diperhatikan antara lain pada baris ke 5, dimana **android:stretchColumns** artinya kolom diset melebar memenuhi layar. Baris ke 11 (**android:padding**) merupakan atribut untuk membuat jarak antara content terhadap tepi kanan kiri textview .

Apabila Ebook ini bermanfaat buat Kamu, Bantu penulis menyempurnakan ebook ini dengan cara
DONASI.

JAM 4 | ANDROID DASAR

Beberapa yang akan Kamu pelajari

1. Menampilkan Gambar
2. Teks Autocomplete

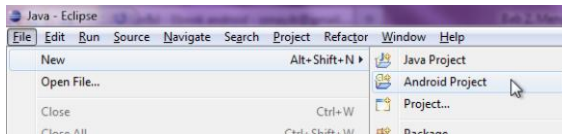
A. Menampilkan Gambar

Pada bab ini kita menampilkan gambar menggunakan widget ImageView. Seperti apa sih hasilnya nanti? Perhatikan Gambar 4.1 berikut ini



Gambar 4. 2

1. Jalankan Enclipse, buat Project baru (Gambar 4.2).

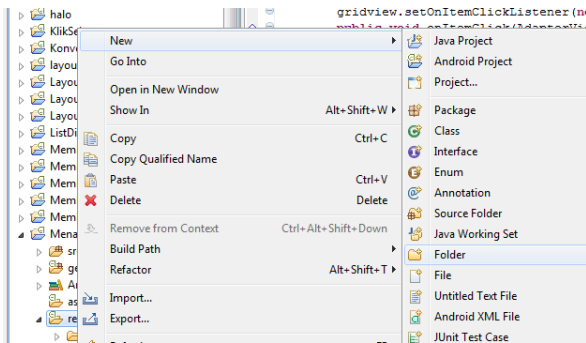


Gambar 4. 3

2. Isilah parameter seperti berikut

Project name	GambarAndroid
Build Target	Android 2.2
Application name	Menampilkan Gambar
Package name	com.gb
Create Activity	TampilanGambar
Min SDK version	8

3. Bikin folder baru dengan nama **drawable** di folder /res. Caranya klik kanan folder res > new> folder (Gambar 4.4).

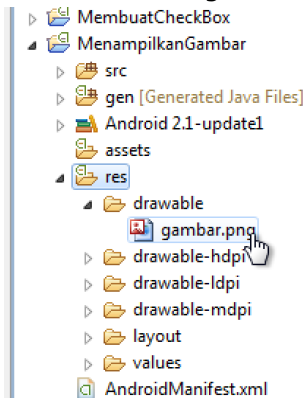


Gambar 4.4. Menambahkan folder baru

4. Copy sebuah gambar PNG atau JPG kemudian paste di dalam folder drawable. **Klik kanan folder drawable > paste.** Project ini memakai gambar format PNG berukuran (250 x 275) px (Gambar 4.6).



Gambar 4.5. file gambar.png



Gambar 4.6. Hierarki folder /res/drawable

5. Ketikkan kode berikut ini pada layout **main.xml**

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <ImageView
  xmlns:android="http://schemas.android.com/apk/res/android"
3:   android:id="@+id/imageView"
4:   android:layout_width="fill_parent"
5:   android:layout_height="fill_parent"
6:   android:src="@drawable/gambar">
7: </ImageView>

```

6. Pastikan kode pada **tampilanGambar.java** seperti berikut

```
1: package contoh.tampilanGambar;
2:
3: import android.app.Activity;
4: import android.os.Bundle;
5:
6: public class tampilanGambar extends Activity {
7:     /** Called when the activity is first created. */
8:     @Override
9:     public void onCreate(Bundle savedInstanceState) {
10:         super.onCreate(savedInstanceState);
11:         setContentView(R.layout.main);
12:     }
13: }
14:
```

7. Lakukan RUN dan lihat hasilnya. Mudah bukan??

PENJELASAN PROGRAM

Perhatikan main.xml, widget ImageView memiliki atribut **android:src="@drawable/gambar"**. Atribut ini digunakan untuk menunjuk lokasi dimana gambar disimpan, biasanya didalam folder drawable.

Dengan cara ini, kita tidak perlu mengubah barisan kode pada tampilanGambar.java karena secara *default* memanggil layout XML yang telah kita buat melalui **setContentView(R.layout.main)**.

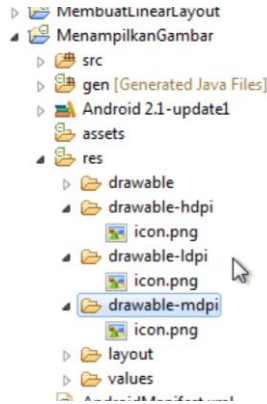
Sekarang kita akan merubah icon aplikasi sederhana ini tanpa coding. Perhatikan gambar 4.7



Gambar 4. 7. Mengganti icon launcher



Gambar 4. 8. Icon.png

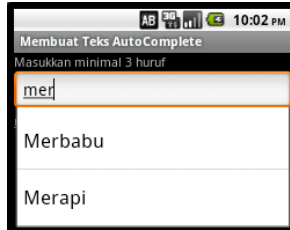


Gambar 4. 9

Siapkan gambar icon.png berukuran 70px x 72 px (gambar 4.8), kemudian paste icon.png ini ke dalam folder res/drawable-lpi. Run aplikasi, sampai disini icon sudah ganti. Kamu juga boleh memasukkan gambar ke folder res/drawable-hpi dan res/drawable-mpi. Yuph! Sangat mudah dan Kamu pasti bisa. 😊

B. MEMBUAT TEKS AUTOCOMPLETE

Sekarang kita belajar bagaimana caranya membuat teks autocomplete. Skenarionya, disediakan sebuah editteks sebagai masukan. Saat Kamu mengetik minimal 3 huruf, secara otomatis akan muncul rekomendasi kata-kata yang berawalan dari 3 huruf tadi. Lebih jelasnya perhatikan Gambar 5.1.



Gambar 5.1

Sudah siap ? mari kita mulai!

1. Jalankan Enclipse, buat project baru, isilah parameter seperti berikut

Project name	AutocompleteSederhana
Build Target	Android 2.2
Application name	Belajar Autocomplete
Package name	com.auto.comp
Create Activity	AutocompleteSederhana
Min SDK version	8

2. Tambahkan kode baris ke 5 berikut pada **String.xml**

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <resources>
3:     <string name="hello">Hello World, autoComplete!</string>
4:     <string name="app_name">Membuat Teks AutoComplete</string>
5:     <string name="perintah">Masukkan minimal 3 huruf</string>
6: </resources>

```

3. Kemudian ketikkan script berikut ini pada **main.xml**.

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
3:     android:orientation="vertical"
  android:layout_width="fill_parent"
4:     android:layout_height="fill_parent">
5:     <TextView android:layout_width="wrap_content"
6:         android:layout_height="wrap_content"
7:         android:id="@+id/textView1"
8:         android:text="@string/perintah"></TextView>
9:     <AutoCompleteTextView android:id="@+id/edit"
10:        android:layout_width="fill_parent"
11:        android:layout_height="wrap_content"
12:        android:completionThreshold="3" />
13:     <TextView android:layout_width="fill_parent"

```

```
14:         android:text="TextView"
15:         android:layout_height="wrap_content"
16:         android:id="@+id/hasil"></TextView>
17: </LinearLayout>
```

4. Ketiklah script **AutocompleteSederhana.java** seperti berikut

```
1: package com.auto.comp;
2:
3: import android.app.Activity;
4: import android.os.Bundle;
5: import android.text.Editable;
6: import android.text.TextWatcher;
7: import android.widget.ArrayAdapter;
8: import android.widget.AutoCompleteTextView;
9: import android.widget.TextView;
10:
11: public class AutocompleteSederhana extends Activity implements
    TextWatcher {
12:     /** Called when the activity is first created. */
13:     TextView hasil;
14:     AutoCompleteTextView edit;
15:     String[] item = { "Merbabu", "Merapi", "Lawu", "Rinjani",
16:         "Sumbing", "Sindoro", "Krakatau", "Selat Sunda", "Selat
17:         Bali", "Selat Malaka", "Kalimantan", "Sulawesi", "Jawa" };
18:
19:     @Override
20:     public void onCreate(Bundle savedInstanceState) {
21:         super.onCreate(savedInstanceState);
22:         setContentView(R.layout.main);
23:
24:         hasil = (TextView) findViewById(R.id.hasil);
25:         edit = (AutoCompleteTextView) findViewById(R.id.edit);
26:         edit.addTextChangedListener(this);
27:         edit.setAdapter(new ArrayAdapter<String>(this,
28:             android.R.layout.simple_dropdown_item_1line, item));
29:     }
30:
31:     public void onTextChanged(CharSequence s, int start, int
32:         before, int count) {
33:         hasil.setText(edit.getText());
34:     }
35:
36:     public void beforeTextChanged(CharSequence s, int start, int
37:         count, int after) {
38:         // not used
39:     }
40:
41:     public void afterTextChanged(Editable s) {
42:         // not used
43:     }
44: }
```

Bila script berantakan, lakukan Format (**source > format**).
Lakukan RUN dan lihat hasilnya.

PENJELASAN PROGRAM

Lihat dulu Main.xml khususnya baris 9-10, merupakan bagian yang menciptakan widget autocomplete. Widget ini memiliki beberapa atribut salah satu contohnya adalah baris 12. Atribut pada baris 12 bernilai 3, berfungsi untuk membatasi jumlah minimal huruf yang nanti dimasukkan. Dengan demikian, seorang user minimal harus memasukkan 3 huruf dulu baru kemudian muncul kata-kata rekomendasi.

Sekarang kita focus ke activity AutocompleteSederhana. Saya bagi menjadi tiga blok

1. **Baris 13-17** : deklarasi object textview, string dan autocompleteTextView
2. **Baris 24-28** : sinkronisasi object terhadap widget di layout xml berdasarkan id, sekaligus menempelkan method pada object edit.
3. **Baris 31-43** : implementasi method TextWatcher

Coba perhatikan dulu baris 14, object 'edit' adalah turunan dari class autoCompleteTextView. Object ini merupakan textview yang bisa diedit kemudian muncul daftar rekomendasi kata secara otomatis saat proses mengetik berlangsung. Daftar rekomendasi kata ditampilkan dalam sebuah drop down menu yang bisa diklik oleh pengguna.

Baris 26, mengaktifkan object 'edit' supaya bisa menangkap sinyal ketika user memasukkan sebuah huruf. Sinyal ini kemudian direspon oleh method-method pada baris 31-43 yaitu

- **afterTextChanged()**
method ini dipanggil setelah edittext selesai diubah
- **beforeTextChanged()**
method ini dipanggil sebelum edittext diubah
- **onTextChanged()**
method ini dipanggil saat text pada edittext sedang diubah

Baris 28 , menempelkan data array bernama 'item' ke object 'edit'. Bagian ini mengakibatkan kata-kata yang disimpan dalam variable item muncul otomatis saat user memasukkan huruf. Baris 33 berperan menduplikasi data, yaitu huruf apa saja yang dimasukkan ke dalam object 'edit' maka tampil juga di object 'hasil'.

Apabila Ebook ini bermanfaat buat Kamu, Bantu penulis menyempurnakan ebook ini dengan cara **DONASI.**

**Kirim KRITIK, SARAN, dan
TESTIMONI Kamu di
www.odayib.com/testimoni**

Ingin mengadakan Workshop/seminar di Kampus
Kamu?? Hubungi
ImagineIT Education Center Yogyakarta

JAM 5 | ANDROID DASAR

Beberapa yang akan Kamu pelajari

1. Kotak Dialog
2. Animasi 2D (not yet available)

A. Membuat Kotak Dialog

Dalam sebuah aplikasi, kotak dialog merupakan sarana interaktif antara user dengan aplikasi itu sendiri. Misalnya jika kita ingin keluar dari aplikasi, biasanya muncul peringatan berisi pertanyaan "Are you sure to quit?" dan tersedia tombol "yes" dan "cancel". Pada bagian ini, kita mengimplementasikan 3 macam dialog yaitu

- Toast
- ListDialog
- AlertDialog

Hasil akhir project ini terlihat seperti gambar 5.2. Oks, langsung kita praktikkan yaap?!



Gambar 5.2. Hasil akhir project Kotak Dialog

1. Bikin sebuah project baru

Project name	DialogBox
Build Target	Android 2.2
Application name	Kotak Dialog
Package name	com.dialog
Create Activity	KotakDialog
Min SDK version	8

2. Tambahkan 3 string element pada **Strings.xml**

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <resources>
3:     <string name="hello">Hello World, kotakDialog!</string>

```

```

4:     <string name="app_name">Membuat Kotak Dialog</string>
5:     <string name="toastTextBtn">Panggil Toast</string>
6:     <string name="exitTextBtn">Keluar Aplikasi</string>
7:     <string name="listDialogTextBtn">Panggil List
    Dialog</string>
8: </resources>

```

3. Bikin dulu layoutnya melalui **Main.xml**

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
3:     android:orientation="vertical"
4:     android:layout_width="fill_parent"
5:     android:layout_height="fill_parent">
6:
7:     <Button android:layout_width="fill_parent"
8:         android:id="@+id/toastBtn"
9:         android:layout_height="wrap_content"
10:        android:text="@string/toastTextBtn"></Button>
11:
12:    <Button android:layout_width="fill_parent"
13:        android:layout_height="wrap_content"
14:        android:id="@+id/listDialogBtn"
15:        android:text="@string/listDialogTextBtn"></Button>
16:
17:    <Button android:layout_width="fill_parent"
18:        android:layout_height="wrap_content"
19:        android:text="@string/exitTextBtn"
20:        android:id="@+id/exitBtn"></Button>
21: </LinearLayout>

```

4. Okey, sekarang saatnya menulis kode di activity **KotakDialog.java**

```

1: package com.dialog;
2:
3: import android.app.Activity;
4: import android.app.AlertDialog;
5: import android.content.DialogInterface;
6: import android.os.Bundle;
7: import android.widget.Button;
8: import android.widget.Toast;
9: import android.view.View;
10:
11: public class KotakDialog extends Activity implements
12:     OnClickListener {
13:     Button pesanToast;
14:     Button keluar;
15:     Button tampilList;
16:     /** Called when the activity is first created. */
17:     @Override
18:     public void onCreate(Bundle savedInstanceState) {
19:         super.onCreate(savedInstanceState);
20:         setContentView(R.layout.main);
21:
22:         pesanToast = (Button) findViewById(R.id.toastBtn);
23:         pesanToast.setOnClickListener(this);

```

```

24:
25:         keluar = (Button) findViewById(R.id.exitBtn);
26:         keluar.setOnClickListener(this);
27:
28:         tampilList = (Button)
29:             findViewById(R.id.listDialogBtn);
30:         tampilList.setOnClickListener(this);
31:     }
32:
33:     public void onClick(View clicked) {
34:         switch (clicked.getId()) {
35:             case R.id.listDialogBtn:
36:                 munculListDialog();
37:                 break;
38:             case R.id.toastBtn:
39:                 Toast.makeText(this, "Kamu memilih Toast",
40:                     Toast.LENGTH_SHORT).show();
41:                 break;
42:             case R.id.exitBtn:
43:                 exit();
44:                 break;
45:         }
46:     }
47:
48:     private void munculListDialog() {
49:         // TODO Auto-generated method stub
50:         final CharSequence[] items = { "Es Teh", "Es Jeruk",
51:             "Lemon Squash", "Soft Drink" };
52:
53:         AlertDialog.Builder kk = new AlertDialog.Builder(this);
54:         kk.setTitle("Pilih Minuman");
55:         kk.setItems(items, new DialogInterface.OnClickListener() {
56:             public void onClick(DialogInterface dialog, int item) {
57:                 Toast.makeText(getApplicationContext(), items[item],
58:                     Toast.LENGTH_SHORT).show();
59:             }
60:         }).show();
61:     }
62:
63:     private void exit() {
64:         AlertDialog.Builder builder = new AlertDialog.Builder(this);
65:         builder.setMessage("Apakah Kamu Benar-Benar ingin keluar?")
66:             .setCancelable(false)
67:             .setPositiveButton("Ya", new
68:                 DialogInterface.OnClickListener() {
69:                     public void onClick(DialogInterface dialog,
70:                         int id) {
71:                         KotakDialog.this.finish();
72:                     }
73:                 })
74:             .setNegativeButton("Tidak", new
75:                 DialogInterface.OnClickListener() {
76:                     @Override
77:                     public void onClick(DialogInterface dialog,
78:                         int arg1) {
79:                         // TODO Auto-generated method stub
80:                         dialog.cancel();
81:                     }

```

```
82:         })).show();
83:     }
84: }
```

Nah, sudah selesai. Silakan dirun , semoga tidak error ☺

PEMBAHASAN PROGRAM

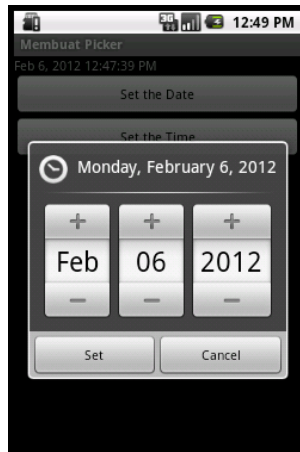
Perhatikan KotakDialog.java, saya bagi menjadi 5 blok yang memiliki peran sendiri-sendiri

1. **Baris 12-14** : deklarasi object Button.
2. **Baris 19-30** : sinkronisasi object terhadap widget button pada layout main.xml sekaligus mengaktifkan button.
3. **Baris 33-45** : fungsi pada masing-masing button.
4. **Baris 48-61** : fungsi yang dipanggil ketika button 'panggil list dialog' diklik.
5. **Baris 63-83** : fungsi yang dipanggil ketika button 'keluar aplikasi' diklik.

Activity ini mengimplementasikan class OnClickListener (baris 11-12). Class ini memiliki sebuah method onClick (baris 33-45) yang bertugas merespon button pada saat diklik. Didalam method onClick(), button direspon berdasarkan idnya melalui perulangan switch-case. Jika button pemilik id 'listDialogBtn' diklik maka kemudian memanggil fungsi munculListDialog() (baris 58-61), jika button ber-id 'toastBtn' diklik, maka activity mengenerate sebuah toast, sedangkan jika yang diklik adalah button ber-id 'exitBtn' maka yang dieksekusi adalah fungsi exit() (baris 63-83).

B. PICKER

Dalam android, picker sering dimanfaatkan untuk melakukan setting tanggal atau jam. Lihat gambar 5.3.



Gambar 5.3. Picker untuk setting tanggal

1. Bikin project baru lagi, dengan parameter seperti berikut

Project name	Picker
Build Target	Android 2.2
Application name	Membuat Picker
Package name	com.picker
Create Activity	picker
Min SDK version	8

2. Kita buat dulu bagian layout **Main.xml**

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
3:   android:orientation="vertical"
4:   android:layout_width="fill_parent"
5:   android:layout_height="fill_parent">
6:   <TextView android:layout_width="fill_parent"
7:     android:layout_height="wrap_content"
8:     android:text="@string/hello"
9:     android:id="@+id/dateAndTime"/>
10:  <Button android:text="Set the Date"
11:    android:layout_width="fill_parent"
12:    android:layout_height="wrap_content"
13:    android:id="@+id/dayBtn"></Button>
14:  <Button android:text="Set the Time"
15:    android:layout_width="fill_parent"
16:    android:layout_height="wrap_content"
17:    android:id="@+id/timeBtn"></Button>

```

```
18: </LinearLayout>
19:
```

3. Berikutnya kita sedikit banyak koding di activity **picker.java**

```
1: package com.picker;
2:
3: import java.text.DateFormat;
4: import java.util.Calendar;
5:
6: import android.app.Activity;
7: import android.app.DatePickerDialog;
8: import android.app.TimePickerDialog;
9: import android.os.Bundle;
10: import android.view.View;
11: import android.view.View.OnClickListener;
12: import android.widget.Button;
13: import android.widget.DatePicker;
14: import android.widget.TextView;
15: import android.widget.TimePicker;
16:
17: public class picker extends Activity
18:     implements OnClickListener {
19:
20:     DateFormat fmtDateAndTime =
21:         DateFormat.getDateTimeInstance();
22:     TextView dateAndTimeLabel;
23:     Calendar dateAndTime = Calendar.getInstance();
24:     DatePickerDialog.OnDateSetListener d =
25:         new DatePickerDialog.OnDateSetListener() {
26:             @Override
27:             public void onDateSet(DatePicker view, int year, int month,
28:                 int day) {
29:                 // TODO Auto-generated method stub
30:                 dateAndTime.set(Calendar.YEAR, year);
31:                 dateAndTime.set(Calendar.MONTH, month);
32:                 dateAndTime.set(Calendar.DAY_OF_MONTH, day);
33:                 updateLabel();
34:             }
35:         };
36:     TimePickerDialog.OnTimeSetListener t =
37:         new TimePickerDialog.OnTimeSetListener() {
38:             @Override
39:             public void onTimeSet(TimePicker view, int jam, int menit) {
40:                 // TODO Auto-generated method stub
41:                 dateAndTime.set(Calendar.HOUR_OF_DAY, jam);
42:                 dateAndTime.set(Calendar.MINUTE, menit);
43:                 updateLabel();
44:             }
45:         };
46:
47:     /** Called when the activity is first created. */
48:     @Override
49:     public void onCreate(Bundle savedInstanceState) {
50:         super.onCreate(savedInstanceState);
51:         setContentView(R.layout.main);
```



```

52:
53:     Button dayBtn = (Button) findViewById(R.id.dayBtn);
54:     dayBtn.setOnClickListener(this);
55:     Button timeBtn = (Button) findViewById(R.id.timeBtn);
56:     timeBtn.setOnClickListener(this);
57:     dateAndTimeLabel = (TextView)
58:         findViewById(R.id.dateAndTime);
59:     updateLabel();
60: }
61:
62: @Override
63: public void onClick(View v) {
64:     // TODO Auto-generated method stub
65:     switch (v.getId()) {
66:         case R.id.dayBtn:
67:             settingTanggal();
68:             break;
69:         case R.id.timeBtn:
70:             setJam();
71:             break;
72:     }
73: }
74:
75: private void updateLabel() {
76:     dateAndTimeLabel.setText(
77:         fmtDateAndTime.format(dateAndTime.getTime()));
78: }
79:
80: private void settingTanggal() {
81:     new DatePickerDialog(picker.this, d,
82:         dateAndTime.get(Calendar.YEAR),
83:         dateAndTime.get(Calendar.MONTH),
84:         dateAndTime.get(Calendar.DAY_OF_MONTH)).show();
85: }
86:
87: private void setJam() {
88:     new TimePickerDialog(picker.this, t,
89:         dateAndTime.get(Calendar.HOUR_OF_DAY),
90:         dateAndTime.get(Calendar.MINUTE), true).show();
91: }
92: }

```

PENJELASAN PROGRAM

Kita mulai dari layout main.xml, di bagian ini Kamu menyiapkan 2 buah button dan sebuah textview. Button pertama diberi id 'dayBtn' (baris 13) nanti difungsikan untuk memanggil picker tanggal, sedangkan button kedua diberi id 'timeBtn' (baris 17) nanti dipakai untuk memanggil picker jam. Ok, sangat simple! ☺

Sekarang lanjut ke activity picker.java, Kamu harus mulai berkonsentrasi. Hehehe.. Ok, DatePickerDialog.onDateSetListener (baris 24) berperan menangkap informasi setelah user selesai melakukan setting

waktu. Pada saat yang bersamaan, method `onDataSet()` dipanggil untuk mengupdate Tahun, bulan, hari, tanggal maupun jam, keudian ditampilkan ke textview melalui method `updateLabel()`.

**Kirim KRITIK, SARAN, dan
TESTIMONI Kamu di
www.odayib.com/testimoni**

BAB 3

DASAR ANDROID #2

JAM 6 | ANDROID WIDGET

Bagian ini akan membahas mengenai

- ✓ CheckBox Widget
- ✓ RadioButton
- ✓ ScrollView

A. MEMBUAT CHECK BOX

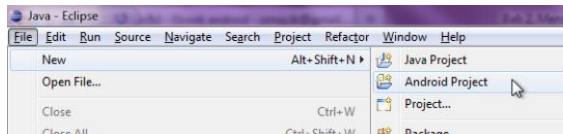
Checkbox adalah salah satu widget berjenis optional, artinya 2 atau lebih item bisa dipilih secara bersamaan. Kita coba dulu dari yang paling sederhana, skenarionya disediakan sebuah checkbox yang belum aktif, ditKamui dengan tulisan "Chackbox ini : Belum dicentang!". Kemudian jika user mengklik, checkbox berubah warna menjadi hijau dan tulisannya ganti "Checkbox ini : dicentang!". Perhatikan Gambar 6.1



Gambar 6. 1

Sebelum mulai siapkan es Teh dan snack disamping komputer, supaya belajarnya lebih menyenangkan. ☺. Nah saatnya dimulai.

1. Jalankan Enclipse, buat Project baru.



Gambar 6. 2. Membuat new Project

2. Isilah parameter seperti berikut

Project name	MembuatCheckBox
Build Target	Android 2.2
Application name	Membuat Check Box
Package name	contoh.checkBox
Create Activity	checkBox
Min SDK version	8

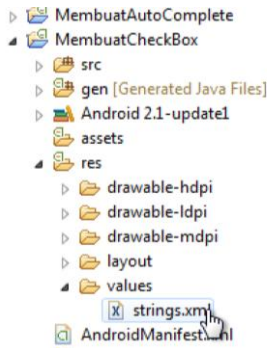
3. Perhatikan kode pada **String.xml** lihat pada res/values/string.xml (Gambar 6.3). Tambahkan kode menjadi seperti berikut

```
1: <?xml version="1.0" encoding="utf-8"?>
2: <resources>
```

```

3:     <string name="hello">Hello World, checkBox!</string>
4:     <string name="app_name">Membuat Check Box</string>
5:     <string name="checkBox">checkBox ini : Tidak
    Dicentang!</string>
6: </resources>
7:

```



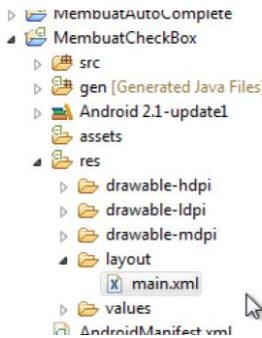
Gambar 6. 3. Letak strings.xml

4. Kemudian ketikkan kode berikut ini pada **main.xml** (Gambar 6.4).

```

8: <?xml version="1.0" encoding="utf-8"?>
9: <CheckBox
    xmlns:android="http://schemas.android.com/apk/res/android"
10:     android:layout width="wrap_content"
11:     android:id="@+id/check"
12:     android:layout_height="wrap_content"
13:     android:text="@string/checkBox">
14: </CheckBox>

```



Gambar 6. 4. Letak main.xml

5. Tuliskan kode **checkBox.java** seperti berikut.

```

1: package contoh.checkBox;
2:
3: import android.app.Activity;
4: import android.os.Bundle;
5: import android.widget.CheckBox;
6: import android.widget.CompoundButton;
7: import android.widget.CompoundButton.OnCheckedChangeListener;
8:
9: public class checkBox extends Activity implements

```

```

OnCheckedChangeListener {
10:   CheckBox cb;
11:   /** Called when the activity is first created. */
12:   @Override
13:   public void onCreate(Bundle savedInstanceState) {
14:       super.onCreate(savedInstanceState);
15:       setContentView(R.layout.main);
16:
17:       cb=(CheckBox) findViewById(R.id.check);
18:       cb.setOnCheckedChangeListener(this);
19:
20:   }
21:   public void onCheckedChanged(CompoundButton buttonView,
22:       boolean isChecked) {
23:       if (isChecked) {
24:           cb.setText("checkBox ini : Dicentang!");
25:       }
26:       else {
27:           cb.setText("checkBox ini : Tidak Dicentang!");
28:       }
29:   }
30: }

```

6. Bila kode berantakan, lakukan Format (**source > format**).
7. Lakukan RUN dan lihat hasilnya. Simple!

PENJELASAN PROGRAM

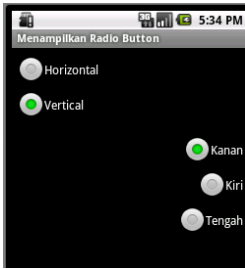
Perhatikan dulu main.xml, disini kita memakai widget checkbox yang diberi id 'check' (lihat baris 4). Widget checkbox memiliki property text, isinya mengambil dari string 'checkBox' pada strings.xml (baris 6).

Widget CheckBox memiliki 2 status yaitu checked dan unchecked. Dengan mengeklik checkBox artinya kita mengubah status dari checked menjadi unchecked atau sebaliknya.

Sekarang kita lihat activity checkbox.java, baris 17 adalah sinkronisasi antara object 'cb' terhadap widget CheckBox dari layout main.xml, sedangkan baris 18 untuk mengaktifkan widget checkbox agar berfungsi saat diklik. Activity ini mengimplementasikan class OnCheckedChangeListener, agar sistem bisa mengenali apakah checkbox diklik atau tidak. Aktivitas ini di baca oleh method onCheckedChanged() pada baris 21-29. Method ini lah yang bertugas mengganti tulisan pada widget ketika checkbox diklik.

B. RADIOBUTTON

Pada bab ini, kita akan membuat aplikasi sederhana menggunakan RadioButton. Skenarionya ada 5 buah RadioButton masing-masing diberi teks Horizontal, Vertical, Kanan, Tengah, dan Kiri. Jika salah satu RadioButton dipilih, maka susunan RadioButton akan berubah sesuai pilihannya.



(a)

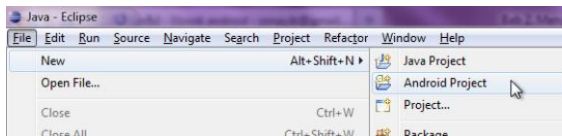


(b)

Gambar 6.5. (a) Kombinasi tampilan secara vertivcal dengan rata kanan (b) kombinasi tampilan horizontal dengan center

Penasaran? Yuk, kita latihan..

1. Jalankan Enclipse, buat Project baru (gambar 6.3).



Gambar 6. 6

2. Isilah parameter seperti berikut

Project name	RadioButton
Build Target	Android 2.2
Application name	Menampilkan Radio Button
Package name	contoh.RadioButton
Create Activity	RadioButton
Min SDK version	8

3. Kemudian ketikkan kode berikut ini pada **main.xml** (Gambar 6.4).

```

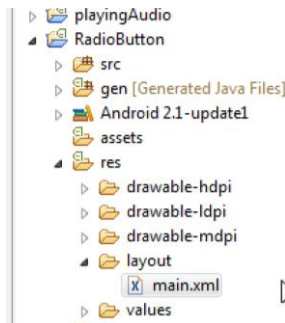
1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
3: xmlns:android="http://schemas.android.com/apk/res/android"
4:     android:layout_height="fill_parent"
5:     android:layout_width="fill_parent"
6:     android:orientation="vertical">
7:     <RadioGroup android:padding="5px"
8:         android:layout_height="wrap_content"
9:         android:layout_width="wrap_content"
10:         android:id="@+id/orientation"
11:         android:orientation="horizontal">
12:         <RadioButton android:layout_height="wrap_content"
13:             android:layout_width="wrap_content"
14:             android:id="@+id/horizontal"
15:             android:text="Horizontal">
16:         </RadioButton>
17:         <RadioButton android:layout_height="wrap_content"
18:             android:layout_width="wrap_content"
19:             android:id="@+id/vertical"
20:             android:text="Vertical">
21:         </RadioButton>
22:     </RadioGroup>

```

```

23:         <RadioGroup android:id="@+id/gravity"
24:             android:orientation="vertical"
25:             android:padding="5px"
26:             android:layout_width="fill_parent"
27:             android:layout_height="wrap_content">
28:             <RadioButton
29:                 android:id="@+id/kanan"
30:                 android:text="Kanan">
31:             </RadioButton>
32:             <RadioButton
33:                 android:text="Kiri"
34:                 android:id="@+id/kiri">
35:             </RadioButton>
36:             <RadioButton
37:                 android:id="@+id/tengah"
38:                 android:text="Tengah">
39:             </RadioButton>
40:         </RadioGroup>
41:     </LinearLayout>
42:

```



Gambar 6. 1. Letak layout main.xml dalam package

4. Ketiklah kode **RadioButton.java** seperti berikut (Gambar 6.5).

```

1: package contoh.radioButton;
2:
3: import android.app.Activity;
4: import android.os.Bundle;
5: import android.view.Gravity;
6: import android.widget.LinearLayout;
7: import android.widget.RadioGroup;
8:
9: public class radioButton extends Activity implements
10:     RadioGroup.OnCheckedChangeListener {
11:     /** Called when the activity is first created. */
12:
13:     RadioGroup orientation;
14:     RadioGroup gravity;
15:
16:     @Override
17:     public void onCreate(Bundle savedInstanceState) {
18:         super.onCreate(savedInstanceState);
19:         setContentView(R.layout.main);
20:
21:         orientation = (RadioGroup) findViewById(R.id.orientation);
22:         orientation.setOnCheckedChangeListener(this);
23:         gravity = (RadioGroup) findViewById(R.id.gravity);
24:         gravity.setOnCheckedChangeListener(this);
25:

```



```

26:     }
27:
28:     public void onCheckedChanged(RadioGroup group,
29:         int checkId) {
30:         switch (checkId) {
31:             case R.id.horizontal:
32:                 orientation.setOrientation(LinearLayout.HORIZONTAL);
33:                 break;
34:             case R.id.vertical:
35:                 orientation.setOrientation(LinearLayout.VERTICAL);
36:                 break;
37:             case R.id.kiri:
38:                 gravity.setGravity(Gravity.LEFT);
39:                 break;
40:             case R.id.tengah:
41:                 gravity.setGravity(Gravity.CENTER);
42:                 break;
43:             case R.id.kanan:
44:                 gravity.setGravity(Gravity.RIGHT);
45:                 break;
46:         }
47:     }
48: }

```



Gambar 6. 2. Activity radioButton.java dalam package

5. Bila kode berantakan, lakukan Format **(source > format)**.
6. Lakukan RUN dan lihat hasilnya. Yah! Sangat mudah.

PENJELASAN PROGRAM

Kita mulai dari layout main.xml. RadioGroup berbeda dengan RadioButton. RadioGroup adalah kumpulan beberapa RadioButton. Dalam layout ini, kita mempunyai 2 buah RadioGroup

1. **Baris 7-22** : RadioGroup pertama dengan id 'orientation'. Memiliki 2 radioButton masing-masing ber-id 'Horizontal' dan 'vertical'.
2. **Baris 23-40**: RadioGroup kedua dengan id 'gravity'. Mempunyai 3 radioButton masing-masing ber-id 'left', 'center', 'right'.

Sekarang kita pindah ke activity radioButton. Activity ini mengimplementasikan class `RadioGroup.OnCheckedChangeListener`, agar supaya sistem dapat mengetahui bahwa ada radioButton yang diklik. Untuk memudahkan pemahaman, saya bagi menjadi 3 bagian

1. **Baris 13-14** :Deklarasi object RadioGroup dan RadioButton
2. **Baris 21-24** :Sinkronisasi object yang sudah di deklarasikan terhadap widget pada main.xml, sekaligus mengaktifkan radioButton agar berfungsi saat diklik.
3. **Baris 28-47** :fungsi yang akan dieksekusi jika salah satu radioButton diklik

Selain melalui xml, ada cara lain untuk menentukan orientasi sebuah layout, yakni melalui method `setOrientation()` seperti pada Baris 32 dan 35. Begitu juga dengan gravity sebuah object, bisa juga diatur dari java melalui method `setGravity()`.

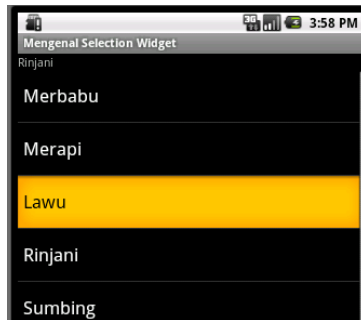
JAM 7 | ANDROID WIDGET

Bagian ini akan membahas mengenai

- ✓ SelectionWidget
- ✓ Menu

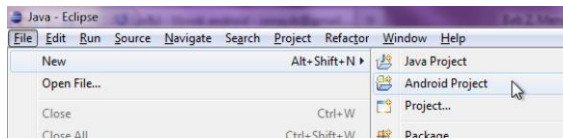
A. MENGGUNAKAN LIST VIEW

Apa list view widget itu? Dengan widget ini, kamu bisa membuat sebuah daftar berisi banyak item. Dua item atau lebih bisa dipilih bersamaan tergantung pada jenis list view widget yang kita gunakan.



Gambar 7.1. Selection widget

1. Jalankan Eclipse, buat Project baru.



Gambar 7.2. New Project

2. Isilah parameter seperti berikut

Project name	MengenalSelectionWidget
Build Target	Android 2.2
Application name	Mengenal Slection Widget
Package name	contoh.seleksi
Create Activity	seleksi
Min SDK version	8

3. Kemudian ketikkan script berikut ini pada **main.xml**

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
3:   xmlns:android="http://schemas.android.com/apk/res/android"
4:   android:orientation="vertical"
5:   android:layout_width="fill_parent"
6:   android:layout_height="fill_parent" >
7:   <TextView

```

```

8: android:id="@+id/yangDipilih"
9: android:layout_width="fill_parent"
10: android:layout_height="wrap_content" />
11: <ListView
12: android:id="@android:id/list"
13: android:layout_width="fill_parent"
14: android:layout_height="fill_parent"
15: android:drawSelectorOnTop="false"
16: />
17: </LinearLayout>

```

4. Ketiklah script **seleksi.java** seperti berikut

```

1: package contoh.seleksi;
2:
3: import android.app.ListActivity;
4: import android.os.Bundle;
5: import android.view.View;
6: import android.widget.ArrayAdapter;
7: import android.widget.ListView;
8: import android.widget.TextView;
9:
10: public class seleksi extends ListActivity {
11:     /** Called when the activity is first created. */
12:     TextView seleksi;
13:     String[] pilihan = {
14:         "Merbabu", "Merapi", "Lawu", "Rinjani",
15:         "Sumbing", "Sindoro", "Krakatau", "Selat Sunda",
16:         "Selat Bali", "Selat Malaka", "Kalimantan",
17:         "Sulawesi", "Jawa" };
18:     @Override
19:     public void onCreate(Bundle savedInstanceState) {
20:         super.onCreate(savedInstanceState);
21:         setContentView(R.layout.main);
22:         setListAdapter(new ArrayAdapter<String>
23:             (this, android.R.layout.simple_list_item_1, pilihan));
24:         seleksi = (TextView) findViewById(R.id.yangDipilih);
25:     }
26:
27:     public void onItemClick(ListView parent, View v,
28:         int position, long id) {
29:         seleksi.setText(pilihan[position]);
30:     }
31: }

```

5. Bila script berantakan, lakukan Format (**source > format**).

6. Lakukan RUN dan lihat hasilnya.

PENJELASAN PROGRAM

Baik, kita mulai dari layout main.xml. Kamu hanya perlu menaruh sebuah TextView (baris 7-10) untuk menampilkan apa saja yang Kamu klik dan sebuah ListView (baris 11-15) untuk menampilkan item apa saja yang dapat diklik. Dua widget diatas, dirangkai dalam satu LinearLayout.

Kita beralaih ke activity, baris 12 merupakan bagian pendeklarasian objek dan pada baris 13-17 kita membuat object jenis array bertipe string isinya adalah nama-nama

gunung dan selat di Indonesia. Method `setListAdapter()` pada baris 22 dapat dianalogikan seperti jembatan penghubung antara `ListView` pada layout xml dengan item array. Baris 23 apabila `simple_list_item_1` diganti dengan `simple_list_single_choice` maka hasilnya terlihat seperti Gambar 6.10(a), sedangkan bila diganti dengan `simple_list_multiple_choice` hasilnya seperti gambar 6.10(b).



(a)



(b)

Gambar 7.3.(a) `ListView` dengan `simple_list_single_choice`, b) `ListView` dengan `simple_list_multiple_choice`

Kami menerima kerjasama untuk mengadakan workshop, seminar, atau pelatihan IT. Jika berminat bisa menghubungi omayib@gmail.com

B. Menu

<Not YET AVAILABEL>

Apabila Ebook ini bermanfaat buat Kamu, Bantu penulis menyempurnakan ebook ini dengan cara **DONASI.**

JAM 8 | ANDROID INTENT

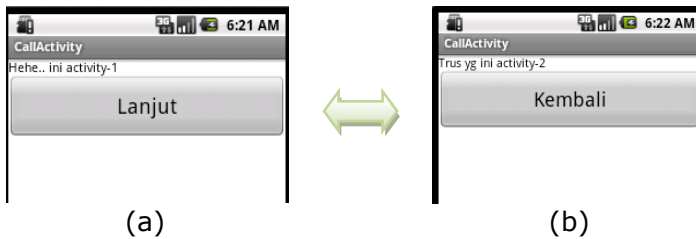
Bagian ini akan membahas mengenai

- ✓ Intent untuk memanggil activity

A. Memanggil activity

Inti dari sebuah aplikasi sebenarnya ada 3 yaitu activity, service dan broadcast. Intent sendiri digunakan untuk memanggil activity, memanggil service atau melakukan broadcast.

Bagian ini kita akan memahami intent dari yang paling sederhana yaitu memanggil activity lain. Skenarionya, kita punya 2 layout dan 2 activity yang saling berpasangan. Pada activity yang pertama disediakan sebuah button, jika diklik lalu pindah ke activity kedua. Pada activity kedua ini juga disediakan button, jika diklik maka activity kedua akan destroy (close) lalu activity pertama muncul kembali. Perhatikan gambar 8.1 berikut.



Gambar 8.1. (a) Tampilan Activity pertama, (b) Tampilan Activity kedua

Nah, sudah tidak sabar kan??? Yuk kita mulai...

1. Bikin project baru dulu ya
2. Kita bikin layout pertama dulu menggunakan **main.xml**

```
1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
3:   android:orientation="vertical"
4:   android:layout_width="fill_parent"
5:   android:layout_height="fill_parent"
6:   android:background="#ffffff">
7:   <TextView android:layout_width="fill_parent"
8:     android:layout_height="wrap_content"
9:     android:textColor="#000000"
10:    android:text="Hehe.. ini activity-1" />
11:   <Button android:text="Lanjut"
12:     android:id="@+id/Button01"
13:     android:layout_width="fill_parent"
14:     android:textSize="18px"
15:     android:layout_height="55px">
16:   </Button>
17: </LinearLayout>
```

3. Kita bikin xml baru diberi nama **main2.xml**

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

```

3:     android:orientation="vertical"
4:     android:layout_width="fill_parent"
5:     android:layout_height="fill_parent"
6:     android:background="#ffffff">
7:     <TextView android:layout_width="fill_parent"
8:         android:layout_height="wrap_content"
9:         android:textColor="#000000"
10:        android:text="Trus yg ini activity-2" />
11:     <Button android:text="Kembali"
12:        android:id="@+id/Button02"
13:        android:layout_width="fill_parent"
14:        android:textSize="18px"
15:        android:layout_height="55px">
16:     </Button>
17: </LinearLayout>

```

4. Sekarang kita coding untuk activity pertama **CallActivity.java**

```

1: package cnt.CallActivity;
2:
3: import android.app.Activity;
4: import android.content.Intent;
5: import android.os.Bundle;
6: import android.view.View;
7: import android.widget.Button;
8:
9: public class CallActivity extends Activity {
10:     /** Called when the activity is first created. */
11:     @Override
12:     public void onCreate(Bundle savedInstanceState) {
13:         super.onCreate(savedInstanceState);
14:         setContentView(R.layout.main);
15:
16:         Button next = (Button) findViewById(R.id.Button01);
17:         next.setOnClickListener(new View.OnClickListener() {
18:
19:             public void onClick(View bebek) {
20:                 Intent myIntent = new
21:                     Intent(bebek.getContext(), CallActivity2.class);
22:                 startActivity(myIntent, 0);
23:             }
24:
25:         });
26:     }
27: }

```

5. Buat class baru beri nama **CallActivity2.java**. Klik kanan package>new>class.

```

1: package cnt.CallActivity;
2:
3: import android.app.Activity;
4: import android.content.Intent;
5: import android.os.Bundle;
6: import android.view.View;
7: import android.widget.Button;
8:
9: public class CallActivity2 extends Activity {
10:     /** Called when the activity is first created. */
11:     @Override
12:     public void onCreate(Bundle savedInstanceState) {
13:         super.onCreate(savedInstanceState);
14:         setContentView(R.layout.main2);
15:
16:         Button prev = (Button) findViewById(R.id.Button02);
17:         prev.setOnClickListener(new View.OnClickListener() {

```



```

18:         public void onClick(View view) {
19:             Intent intent = new Intent();
20:             setResult(RESULT_OK, intent);
21:             finish();
22:         }
23:
24:     });
25: }
26: }
    
```

6. Sekarang perhatikan **AndroidManifest.xml**, daftarkan dulu activity kedua tadi disini

```

27: <?xml version="1.0" encoding="utf-8"?>
28: <manifest
29:     xmlns:android="http://schemas.android.com/apk/res/android"
30:     package="cnt.CallActivity"
31:     android:versionCode="1"
32:     android:versionName="1.0">
33:     <uses-sdk android:minSdkVersion="7" />
34:     <application android:icon="@drawable/icon"
35:         android:label="@string/app_name">
36:         <activity android:name=".CallActivity"
37:             android:label="@string/app_name">
38:             <intent-filter>
39:                 <action android:name="android.intent.action.MAIN" />
40:                 <category
41:                     android:name="android.intent.category.LAUNCHER" />
42:             </intent-filter>
43:         </activity>
44:         <activity android:name="CallActivity2">
45:             <intent-filter>
46:                 <action android:name="android.intent.action.MAIN2" />
47:             </intent-filter>
48:         </activity>
49:     </application>
50: </manifest>
    
```

7. Selesai! Silakan di run.

PEMBAHASAN PROGRAM

Kita mulai dari activity pertama callActivity.java, seperti biasa terlebih dahulu kita melakukan sinkronisasi object terhadap widget pada xml. Sinkronisasi button kita lakukan di baris 11, sedangkan baris 12 berperan untuk mengaktifkan button. Baris 15-16 adalah deklarasi intent sedangkan baris 17 mengaktifkan intent untuk memanggil activity kedua menggunakan method `startActivityForResult()`. Dengan method ini, activity baru akan muncul diatas activity sebelumnya.

Sekarang kita bahas activity yang kedua. Baris 19 merupakan deklarasi intent, sedangkan `setResult()` pada baris 20 merupakan jawaban atas panggilan `startActivityForResult()` pada activity pertama. `Finish()` pada baris 21 adalah method untuk mengakhiri activity. Dengan finish-nya activity kedua ini, maka yang muncul dilayar adalah activity sebelumnya.

Apabila Ebook ini bermanfaat buat Kamu, Bantu penulis menyempurnakan ebook ini dengan cara **DONASI.**

Kami menerima kerjasama untuk mengadakan workshop, seminar, atau pelatihan IT. Jika berminat bisa menghubungi omayib@gmail.com

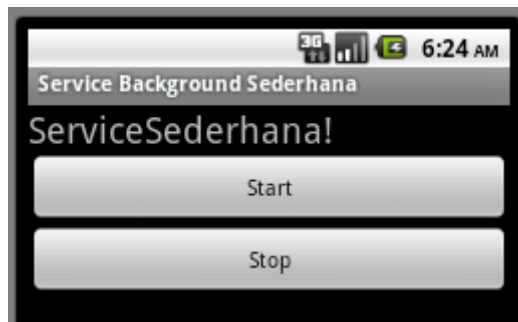
JAM 8 | BACKGROUND SERVICE

Bagian ini akan membahas mengenai

✓ Background service <NOT YET AVAILABLE>

Untuk apa sih background service? Background service adalah salah satu cara system untuk menjalankan fungsinya tanpa sepengetahuan user. Selama melakukan proses, tidak ada interaksi antara user dengan aplikasi. Implementasi paling sederhana adalah sound pada sebuah game, atau fungsi auto update pada aplikasi tertentu.

Nah kesempatan ini kita belajar implementasi background service menggunakan sound. Identya pada UI aplikasi disediakan 2 buah button yaitu start dan stop. Apabila button start diklik maka dari aplikasi akan terdengar suara music. Musik akan terus terdengar meskipun Kamu mengklik tombol back atau home. Lalu bagaimana untuk menghentikannya? Yap, Kamu harus kembali ke aplikasi ini kemudian mengklik button stop. Perhatikan Gambar 8.1 adalah tampilan aplikasi background service sederhana.



Gambar 8.1. Implementasi background service menggunakan sound

Are your ready??

1. Bikin project baru

Project name	ServiceBackground
Build Target	Android 2.2
Application name	ServiceSederhana
Package name	Com.serv.bg
Create Activity	ServiceSederhana
Min SDK version	8

2. Siapkan dulu strings.xml

Dapatkan materi terbaru di
www.omayib.com

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <resources>
3:     <string name="hello">ServiceSederhana!</string>
4:     <string name="app_name">Service Background
5:         Sederhana</string>
6:     <string name="startBtn">Start</string>
7:     <string name="stopBtn">Stop</string>
8: </resources>
9:

```

3. Kita bikin layoutnya di main.xml

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
3:     xmlns:android="http://schemas.android.com/apk/res/android"
4:     android:layout_width="fill_parent"
5:     android:layout_height="fill_parent"
6:     android:orientation="vertical" >
7:     <TextView
8:         android:layout_width="fill_parent"
9:         android:layout_height="wrap_content"
10:        android:text="@string/hello"
11:        android:textSize="24dip" />
12:
13:     <Button
14:         android:id="@+id/startBtn"
15:         android:layout_width="match_parent"
16:         android:layout_height="wrap_content"
17:         android:text="@string/startBtn" >
18:     </Button>
19:
20:     <Button
21:         android:id="@+id/stopBtn"
22:         android:layout_width="match_parent"
23:         android:layout_height="wrap_content"
24:         android:text="@string/stopBtn" >
25:     </Button>
26:
27: </LinearLayout>

```

4. Buat folder raw didalam folder res. Taruh saja file mp3nya di folder raw.

5. Membuat class MyService.java

```

1: package com.serv.bg;
2:
3: import android.app.Service;
4: import android.content.Intent;
5: import android.media.MediaPlayer;
6: import android.os.IBinder;
7:
8: public class MyService extends Service{
9:     MediaPlayer mp;
10:
11:     @Override
12:     public IBinder onBind(Intent arg0) {
13:         // TODO Auto-generated method stub
14:         return null;
15:     }
16:     @Override
17:     public void onCreate(){
18:         mp=MediaPlayer.create(this, R.raw.beraksi);
19:         mp.setLooping(false);
20:     }
21:

```

```

22:     public void onStart(Intent intent,int startId){
23:         mp.start();
24:     }
25:
26:     @Override
27:     public void onDestroy(){
28:         mp.stop();
29:     }
30: }
31:

```

6. Membuat activity ServiceSederhana.java

```

1: package com.serv.bg;
2:
3: import android.app.Activity;
4: import android.content.Intent;
5: import android.os.Bundle;
6: import android.view.View;
7: import android.view.View.OnClickListener;
8: import android.widget.Button;
9:
10: public class ServiceSederhana extends Activity implements
    OnClickListener {
11:     Button startBtn,stopBtn;
12:     /** Called when the activity is first created. */
13:     @Override
14:     public void onCreate(Bundle savedInstanceState) {
15:         super.onCreate(savedInstanceState);
16:         setContentView(R.layout.main);
17:
18:         startBtn=(Button)findViewById(R.id.startBtn);
19:         stopBtn=(Button)findViewById(R.id.stopBtn);
20:         startBtn.setOnClickListener(this);
21:         stopBtn.setOnClickListener(this);
22:
23:     }
24:     @Override
25:     public void onClick(View v) {
26:         switch(v.getId()){
27:             case R.id.startBtn:
28:                 startService(new
    Intent(this,MyService.class));
29:                 break;
30:             case R.id.stopBtn:
31:                 stopService(new
    Intent(this,MyService.class));
32:                 break;
33:         }
34:         // TODO Auto-generated method stub
35:
36:     }
37: }

```

BAB 4

MULTIMEDIA

JAM 10 | MULTIMEDIA AUDIO

Bagian ini akan membahas tentang

1. Mengakses sound
2. Record <NOT YET AVAILABLE>
3. Mengakses video <NOT YET AVAILABLE>
4. Mengakses streaming audio dan video <NOT YET AVAILABLE>

A. Mengakses sound

Kita akan membuat aplikasi sederhana untuk memutar file .mp3 melalui sebuah tombol play. Skenarionya, jika tombol play diklik, mp3 akan dimainkan. Pada saat bersamaan tombol play menjadi *disable* . Namun jika mp3 selesai berputar, baru kemudian tombol play *enable* kembali. Perhatikan gambar 10.1.



Gambar 10. 1

Sudah siap ? mari kita mulai!

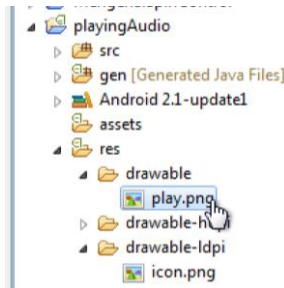
1. Jalankan Enclipse, buat Project baru.
2. Isilah parameter seperti berikut

Project name	playingAudio
Build Target	Android 2.2
Application name	Memutar file audio
Package name	contoh.playingAudio
Create Activity	playingAudio
Min SDK version	8

3. Dalam folder res, buat folder baru dengan nama drawable. Masukkan gambar play (atau apa saja untuk mewakili icon play) dalam format * .png (gambar 10.2).

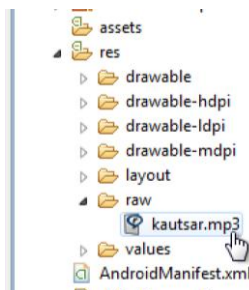


Gambar 10. 2 Gambar play.png



Gambar 10. 3

4. Buat folder baru lagi di res, kali ini beri nama raw. Masukkan file mp3 ke dalam folder raw. Latihan ini menggunakan file kautsar.mp3.



Gambar 10. 4

5. Kemudian ketikkan script berikut ini pada **main.xml**.

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
3:     android:orientation="vertical"
4:     android:layout_width="fill_parent"
5:     android:layout_height="fill_parent"
6:     >
7:     <TextView android:textSize="15px"
8:         android:layout_width="wrap_content"
9:         android:layout_height="wrap_content"
10:        android:id="@+id/textView"
11:        android:text="Memainkan Musik"></TextView>
12:     <ImageButton android:id="@+id/putarMusik"
13:         android:layout_height="wrap_content"
14:         android:adjustViewBounds="false"
15:         android:src="@drawable/play"
16:         android:layout_gravity="center_vertical|center_horizontal"
17:         android:layout_width="fill_parent"></ImageButton>
18:     <TextView android:text=""
19:         android:id="@+id/ket"
20:         android:layout_width="wrap_content"
21:         android:layout_height="wrap_content"
22:         android:layout_gravity="center_horizontal"
23:         android:textSize="15px"></TextView>
24: </LinearLayout>

```

6. Ketiklah kode **playingAudio.java** seperti berikut

```

1: package contoh.playingAudio;
2:

```



```

1: import java.io.IOException;
2: import android.app.Activity;
3: import android.media.MediaPlayer;
4: import android.media.MediaPlayer.OnCompletionListener;
5: import android.os.Bundle;
6: import android.view.View;
7: import android.view.View.OnClickListener;
8: import android.widget.ImageButton;
9: import android.widget.TextView;
10:
11: public class playingAudio extends Activity{
12:     ImageButton mainkan;
13:     TextView keterangan;
14:     MediaPlayer mp;
15:     /** Called when the activity is first created. */
16:     @Override
17:     public void onCreate(Bundle savedInstanceState) {
18:         super.onCreate(savedInstanceState);
19:         setContentView(R.layout.main);
20:
21:         keterangan=(TextView) findViewById(R.id.ket);
22:         keterangan.setText("Silakan klik tombol play");
23:
24:         mainkan=(ImageButton) findViewById(R.id.putarMusik);
25:         mainkan.setOnClickListener(new OnClickListener() {
26:             public void onClick(View arg0) {
27:                 mainkan.setEnabled(false);
28:                 keterangan.setText("Tombol play tidak aktif");
29:                 go();
30:             }
31:         });
32:     }
33:     public void go() {
34:         mp=MediaPlayer.create(playingAudio.this, R.raw.kautsar);
35:         try {
36:             mp.prepare();
37:         } catch (IllegalStateException e) {
38:             // TODO Auto-generated catch block
39:             e.printStackTrace();
40:         } catch (IOException e) {
41:             // TODO Auto-generated catch block
42:             e.printStackTrace();
43:         }
44:         mp.start();
45:         mp.setOnCompletionListener(new OnCompletionListener() {
46:             public void onCompletion(MediaPlayer arg0) {
47:                 mainkan.setEnabled(true);
48:                 keterangan.setText("Silakan klik tombol play");
49:             }
50:         });
51:     }
52: }

```

7. Bila script berantakan, lakukan Format (**source > format**).
8. Lakukan RUN dan lihat hasilnya.

PENJELASAN PROGRAM

1. Pada layout main.xml, Kita memasukkan 3 widget, yaitu
 1. TextView pada baris 7-11 untuk memunculkan tulisan "Memainkan Musik"

2. ImageButton pada baris 12-17 untuk membuat button yang didalamnya bisa diberi gambar.
3. TextView lagi pada baris 18-23 untuk menampilkan status button aktif atau tidak.

ImageButton mengambil gambar dari folder res/drawable, ditunjukkan pada baris 15. Sudah jelas kan?

Sekarang kita ke activity nya. Agar lebih mudah memahami, saya membuat activity ini menjadi 3 bagian yaitu

1. Baris 12-14 adalah deklarasi objek
2. Baris 21-31 adalah sinkronisasi objek terhadap widget di xml sekaligus mengaktifkan ImageButton. Apabila ImageButton diklik, maka method go() dipanggil.
3. Baris 33-51 adalah method go(), yang berisi fungsi-fungsi untuk memainkan media player.

Method go() berisi barisan perintah untuk memanggil kelas MediaPlayer. Kelas MediaPlayer bertugas memanggil dan memainkan file audio yang kita simpan di folder raw. Dalam latihan ini, penulis menggunakan file kautsar.mp3. Sebenarnya ada tahapan-tahapan (state diagram) dalam memainkan file audio, namun tidak dijelaskan pada buku ini karena dianggap rumit bagi level pemula. Sehingga penulis menyajikan dalam contoh latihan yang sederhana dan mudah dipahami.

Cara memanggilnya melalui baris 34. Mp adalah object MediaPlayer yang sudah dideklarasikan di awal program. Setelah file mp3 dipanggil, file audio memasuki tahap prepared (mp.prepared()). Pada tahap ini menggunakan try-catch. Try-catch adalah cara java untuk mengeksekusi suatu perintah yang ada didalam try. Jika eksekusi gagal, langsung ditangani oleh perintah yang ada didalam catch.

Setelah memaskui mp.prepared(), kemudian menggunakan mp.start() untuk mulai memainkan mp3. Sampai disini sudah terlihat, begitu ImageButton diclick, file mp3 akan diputar, ImageButton menjadi tidak aktif dan tulisan yang tampil adalah "Tombol play tidak aktif". Ide berikutnya adalah mengaktifkan kembali ImageButton jika file mp3 sudah selesai diputar. Maka digunakan method setOnCompletionListener().

Untuk mengetahui apakah mp3 selesai diputar atau belum, kita gunakan method onCompletion(). Disinilah kita kembali mengaktifkan imageButton (baris 47) dan mengubah status imageButton menjadi "Silakan Klik tombol play" (baris 48). Dengan demikian, begitu mp3 selesai diputar, maka ImageButton mainkan kembali aktif, dan TextView keterangan kembali menampilkan tulisan "Silakan klik tombol play".

JAM 11 | MULTIMEDIA VIDEO

Bagian ini akan membahas tentang

1. Mengakses video <NOT YET AVAILABLE>
2. Mengakses streaming audio dan video <NOT YET AVAILABLE>

Apabila Ebook ini bermanfaat buat Kamu, Bantu penulis menyempurnakan ebook ini dengan cara **DONASI.**

**Kirim TESTIMONI, KRITIK, SARAN
Kamu di
www.odayib.com/testimoni**

Kami menerima kerjasama untuk mengadakan workshop, seminar, atau pelatihan IT. Jika berminat bisa menghubungi omayib@gmail.com

BAB 5

ANDROID SQLite DATABASE

Kirim KRITIK, SARAN, dan
TESTIMONI Kamu di
www.omyib.com/testimoni

JAM 12 | ANDROID SQLite Database

Pada bagian ini, Kamu akan mempelajari mengenai

✓ Membuat database

SQLite adalah database yang bisa dibangun di android. SQLite memiliki fitur relasional database, hampir sama dengan SQL pada desktop hanya saja SQLite membutuhkan memori yang sedikit.

SQLite terdapat pada semua perangkat android, Kamu cukup mendefinisikan perintah SQL untuk meng-create atau meng-update database, selanjutnya system pada android akan menangani hal-hal yang berhubungan dengan database. SQLite database otomatis akan tersimpan didalam path data/data/nama_package/database/nama_database.

Sebelum mulai membuat aplikasi SQLite database, ada 3 kelas yang harus Kamu ketahui yaitu

1. SQLiteDatabase

SQLiteDatabase merupakan kelas yang mempunyai method seperti

- Insert() untuk menambahkan baris ke database
- Update() untuk memperbarui baris pada database
- Delete() untuk menghapus baris pada database
- execSQL() untuk mengeksekusi sintak SQL

2. SQLiteOpenHelper

SQLiteOpenHelper adalah subclass yang memiliki beberapa method seperti

- onCreate() dijalankan jika sebelumnya belum ada database
- onUpgrade() dijalankan jika sebelumnya sudah ditemukan database yang sama namun beda versi. Method ini bisa dimanfaatkan untuk mengubah skema database.
- onOpen() dijalankan jika database sudah dalam keadaan open

- `getWritableDatabase()` memanggil database agar bisa dimasuki data
- `getReadableDatabase()` memanggil database agar bisa membaca datanya

3. Cursor

Setiap query yang dieksekusi pasti membawa nilai kembalian atau feedback. Feedback yang dihasilkan query ini disebut cursor.

Jadi dengan kata lain, cursor merepresentasikan hasil query yang dieksekusi pada baris dan kolom tertentu. Gambar 12.1 berikut ini akan membantu Kamu memahami istilah cursor pada SQLite.

[fitri,masak] →	nama	hobi
	Ayib	makan
	fitri	masak
	ridhani	baca

Gambar 12.1. Analogi cursor pada SQLite database

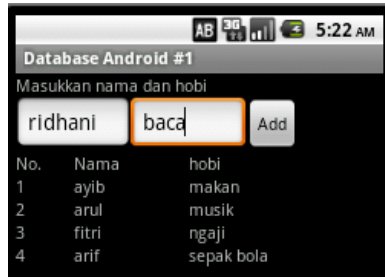
Cursor diwakili dengan tkamu panah berada pada posisi baris ke 2 dengan membawa data [fitri,masak]. Berikut ini beberapa method yang digunakan dalam latihan ini

- `moveToFirst()` untuk pindah ke baris pertama
- `isAfterLast()` akan mengirim pesan balik jika posisi cursor sudah berada di baris terakhir
- `getLong()` untuk mengambil data pada kolom yang mempunyai tipe data long
- `getString()` untuk mengambil data pada kolom yang mempunyai tipe data String

A. Membuat database

Sebelum mulai membuat project, ada baiknya kamu lihat dulu hasil akhir project Database Android #1 pada gambar 12.2

Kita akan membuat database hobi seseorang. Data dimasukkan melalui 2 buah edittext, kemudian tombol add dipakai untuk menyimpan data kedalam database sekaligus mengupdate tabel jika data berhasil disimpan.



Gambar 12.2 Android database #1

Sudah siap?? Mari kita praktekkan ☺

1. Siapkan project baru dengan nama sebagai berikut

Project name	DatabaseAndroid
Build Target	Android 2.2
Aplication name	Database Android #1
Package name	Com.db.satu
Create Activity	DatabaseAndroidSatu
Min SDK version	8

2. Ubah baigan **String.xml** seperti berikut

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <resources>
3:     <string name="hello">Masukkan nama dan hobi</string>
4:     <string name="app_name">Database Android #1</string>
5:     <string name="btnAddtxt">Add</string>
6:     <string name="namaLabel">Nama</string>
7:     <string name="hobiLabel">hobi</string>
8:     <string name="nomorLabel">No.</string>
9: </resources>
    
```

3. Ubah **main.xml** seperti berikut ini

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
3:     xmlns:android="http://schemas.android.com/apk/res/android"
4:     android:orientation="vertical"
5:     android:layout_width="fill_parent"
6:     android:layout_height="fill_parent">
7:     <TextView android:layout_width="fill_parent"
8:         android:layout_height="wrap_content"
9:         android:text="@string/hello" />
10:    <LinearLayout android:layout_width="match_parent"
11:        android:layout_height="wrap_content"
12:        android:id="@+id/linearLayout1">
13:        <EditText android:id="@+id/inNama"
14:            android:layout_height="wrap_content"
15:            android:layout_width="100dip"></EditText>
16:        <EditText android:id="@+id/inHobi"
17:            android:layout_height="wrap_content"
    
```



```

17:         android:layout_width="100dip"></EditText>
18:     <Button android:layout_width="wrap_content"
19:         android:id="@+id/btnAdd"
20:         android:layout_height="wrap_content"
21:         android:text="@string/btnAddtxt"></Button>
22: </LinearLayout>
23: <TableLayout android:layout_height="wrap_content"
24:     android:layout_width="match_parent"
25:     android:id="@+id/tabel_data">
26: <TableRow android:id="@+id/tableRow1"
27:     android:layout_width="wrap_content"
28:     android:layout_height="wrap_content">
29: <TextView android:layout_height="wrap_content"
30:     android:layout_width="50dip"
31:     android:text="@string/nomorLabel"
32:     android:id="@+id/no_id"></TextView>
33: <TextView android:layout_height="wrap_content"
34:     android:layout_width="100dip"
35:     android:text="@string/namaLabel"
36:     android:id="@+id/nama_id"></TextView>
37: <TextView android:layout_width="100dip"
38:     android:layout_height="wrap_content"
39:     android:text="@string/hobiLabel"
40:     android:id="@+id/hobi_id"></TextView>
41: </TableRow>
42: </TableLayout>
43: </LinearLayout>

```

4. Buat kelas baru kemudian beri nama **DatabaseManager.java**, setelah itu ketikkan baris kode berikut ini.

```

1: package com.db.satu;
2:
3: import java.util.ArrayList;
4:
5: import android.content.ContentValues;
6: import android.content.Context;
7: import android.database.Cursor;
8: import android.database.sqlite.SQLiteDatabase;
9: import android.database.sqlite.SQLiteOpenHelper;
10: import android.util.Log;
11:
12: public class DatabaseManager {
13:
14:     private static final String ROW_ID = "_id";
15:     private static final String ROW_NAMA = "nama";
16:     private static final String ROW_HOBI = "hobi";
17:
18:     private static final String NAMA_DB = "DatabaseAndroidSatu";
19:     private static final String NAMA_TABEL = "hobiku";
20:     private static final int DB_VERSION = 1;
21:
22:     private static final String CREATE_TABLE = "create table
23:         "+NAMA_TABEL+" (" +ROW_ID+" integer PRIMARY KEY autoincrement,
24:         "+ROW_NAMA+" text,"+ROW_HOBI+" text)";

```

```

24:     private final Context context;
25:     private DatabaseOpenHelper dbHelper;
26:     private SQLiteDatabase db;
27:
28:     public DatabaseManager(Context ctx) {
29:         this.context = ctx;
30:         dbHelper = new DatabaseOpenHelper(context);
31:         db = dbHelper.getWritableDatabase();
32:     }
33:
34:     private static class DatabaseOpenHelper extends
        SQLiteOpenHelper {
35:
36:     public DatabaseOpenHelper(Context context) {
37:         super(context, NAMA_DB, null, DB_VERSION);
38:         // TODO Auto-generated constructor stub
39:     }
40:
41:     @Override
42:     public void onCreate(SQLiteDatabase db) {
43:         // TODO Auto-generated method stub
44:         db.execSQL(CREATE_TABLE);
45:     }
46:
47:     @Override
48:     public void onUpgrade(SQLiteDatabase db, int oldVer, int
        newVer) {
49:         // TODO Auto-generated method stub
50:         db.execSQL("DROP TABLE IF EXISTS "+NAMA_DB);
51:         onCreate(db);
52:     }
53:     }
54: }
55: public void close() {
56:     dbHelper.close();
57: }
58:
59: public void addRow(String nama, String hobi) {
60:     ContentValues values = new ContentValues();
61:     values.put(ROW_NAMA, nama);
62:     values.put(ROW_HOBI, hobi);
63:     try {
64:         db.insert(NAMA_TABEL, null, values);
65:     } catch (Exception e) {
66:         Log.e("DB ERROR", e.toString());
67:         e.printStackTrace();
68:     }
69: }
70:
71: public ArrayList<ArrayList<Object>> ambilSemuaBaris() {
72:     ArrayList<ArrayList<Object>> dataArray = new
        ArrayList<ArrayList<Object>>();
73:     Cursor cur;
74:     try {
75:         cur = db.query(NAMA_TABEL,
76:             new String[] { ROW_ID, ROW_NAMA, ROW_HOBI }, null, null,
77:             null, null, null);
78:         cur.moveToFirst();

```

```

79: if (!cur.isAfterLast()) {
80: do {
81: ArrayList<Object> dataList = new ArrayList<Object>();
82:     dataList.add(cur.getLong(0));
83:     dataList.add(cur.getString(1));
84:     dataList.add(cur.getString(2));
85:     dataArray.add(dataList);
86: } while (cur.moveToNext());
87: }
88: } catch (Exception e) {
89: // TODO Auto-generated catch block
90:     e.printStackTrace();
91:     Log.e("DEBE ERROR", e.toString());
92: }
93: return dataArray;
94: }
95: }

```

5. Nah sekarang kita menambahkan kode di activity **DatabaseAndroidSatu.java** seperti dibawah ini

```

1: package com.db.satu;
2:
3: import java.util.ArrayList;
4:
5: import android.app.Activity;
6: import android.os.Bundle;
7: import android.view.View;
8: import android.widget.Button;
9: import android.widget.EditText;
10: import android.widget.TableLayout;
11: import android.widget.TableRow;
12: import android.widget.TextView;
13: import android.widget.Toast;
14:
15: public class DatabaseAndroidSatu extends Activity {
16:     DatabaseManager dm;
17:     EditText nama, hobi;
18:     Button addBtn;
19:     TableLayout tabel4data;// tabel for data
20:
21: /** Called when the activity is first created. */
22: @Override
23: public void onCreate(Bundle savedInstanceState) {
24:     super.onCreate(savedInstanceState);
25:     setContentView(R.layout.main);
26:
27:     dm = new DatabaseManager(this);
28:     tabel4data = (TableLayout) findViewById(R.id.tabel_data);
29:     nama = (EditText) findViewById(R.id.inNama);
30:     hobi = (EditText) findViewById(R.id.inHobi);
31:     addBtn = (Button) findViewById(R.id.btnAdd);
32:     addBtn.setOnClickListener(new View.OnClickListener() {
33:         @Override
34:         public void onClick(View v) {
35:             simpKamuta();
36:         }
37:     });
38: }

```

```

37:     });
38:     updateTable();
39: }
40:
41: protected void simpKamuta() {
42:     try {
43:         dm.addRow(nama.getText().toString(), hobi.getText().toString());
44:         Toast.makeText(getApplicationContext(),
45:             nama.getText().toString() + ", berhasil disimpan",
46:             Toast.LENGTH_SHORT).show();
47:         updateTable();
48:         kosongkanField();
49:     } catch (Exception e) {
50:         e.printStackTrace();
51:         Toast.makeText(getApplicationContext(), "gagal simpan, " +
52:             e.toString(), Toast.LENGTH_LONG).show();
53:     }
54:     protected void kosongkanField() {
55:         nama.setText("");
56:         hobi.setText("");
57:     }
58:     protected void updateTable() {
59:         // TODO Auto-generated method stub
60:         while (tabel4data.getChildCount() > 1) {
61:             tabel4data.removeViewAt(1);
62:         }
63:
64:         ArrayList<ArrayList<Object>> data = dm.ambilSemuaBaris();//
65:
66:         for (int posisi = 0; posisi < data.size(); posisi++) {
67:             TableRow tabelBaris = new TableRow(this);
68:             ArrayList<Object> baris = data.get(posisi);
69:
70:             TextView idTxt = new TextView(this);
71:             idTxt.setText(baris.get(0).toString());
72:             tabelBaris.addView(idTxt);
73:
74:             TextView namaTxt = new TextView(this);
75:             namaTxt.setText(baris.get(1).toString());
76:             tabelBaris.addView(namaTxt);
77:
78:             TextView hobiTxt = new TextView(this);
79:             hobiTxt.setText(baris.get(2).toString());
80:             tabelBaris.addView(hobiTxt);
81:
82:             tabel4data.addView(tabelBaris);
83:         }
84:     }
85: }

```

Jika tidak ada error, maka aplikasi AndroidDatabaseSatu siap dijalankan.

PENJELASAN PROGRAM

Pada project database kita mempunyai kelas java **DatabaseManager.java** (bukan activity). Kelas ini terdiri dari

- Deklarasi variable pada baris 14-26
- Constructor DatabaseManager pada baris 28-32
- subClass DatabaseOpenHelper pada baris 34-54
- Method close pada baris 55-57
- Method addRow pada baris 59-69
- Method ambilSemuaBaris pada baris 71-94

Didalam cunstructor DatabaseManager, kita mendefinisikan subclass DatabaseOpenHelper dan menaruh method getWritableDatabase sehingga ketika project ini dijalankan database langsung diset agar bisa ditulisi data.

Subclass DatabaseOpenHelper juga memiliki cunstructor DatabaseOpenHelper yang didalamnya terdapat method super(). Method ini dipanggil untuk mengidedntifikasi nama dan versi database. Pada subclass ini juga terdapat method onCreate() untuk men-generate database jika sebelumnya tidak ada database yang sama dan juga method onUpdgrate() yang akan dieksekusi jika dijumpai database dengan versi lebih baru.

Method addRow bertugas untuk menambahkan data kedalam database. Didalamnya terdapat class ContentValues(), dipakai untuk menyimpan sepasang data. Baris put(**ROW_NAMA**, nama) artinya menambahkan data yang dibawa oleh variable nama kedalam kolom nama.

Method ambilSemuaBaris digunakan untuk membaca seluruh isi database. Data yang terbaca kemudian disimpan kedalam array bertingkat. Output dari method ini adalah data [_id,nama,hobi].

Berikutnya kita bahas listing dalam activity **AndroidDatabaseSatu.java**. Acitivity ini memiliki 4 method yaitu onCreate(), simpKamuta(),updateTable(), kosongkanField(). Semua method diawali dengan void karena tidak mempunyai nilai kembalian apapun.

Pada method simpKamuta(), kita memakai method addRow() milik class DatabaseManager untuk menambahkan data ke dalam database. Sedangkan didalam method updateTable() kita memanggil method ambilSemuaBaris()

untuk menampilkan data ke tabel. Method `kosongkanField()` dipakai untuk mengosongkan editteks kembali.

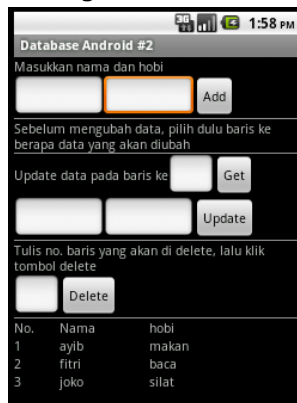
JAM 13 | ANDROID SQLite Database #2

Pada bagian ini, Kamu akan mempelajari mengenai

- ✓ Update database
- ✓ Hapus database

A. Android Database #2

Nah sekarang kita kembangkan lagi pemahaman database kita dengan menambahkan beberapa fitur seperti mengubah data dan menghapus data yang sebelumnya terdapat pada database. Perhatikan dulu gambar 12.3 berikut



1. Buat project baru dengan parameter sebagai berikut

Project name	DatabaseAndroid
Build Target	Android 2.2
Aplication name	Database Android #2
Package name	Com.db.dua
Create Activity	DatabaseAndroidDua
Min SDK version	8

2. Ubah **Strings.xml** seperti berikut

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <resources>
3:     <string name="hello">Masukkan nama dan hobi</string>
4:     <string name="app_name">Database Android #2</string>
5:     <string name="btnAddtxt">Add</string>
6:     <string name="namaLabel">Nama</string>
7:     <string name="hobiLabel">hobi</string>

```

```

8:      <string name="nomorLabel">No.</string>
9:      <string name="btnGetRow">Get</string>
10:     <string name="btnUpdateRow">Update</string>
11:     <string name="ketUpdate">Sebelum mengubah data,
12:         pilih dulu baris ke berapa
13:         data yang akan diubah</string>
14:     <string name="ketAmbilBaris">Update data pada baris
15:         ke</string>
16:     <string name="ketDelete">Tulis no. baris
17:         yang akan di delete,
18:         lalu klik tombol &quot;delete&quot;</string>
19:     <string name="btnDel">Delete</string>
20: </resources>

```

3. Buatlah layout **Main.xml** seperti ini

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
3:     xmlns:android="http://schemas.android.com/apk/res/android"
4:     android:orientation="vertical"
5:     android:layout_width="fill_parent"
6:     android:layout_height="fill_parent">
7:     <TextView android:layout_width="fill_parent"
8:         android:layout_height="wrap_content"
9:         android:text="@string/hello" />
10:    <LinearLayout android:layout_width="match_parent"
11:        android:layout_height="wrap_content"
12:        android:id="@+id/linearLayout1">
13:        <EditText android:id="@+id/inNama"
14:            android:layout_height="wrap_content"
15:            android:layout_width="100dip"></EditText>
16:        <EditText android:id="@+id/inHobi"
17:            android:layout_height="wrap_content"
18:            android:layout_width="100dip"></EditText>
19:        <Button android:layout_width="wrap_content"
20:            android:id="@+id/btnAdd"
21:            android:layout_height="wrap_content"
22:            android:text="@string/btnAddtxt"></Button>
23:    </LinearLayout>
24:    <View android:layout_width="wrap_content"
25:        android:id="@+id/view2"
26:        android:layout_height="1dip"
27:        android:background="#FF909090"></View>
28:    <TextView android:layout_width="wrap_content"
29:        android:layout_height="wrap_content"
30:        android:id="@+id/textView1"
31:        android:text="@string/ketUpdate"></TextView>
32:    <View android:layout_width="wrap_content"
33:        android:id="@+id/view2"
34:        android:layout_height="1dip"
35:        android:background="#FF909090"></View>
36:    <LinearLayout android:layout_width="match_parent"
37:        android:id="@+id/linearLayout2"
38:        android:layout_height="wrap_content">
39:        <TextView android:layout_width="wrap_content"
40:            android:text="@string/ketAmbilBaris"
41:            android:layout_height="wrap_content"

```

```

42:         android:id="@+id/textView2"></TextView>
43:     <EditText android:layout_height="wrap_content"
44:         android:id="@+id/inGetId"
45:         android:layout_width="50dip"></EditText>
46:     <Button android:layout_width="wrap_content"
47:         android:layout_height="wrap_content"
48:         android:text="@string/btnGetRow"
49:         android:id="@+id/btnGetId"></Button>
50: </LinearLayout>
51: <LinearLayout android:layout_width="match_parent"
52:     android:id="@+id/linearLayout3"
53:     android:layout_height="wrap_content">
54:     <EditText android:layout_height="wrap_content"
55:         android:layout_width="100dip"
56:         android:id="@+id/inUpdateNama"></EditText>
57:     <EditText android:layout_height="wrap_content"
58:         android:layout_width="100dip"
59:         android:id="@+id/inUpdateHobi"></EditText>
60:     <Button android:layout_width="wrap_content"
61:         android:layout_height="wrap_content"
62:         android:text="@string/btnUpdateRow"
63:         android:id="@+id/btnUpdate"></Button>
64: </LinearLayout>
65: <View android:layout_width="wrap_content"
66:     android:id="@+id/view2"
67:     android:layout_height="1dip"
68:     android:background="#FF909090"></View>
69: <TextView android:layout_height="wrap_content"
70:     android:layout_width="wrap_content"
71:     android:id="@+id/textView3"
72:     android:text="@string/ketDelete"></TextView>
73: <LinearLayout android:id="@+id/linearLayout4"
74:     android:layout_height="wrap_content"
75:     android:layout_width="match_parent">
76:     <EditText android:layout_height="wrap_content"
77:         android:layout_width="50dip"
78:         android:id="@+id/idDelete"></EditText>
79:     <Button android:layout_height="wrap_content"
80:         android:layout_width="wrap_content"
81:         android:id="@+id/button1"
82:         android:text="@string/btnDel"></Button>
83: </LinearLayout>
84: <View android:layout_width="wrap_content"
85:     android:id="@+id/view2"
86:     android:layout_height="1dip"
87:     android:background="#FF909090"></View>
88: <TableLayout android:layout_height="wrap_content"
89:     android:layout_width="match_parent"
90:     android:id="@+id/tabel_data">
91:     <TableRow android:id="@+id/tableRow1"
92:         android:layout_width="wrap_content"
93:         android:layout_height="wrap_content">
94:         <TextView android:layout_height="wrap_content"
95:             android:layout_width="50dip"
96:             android:text="@string/nomorLabel"
97:             android:id="@+id/no_id"></TextView>
98:         <TextView android:layout_height="wrap_content"
99:             android:layout_width="100dip"

```



```

100:         android:text="@string/namaLabel"
101:         android:id="@+id/nama_id"></TextView>
102:     <TextView android:layout_width="100dip"
103:         android:layout_height="wrap_content"
104:         android:text="@string/hobiLabel"
105:         android:id="@+id/hobi_id"></TextView>
106:     </TableRow>
107: </TableLayout>
108: </LinearLayout>

```

4. Selanjutnya kita sedikit mengubah Databasemanager.java

```

1: package com.db.dua;
2:
3: import java.util.ArrayList;
4:
5: import android.content.ContentValues;
6: import android.content.Context;
7: import android.database.Cursor;
8: import android.database.sqlite.SQLiteDatabase;
9: import android.database.sqlite.SQLiteOpenHelper;
10: import android.util.Log;
11: import android.widget.Toast;
12:
13: public class DatabaseManager {
14:
15:     private static final String ROW_ID = "_id";
16:     private static final String ROW_NAMA = "nama";
17:     private static final String ROW_HOBI = "hobi";
18:
19:     private static final String NAMA_DB = "DatabaseAndroidDua";
20:     private static final String NAMA_TABEL = "hobiku";
21:     private static final int DB_VERSION = 1;
22:
23:     private static final String CREATE_TABLE = "create table " +
24:     NAMA_TABEL+" (" + ROW_ID + " integer PRIMARY KEY autoincrement,"
25:     + ROW_NAMA+ " text," + ROW_HOBI + " text)";
26:
27:     private final Context context;
28:     private DatabaseOpenHelper dbHelper;
29:     private SQLiteDatabase db;
30:
31:     public DatabaseManager(Context ctx) {
32:         this.context = ctx;
33:         dbHelper = new DatabaseOpenHelper(ctx);
34:         db = dbHelper.getWritableDatabase();
35:     }
36:
37:     private static class DatabaseOpenHelper extends
38:     SQLiteOpenHelper {
39:     public DatabaseOpenHelper(Context context) {
40:         super(context, NAMA_DB, null, DB_VERSION);
41:     }
42:
43:     @Override
44:     public void onCreate(SQLiteDatabase db) {

```

```

43:         db.execSQL(CREATE_TABLE);
44:     }
45:
46:     @Override
47:     public void onUpgrade(SQLiteDatabase db, int oldVer, int
newVer) {
48:         db.execSQL("DROP TABLE IF EXISTS " + NAMA_DB);
49:         onCreate(db);
50:     }
51: }
52:
53: public void close() {
54:     dbHelper.close();
55: }
56:
57: public void addRow(String nama, String hobi) {
58:
59:     ContentValues values = new ContentValues();
60:     values.put(ROW_NAMA, nama);
61:     values.put(ROW_HOBI, hobi);
62:
63:     try {
64:         db.insert(NAMA_TABEL, null, values);
65:     } catch (Exception e) {
66:         Log.e("DB ERROR", e.toString());
67:         e.printStackTrace();
68:     }
69: }
70:
71: public ArrayList<ArrayList<Object>> ambilSemuaBaris() {
72:     ArrayList<ArrayList<Object>> dataArray = new
73:     ArrayList<ArrayList<Object>>();
74:     Cursor cur;
75:     try {
76:         cur = db.query(NAMA_TABEL, new String[] { ROW_ID, ROW_NAMA,
77:             ROW_HOBI }, null, null, null, null, null);
78:         cur.moveToFirst();
79:         if (!cur.isAfterLast()) {
80:             do {
81:                 ArrayList<Object> dataList = new ArrayList<Object>();
82:                 dataList.add(cur.getLong(0));
83:                 dataList.add(cur.getString(1));
84:                 dataList.add(cur.getString(2));
85:                 dataArray.add(dataList);
86:             } while (cur.moveToNext());
87:         }
88:     } catch (Exception e) {
89:         e.printStackTrace();
90:         Log.e("DEBE ERROR", e.toString());
91:         Toast.makeText(context, "gagal ambil semua
baris:" + e.toString(), Toast.LENGTH_SHORT).show();
92:     }
93:     return dataArray;
94: }
95:
96: public ArrayList<Object> ambilBaris(long rowId) {
97:
98:     ArrayList<Object> arrbaris = new ArrayList<Object>();

```

```

99:   Cursor cursor;
100:   try {
101:       cursor = db.query(NAMA_TABEL, new String[] { ROW_ID,
102:           ROW_NAMA, ROW_HOBI }, ROW_ID + "=" + rowId, null, null, null,
103:           null, null);
104:       cursor.moveToFirst();
105:
106:       if (!cursor.isAfterLast()) {
107:           do {
108:               arrbaris.add(cursor.getLong(0));
109:               arrbaris.add(cursor.getString(1));
110:               arrbaris.add(cursor.getString(2));
111:           } while (cursor.moveToNext());
112:           String r = String.valueOf(arrbaris);
113:           Toast.makeText(context, "haha" + r,
114:               Toast.LENGTH_SHORT).show();
115:       }
116:       cursor.close();
117:   } catch (Exception e) {
118:       e.printStackTrace();
119:       Log.e("error", e.toString());
120:       Toast.makeText(context, "hhii" + e.toString(),
121:           Toast.LENGTH_SHORT).show();
122:   }
123:   return arrbaris;
124: }
125:
126: public void updateBaris(long rowId, String nama, String hobi) {
127:
128:   ContentValues cv = new ContentValues();
129:   cv.put(ROW_NAMA, nama);
130:   cv.put(ROW_HOBI, hobi);
131:
132:   try {
133:       db.update(NAMA_TABEL, cv, ROW_ID + "=" + rowId, null);
134:   } catch (Exception e) {
135:       e.printStackTrace();
136:       Log.e("Db Error", e.toString());
137:   }
138: }
139:
140: public void deleteBaris(long idBaris){
141:   try {
142:       db.delete(NAMA_TABEL, ROW_ID+"="+idBaris, null);
143:   } catch (Exception e) {
144:       e.printStackTrace();
145:       Log.e("Error", e.toString());
146:   }
147: }
148: }

```

5. Terakhir kita membuat activity DatabaseAndroidDua.java

```

1: package com.db.dua;
2:
3: import java.util.ArrayList;

```

```

4: import android.app.Activity;
5: import android.os.Bundle;
6: import android.util.Log;
7: import android.view.View;
8: import android.widget.Button;
9: import android.widget.EditText;
10: import android.widget.TableLayout;
11: import android.widget.TableRow;
12: import android.widget.TextView;
13: import android.widget.Toast;
14:
15: public class DatabaseAndroidDua extends Activity {
16:     DatabaseManager dm;
17:     EditText nama, hobi, GetId, updateNama, updateHobi, idDel;
18:     Button addBtn, getIdBtn, updateBtn, delBtn;
19:     TableLayout tabel4data; // tabel for data
20:
21:     /** Called when the activity is first created. */
22:     @Override
23:     public void onCreate(Bundle savedInstanceState) {
24:         super.onCreate(savedInstanceState);
25:         setContentView(R.layout.main);
26:
27:         dm = new DatabaseManager(this);
28:         setupView();
29:         fungsiBtn();
30:         updateTable();
31:     }
32:
33:     public void setupView() {
34:         tabel4data = (TableLayout) findViewById(R.id.tabel_data);
35:         nama = (EditText) findViewById(R.id.inNama);
36:         hobi = (EditText) findViewById(R.id.inHobi);
37:         updateNama = (EditText) findViewById(R.id.inUpdateNama);
38:         updateHobi = (EditText) findViewById(R.id.inUpdateHobi);
39:         GetId = (EditText) findViewById(R.id.inGetId);
40:         idDel = (EditText) findViewById(R.id.idDelete);
41:
42:         addBtn = (Button) findViewById(R.id.btnAdd);
43:         getIdBtn = (Button) findViewById(R.id.btnGetId);
44:         updateBtn = (Button) findViewById(R.id.btnUpdate);
45:         delBtn = (Button) findViewById(R.id.btnDel);
46:     }
47:
48:     public void fungsiBtn() {
49:         addBtn.setOnClickListener(new View.OnClickListener() {
50:             @Override
51:             public void onClick(View v) {
52:                 simpKamuta();
53:                 kosongkanField();
54:             }
55:         });
56:         getIdBtn.setOnClickListener(new View.OnClickListener() {
57:             @Override
58:             public void onClick(View b) {
59:                 ambilBaris();
60:             }
61:         });

```

```

62:     updateBtn.setOnClickListener(new View.OnClickListener() {
63:         @Override
64:         public void onClick(View c) {
65:             updateBaris();
66:             kosongkanField();
67:         }
68:     });
69:     delBtn.setOnClickListener(new View.OnClickListener() {
70:         @Override
71:         public void onClick(View d) {
72:             // TODO Auto-generated method stub
73:             deleteData();
74:             kosongkanField();
75:         }
76:     });
77: }
78: protected void kosongkanField(){
79:     nama.setText("");
80:     hobi.setText("");
81:     updateNama.setText("");
82:     updateHobi.setText("");
83:     GetId.setText("");
84:     idDel.setText("");
85: }
86:
87: private void deleteData(){
88:     dm.deleteBaris(Long.parseLong(idDel.getText().toString()));
89:     updateTable();
90: }
91:
92: protected void updateBaris() {
93:     dm.updateBaris(Long.parseLong(GetId.getText().toString()),
94:         updateNama.getText().toString(),
95:         updateHobi.getText().toString());
96:     updateTable();
97: }
98:
99: private void ambilBaris() {
100:     try {
101:         ArrayList<Object> baris;
102:         baris =
103:         dm.ambilBaris(Long.parseLong(GetId.getText().toString()));
104:         updateNama.setText((String) baris.get(1));
105:         updateHobi.setText((String) baris.get(2));
106:     } catch (NumberFormatException e) {
107:         e.printStackTrace();
108:         Log.e("eror db", e.toString());
109:         Toast.makeText(getApplicationContext(), e.toString(),
110:             Toast.LENGTH_LONG).show();
111:     }
112: }
113:
114: protected void simpKamuta() {
115:     try {
116:         dm.addRow(nama.getText().toString(),
117:             hobi.getText().toString());
118:         updateTable();
119:     } catch (Exception e) {

```

```

118:     e.printStackTrace();
119:     Toast.makeText(getApplicationContext(), "gagal simpan,"+
120:         e.toString(), Toast.LENGTH_LONG).show();
121:     }
122: }
123:
124: protected void updateTable() {
125:     while (tabel4data.getChildCount() > 1) {
126:         tabel4data.removeViewAt(1);
127:     }
128:     ArrayList<ArrayList<Object>> data = dm.ambilSemuaBaris();//
129:     for (int posisi = 0; posisi < data.size(); posisi++) {
130:         TableRow tabelBaris = new TableRow(this);
131:         ArrayList<Object> baris = data.get(posisi);
132:
133:         TextView idTxt = new TextView(this);
134:         idTxt.setText(baris.get(0).toString());
135:         tabelBaris.addView(idTxt);
136:
137:         TextView namaTxt = new TextView(this);
138:         namaTxt.setText(baris.get(1).toString());
139:         tabelBaris.addView(namaTxt);
140:
141:         TextView hobiTxt = new TextView(this);
142:         hobiTxt.setText(baris.get(2).toString());
143:         tabelBaris.addView(hobiTxt);
144:
145:         tabel4data.addView(tabelBaris);
146:     }
147: }
148: }

```

PENJELASAN PROGRAM

Ok, pada bagian kedua ini kita menambah fitur agar bisa menghapus data dan mengubah isi tabel. Lihat kembali class **AndroidManager.java**. Disini kita menambahkan 3 method baru yaitu

1. Method `deleteBaris()` berfungsi untuk menghapus data. Terdapat pada baris 140-148
2. Method `ambilBaris()` digunakan untuk mengambil data berdasarkan id. Terdapat pada baris 96-124
3. Method `updateBaris()` berfungsi untuk mengubah data pada baris tertentu. Terdapat pada baris 126-138

Sekarang kita lihat activity **AndroidDatabaseDua.java**. disini kita menambahkan fungsi baru untuk menghapus data yaitu method `deleteData()` pada baris 87-90. Data dihapus berdasarkan kolom id. Misalkan id yang kamu masukkan bernilai 2 maka data yang dihapus adalah data pada baris 2. Nomor id diambil dari editteks `idDel`, kemudian dikirim ke method `deleteBaris()` milik `AndroidManager.java`.

Skeneraio berikutnya untuk mengubah data pada tabel dibagi menjadi 2 tahap, pertama memanggil datanya dulu berdasarkan id kemudian data ditangkap dan ditampilkan ke editteks. Disinilah user bisa mengubah data nama dan hobi, kemudian setelah menekan tombol "update", barulah method updateBaris() pada baris 92-95 bekerja menangkap isi editteks nama dan hobi kemudian dikirimkan ke method updateBaris() milik AndroidManager.java. Terakhir memanggil method updateTable untuk meperbarui isi tabel.

Apabila Ebook ini bermanfaat buat Kamu, Bantu penulis menyempurnakan ebook ini dengan cara
DONASI.

ImagineIT Education Center membuka kelas
Android Application Development
Mau?? 😊

**Kirim TESTIMONI, KRITIK, SARAN
Kamu di
www.odayib.com/testimoni**

BAB 6

ANDROID Location Based Service

JAM 14 | ANDROID GPS

Pada bagian ini, Kamu akan mempelajari mengenai

- ✓ GPS pada android
- ✓ Google Map
- ✓ Map, Marker dan GPS

A. GPS Android

Trend teknologi sekarang dan kedepan salah satunya adalah teknologi mobile dengan fitur LBS nya. Dengan LBS, kita bisa mencari suatu lokasi dengan sangat mudah, mencari posisi teman, mempublish lokasi keberadaan kita sendiri, mencari rute jalan atau lainnya.

Suatu gadget katakanlah Android mendapatkan lokasi bisa dari GPS, Network Location Provider atau bisa juga keduanya. Cell-id (Simcard) dan wifi adalah contoh Network Location Provider.

Menentukan lokasi menggunakan Android itu gampang-gampang susah, karena semakin akurat suatu lokasi semakin boros juga baterainya. Berikut ini adalah hal-hal yang menjadi kendala untuk menentukan lokasi

1. Multitude sumber lokasi

GPS, simcard dan wifi dapat dikombinasikan untuk mendapatkan lokasi yang akurat, namun efeknya jatuh ke baterai.

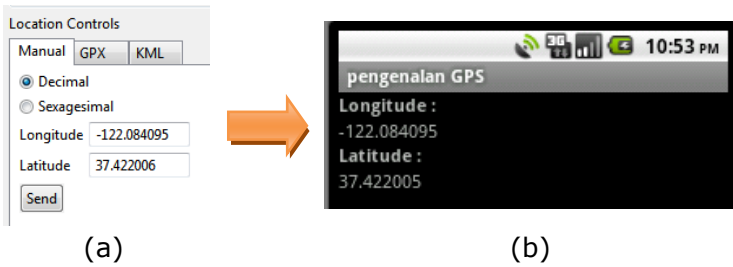
2. Perpindahan pengguna

Karena seorang pengguna melakukan perpindahan, maka aplikasi juga akan merefresh lokasi secara berulang.

3. Akurasi

Akurasi terhadap posisi suatu lokasi tidak konsisten. Posisi 10 menit yang lalu bisa jadi lebih akurat dibandingkan posisi yang terbaru.

Baik, agar cepat menjadi jago android, sekarang saatnya membuat aplikasi menentukan Latitude dan Longitude. Skenarionya kita bikin aplikasi yang bisa menangkap perubahan nilai longitude dan latitude. Setiap terjadi perubahan, nilai latitude dan longitude yang baru ditampilkan melalui Toast dan TextView. Kita bisa mengirim lokasi ke emulator menggunakan DDMS. DDMS dibahas lebih dalam di bagian xx. Hasil akhirnya tampak seperti gambar 14.1.



Gambar 14.1. Mengirim Lokasi ke emulator. (a) Bagian DDMS, (b) Emulator

1. Buat project baru

Project name	PengenalanGPS
Build Target	Android 2.2
Application name	Pengenalan GPS
Package name	Com.gps.sederhana
Create Activity	Lokasiku
Min SDK version	8

2. Tambahkan beberapa baris kode pada Strings.xml

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <resources>
3:     <string name="hello">Lokasiku!</string>
4:     <string name="app_name">pengenalan GPS</string>
5:     <string name="latText">Latitude :</string>
6:     <string name="lonText">Longitude :</string>
7: </resources>
    
```

3. Buat tampilan main.xml seperti berikut

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
3:     xmlns:android="http://schemas.android.com/apk/res/android"
4:     android:orientation="vertical"
5:     android:layout_width="fill_parent"
6:     android:layout_height="fill_parent">
7:     <TextView android:id="@+id/textView1"
8:         android:layout_width="wrap_content"
9:         android:layout_height="wrap_content"
10:         android:textStyle="bold"
11:         android:text="@string/lonText"></TextView>
12:     <TextView android:text="unknown"
13:         android:id="@+id/longitudetxt"
14:         android:layout_width="wrap_content"
15:         android:layout_height="wrap_content"></TextView>
16:     <TextView android:id="@+id/textView3"
17:         android:layout_width="wrap_content"
    
```

```

18:         android:layout_height="wrap_content"
19:         android:textStyle="bold"
20:         android:text="@string/latText"></TextView>
21:     <TextView android:text="unknown"
22:         android:id="@+id/latitudeTxt"
23:         android:layout_width="wrap_content"
24:         android:layout_height="wrap_content"></TextView>
25: </LinearLayout>
26:

```

4. Sekarang kita ke activity Lokasiku.java, tambahkan kode berikut

```

1: package com.gps.sederhana;
2:
3: import android.app.Activity;
4: import android.content.Context;
5: import android.location.Location;
6: import android.location.LocationListener;
7: import android.location.LocationManager;
8: import android.os.Bundle;
9: import android.widget.TextView;
10: import android.widget.Toast;
11:
12: public class Lokasiku extends Activity {
13:     private LocationManager lm;
14:     private LocationListener locListener;
15:     private TextView latTxt, lonTxt;
16:
17:     /** Called when the activity is first created. */
18:     @Override
19:     public void onCreate(Bundle savedInstanceState) {
20:         super.onCreate(savedInstanceState);
21:         setContentView(R.layout.main);
22:
23:         latTxt = (TextView) findViewById(R.id.latitudeTxt);
24:         lonTxt = (TextView) findViewById(R.id.longitudetext);
25:         lm = (LocationManager)
26:             getSystemService(Context.LOCATION_SERVICE);
27:         locListener = new MyLocationListener();
28:         lm.requestLocationUpdates(LocationManager.GPS_PROVIDER, 0,
29:             200, locListener);
30:     }
31:
32:     private class MyLocationListener implements LocationListener
33:     {
34:         @Override
35:         public void onLocationChanged(Location loc) {
36:             if (loc != null) {
37:                 latTxt.setText(String.valueOf(loc.getLatitude()));
38:                 lonTxt.setText(String.valueOf(loc.getLongitude()));
39:                 Toast.makeText(getApplicationContext(),
40:                     "Location Changed : Lat " + loc.getLatitude() +
41:                     "lgt: " + loc.getLongitude(), Toast.LENGTH_SHORT).show();
42:             }
43:         }
44:         @Override
45:         public void onProviderDisabled(String arg0) {}
46:         @Override

```

```

46:         public void onProviderEnabled(String arg0) {}
47:         @Override
48:         public void onStatusChanged(String provider, int status,
49:             Bundle extras){}
50:     }
51: }

```

5. Terakhir, tambahkan user.permission untuk mengakses GPS ke dalam Manifest.

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <manifest
   xmlns:android="http://schemas.android.com/apk/res/android"
3:     package="com.gps.sederhana" android:versionCode="1"
4:     android:versionName="1.0">
5:     <application android:icon="@drawable/icon"
6:         android:label="@string/app_name">
7:         <activity android:name=".Lokasiku"
8:             android:label="@string/app_name">
9:             <intent-filter>
10:             <action android:name="android.intent.action.MAIN" />
11:             <category
12:                 android:name="android.intent.category.LAUNCHER" />
13:             </intent-filter>
14:         </activity>
15:     </application>
16:     <uses-sdk android:minSdkVersion="8" />
17:     <uses-permission
18:         android:name="android.permission.ACCESS_FINE_LOCATION">
19:     </uses-permission>
20:     <uses-permission
21:         android:name="android.permission.ACCESS_COARSE_LOCATION">
22:     </uses-permission>
23:     <uses-permission
24:         android:name="android.permission.INTERNET">
25:     </uses-permission>
26: </manifest>

```

Silakan dirun, sambil berpindah ke tampilan DDMS.

PENJELASAN PROGRAM

Kita mulai dari Manifest, berikut ini rincian 3 permission yang baru saja ditambahkan

ACCESS_COARSE_LOCATION	Izin untuk menentukan lokasi dari cell-id, wifi
ACCESS_FINE_LOCATION	Izin untuk menentukan lokasi dari GPS
INTERNET	Izin untuk mengakses internet

Tanpa menambahkan permission diatas, aplikasi yang kamu buat tidak akan gagal dalam menentukan lokasi.

Berikutnya lihat lagi activity Lokasiku.java, kita menggunakan 2 kelas baru yaitu class `LocationListener` dan class `LocationManager`. Class `LocationManager` berperan sebagai pintu masuk aplikasi untuk mengakses service lokasi pada Android. Service ini memungkinkan aplikasi untuk meng-update posisi secara periodik. Perhatikan baris 54, Parameter pertama pada `requestLocationUpdates()` adalah jenis system yang digunakan untuk menentukan lokasi, dalam hal ini kita menggunakan GPS. Jika ingin menentukan lokasi berdasarkan wifi, gantilah `GPS_PROVIDER` dengan `NETWORK_PROVIDER`. Parameter berikutnya adalah minimal waktu aplikasi melakukan update, sedangkan parameter ke 3 adalah jarak minimal. Jika minimal waktu diisi 60000 (ms) maka aplikasi akan melakukan update setiap 6 detik. Dalam project ini, jarak diisi dengan angka 200 (m) maka aplikasi akan melakukan update setiap melakukan perpindahan sejauh lebih dari 200m. Parameter terakhir adalah `LocationListener` yang akan menangkap setiap terjadi proses update lokasi.

Saat terjadi perubahan lokasi atau saat proses update lokasi berlangsung, aplikasi mengirim sebuah sinyal yang kemudian ditangkap oleh class `LocationListener`. Didalam class ini, terdapat 4 method dengan rincian sebagai berikut

1. `onLocationChanged()`, dieksekusi ketika terjadi perubahan lokasi
2. `onProviderDisabled()`, dieksekusi saat provider tidak aktif
3. `onProviderEnabled()`, dipanggil saat provider aktif
4. `onStatusChanged()`, dipanggil saat terjadi perubahan status pada provider seperti `OUT_OF_SERVICE`, `AVAILABLE`, atau `TEMPORARILY_UNAVAILABLE`.

Apabila Ebook ini bermanfaat buat Kamu, Bantu penulis menyempurnakan ebook ini dengan cara **DONASI.**

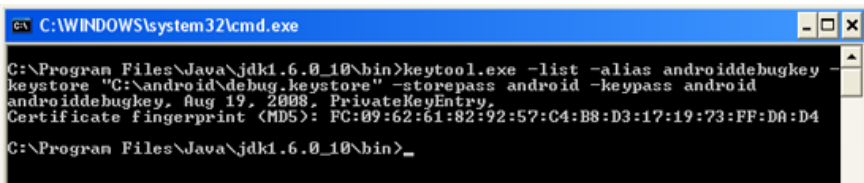
JAM 15 | GOOGLE MAP

A. GOOGLE MAP

Kali ini kita akan menggunakan fasilitas API Google Map untuk itu setiap kali mencoba kita memerlukan sebuah koneksi internet. Langkah pertama yang harus kamu lakukan adalah mendapatkan kode Google API. Berikut cara melakukan generate di windows seven.

1. Cari file debug.keystore , dalam laptop saya terletak di C:\User\acer\.android. Kalo tidak ketemu, gunakan fasilitas search di windows explorer dengan lingkup pencarian drive C.
2. Copy dulu file debug.keystore kemudian paste ke dalam folder C:\android
3. Melalui CMD, masuk dulu ke path C:\Program Files\Java\jdk1.6.0_10\bin
4. Kemudian ketikkan perintah berikut lalu tekan enter. Perhatikan gambar 14.2 ini.

```
keytool.exe -list -alias androiddebugkey -keystore
"C:\android\debug.keystore" -storepass android -keypass
android
```



Gambar 14.2. Proses generate MD5 Fingerprint

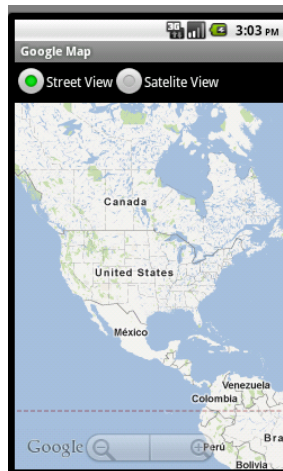
5. Jika berhasil, Kamu mendapatkan sebuah kode MD5 Fingerprint. Kode ini digunakan untuk membuat Key Google Map
6. Sekarang masuklah ke <http://code.google.com/android/maps-api-signup.html> , ikuti petunjuknya.
7. Selamat, Kamu sudah memiliki Key Google Map. Simpan baik-baik key ini karena akan diletakkan di bagian main.xml untuk membuat aplikasi di Android. Berikut ini adalah cuplikan key yang dimasukkan ke main.xml

```
1: ...
2: <com.google.android.maps.MapView
3:     android:id="@+id/mapView"
4:     android:layout_width="match_parent"
5:     android:layout_height="match_parent"
6:     android:enabled="true"
7:     android:clickable="true"
8:     android:apiKey="0CyuBP8zNhMbsh0kiDX7go-37s8g81rYQQoldrQ" />
```

Sekarang baru kita mulai project membuat Map paling sederhana di Android. Skenarionya, begitu aplikasi ini dilaunch, maka akan muncul map google (terhubung dengan internet). Pada layout tersedia 2 buah radio button untuk memilih jenis tampilan antara satellite dan street. Map juga bisa di zoom. Hasil akhirnya seperti pada Gambar 14.3. Are you ready?? Yup, kita mulai!.

1. Buat project baru

Project name	mapSederhana
Build Target	Android 2.2
Aplication name	Google Map
Package name	Com.map
Create Activity	TampilkanMap
Min SDK version	8



Gambar 14.3. Menampilkan map pada emulator

2. Kita ubah dulu **Main.xml**, masukkan key Google Api tadi ke android:apiKey

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
  xmlns:android="http://schemas.android.com/apk/res/android"
3:   android:id="@+id/linearLayout1"
4:   android:layout_width="match_parent"
5:   android:layout_height="match_parent"

```

```

6:     android:orientation="vertical">
7:     <LinearLayout android:id="@+id/linearLayout2"
8:     android:layout_width="match_parent"
9:     android:layout_height="wrap_content">
10:         <RadioGroup android:layout_height="wrap_content"
11:         android:orientation="horizontal"
12:         android:layout_width="match_parent"
13:         android:id="@+id/viewRG">
14:             <RadioButton android:layout_height="wrap_content"
15:             android:layout_width="wrap_content"
16:             android:checked="true"
17:             android:text="@string/streetTxt"
18:             android:id="@+id/streetRB"></RadioButton>
19:             <RadioButton android:layout_height="wrap_content"
20:             android:layout_width="wrap_content"
21:             android:text="@string/satelitTxt"
22:             android:layout_gravity="right"
23:             android:id="@+id/sateliteRB"></RadioButton>
24:         </RadioGroup>
25:     </LinearLayout>
26: <RelativeLayout android:layout_width="match_parent"
27: android:id="@+id/relativeLayout1"
28: android:layout_height="match_parent">
29: <com.google.android.maps.MapView
30: android:id="@+id/mapView"
31: android:layout_width="match_parent"
32: android:layout_height="match_parent"
33: android:enabled="true"
34: android:clickable="true"
35: android:apiKey="0CYuBP8zNhMbsh0kiDX7go-37s8g81rYQQoIdrQ" />
36: </RelativeLayout>
37: </LinearLayout>

```

3. Tambahkan kode berikut ke Activity **TampilkanMap.java**

```

1: package com.map;
2:
3: import com.google.android.maps.MapActivity;
4: import com.google.android.maps.MapController;
5: import com.google.android.maps.MapView;
6:
7: import android.os.Bundle;
8: import android.widget.RadioGroup;
9: import android.widget.RadioGroup.OnCheckedChangeListener;
10:
11: public class TampilkanMap extends MapActivity
12:     implements OnCheckedChangeListener {
13:     MapView mp=null;
14:     MapController mc;
15:     RadioGroup rg;
16:     /** Called when the activity is first created. */
17:     @Override
18:     public void onCreate(Bundle savedInstanceState) {
19:         super.onCreate(savedInstanceState);
20:         setContentView(R.layout.main);
21:
22:         mp=(MapView) findViewById(R.id.mapView);
23:         mp.setBuiltInZoomControls(true);

```



```

24:         rg=(RadioGroup) findViewById(R.id.viewRG);
25:         rg.setOnCheckedChangeListener(this);
26:     }
27:     @Override
28:     protected boolean isRouteDisplayed() {
29:         return false;
30:     }
31:     @Override
32:     public void onCheckedChanged(RadioGroup group, int chekId) {
33:         // TODO Auto-generated method stub
34:         switch(chekId){
35:             case R.id.sateliteRB:
36:                 mp.setStreetView(false);
37:                 mp.setSatellite(true);
38:                 break;
39:             case R.id.streetRB:
40:                 mp.setSatellite(false);
41:                 mp.setStreetView(true);
42:                 break;
43:         }
44:     }
45: }
46: }

```

4. Jangan lupa dibagian **AndroidManifest.xml** Kamu harus menambahkan library google map (lihat baris 9).

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <manifest
3:     xmlns:android="http://schemas.android.com/apk/res/android"
4:     package="com.map" android:versionCode="1"
5:     android:versionName="1.0">
6:
7:     <application android:icon="@drawable/icon"
8:         android:label="@string/app_name">
9:         <uses-library android:name="com.google.android.maps"></uses-
10:     library>
11:     <activity android:name="com.map.TampilkanMap"
12:         android:label="@string/app_name">
13:     <intent-filter>
14:     <action android:name="android.intent.action.MAIN" />
15:     <category android:name="android.intent.category.LAUNCHER" />
16:     </intent-filter>
17:     </activity>
18:     </application>
19:
20:     <uses-sdk android:minSdkVersion="8" />
21:     <uses-permission
22:     android:name="android.permission.INTERNET"></uses-permission>
23: </manifest>

```

Selamat , Kamu sudah bisa mengakses Mapnya google!!
Hehehe...

PEMBAHASAN PROGRAM

Mari kita lihat kembali activity TampilkanMap. Seperti biasa, deklarasi dilakukan dibagian awal program pada baris 13-14. MapView adalah object yang dapat menampilkan map yang diambil dari google. Object ini bisa merespon sentuhan, bisa juga ditambah sebuah lapisan untuk tempat marker. MapView dapat menampilkan 3 mode yaitu mode satellite, street, dan traffic. Class MapController dipakai untuk mengontrol map agar bisa di zoom dengan perbesaran sekian kali. MapController juga berfungsi untuk menambahkan sebuah animasi kecil. Jika Kamu pernah menggunakan google map, kemudian mengklik sebuah marker maka map akan menggeser focus ke tempat yang baru. Proses ini adalah contoh implementasi MapController.

Baris 22-25 adalah sinkronisasi object terhadap id nya pada layout xml. Baris 25 sendiri berfungsi menambahkan method kedalam button group agar aktif saat diklik.

Saat radio button diklik, maka kemudian memanggil fungsi `onCheckedChanged()` pada baris 32-43. Didalamnya terdapat sebuah pilihan kondisi menggunakan switch-case. Pilihan kondisi ini dedasarkan pada id radio button yang diklik. Jika yang diklik adalah id `sateliteRB`, maka `setStreetView()` diset false dan `setSateliteView()` diset true. Sebaliknya, jika yang diklik adalah id `streetRB`, maka `setStreetView()` diset true dan `setSateliteView()` diset false. `setStreetView()` dan `setSateliteView()` adalah methodnya class MapView.

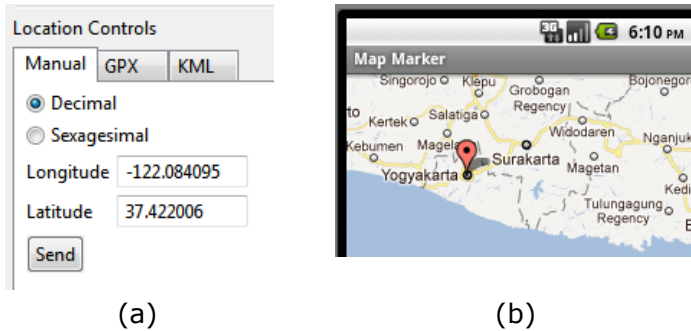
B. MAP, MARKER, dan GPS

Di bagian ini, kita naik satu tingkat lagi dengan menggabungkan 2 project sebelumnya ditambah menampilkan sebuah marker atau pin sebagai tKamu suatu tempat, dengan demikian kita menggabungkan 3 dasar

1. Menampilkan map
2. Menentukan lokasi dari gps
3. Member tKamu lokasi kita dengan sebuah marker.

Ups, tapi sabar dulu..! sebelumnya kita bicara scenarionya dulu. Begitu dijalankan, bersamaan dengan menampilkan map, muncul juga sebuah marker atau pin pada lokasi tertentu. Dalam hal ini defaultnya marker berada di atas kota Yogyakarta. Begitu GPS menangkap latitude dan longitude yang baru (disimulasikan dengan mengirim lokasi melalui DDMS),

maka marker akan berpindah ke lokasi baru, diikuti animasi sederhana map bergeser mengikuti marker sekaligus melakukan zoom. Perhatikan Gambar 15.1.



Gambar 15.1. Kombinasi map, marker dan GPS , (a) DDMS untuk emulator GPS, (b) Map dan markernya

1. Bikin project baru lagi

Project name	mapMarkerGps
Build Target	Android 2.2
Application name	Google Map
Package name	Com.map.marker
Create Activity	MapMarker
Min SDK version	8

2. Masukkan gambar marker/pin ke drawable

3. Main.xml

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <RelativeLayout
3:     xmlns:android="http://schemas.android.com/apk/res/android"
4:     android:orientation="vertical"
5:     android:layout_width="fill_parent"
6:     android:layout_height="fill_parent">
7:
8:     <com.google.android.maps.MapView
9:
10:        xmlns:android="http://schemas.android.com/apk/res/android"
11:        android:id="@+id/map_view"
12:        android:layout_width="fill_parent"
13:        android:layout_height="fill_parent"
14:        android:clickable="true"
15:        android:enabled="true"
16:        android:apiKey="0CyuBP8zNhMbsh0kiDX7go-
17:        37s8g81rYQQoldrQ" />
18: </RelativeLayout>
19: 
```

4. userPosition

```

1: package com.map.marker;
2:
3: public class userPosition {
4:     static double latitude;
5:     static double longitude;
6:
7:     public double getLatitude() {
8:         return latitude;
9:     }
10:
11:    public double getLongitude() {
12:        return longitude;
13:    }
14:    public void setLatitude(double latitude) {
15:        userPosition.latitude = latitude;
16:    }
17:    public void setLongitude(double longitude) {
18:        userPosition.longitude = longitude;
19:    }
20: }
21:

```

5. Bikin dulu class baru CustomItemizedOverlay.java

```

1: package com.map.marker;
2:
3: import java.util.ArrayList;
4:
5: import android.app.AlertDialog;
6: import android.content.Context;
7: import android.graphics.drawable.Drawable;
8:
9: import com.google.android.maps.ItemizedOverlay;
10: import com.google.android.maps.OverlayItem;
11:
12: public class CustomItemizedOverlay extends
13: ItemizedOverlay<OverlayItem> {
14:
15:     private ArrayList<OverlayItem> mapOverlays = new
16: ArrayList<OverlayItem>();
17:
18:     private Context context;
19:
20:     public CustomItemizedOverlay(Drawable defaultMarker) {
21:         super(boundCenterBottom(defaultMarker));
22:         // TODO Auto-generated constructor stub
23:     }
24:
25:     public CustomItemizedOverlay(Drawable defaultMarker, Context
26: context) {
27:         this(defaultMarker);
28:         this.context = context;
29:     }
30:
31:     @Override

```

```

32:     protected OverlayItem createItem(int i) {
33:         // TODO Auto-generated method stub
34:         return mapOverlays.get(i);
35:     }
36:
37:     @Override
38:     public int size() {
39:         // TODO Auto-generated method stub
40:         return mapOverlays.size();
41:     }
42:
43:     @Override
44:     protected boolean onTap(int index) {
45:         OverlayItem item = mapOverlays.get(index);
46:         AlertDialog.Builder ad = new AlertDialog.Builder(context);
47:         ad.setTitle(item.getTitle());
48:         ad.setMessage(item.getSnippet());
49:         ad.show();
50:         return true;
51:     }
52:
53:     public void addOverlay(OverlayItem overlay) {
54:         mapOverlays.add(overlay);
55:         this.populate();
56:     }
57: }
58:

```

6. Tulis kode Activity MapMarker seperti ini

```

1: package com.map.marker;
2:
3: import java.util.List;
4:
5: import com.google.android.maps.GeoPoint;
6: import com.google.android.maps.MapActivity;
7: import com.google.android.maps.MapController;
8: import com.google.android.maps.MapView;
9: import com.google.android.maps.Overlay;
10: import com.google.android.maps.OverlayItem;
11: import android.content.Context;
12: import android.location.Location;
13: import android.location.LocationListener;
14: import android.location.LocationManager;
15: import android.os.Bundle;
16: import android.widget.Toast;
17:
18: public class MapMarker extends MapActivity {
19:
20:     private MapView mapView;
21:     MapController mapController;
22:     LocationListener locationListener;
23:     CustomItemizedOverlay itemizedOverlay;
24:     List<Overlay> mapOverlays;
25:     userPosition pos;
26:
27:     /** Called when the activity is first created. */
28:     @Override

```

```

29: public void onCreate(Bundle savedInstanceState) {
30:     super.onCreate(savedInstanceState);
31:     setContentView(R.layout.main);
32:
33:     mapView = (MapView) findViewById(R.id.map_view);
34:     mapView.setBuiltInZoomControls(true);
35:     mapOverlays = mapView.getOverlays();
36:     mapController = mapView.getController();
37:     pos = new userPosition();
38:
39:     LocationManager locationManager = (LocationManager)
40:     getSystemService(Context.LOCATION_SERVICE);
41:     locationListener = new MyLocationListener();
42:     locationManager.requestLocationUpdates
43:         (LocationManager.GPS_PROVIDER, 0,
44:          0, locationListener);
45:
46:     if (pos.getLatitude() > 0) {
47:         getGeoPointUser(pos.getLatitude(),
48:             pos.getLongitude());
49:
50:     } else {
51:         getGeoPointUser(-7.801307, 110.364756); //yogya
52:     }
53: }
54:
55: @Override
56: protected boolean isRouteDisplayed() {
57:     // TODO Auto-generated method stub
58:     return false;
59: }
60:
61: private void getGeoPointUser(double lat, double lon) {
62:     // TODO Auto-generated method stub
63:     GeoPoint point = new GeoPoint((int) (lat * 1E6),
64:         (int) (lon * 1E6));
65:     OverlayItem overlayitem = new OverlayItem(point, "Hai..",
66:         "Saya omayib");
67:     itemizedOverlay = new
68:         CustomItemizedOverlay(this.getResources()
69:             .getDrawable(R.drawable.marker), MapMarker.this);
70:     itemizedOverlay.addOverlay(overlayitem);
71:     mapOverlays.add(itemizedOverlay);
72:     mapController.animateTo(point);
73:     mapController.setZoom(6);
74: }
75:
76: public class MyLocationListener
77:     implements LocationListener {
78:
79:     @Override
80:     public void onLocationChanged(Location loc) {
81:         // TODO Auto-generated method stub
82:         if (loc != null) {
83:             mapOverlays.remove(itemizedOverlay);
84:             Toast.makeText(getApplicationContext(), "Location
85:                 changed : Lat: " + loc.getLatitude() + " Lng: "
86:                 + loc.getLongitude(),

```

```

87:         Toast.LENGTH_SHORT).show();
88:         double lat = loc.getLatitude();
89:         double longi = loc.getLongitude();
90:         pos.setLatitude(lat * 1E6);
91:         pos.setLongitude(longi * 1E6);
92:         getGeoPointUser(lat, longi);
93:     }
94: }
95:
96: @Override
97:     public void onProviderDisabled(String arg0) {
98:         // TODO Auto-generated method stub
99:     }
100: @Override
101:     public void onProviderEnabled(String arg0) {
102:         // TODO Auto-generated method stub
103:     }
104:
105: @Override
106:     public void onStatusChanged(String arg0, int arg1, Bundle
107:     arg2) {
108:         // TODO Auto-generated method stub
109:     }
110: }
111: }
112:
113: }

```

Setelah tidak ada error, coba di run. Wow!! Android memang luar biasa! 😊

PEMBAHASAN PROGRAM

<NOT YET AVAILABLE>

Apabila Ebook ini bermanfaat buat Kamu, Bantu penulis menyempurnakan ebook ini dengan cara **DONASI.**

BAB 7

ANDROID

Client Server

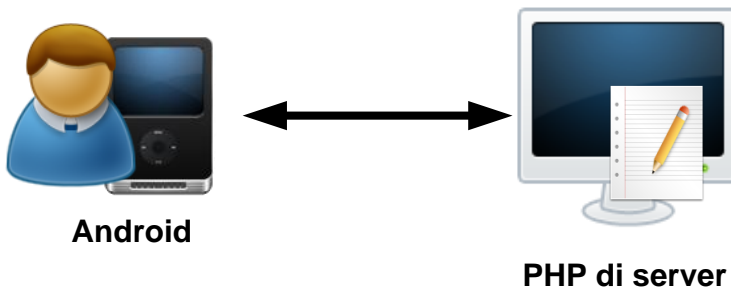
JAM 17 | ANDROID dan PHP

Pada bagian ini, Kamu akan mempelajari mengenai

- ✓ Login menggunakan php
- ✓ Login menggunakan mySQL server
- ✓ Android dan mySQL server (NOT YET AVAILABLE)
- ✓ JSOIN WebService (coming soon)
- ✓ Facebook-Twitter API (coming soon)

A. Login menggunakan php

Pada bagian ini, Kamu akan membuat project sederhana berupa halaman login yang autentifikasinya melalui PHP di server. Skenarionya seperti pada gambar 16.1.



Gambar 16.1. Skenario client server

Dalam hal ini, Android bertindak sebagai client. Teks masukan berupa *username* dan *password* pada halaman login dikemas sedemikian rupa kemudian dikirim keserver.

Pada bagian server, data yang dikirim android ditangkap oleh file PHP kemudian diolah apakah datanya cocok atau tidak. Selanjutnya PHP mengeluarkan jawaban yang selanjutnya akan ditangkap oleh android.

Langkah	Client	Server
1	Mengambil teks <i>username</i> dan <i>password</i>	-
2	Megirim data <i>username</i> dan <i>password</i> ke server	Data ditangkap oleh PHP

3	-	PHP mengeluarkan jawaban berhasil login atau tidak
4	Jawaban dari server ditangkap oleh android kemudian ditampilkan	

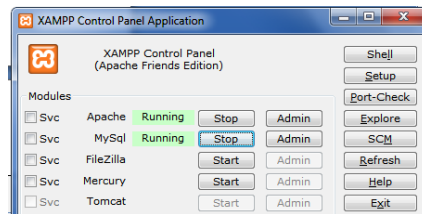
Project ini diselesaikan melalui 3 tahap,

1. Instalasi XAMPP untuk mengaktifkan server local (localhost)
2. Membuat file PHP
3. Membuat project di Android

Nah, saatnya kita mulai ☺

Instalasi XAMPP

1. Download xampp di website resminya
2. Cari file XAMPP-Control, start **APACHE** dan **MYSQL** (Gambar 16.2)



Gambar 16.2 Saat xampp control aktif

3. Masuklah ke direktori c:/xampp/htdocs , bikin folder baru bernama **android**

Membuat file PHP

1. Ketik file **login.php** berikut (bisa menggunakan notepad)

```

1: <?php
2: $user=$_POST['auser'];
3: $pass=$_POST['apass'];
4:
5: if($user=='omayib'&&$pass='1234'){
6:     $ket='Login sukses';
7: }else{
8:     $ket='gagal login';
9: }
10: echo $ket;
11: ?>

```

2. simpan di direktori C:/xampp/htdocs/android

Membuat Project di Android

1. Siapkan project baru dengan seperti ketentuan berikut

Project name	PHPLogin
Build Target	Android 2.2
Aplication name	Login melalui PHP
Package name	omayib.com.phplogin
Create Activity	Phplogin
Min SDK version	8

2. Ketik **strings.xml** seperti berikut

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <resources>
3: <string name="hello">Hello World, Phplogin!</string>
4: <string name="app_name">Login melalui PHP</string>
5:   <string name="txtPesan">Silakan Login dulu</string>
6:   <string name="txtUser">Username</string>
7:   <string name="txtPass">Password</string>
8:   <string name="btnLogin">Login</string>
9:   <string name="btnSignUp">Sign Up</string>
10: </resources>

```

3. Ketik **main.xml** seperti berikut

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <LinearLayout
3:   xmlns:android="http://schemas.android.com/apk/res/android"
4:   android:orientation="vertical"
5:   android:layout_width="fill_parent"
6:   android:layout_height="fill_parent">
7:   <TextView
8:     android:layout_width="fill_parent"
9:     android:layout_height="wrap_content"
10:    android:text="@string/hello" />
11: <TextView
12:   android:layout_width="wrap_content"
13:   android:layout_height="wrap_content"
14:   android:id="@+id/textView2">
15:   android:text="@string/txtUser"></TextView>
16: <EditText
17:   android:layout_width="match_parent"
18:   android:layout_height="wrap_content"
19:   android:id="@+id/userInput"></EditText>
20: <TextView
21:   android:layout_width="wrap_content"
22:   android:layout_height="wrap_content"
23:   android:id="@+id/textView1">
24:   android:text="@string/txtPass"></TextView>
25: <EditText

```

```

26: android:layout_width="match_parent"
27: android:layout_height="wrap_content"
28: android:id="@+id/passInput"></EditText>
29: <Button android:id="@+id/button1"
30: android:layout_width="match_parent"
31: android:layout_height="wrap_content"
32: android:text="@string/btnLogin"></Button>
33: <EditText
34: android:layout_width="match_parent"
35: android:layout_height="wrap_content"
36: android:id="@+id/status"
37: android:text="@string/txtPesan"></EditText>
38: </LinearLayout>

```

4. Membuat class ClientToserver.java yang nantinya berperan sebagai jembatan antara php deserver dengan activity phplogin

```

1: package omayib.com.phplogin;
2:
3: import java.io.BufferedReader;
4: import java.io.InputStreamReader;
5: import java.net.URI;
6: import java.util.ArrayList;
7:
8: import org.apache.http.HttpResponse;
9: import org.apache.http.NameValuePair;
10: import org.apache.http.client.HttpClient;
11: import
    org.apache.http.client.entity.UrlEncodedFormEntity;
12: import org.apache.http.client.methods.HttpGet;
13: import org.apache.http.client.methods.HttpPost;
14: import org.apache.http.conn.params.ConnManagerParams;
15: import org.apache.http.impl.client.DefaultHttpClient;
16: import org.apache.http.params.HttpConnectionParams;
17: import org.apache.http.params.HttpParams;
18:
19: public class ClientToServer {
20:     public static final int HTTP_TIMEOUT = 30 * 1000;
21:     private static HttpClient client;
22:     private static HttpClient getHttpClient() {
23:         if (client == null) {
24:             client = new DefaultHttpClient();
25:             final HttpParams parameterHttp = client.getParams();
26:             HttpConnectionParams.setConnectionTimeout(parameterHttp,
27:                 HTTP_TIMEOUT);
28:             ConnManagerParams.setTimeout(parameterHttp,
29:                 HTTP_TIMEOUT);
30:         }
31:         return client;
32:     }
33:     public static String eksekusiHttpPost(String url,
34:     ArrayList<NameValuePair> postParameter) throws Exception
35:     {
36:         BufferedReader in = null;
37:         try {

```

```

37: HttpClient klien = getHttpClient();
38: HttpPost req = new HttpPost(url);
39: UrlEncodedFormEntity formEntity = new
    UrlEncodedFormEntity(
40:     postParameter);
41: req.setEntity(formEntity);
42: HttpResponse jawaban = klien.execute(req);
43: in = new BufferedReader(new

    InputStreamReader(jawaban.getEntity().getContent()));
44: StringBuffer sb = new StringBuffer("");
45: String line = "";
46: String NL = System.getProperty("line.separator");
47: while ((line = in.readLine()) != null) {
48:     sb.append(line + NL);
49: }
50: in.close();
51: String hasil = sb.toString();
52: return hasil;
53: } finally {
54:     if (in != null) {
55:         in.close();
56:     }
57: }
58:
59: }
60:
61: public static String eksekusiHttpGet(String url) throws
    Exception {
62:     BufferedReader in = null;
63:     try {
64:         HttpClient hc = getHttpClient();
65:         HttpGet req = new HttpGet();
66:         req.setURI(new URI(url));
67:         HttpResponse resp = hc.execute(req);
68:         in = new BufferedReader(new

            InputStreamReader(resp.getEntity().getContent()));
69:         StringBuffer sb = new StringBuffer("");
70:         String line = "";
71:         String NL = System.getProperty("line.separator");
72:         while ((line = in.readLine()) != null) {
73:             sb.append(line + NL);
74:         }
75:         in.close();
76:         String hasil = sb.toString();
77:         return hasil;
78:     } finally {
79:         if (in != null) {
80:             in.close();
81:         }
82:     }
83: }
84: }
85:

```

5. Pada **phplogin.java** ubah kodenya seperti dibawah ini

```

1: package omayib.com.phplogin;
2:
3: import java.util.ArrayList;
4:
5: import org.apache.http.NameValuePair;
6: import org.apache.http.message.BasicNameValuePair;
7:
8: import android.app.Activity;
9: import android.os.Bundle;
10: import android.view.View;
11: import android.view.View.OnClickListener;
12: import android.widget.Button;
13: import android.widget.EditText;
14: import android.widget.Toast;
15:
16: public class Phplogin extends Activity implements
    OnClickListener {
17:     Button loginBtn;
18:     EditText user, pass;
19:     EditText status;
20:     private String url = "http://10.0.2.2/android/login.php";
21:
22:     /** Called when the activity is first created. */
23:     @Override
24:     public void onCreate(Bundle savedInstanceState) {
25:         super.onCreate(savedInstanceState);
26:         setContentView(R.layout.main);
27:
28:         user = (EditText) findViewById(R.id.userInput);
29:         pass = (EditText) findViewById(R.id.passInput);
30:         status = (EditText) findViewById(R.id.status);
31:         loginBtn = (Button) findViewById(R.id.button1);
32:         loginBtn.setOnClickListener(this);
33:     }
34:
35:     @Override
36:     public void onClick(View v) {
37:         // TODO Auto-generated method stub
38:         kirimData();
39:     }
40:
41:     private void kirimData() {
42:         // TODO Auto-generated method stub
43:         ArrayList<NameValuePair> kirimkephp = new
44:         ArrayList<NameValuePair>();
45:         kirimkephp.add(new
46:         BasicNameValuePair("auser",user.getText().toString()));
47:         kirimkephp.add(new
48:         BasicNameValuePair("apass",pass.getText().toString()));
49:         String respon=null;
50:         try {
51:             respon=ClientToServer.eksekusiHttpPost(url,
52:             kirimkephp);
53:             String res=respon.toString();
54:             res=res.trim();
55:             Toast.makeText(this, res.toString(),
56:             Toast.LENGTH_SHORT).show();
57:             status.setText(res.toString());

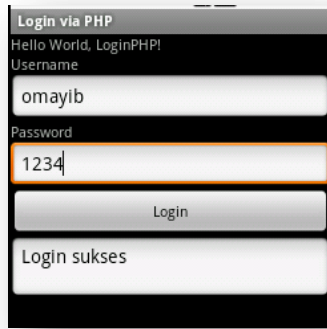
```

```
58:     } catch (Exception e) {
59:         // TODO Auto-generated catch block
60:         e.printStackTrace();
61:     }
62: }
63: }
```

6. Bagian **AndroidManifest.xml**, tambahkan **permission.internet**

```
1: <?xml version="1.0" encoding="utf-8"?>
2: <manifest
   xmlns:android="http://schemas.android.com/apk/res/android"
3:     package="com.login.php"
4:     android:versionCode="1"
5:     android:versionName="1.0">
6:     <uses-sdk android:minSdkVersion="8" />
7:
8:     <uses-permission
9:         android:name="android.permission.INTERNET">
10:     </uses-permission>
11:
12:     <application android:icon="@drawable/icon"
13:         android:label="@string/app_name">
14:         <activity android:name=".LoginPHP"
15:             android:label="@string/app_name">
16:             <intent-filter>
17:                 <action
18:                     android:name="android.intent.action.MAIN"
19:                     />
20:                 <category
21:                     android:name="android.intent.category.LAUNCHER"
22:                     />
23:             </intent-filter>
24:         </activity>
25:     </application>
26: </manifest>
```

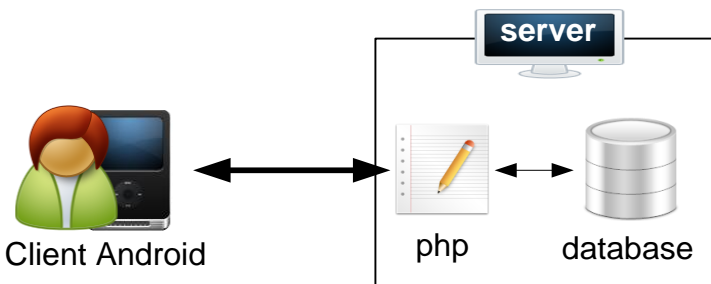
7. Lakukan run, jika berhasil maka hasilnya seperti gambar 16.3



Gambar 16.3 Hasil akhir project phplogin

B. Login menggunakan database

Nah sekarang skenarionya kita ubah sedikit. File PHP bertugas mencocokkan data dari android dengan data di database. Dengan lain kata dalam bagian ini file PHP berfungsi sebagai jembatan antara client android dengan database di server.



Gambar 16.4. Skenario login menggunakan database

Secara garis besar, langkah-langkah yang harus dilakukan sbbagai berikut

1. Membuat database di server localhost
2. Membuat php
3. Membuat project di Android

Membuat database di server localhost

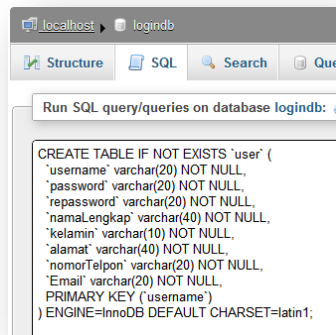
1. Jalankan xampp
2. Jalankan browser, ketik "localhost/phpmyadmin"
3. Membuat database, pada bagian SQL ketikkan sebagai berikut

```
Create database logindb;
```


4. Klik database logindb, kemudian kita akan membuat table user. Pada bagian SQL ketikkan sebagai berikut

```
CREATE TABLE IF NOT EXISTS `user` (
  `_id` int(4) NOT NULL,
  `username` varchar(32) NOT NULL,
  `password` varchar(32) NOT NULL,
  `email` varchar(32) NOT NULL,
  `akun twitter` varchar(32) NOT NULL
) ENGINE=MyISAM DEFAULT CHARSET=latin1;
```

Untuk lebih jelasnya, langkah 4 ini tampak seperti gambar 6.15.



Gambar 16.5 SQL untuk membuat table user

5. Memasukkan data ke dalam database

```
INSERT INTO `logindb`.`user` (`_id`, `username`, `password`, `email`, `akun twitter`)
VALUES (
  '1', 'omayib', '1234r', 'omayib@gmail.com', '@omayib'
);
```

Jika sudah benar, maka hasil akhirnya seperti gambar 16.6

_id	username	password	email	akun twitter
1	omayib	1234r	omayib@gmail.com	@omayib

Gambar 16.6 Tabel user pada database dan datanya

Membuat php

Langkah selanjutnya membuat file **logindb.php** yang nantinya bertugas untuk mencocokkan data dari server dan client. Berikut ini adalah **logindb.php**. Simpan file logindb.php di direktori c:/xampp/htdocs/android/.

```

1: <?php
2: $user=$_POST['user'];
3: $pass=$_POST['pass'];
4:
5: $koneksi=mysql_connect("localhost","root","");
6: mysql_select_db("latihanDB");
7:
8: $query="SELECT * FROM user WHERE username='$user' AND
password='$pass'";
9: $hasil=mysql_query($query) or die("Kesalahan query
: ".mysql_error());
10:
11: if(mysql_num_rows($hasil)==1){
12:     echo 1;
13: }else{
14:     echo 0;
15: }
16: ?>

```

Membuat project di Android

Bagian ini masih menggunakan project login sebelumnya yaitu project PHPLogin. Hanya saja url diganti seperti ini
Sebelumnya

```

private String url =
"http://10.0.2.2/android/login.php";

```

Kemudian diganti

```

private String url =
"http://10.0.2.2/android/loginDB.php";

```

JAM 18 | ANDROID dan MYSQL

Apabila Ebook ini bermanfaat buat Kamu, Bantu penulis menyempurnakan ebook ini dengan cara **DONASI.**

**Kirim TESTIMONI, KRITIK, SARAN
Kamu di
www.odayib.com/testimoni**

BAB 8

ANDROID HADWARE

ImagineIT Education Center membuka kelas
Android Application Development
Mau?? 😊

**Kirim TESTIMONI, KRITIK, SARAN
Kamu di
www.odayib.com/testimoni**

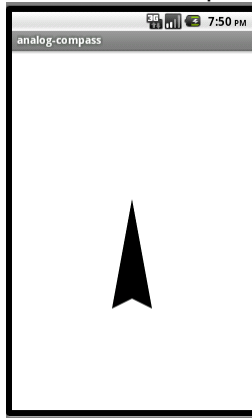
JAM 19 | COMPASS ANDROID

Bagian ini, Kita belajar mengakses

1. Compass
2. Camera <not yet available>
3. Sensor <not yet available>

A. Analog Compass

Waah, tak terasa metri belajar kita sudah jauh. Sekarang saatnya kita belajar mengakses hardware Android. Kita belajar dari yang paling sederhana yaitu membuat analog compass, hasil akhir project ini bisa Kamu lihat pada Gambar 18.1



Gambar 18.1. Analog Compass

Kita tidak memerlukan layout main.xml seperti project-project sebelumnya, karena gambar segitiga diatas dibuat di java menggunakan canvas. Yuuk langsung kita coba..

1. Bikin projectnya dulu

Project name	Compass
Build Target	Android 2.2
Aplication name	Analog Compass
Package name	Cont.AnalogCompass
Create Activity	AnalogCompass
Min SDK version	8

2. Langsung buka activity AnalogCompass.java, lalu ketik barisan kode berikut.

```
1: package cont.AnalogCompass;
2:
3: import android.app.Activity;
4: import android.os.Bundle;
5: import android.content.Context;
6: import android.graphics.*;
```

```

7: import android.hardware.Sensor;
8: import android.hardware.SensorEvent;
9: import android.hardware.SensorEventListener;
10: import android.hardware.SensorManager;
11: import android.util.Config;
12: import android.util.Log;
13: import android.view.View;
14:
15: public class AnalogCompass extends Activity {
16:     private static final String TAG = "Compass";
17:
18:     private SensorManager mSensorManager;
19:     private Sensor mSensor;
20:     private SampleView mView;
21:     private float[] mValues;
22:
23:     private final SensorEventListener mListener = new
SensorEventListener() {
24:         public void onSensorChanged(SensorEvent event) {
25:             if (Config.DEBUG)
26:                 Log.d(TAG, "sensorChanged (" +
event.values[0] + ", "
27:                                     + event.values[1] + ",
" + event.values[2] + ")");
28:             mValues = event.values;
29:             if (mView != null) {
30:                 mView.invalidate();
31:             }
32:         }
33:
34:         public void onAccuracyChanged(Sensor sensor, int
accuracy) {
35:         }
36:     };
37:
38:     /** Called when the activity is first created. */
39:     @Override
40:     public void onCreate(Bundle savedInstanceState) {
41:         super.onCreate(savedInstanceState);
42:         // setContentView(R.layout.main);
43:         mSensorManager = (SensorManager)
getSystemService(Context.SENSOR_SERVICE);
44:         mSensor =
mSensorManager.getDefaultSensor(Sensor.TYPE_ORIENTATION);
45:         mView = new SampleView(this);
46:         setContentView(mView);
47:
48:     }
49:
50:     @Override
51:     protected void onResume() {
52:         if (Config.DEBUG)
53:             Log.d(TAG, "onResume");
54:         super.onResume();
55:
56:         mSensorManager.registerListener(mListener, mSensor,
SensorManager.SENSOR_DELAY_GAME);
57:
58:     }
59:
60:     @Override
61:     protected void onStop() {
62:         if (Config.DEBUG)
63:             Log.d(TAG, "onStop");
64:         mSensorManager.unregisterListener(mListener);
65:         super.onStop();
66:     }

```

```

67:
68:     private class SampleView extends View {
69:         private Paint mPaint = new Paint();
70:         private Path mPath = new Path();
71:         private boolean mAnimate;
72:
73:         public SampleView(Context context) {
74:             super(context);
75:
76:             // Construct a wedge-shaped path
77:             mPath.moveTo(0, -50);
78:             mPath.lineTo(-20, 60);
79:             mPath.lineTo(0, 50);
80:             mPath.lineTo(20, 60);
81:             mPath.close();
82:         }
83:
84:         @Override
85:         protected void onDraw(Canvas canvas) {
86:             Paint paint = mPaint;
87:
88:             canvas.drawColor(Color.WHITE);
89:
90:             paint.setAntiAlias(true);
91:             paint.setColor(Color.BLACK);
92:             paint.setStyle(Paint.Style.FILL);
93:
94:             int w = canvas.getWidth();
95:             int h = canvas.getHeight();
96:             int cx = w / 2;
97:             int cy = h / 2;
98:
99:             canvas.translate(cx, cy);
100:             if (mValues != null) {
101:                 canvas.rotate(-mValues[0]);
102:             }
103:             canvas.drawPath(mPath, mPaint);
104:         }
105:
106:         @Override
107:         protected void onAttachedToWindow() {
108:             mAnimate = true;
109:             if (Config.DEBUG)
110:                 Log.d(TAG,
111:                     "onAttachedToWindow. mAnimate=" + mAnimate);
112:             super.onAttachedToWindow();
113:         }
114:
115:         @Override
116:         protected void onDetachedFromWindow() {
117:             mAnimate = false;
118:             if (Config.DEBUG)
119:                 Log.d(TAG,
120:                     "onDetachedFromWindow. mAnimate=" + mAnimate);
121:             super.onDetachedFromWindow();
122:         }
123:     }

```

JAM 20 | SENSOR ANDROID

<not yet available>

Apabila Ebook ini bermanfaat buat Kamu, Bantu penulis menyempurnakan ebook ini dengan cara **DONASI.**

ImagineIT Education Center membuka kelas
Android Application Development
Mau?? 😊

**Kirim TESTIMONI, KRITIK, SARAN
Kamu di
www.odayib.com/testimoni**

JAM 21 | CAMERA ANDROID

<not yet available>

ImagineIT Education Center membuka kelas
Android Application Development
Mau?? ☺

Apabila Ebook ini bermanfaat buat Kamu, Bantu
penulis menyempurnakan ebook ini dengan cara
DONASI.

JAM 22 | BLUETOOTH ANDROID

<not yet available>

ImagineIT Education Center membuka kelas
Android Application Development
Mau?? 😊

Apabila Ebook ini bermanfaat buat Kamu, Bantu
penulis menyempurnakan ebook ini dengan cara
DONASI.

BAB9

Membangun Aplikasi Android

JAM 23-24

Dapatkan materi terbaru di
www.omyib.com

131

Bagian ini kita akan membuat aplikasi-aplikasi sederhana

1. Aplikasi "DISKON"
2. Aplikasi GUITAR TUNER"
3. Aplikasi Peta Digital Kota Solo

A. Aplikasi Diskon

Bagian ini kita belajar merancang sebuah aplikasi yang berguna bagi orang umum. Usahakan dalam mengembangkan aplikasi, kita berangkat dari masalah sederhana dalam kehidupan sehari-hari.

Apa sih yang paling dicari setiap orang saat berbelanja? Yah! DISKON!! Barang semahal apapun kalau dikasih label DISKON!! Pasti dilirik orang. Betul?? Masalahnya tidak semua orang bisa menghitung harga setelah kena diskon dengan cepat. Nah, kita sebagai orang IT bisa dong membantu orang-orang ini menghitung dengan cepat (selain bertanya pada pelayan,hehehe..).

Ini lah ide sederhana untuk membuat aplikasi DISKON!!. Skenarionya apabila aplikasi di-launch, disediakan 2 buah editteks untuk memasukkan harga awal dan besar diskon. Kemudian sebuah tombol, jika diklik maka muncul nilai berapa harga setelah kena diskon. Lihat Gambar 9.1 dulu ya biar tau hasil akhirnya nanti. Yuuk langsung kita buat.



Gambar 20.1. Hasil Akhir aplikasi DISKON!!

1. Bikin project baru dulu ya
2. Masukkan icon gambar apa saja yang mewakili ikon DISKON. Gambar ini akan kita jadikan button, simpan file icon di folder res/drawable, beri nama file diskon.png
3. Tambahkan resources di Strings.xml seperti berikut

```
1: <?xml version="1.0" encoding="utf-8"?>
2: <resources>
3:     <string name="hello">HargaDiskon!</string>
4:     <string name="app_name">Diskon!</string>
5:     <color name="trans">#00330000</color>
6:     <drawable name="klik">#CCFFFFFF</drawable>
```

7: </resources>

4. Sekarang kita bermain koding dulu di main.xml

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <ScrollView android:id="@+id/scrollView1"
3: xmlns:android="http://schemas.android.com/apk/res/android"
4: android:layout_width="wrap_content"
5: android:layout_height="wrap_content">
6: <TableLayout android:orientation="vertical"
7:     android:layout_height="wrap_content"
8:     android:layout_width="fill_parent">
9:     <TableRow android:layout_width="fill_parent"
10:         android:layout_height="wrap_content"
11:         android:id="@+id/tableRow1"
12:         android:layout_weight="1">
13:         <TextView android:id="@+id/textView1"
14:             android:layout_width="fill_parent"
15:             android:layout_height="wrap_content"
16:             android:text="Harga Asli    Rp."></TextView>
17:         <EditText android:text=""
18:             android:layout_height="wrap_content"
19:             android:id="@+id/harga"
20:             android:inputType="numberDecimal"
21:             android:layout_width="40dip"></EditText>
22:         <TextView android:text=","00"
23:             android:id="@+id/textView2"
24:             android:layout_width="wrap_content"
25:             android:layout_height="wrap_content"></TextView>
26:     </TableRow>
27:     <TableRow android:layout_width="fill_parent"
28:         android:layout_height="wrap_content"
29:         android:id="@+id/tableRow1"
30:         android:layout_weight="1">
31:         <TextView android:id="@+id/textView1"
32:             android:layout_width="fill_parent"
33:             android:layout_height="wrap_content"
34:             android:text="Besar Diskon "></TextView>
35:         <EditText android:text=""
36:             android:layout_height="wrap_content"
37:             android:id="@+id/diskon"
38:             android:inputType="numberDecimal"
39:             android:layout_width="50dip"></EditText>
40:         <TextView android:text="%"
41:             android:id="@+id/textView2"
42:             android:layout_width="wrap_content"
43:             android:layout_height="wrap_content"></TextView>
44:     </TableRow>
45:     <View android:background="#FF909090"
46:         android:layout_width="wrap_content"
47:         android:id="@+id/view1"
48:         android:layout_height="2dip"></View>
49:     <TextView android:id="@+id/tekshitung"
50:         android:layout_width="fill_parent"
51:         android:layout_height="wrap_content"
52:         android:gravity="center_horizontal"
53:         android:text=".: Hitung .:"></TextView>
54:     <View android:background="#FF909090"
55:         android:layout_width="wrap_content"
56:         android:id="@+id/view1"
57:         android:layout_height="2dip"></View>
58:     <TableRow android:id="@+id/tableRow2"
59:         android:layout_width="wrap_content"
60:         android:layout_height="wrap_content">
61:         <ImageButton android:layout_width="wrap_content"
62:             android:layout_height="wrap_content"

```

```

63:         android:src="@drawable/diskon"
64:         android:id="@+id/imageButton1"
65:         android:background="@color/trans"
66:         android:paddingTop="10dip"
67:         android:paddingBottom="10dip"></ImageButton>
68:     <Button android:text="Reset"
69:         android:id="@+id/reset"
70:         android:layout_marginLeft="50dip"
71:         android:layout_width="wrap_content"
72:         android:layout_marginTop="7dip"
73:         android:gravity="center_vertical|left"
74:         android:tag="reset all"
75:         android:layout_gravity="left|center vertical"
76:         android:layout_height="wrap_content"></Button>
77: </TableRow>
78: <View android:background="#FF909090"
79:     android:layout_width="wrap_content"
80:     android:id="@+id/view1"
81:     android:layout_height="2dip"></View>
82:
83: <TableRow android:id="@+id/tableRow2"
84:     android:layout_width="wrap_content"
85:     android:layout_height="wrap_content">
86:     <TextView android:id="@+id/textView1"
87:         android:layout_width="fill_parent"
88:         android:layout_height="wrap_content"
89:         android:text="Harga sekarang :"
90:         android:maxLines="2"
91:         android:textStyle="bold"
92:         android:textSize="15dip"></TextView>
93:     <TextView android:text=""
94:         android:layout_height="wrap_content"
95:         android:id="@+id/hasil"
96:         android:inputType="numberDecimal"
97:         android:textStyle="bold"
98:         android:textSize="20dip"
99:         android:layout_width="wrap_content"></TextView>
100: </TableRow>
101: <TextView android:layout_width="fill_parent"
102:     android:layout_height="wrap_content"
103:     android:maxLines="2"
104:     android:textStyle="bold"
105:     android:textSize="15dip"
106:     android:id="@+id/hemat"
107:     android:gravity="center_horizontal"
108:     android:paddingTop="10dip"></TextView>
109: </TableLayout>
110: </ScrollView>

```

Buka activity kemudian tulis kode berikut ini

```

1: package com.diskon;
2:
3: import java.text.DecimalFormat;
4:
5: import android.app.Activity;
6: import android.app.AlertDialog;
7: import android.content.DialogInterface;
8: import android.os.Bundle;
9: import android.text.Editable;
10: import android.view.Menu;
11: import android.view.MenuItem;
12: import android.view.View;
13: import android.view.View.OnClickListener;
14: import android.widget.Button;
15: import android.widget.EditText;

```

```

16: import android.widget.ImageButton;
17: import android.widget.TextView;
18:
19: public class hargaDiskon extends Activity {
20:     EditText harga, diskon;
21:     TextView hargabaru, hmt;
22:     Editable isiharga, isidiskon;
23:     Button reset;
24:     ImageButton htng;
25:     String sharga, sdiskon;
26:
27:     /** Called when the activity is first created. */
28:     @Override
29:     public void onCreate(Bundle savedInstanceState) {
30:         super.onCreate(savedInstanceState);
31:         setContentView(R.layout.main);
32:
33:         hargabaru = (TextView) findViewById(R.id.hasil);
34:         harga = (EditText) findViewById(R.id.harga);
35:         diskon = (EditText) findViewById(R.id.diskon);
36:         hmt = (TextView) findViewById(R.id.hemat);
37:         reset = (Button) findViewById(R.id.reset);
38:         reset.setOnClickListener(new reset());
39:         htng = (ImageButton) findViewById(R.id.imageButton1);
40:         htng.setOnClickListener(new itung());
41:
42:     }
43:
44:     private class itung implements OnClickListener {
45:         public void onClick(View v) {
46:             try {
47:                 Double h =
48:                     Double.parseDouble(harga.getText().toString());
49:                 Double d =
50:                     Double.parseDouble(diskon.getText().toString());
51:                 Double nd = (d / 100) * h;
52:                 double hsl = h - nd;
53:                 Number irit = h - hsl;
54:                 DecimalFormat df = new DecimalFormat("@@##");
55:                 hargabaru.setText("Rp." + df.format(hsl) + ",00");
56:                 hmt.setText("Wow!! Kamu Hemat Rp."
57:                     + df.format(irit) + " !!");
58:             } catch (Exception e) {
59:             };
60:         }
61:     }
62:
63:     public class reset implements OnClickListener {
64:         public void onClick(View v) {
65:             hargabaru.setText("");
66:             harga.setText("");
67:             diskon.setText("");
68:             hmt.setText("");
69:         }
70:     }
71:
72:     public boolean onCreateOptionsMenu(Menu menu) {
73:         super.onCreateOptionsMenu(menu);
74:         menu.add(0, 0, 0, "Tutorial");
75:         menu.add(0, 1, 0, "Info");
76:         menu.add(0, 2, 0, "Keluar");
77:         return true;
78:     }
79:
80:     public boolean onOptionsItemSelected(MenuItem item) {
81:

```

```

82:     if (item.getItemId() == 0) {
83:         new AlertDialog.Builder(this)
84:             .setTitle("Info Aplikasi")
85:             .setMessage("1. Masukkan harga asli"
86:                 + "\n2. Masukkan besar diskon"
87:                 + "\n3. Klik gambar HITUNG DISKON!"
88:                 + "\n4. Klik tombol Reset untuk mengosongkan data")
89:             .setNeutralButton("Kembali",
90:                 new DialogInterface.OnClickListener() {
91:                     @Override
92:                     public void onClick(DialogInterface arg0,
93:                         int arg1) { //
94:                         // TODO Auto-generated method stub
95:                     }
96:                 }).show();
97:     } else if (item.getItemId() == 1) {
98:         new AlertDialog.Builder(this)
99:             .setTitle("Info Aplikasi")
100:             .setMessage(
101:                 "Aplikasi DISKON! dibuat oleh OmAyib2011."
102:                 + "Kunjungi www.omayib.com.")
103:             .setNeutralButton("Kembali",
104:                 new DialogInterface.OnClickListener() {
105:                     @Override
106:                     public void onClick(DialogInterface arg0,
107:                         int arg1) { //
108:                         // TODO Auto-generated method stub
109:                     }
110:                 }).show();
111:     } else if (item.getItemId() == 2) {
112:         hargaDiskon.this.finish();
113:     }
114:     return true;
115: }

```

Selesai! Silakan diRUN, semoga tidak ada yang error 😊

**Kirim TESTIMONI, KRITIK, SARAN
Kamu di
www.omayib.com/testimoni**

B. Aplikasi Gitar TUNER

Siapa sih yang tidak kenal gitar? Dari yang muda sampai tua, dari desa hingga kota hampir setiap saat kita menemukan orang bermain gitar. Masalahnya kadang mereka tidak tau apakah nada gitarnya sudah pas sedangkan harga efek atau tuner gitar relative mahal.



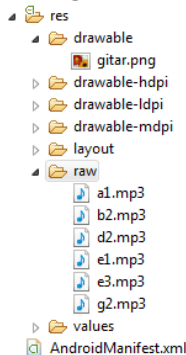
Tidak ada salahnya kita sebagai orang IT, membantu mereka membuat aplikasi sederhana untuk mencocokkan nada-nada gitar mereka secara standar.

Aplikasi ini dilengkapi 6 button masing-masing mewakili 1 senar, dan sebuah button untuk keluar. Jika button "Senar A" di klik, maka android mengeluarkan bunyi senar gitar nada A. Kira-kira seperti itulah idenya.

Untuk project ini, silakan mendownload dari internet bunyi string gitar (www.omayib.com/downloads).

Mari, sekarang kita lihat code aplikasinya

1. Bikin project baru
2. Bikin folder baru /res/raw. Paste file suara string gitar didalam folder ini, lihat gambar



Gambar 21.1. Isi folder /res/raw

3. Siapkan dulu data string di **Strings.xml**

```

1:  <?xml version="1.0" encoding="utf-8"?>
2:  <resources>
3:    <string name="hello">Pilih salah satu string</string>
4:    <string name="app_name">Guitar Tuner </string>
5:    <color name="bg">#ffffff</color>
6:    <color name="textColor">#ff0000</color>
7:    <color name="textColor2">#000000</color>
8:    <string name="e1">Senar E</string>
9:    <string name="a">Senar A</string>
10:   <string name="d">Senar D</string>

```

```

11: <string name="g">Senar G</string>
12: <string name="b">Senar B</string>
13: <string name="e2">Senar E</string>
14: <string name="keluarBtn">Keluar</string>
15: <string name="Deskripsi_Gambar">Guitar Image</string>
16: </resources>

```

4. Sekarang kita bikin layoutnya di **Main.xml**

```

1: <?xml version="1.0" encoding="utf-8"?>
2: <ScrollView
  xmlns:android="http://schemas.android.com/apk/res/android"
3:   android:id="@+id/scrollView1"
4:   android:layout_width="fill_parent"
5:   android:layout_height="fill_parent" >
6:
7:   <LinearLayout
8:     android:layout_width="fill_parent"
9:     android:layout_height="wrap_content"
10:    android:background="@color/bg"
11:    android:orientation="vertical" >
12:
13:    <TextView
14:      android:layout_width="fill_parent"
15:      android:layout_height="wrap_content"
16:      android:fitsSystemWindows="true"
17:      android:focusable="true"
18:      android:focusableInTouchMode="true"
19:      android:gravity="top"
20:      android:text="@string/hello"
21:      android:textColor="@color/textColor2" />
22:
23:    <ImageView
24:      android:id="@+id/imageView1"
25:      android:layout_width="wrap_content"
26:      android:layout_height="wrap_content"
27:      android:layout_gravity="center_horizontal"
28:      android:contentDescription="@string/Deskripsi_Gb"
29:      android:scaleType="center"
30:      android:src="@drawable/gitar" />
31:
32:    <Button
33:      android:id="@+id/stringE1"
34:      android:layout_width="fill_parent"
35:      android:layout_height="wrap_content"
36:      android:text="@string/e1" >
37:    </Button>
38:
39:    <Button
40:      android:id="@+id/stringA"
41:      android:layout_width="fill_parent"
42:      android:layout_height="wrap_content"
43:      android:text="@string/a" >
44:    </Button>
45:
46:    <Button
47:      android:id="@+id/stringD"
48:      android:layout_width="fill_parent"
49:      android:layout_height="wrap_content"
50:      android:text="@string/d" >
51:    </Button>
52:
53:    <Button
54:      android:id="@+id/stringG"
55:      android:layout_width="fill_parent"
56:      android:layout_height="wrap_content"

```

```

57:         android:text="@string/g" >
58:     </Button>
59:
60:     <Button
61:         android:id="@+id/stringB"
62:         android:layout_width="fill_parent"
63:         android:layout_height="wrap_content"
64:         android:text="@string/b" >
65:     </Button>
66:
67:     <Button
68:         android:id="@+id/stringE"
69:         android:layout_width="fill_parent"
70:         android:layout_height="wrap_content"
71:         android:text="@string/e2" >
72:     </Button>
73:
74:     <TextView
75:         android:id="@+id/keterangan"
76:         android:layout_width="fill_parent"
77:         android:layout_height="wrap_content"
78:         android:layout_gravity="center_horizontal"
79:         android:text=""
80:         android:textColor="@color/textColor"
81:         android:textStyle="bold" >
82:     </TextView>
83:
84:     <Button
85:         android:id="@+id/keluar"
86:         android:layout_width="wrap_content"
87:         android:layout_height="wrap_content"
88:         android:layout_gravity="right"
89:         android:text="@string/keluarBtn" >
90:     </Button>
91:
92: </LinearLayout>
93:
94: </ScrollView>

```

5. Berikut adalah activity tunerGitar.java

```

1: package code.tunerGitar;
2:
3: import android.app.Activity;
4: import android.media.MediaPlayer;
5: import android.os.Bundle;
6: import android.view.View;
7: import android.view.View.OnClickListener;
8: import android.widget.Button;
9: import android.widget.TextView;
10: import android.widget.Toast;
11:
12: public class tunerGitar extends Activity {
13:
14:     private static final String isPlaying = "Kamu
15:         mendengarkan ";
16:     private MediaPlayer player;
17:     private Button EButton;
18:     private Button BButton;
19:     private Button GButton;
20:     private Button DButton;
21:     private Button AButton;
22:     private Button E2Button;
23:     private Button keluar;
24:     TextView keterangan;
25:

```

```

26:  /** Called when the activity is first created. */
27:  @Override
28:  public void onCreate(Bundle savedInstanceState) {
29:      super.onCreate(savedInstanceState);
30:      setContentView(R.layout.main);
31:      EButton = (Button) this.findViewById(R.id.stringE);
32:      EButton.setOnClickListener(new OnClickListener() {
33:          @Override
34:          public void onClick(View arg0) {
35:              playSound(1);
36:          }
37:      });
38:      BButton = (Button) this.findViewById(R.id.stringB);
39:      BButton.setOnClickListener(new OnClickListener() {
40:          @Override
41:          public void onClick(View arg0) {
42:              playSound(2);
43:          }
44:      });
45:      GButton = (Button) this.findViewById(R.id.stringG);
46:      GButton.setOnClickListener(new OnClickListener() {
47:          @Override
48:          public void onClick(View arg0) {
49:              playSound(3);
50:          }
51:      });
52:      DButton = (Button) this.findViewById(R.id.stringD);
53:      DButton.setOnClickListener(new OnClickListener() {
54:          @Override
55:          public void onClick(View arg0) {
56:              playSound(4);
57:          }
58:      });
59:      AButton = (Button) this.findViewById(R.id.stringA);
60:      AButton.setOnClickListener(new OnClickListener() {
61:          @Override
62:          public void onClick(View arg0) {
63:              playSound(5);
64:          }
65:      });
66:      E2Button = (Button) this.findViewById(R.id.stringE1);
67:      E2Button.setOnClickListener(new OnClickListener() {
68:          @Override
69:          public void onClick(View arg0) {
70:              playSound(6);
71:          }
72:      });
73:      keluar = (Button) this.findViewById(R.id.keluar);
74:      keluar.setOnClickListener(new OnClickListener() {
75:          @Override
76:          public void onClick(View arg0) {
77:              tunerGitar.this.finish();
78:          }
79:      });
80:      keterangan = (TextView)
81:          this.findViewById(R.id.keterangan);
82:  }
83:
84:  @Override
85:  public void onPause() {
86:      try {
87:          super.onPause();
88:          player.pause();
89:      } catch (Exception e) {
90:      }
91:  }

```

```

92:
93:     public void onStop() {
94:         super.onStop();
95:         player.stop();
96:     }
97:
98:     private void playSound(int arg) {
99:         try {
100:             if (player.isPlaying()) {
101:                 player.stop();
102:                 player.release();
103:             }
104:         } catch (Exception e) {
105:             Toast.makeText(this, e.toString(),
106:                 Toast.LENGTH_SHORT).show();
107:         }
108:         if (arg == 1) {
109:             player = MediaPlayer.create(this, R.raw.e3);
110:             keterangan.setText(isPlaying + "String E");
111:         } else if (arg == 2) {
112:             player = MediaPlayer.create(this, R.raw.b2);
113:             keterangan.setText(isPlaying + "String B");
114:         } else if (arg == 3) {
115:             player = MediaPlayer.create(this, R.raw.g2);
116:             keterangan.setText(isPlaying + "String G");
117:         } else if (arg == 4) {
118:             player = MediaPlayer.create(this, R.raw.d2);
119:             keterangan.setText(isPlaying + "String D");
120:         } else if (arg == 5) {
121:             player = MediaPlayer.create(this, R.raw.a1);
122:             keterangan.setText(isPlaying + "String A");
123:         } else if (arg == 6) {
124:             player = MediaPlayer.create(this, R.raw.e1);
125:             keterangan.setText(isPlaying + "String E");
126:         }
127:         player.setLooping(true);
128:         player.start();
129:     }
130: }

```

Selamat bermain gitar ☺

