

# BAB 1

## ANDROID PROGRAMMING

Bab ini telah memformulasikan inti teknik dasar dari Android Programming dengan Java Eclipse dan mengimplementasikannya dalam bentuk sebuah aplikasi manipulasi database sederhana. Aplikasinya hanya terdiri atas 2 Form, yaitu:

1. Form registrasi. Digunakan untuk meng-input sebuah data user account ke database (data yang dimaksud adalah username, password, sex, tempat tinggal, dan hobi).
2. Form login. Digunakan untuk mengecek availability sebuah data user account pada database.

Detail pembuatan form registrasi ada pada Subbab 1.1. Detail pembuatan form login ada pada Subbab 1.2. Menghubungkan kedua form tersebut dengan database tersebut ada pada Subbab 1.3. Jangan melangkah langsung ke Proyek 1 s.d. Proyek 5 (Android) sebelum memahami intinya tersebut.

Berikut adalah inti teknik dasar Android Programming. Ada banyak Proyek Android yang dapat dihasilkan jika menerapkannya. Proyek 1 s.d. Proyek 5 (Android) pada buku ini adalah beberapa buktinya.

**STEP 1.** Siapkan file graphic layout (berupa source code \*.xml) pada folder *res/layout/\*.xml*.

**STEP 2.** Siapkan file class (berupa source code \*.java) pada folder *src/nama\_package/\*.java*. Pada umumnya, file class ini digunakan untuk menampilkan graphic layout (Step1) dan menangani semua event yang mungkin bisa terjadi padanya. Event yang dimaksud di antaranya adalah seperti berikut:

- Event klik menu yang ada di form.
- Event klik button yang ada di form.
- Event klik item listview yang ada di form.
- dan masih banyak event lainnya.

**STEP 3.** Update file menu (berupa source code \*.xml) pada folder *res/menu/\*.xml*. Pada umumnya, hal tersebut bertujuan untuk menyediakan menu redirect ke semua graphic layout yang telah disiapkan di Step1.

**STEP 4.** Update file AndroidManifest (berupa source code \*.xml) supaya apk-nya bisa membuka (tanpa error) semua graphic layout yang telah disiapkan di Step 1, dan juga bisa kompatibel dengan target device HP Android-nya.

**STEP 5.** Build Proyek dan Upload apk-nya ke device.

Dalam DVD buku ini, terdapat folder *android\_basic (adt-bundle-windows-x86\Proyek\_android\android\_basic)*. Folder tersebut berisi aplikasi yang dibahas pada bab ini. Jadi, tidak perlu dibuat dari 0. Namun, tidak ada salahnya untuk mengetahui bagaimana cara membuatnya dari 0 sehingga memberikan gambaran pengimplementasian inti teknik dasar Android Programming tersebut.

## 1.1 Membuat Form (1)

Pada Subbab ini, akan diketahui bagaimana cara membuat sebuah form (graphic layout). Kita bisa menambahkan beragam field (komponen) ke dalam form tersebut, beberapa di antaranya adalah seperti berikut:

TextView	WebView	ScrollView
TextBox	CheckBox	
ComboBox (Spinner)	Button	
ListView	GridView	

Hal tersebut juga dapat diterapkan pada Smartphone Programming lainnya (Blackberry, iOS, Windows Phone, J2ME, dan lainnya).

Mari terapkan inti teknik yang telah dibuat pada awal bab dalam rangka membuat form registrasi. Sebelumnya, pastikan Anda telah mengikuti step instalasi JAVA yang ada pada buku.

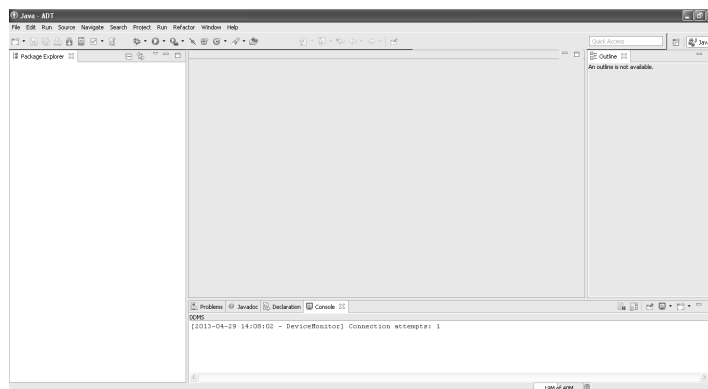
**STEP 1.** Buat folder baru dengan nama *android\_basic* pada directory *C:\adt-bundle-windows-x86\Proyek\_android*.

**STEP 2.** Buka Eclipse-nya di *C:\adt-bundle-windows-x86\eclipse\ eclipse.exe*, muncul **Workspace Launcher** seperti gambar berikut.



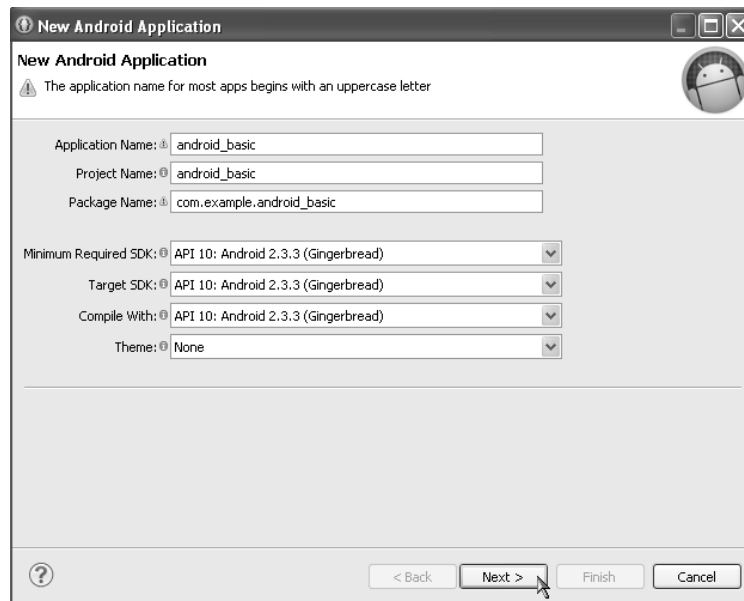
*Gambar 1.1 New Proyek android\_basic*

Ketik *C:\adt-bundle-windows-x86\Proyek\_android\android\_basic* di kotak **Workspace**. Kemudian, klik tombol **OK** sampai muncul ADT-nya seperti pada gambar berikut.



*Gambar 1.2 Area kerja ADT*

**STEP 4.** Buat New Proyek dengan nama *android\_basic*. Caranya: Pada ADT-nya (Gambar 1.2), klik menu **File / New / Android Application Proyek** sampai muncul **New Android Application** seperti gambar berikut.

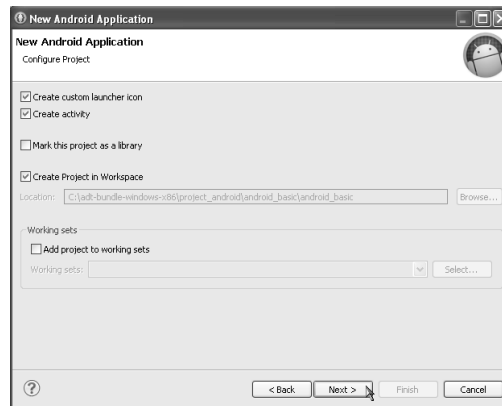


*Gambar 1.3 Setting Proyek android\_basic(1)*

Bila gambarnya terlihat kurang jelas, setting pada gambar tersebut adalah seperti berikut:

<b>Application Name</b>	: android_basic
<b>Proyek Name</b>	: android_basic
<b>Package Name</b>	: com.example.android_basic
<b>Minimum Required SDK</b>	: API 10: Android 2.3.3 (Gingerbread)
<b>Target SDK</b>	: API 10: Android 2.3.3 (Gingerbread)
<b>Compile With</b>	: API 10: Android 2.3.3 (Gingerbread)
<b>Theme</b>	: None

Setelah itu, klik tombol **Next**. Maka, akan muncul tampilan seperti gambar berikut.

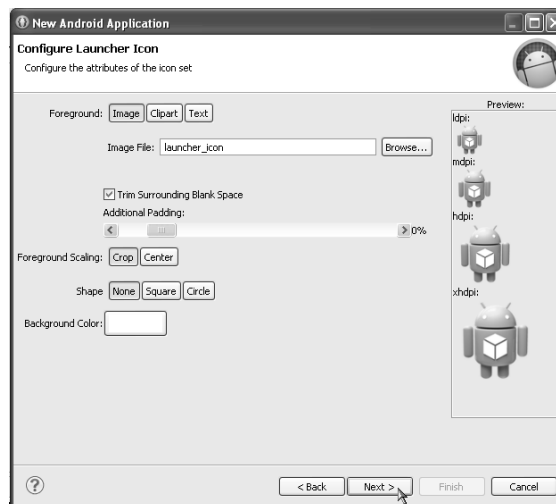


**Gambar 1.4** Setting Proyek android\_basic(2)

Biarkan setting default seperti Gambar 1.5 (bila gambar kurang jelas, setting pada gambar tersebut adalah seperti berikut):

Pastikan checkbox **Create Custom Launcher Icon**, **Create Activity**, dan **Create Proyek in Workspace** ter-contreng.  
**Location:**  
*C:\adt-bundle-windows-x86\Proyek\_android\android\_basic\android\_basic*

Kemudian, klik tombol **Next**. Maka, akan muncul tampilan berikut.

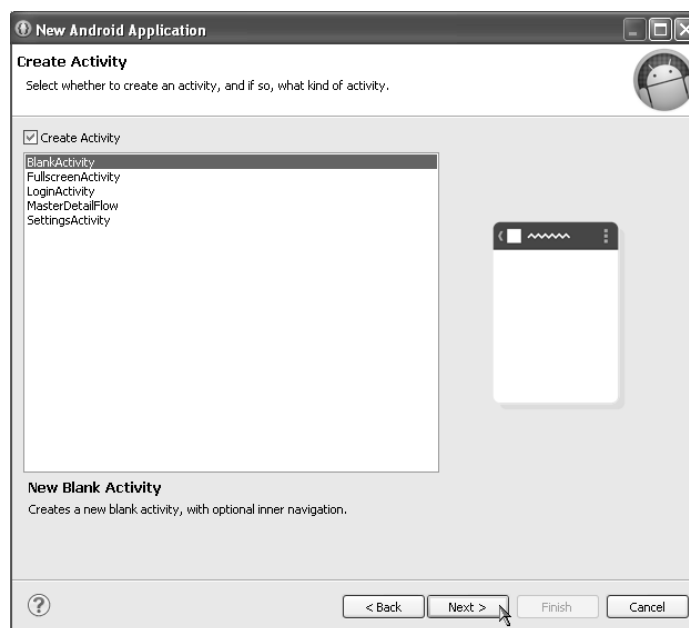


**Gambar 1.5** Setting Proyek android\_basic(3)

Biarkan setting default seperti pada Gambar 1.5, bila gambar kurang jelas, setting pada gambar tersebut adalah seperti berikut:

Pastikan tombol **Image**, **Crop**, **None** dalam mode terklik.  
**Image File:** launcher\_icon.  
Contreng checkbox **Trim Surrounding Blank Space**.  
**Background Color:** white.

Setelah itu, klik tombol **Next**. Maka, akan muncul tampilan seperti gambar berikut.



*Gambar 1.6 Setting Proyek android\_basic(4)*

Biarkan setting default seperti pada Gambar 1.6, bila gambar kurang jelas, setting pada gambar tersebut adalah seperti berikut:

Contreng checkbox **Create Activity**  
Pilih **Blank Activity** yang ada pada List

Setelah itu, klik tombol **Next**. Maka, akan muncul tampilan seperti gambar berikut.

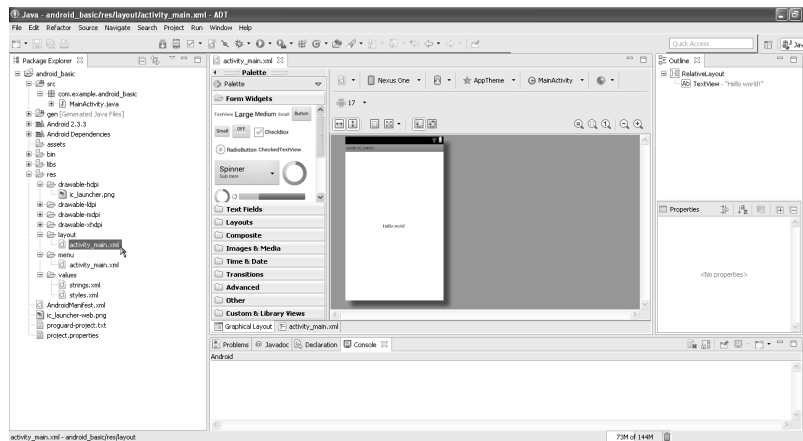


Gambar 1.7 Setting Proyek android\_basic(5)

Biarkan setting default seperti pada Gambar 1.7, bila gambar kurang jelas, setting pada gambar tersebut adalah seperti berikut:

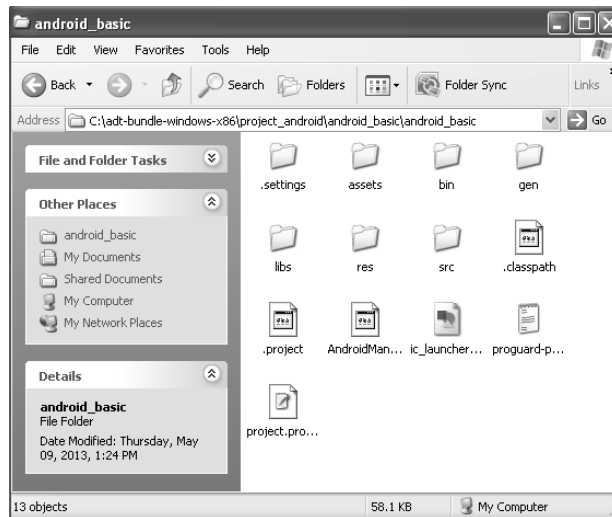
<b>Activity Name</b>	: MainActivity
<b>Layout Name</b>	: activity_main
<b>Navigation Type</b>	: None

Setelah itu, klik tombol **Finish**. Maka, Step 4 selesai. Hal tersebut ditandai dengan munculnya tampilan seperti gambar berikut. Inilah ruang kerja development aplikasi yang dibahas pada Bab1.



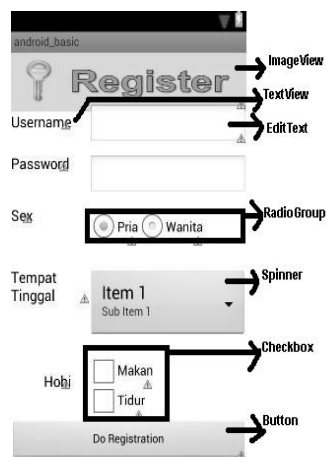
Gambar 1.8 Ruang kerja development aplikasi android\_basic

Pada Gambar 1.8, di sebelah kiri terlihat ada **Package Explorer**. Beberapa itemnya bisa dilihat via Windows Explorer pada direktori `C:\adt-bundle-windows-x86\Proyek_android\android_basic\android_basic`, di antaranya yaitu `res`, `src`, dan `bin` seperti gambar berikut.



**Gambar 1.9** Package Explorer aplikasi android\_basic (via Windows Explorer)

**STEP 5.** Setelah dari Step 4, buatlah desain form registrasi seperti gambar berikut.



**Gambar 1.10** Desain form registrasi (android\_basic)



Untuk mendesain form registrasi seperti Gambar 1.10, maka pada **Package Explorer** (Gambar 1.8), klik 2x pada file *activity\_main.xml* yang ada di dalam folder *res/layout*, sampai di sebelah kanannya muncul tampilan **Graphic Layout** yang masih kosong (seperti Gambar 1.8).

Pada tampilan **Graphic Layout** yang masih kosong tersebut, persis di sebelah kirinya ada **Palette** seperti gambar berikut.



**Gambar 1.11** *Palette*

Persis di sebelah bawah **Palette** tersebut ada 2 tab, yaitu tab *Graphic Layout* dan tab *activity\_main.xml*. Bila seperti Gambar 1.8, maka sekarang kita sedang berada pada tab *Graphic Layout*. Untuk melihat source code dari tampilan **Graphic Layout** yang masih kosong tersebut, klik saja tab *activity\_main.xml*. Pastikan source codenya adalah seperti berikut.

1	<code>&lt;RelativeLayout</code>
2	<code>xmlns:android="http://schemas.android.com/apk/res/android"</code>
3	<code>xmlns:tools="http://schemas.android.com/tools"</code>
4	<code>android:layout_width="match_parent"</code>
5	<code>android:layout_height="match_parent"</code>
6	<code>tools:context=".MainActivity" &gt;</code>
	<code>&lt;/RelativeLayout&gt;</code>

Kembali lagi ke tab Graphic Layout. Pada tab Graphic Layout, tambahkanlah satu persatu field (komponen) dari Pallette ke tampilan Graphic Layout yang masih kosong tersebut.

Misalnya, kita akan menambahkan button 'Do Registration' ke tampilan Graphic Layout yang masih kosong tersebut. Maka, pada Pallette, klik dan drag icon Button ke tampilan Graphic Layout yang masih kosong.

Atur letaknya sebisa mungkin seperti Gambar 1.10. Pastikan akibat penambahan button tersebut, source code dari tampilan Graphic Layout (pada tab *activity\_main.xml*) otomatis berubah menjadi seperti berikut.

1	<code>&lt;RelativeLayout</code>
	<code>xmlns:android="http://schemas.android.com/apk/res/android"</code>
2	<code>xmlns:tools="http://schemas.android.com/tools"</code>
3	<code>android:layout_width="match_parent"</code>
4	<code>android:layout_height="match_parent"</code>
5	<code>tools:context=".MainActivity" &gt;</code>
6	
7	<code>&lt;Button</code>
8	<code>android:id="@+id/button1"</code>
9	<code>android:layout_width="wrap_content"</code>
10	<code>android:layout_height="wrap_content"</code>
11	<code>android:layout_alignParentBottom="true"</code>
12	<code>android:layout_alignParentLeft="true"</code>
13	<code>android:layout_alignParentRight="true"</code>
14	<code>android:text="Do Registration" /&gt;</code>
15	
16	<code>&lt;/RelativeLayout&gt;</code>

Terlihat bahwa button 'Do Registration' yang baru saja ditambahkan ke tampilan Graphic Layout secara otomatis dinamai *button1*. Hal tersebut dilihat dari deklarasi yang ada pada *android:id*-nya (baris 8).

Dari contoh menambahkan button 'Do Registration' ke tampilan Graphic Layout yang masih kosong, terlihat bahwa setiap penambahan field (komponen) ke tampilan Graphic Layout akan otomatis meng-update source code Graphic Layout tersebut.

Karena sudah tau kosepnya seperti itu, maka untuk hemat waktu pembahasan langsung ubah semua source code-nya dengan yang berikut, sehingga tampilannya semirip mungkin seperti Gambar 1.10.

1	<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
2	xmlns:tools="http://schemas.android.com/tools"
3	android:layout width="match parent"
4	android:layout height="match parent"
5	tools:context=".MainActivity" >
6	
7	<ImageView
8	android:id="@+id/imageView1"
9	android:layout width="70dp"
10	android:layout height="70dp"
11	android:layout alignParentLeft="true"
12	android:layout alignParentRight="true"
13	android:scaleType="fitXY"
14	android:src="@drawable/ico_register" />
15	
16	<EditText
17	android:id="@+id/editText1"
18	android:layout width="wrap content"
19	android:layout height="wrap content"
20	android:layout alignBaseline="@+id/textView1"
21	android:layout alignBottom="@+id/textView1"
22	android:layout alignParentRight="true"
23	android:ems="10" />
24	
25	<EditText
26	android:id="@+id/editText2"
27	android:layout width="wrap content"
28	android:layout height="wrap content"
29	android:layout alignLeft="@+id/editText1"
30	android:layout alignTop="@+id/textView3"
31	android:ems="10"
32	android:inputType="textPassword" />
33	
34	<RadioGroup
35	android:id="@+id/radioGroup1"
36	android:layout width="wrap content"
37	android:layout height="wrap content"
38	android:layout alignLeft="@+id/editText2"
39	android:layout alignTop="@+id/textView2" >
40	<RadioButton
41	android:id="@+id/radio0"
42	android:layout width="wrap content"
43	android:layout height="wrap content"
44	android:checked="true"
45	android:text="Pria" />
46	</RadioGroup>
47	

48	<Spinner
49	android:id="@+id/spinner1"
50	android:layout_width="wrap_content"
51	android:layout_height="wrap_content"
52	android:layout_alignTop="@+id/textView4"
53	android:layout_toRightOf="@+id/textView4" />
54	
55	<TextView
56	android:id="@+id/textView5"
57	android:layout_width="wrap_content"
58	android:layout_height="wrap_content"
59	android:layout_above="@+id/button1"
60	android:layout_alignRight="@+id/textView1"
61	android:layout_marginBottom="35dp"
62	android:text="Hobi"
63	android:textAppearance="?android:attr/textAppearanceMedium" />
64	
65	<CheckBox
66	android:id="@+id/checkbox1"
67	android:layout_width="wrap_content"
68	android:layout_height="wrap_content"
69	android:layout_alignBottom="@+id/textView5"
70	android:layout_alignLeft="@+id/spinner1"
71	android:text="Makan" />
72	
73	<CheckBox
74	android:id="@+id/checkbox2"
75	android:layout_width="wrap_content"
76	android:layout_height="wrap_content"
77	android:layout_above="@+id/button1"
78	android:layout_alignLeft="@+id/checkbox1"
79	android:text="Tidur" />
80	
81	
82	<Button
83	android:id="@+id/button1"
84	android:layout_width="wrap_content"
85	android:layout_height="wrap_content"
86	android:layout_alignParentBottom="true"
87	android:layout_alignParentLeft="true"
88	android:layout_alignParentRight="true"
89	android:text="Do Registration" />
90	
91	<RadioButton
92	android:id="@+id/radiol"
93	android:layout_width="wrap_content"
94	android:layout_height="wrap_content"
95	android:layout_alignTop="@+id/radioGroup1"

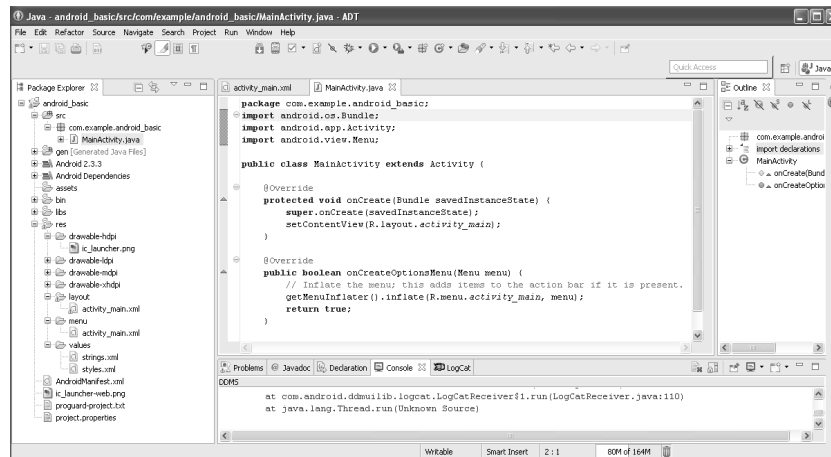
96	android:layout_toRightOf="@+id/radioGroup1"
97	android:text="Wanita" />
98	
99	<TextView
100	android:id="@+id/textView4"
101	android:layout_width="wrap_content"
102	android:layout_height="wrap_content"
103	android:layout_above="@+id/checkbox1"
104	android:layout_marginBottom="54dp"
105	android:layout_toLeftOf="@+id/radioGroup1"
106	android:text="Tempat Tinggal"
107	android:textAppearance="?android:attr/textAppearanceMedium" />
108	
109	<TextView
110	android:id="@+id/textView2"
111	android:layout_width="wrap_content"
112	android:layout_height="wrap_content"
113	android:layout_above="@+id/textView4"
114	android:layout_alignParentLeft="true"
115	android:layout_marginBottom="52dp"
116	android:text="Sex"
117	android:textAppearance="?android:attr/textAppearanceMedium" />
118	
119	<TextView
120	android:id="@+id/textView3"
121	android:layout_width="wrap_content"
122	android:layout_height="wrap_content"
123	android:layout_above="@+id/textView2"
124	android:layout_alignParentLeft="true"
125	android:layout_marginBottom="40dp"
126	android:text="Password"
127	android:textAppearance="?android:attr/textAppearanceMedium" />
128	
129	<TextView
130	android:id="@+id/textView1"
131	android:layout_width="wrap_content"
132	android:layout_height="wrap_content"
133	android:layout_above="@+id/textView3"
134	android:layout_alignParentLeft="true"
135	android:layout_marginBottom="28dp"
136	android:text="Username"
137	android:textAppearance="?android:attr/textAppearanceMedium" />
138	
139	</RelativeLayout>

Pada source code tersebut, terlihat jelas bahwa di form registrasi ada 1 ImageView (yang dinamai *imageView1*). 5 TextView (yang dinamai *textView1*, *textView2*, *textView3*, *textView4*, dan *textView5*). 2 EditText (yang dinamai *editText1* dan *editText2*). 1 RadioGroup (yang dinamai *radioGroup1*). 2 CheckBox (yang dinamai *checkBox1* dan *checkBox2*). 1 Spinner (yang dinamai *spinner1*). Dan 1 Button (yang dinamai *button1*).

**STEP 6.** Pada Step 5, telah disiapkan file Graphic Layout. Maka, sesuai inti teknik Android Programming yang telah dibuat pada awal bab, langkah selanjutnya adalah siapkan file class untuknya (berupa source code *\*.java* yang ada di dalam folder *src/com.example.android\_basic*).

Sebagai permulaan, file class tersebut disiapkan dalam rangka menampilkan beberapa item pada ComboBox (Spinner) yang ada di form registrasi yang telah dibuat di Step 5.

Cara merealisasinya adalah, pada **Package Explorer** (Gambar 1.8), klik 2x file *MainActivity.java* yang ada di dalam folder *src/com.example.android\_basic* sampai di sebelah kanannya muncul source code-nya seperti gambar berikut.



**Gambar 1.12 Source Code File Class (MainActivity.java)**

Untuk menghemat waktu pembahasan, ganti semua source code yang ada padanya dengan berikut.

1	<code>package com.example.android_basic;</code>
2	<code>//persiapkan semua library yang dibutuhkan</code>
3	<code>import android.os.Bundle;</code>
4	<code>import android.app.Activity;</code>
5	<code>import android.view.Menu;</code>
6	<code>import android.view.MenuItem;</code>
7	<code>import android.widget.AdapterView;</code>
8	<code>import android.widget.Spinner;</code>
10	
11	<code>public class MainActivity extends Activity</code>
12	<code>{</code>
13	<code>//deklarasi variable</code>
14	<code>Spinner CBTempatTinggal;</code>
15	<code>String[] CBitem = { "Sumatra", "Jawa", "Kalimantan", "Nusa Tenggara", "Sulawesi", "Papua"};</code>
16	<code>@Override</code>
17	<code>protected void onCreate(Bundle savedInstanceState)</code>
18	<code>{</code>
19	<code>super.onCreate(savedInstanceState);</code>
20	<code>//set tampilan form dengan res/layout/activity_main.xml</code>
21	<code>setContentView(R.layout.activity_main);</code>
22	<code>//objek combobox yang ada pada form</code>
23	<code>CBTempatTinggal=(Spinner)findViewById(R.id.spinner1);</code>
24	
25	<code>//set combobox item yang ada pada form</code>
26	<code>ArrayAdapter aa = new ArrayAdapter(this,android.R.layout.simple_spinner_item,CBitem);</code>
27	<code>aa.setDropDownViewResource(android.R.layout.simple_spinner_dropdown_item);</code>
28	<code>CBTempatTinggal.setAdapter(aa);</code>
29	<code>}</code>
30	<code>@Override</code>
31	<code>public boolean onCreateOptionsMenu(Menu menu)</code>
32	<code>{</code>
33	<code>// set menu form dengan res/menu/activity main.xml</code>
34	<code>getMenuInflater().inflate(R.menu.activity_main, menu);</code>
35	<code>return true;</code>
36	<code>}</code>
37	<code>@Override</code>
38	<code>public boolean onOptionsItemSelected(MenuItem item)</code>
39	<code>{</code>
40	<code>//jika klik menu form</code>
41	<code>switch (item.getItemId())</code>
42	<code>{</code>
43	<code>default:</code>
44	<code>return super.onOptionsItemSelected(item);</code>
45	<code>}</code>
46	<code>}</code>
47	<code>}</code>

Terlihat bahwa file class *MainActivity.java* tersebut digunakan untuk menampilkan form registrasi (yang telah dibuat di Step 5) via potongan source code berikut.

```
setContentView(R.layout.activity_main);
```

Pada potongan source code tersebut, *R.layout.activity\_main* sama dengan direktori *res/layout/activity\_main.xml*. Yang mana file *activity\_main.xml* adalah nama file layout form registrasi (yang telah dibuat di Step 5).

Setelah ditentukan form mana yang akan ditampilkan, maka baru-lah file class bisa meng-handle semua event yang mungkin bisa terjadi pada form tersebut.

Sebagai contohnya, terlihat bahwa file class *MainActivity.java* digunakan untuk menampilkan item ke objek *ComboBox* yang ada di form registrasi via potongan source code berikut.

```
CBTempatTinggal= (Spinner) findViewById(R.id.spinner1);  
CBTempatTinggal.setAdapter(aa);
```

Untuk memudahkan membaca source code file class tersebut, biasanya kita buat sendiri perkiraan kerangka sederhananya (misalnya seperti berikut).

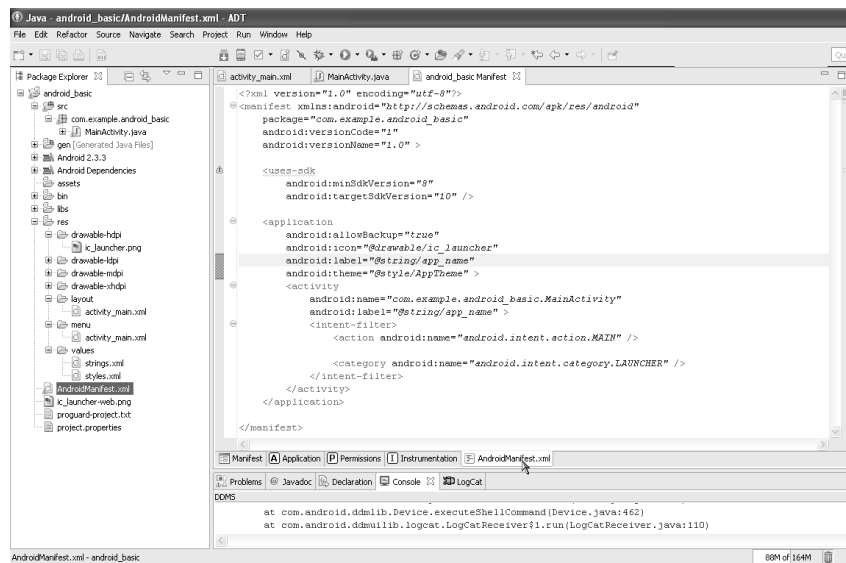
```
01 package com.example.android_basic;  
02 //siapkan semua library yang dibutuhkan di sini  
03 public class MainActivity extends Activity  
04 {  
05     //tambahkan source code di sini  
06     @Override  
07     protected void onCreate(Bundle savedInstanceState)  
08     {  
09         //tambahkan source code di sini  
10     }  
11     @Override  
12     public boolean onCreateOptionsMenu(Menu menu)  
13     {  
14         //tambahkan source code di sini  
15     }  
16     @Override  
17     public boolean onOptionsItemSelected(MenuItem item)  
18     {  
19         //tambahkan source code di sini  
20     }  
21 }
```

Untuk bisa membuat seperti kerangka tersebut, maka biasanya saya menyisakan **public class**, **protected void**, **public Boolean**, **private class**, **public void**, **private void** dari source code-nya.



**STEP 7.** Pada Step 5 telah disiapkan file Graphic Layout. Pada Step 6 telah disiapkan file class-nya. Maka, sesuai inti teknik Android Programming yang telah dibuat pada awal bab, langkah berikutnya adalah update file *AndroidManifest*-nya.

Cara merealisasikannya adalah, pada **Package Explorer** (Gambar 1.9), klik 2x file *AndroidManifest.xml* sampai di sebelah kanannya muncul source codenya seperti gambar berikut.



**Gambar 1.13** *AndroidManifest.xml*

Untuk menghemat waktu pembahasan, ganti semua source code-nya dengan berikut.

1	<code>&lt;?xml version="1.0" encoding="utf-8"?&gt;</code>
2	<code>&lt;manifest</code>
3	<code>  xmlns:android="http://schemas.android.com/apk/res/android"</code>
4	<code>  package="com.example.android.basic"</code>
5	<code>  android:versionCode="1"</code>
6	<code>  android:versionName="1.0" &gt;</code>
7	<code>  &lt;uses-sdk</code>
8	<code>    android:minSdkVersion="8"</code>
9	<code>    android:targetSdkVersion="10" /&gt;</code>
10	
11	<code>  &lt;application</code>
12	<code>    android:allowBackup="true"</code>

13	<code>android:icon="@drawable/ic_launcher"</code>
14	<code>android:label="@string/app_name"</code>
15	<code>android:theme="@style/AppTheme" &gt;</code>
16	<code>&lt;activity</code>
17	<code>android:name="com.example.android_basic.MainActivity"</code>
18	<code>android:label="@string/app_name" &gt;</code>
19	<code>&lt;intent-filter&gt;</code>
20	<code>&lt;action</code> <code>android:name="android.intent.action.MAIN" /&gt;</code>
21	<code>&lt;category</code> <code>android:name="android.intent.category.LAUNCHER" /&gt;</code>
22	<code>&lt;/intent-filter&gt;</code>
23	<code>&lt;/activity&gt;</code>
24	<code>&lt;/application&gt;</code>
25	
26	<code>&lt;/manifest&gt;</code>

Terlihat bahwa pada file *AndroidManifest.xml* tersebut ada:

1. `<uses-sdk android:minSdkVersion="8"`  
`android:targetSdkVersion="10" />`

Supaya apk-nya bisa kompatibel dengan device HP Android OS 2.2 atau versi di atasnya.

2. `<activity`  
`android:name="com.example.android_basic.MainActivity"`  
`<action android:name="android.intent.action.MAIN" />`

Supaya form yang ditampilkan oleh file class *MainActivity.java* adalah form yang muncul saat pertama kali apk-nya dijalankan.

**STEP 8.** Pada Step 5 telah disiapkan file Graphic Layout. Pada Step 6 telah disiapkan file class-nya. Pada Step 7 telah di-update file *AndroidManifest*-nya. Maka, sesuai dengan inti teknik Android Programming yang telah dibuat, langkah selanjutnya adalah Build Proyek dan Upload apk-nya ke device.

Adapun cara Build Proyeknya adalah seperti berikut.

Klik menu **Proyek/Build Proyek** (atau tekan Ctrl+B pada keyboard). Maka akan dilakukan proses Build Proyek. Tunggu sampai proses Build Proyek selesai.

Kemudian, klik menu **Run/Run** (atau tekan Ctrl+F11 pada keyboard) sampai muncul di emulatorenya seperti gambar berikut.



**Gambar 1.14 Hasil Building & Running Proyek**

Sebelum muncul di emulatomnya seperti Gambar 1.14, terkadang muncul warning (huruf berwarna merah) pada tab **Console** seperti gambar berikut.



**Gambar 1.15 Warning sebelum emulator muncul**

Jika demikian, maka kita harus ulangi Step 8 sampai benar-benar muncul di emulatomnya seperti Gambar 1.14.

Setelah muncul di emulatomnya seperti Gambar 1.14, sekarang upload apk-nya ke device. Caranya, copy apk-nya dari folder *bin* (tepatnya pada direktori *C:\adt-bundle-windows-x86\Proyek\_android\android\_basic\android\_basic\bin*) ke device HP Android Anda, via kabel data. Kemudian install-lah apk-nya. Sehingga hasil akhirnya (pada device) adalah seperti gambar berikut.



**Gambar 1.16 Menjalankan aplikasi pada device HP Android**



**CATATAN**

Beberapa poin penting pada Subbab 1.1 adalah:

- `src/<package_name>/ *.java` -> adalah file class. Digunakan untuk menampilkan form dan mendeteksi semua event yang mungkin terjadi pada form(layout) tersebut.
- `res/ layout/ *.xml` -> adalah file layout (form).
- `res/ menu/ *.xml` -> adalah file menu form.
- `AndroidManifest.xml` -> di-update supaya apk-nya aplikasi bisa kompatibel dengan device HP Android targetnya.
- Build Proyek, Run, dan Upload \*.apk ke device.

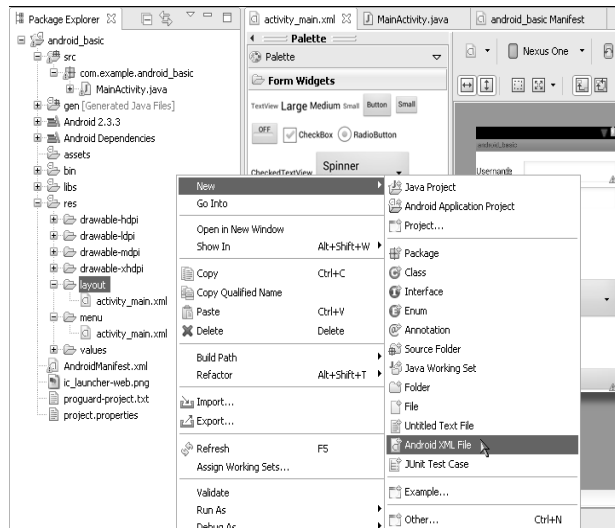
## 1.2 Membuat Form (2)

Pada Subbab 1.1, telah dibuat form Registrasi. Maka, sesuai rencana pembahasan yang telah dibuat, Subbab 1.2 membahas proses pembuatan form login.

Sehingga memberikan gambaran kepada kita bagaimana membuat lebih dari 1 form dan menghubungkan semuanya dalam sebuah aplikasi.

Mari terapkan inti teknik yang telah dibuat pada awal bab dalam rangka membuat form login.

**STEP 1.** Buat XML baru dengan nama Form2.xml. Caranya, pada **Package Explorer** (Gambar 1.8), klik kanan pada folder **res/ Layout** sampai muncul menunya seperti gambar berikut.



*Gambar 1.17 klik-kanan res/layout - New/Android XML File*

Pada menu yang muncul tersebut, klik **New/ Android XML File**. Maka, akan muncul **New Android XML File** seperti gambar berikut.

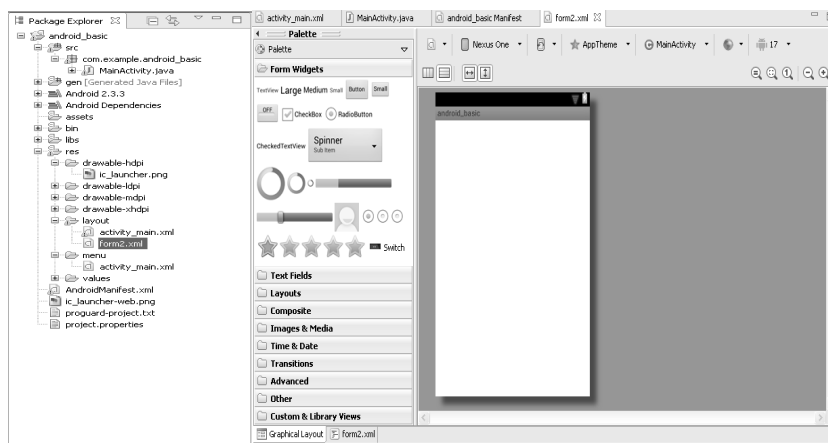


*Gambar 1.18 Setting New Android XML File*

Biarkan saja setting default seperti Gambar 1.18 (bila gambar kurang jelas, setting pada gambar tersebut adalah seperti berikut).

**Resource Type** : Layout  
**Proyek** : android\_basic  
**File** : form2  
**Root Element** : Linear Layout

Setelah diatur seperti demikian, klik tombol **Finish** yang ada padanya, maka pada **Package Explorer** muncul *Form2.xml* di dalam folder *res/layout*, dan di sebelah kanannya muncul tampilan **Graphic Layout** darinya yang masih kosong (seperti gambar berikut). Tampilan Graphic Layout yang masih kosong tersebutlah yang akan digunakan untuk mendesain form login.

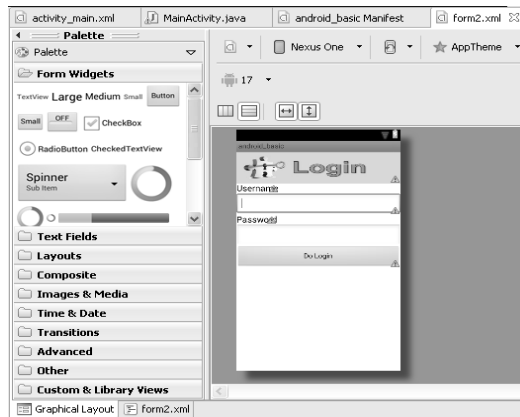


**Gambar 1.19** Akibat menambahkan *Form2.xml*

Dan juga pastikan bahwa source code dari tampilan **Graphic Layout** yang masih kosong tersebut adalah seperti berikut.

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

**STEP 2.** Pada Step 1 telah dibuatkan XML-nya. Maka, langkah selanjutnya adalah membuatkan desain formnya ke XML tersebut. Adapun tampilan Form Login yang akan dibuatkan ke XML tersebut adalah seperti gambar berikut.



*Gambar 1.20 Desain form login yang akan dibuat*

Konsep pembuatannya mirip seperti pada pembahasan Step 5 (Subbab 1.1). Oleh karena itu, untuk menghemat waktu pembahasan, ganti semua source code dari Graphic Layout *Form2.xml* tersebut dengan berikut.

1	<code>&lt;?xml version="1.0" encoding="utf-8"?&gt;</code>
2	<code>&lt;LinearLayout</code> <code>xmlns:android="http://schemas.android.com/apk/res/android"</code>
3	<code>android:layout_width="match_parent"</code>
4	<code>android:layout_height="match_parent"</code>
5	<code>android:orientation="vertical" &gt;</code>
6	
7	<code>&lt;TextView</code>
8	<code>android:id="@+id/textView1"</code>
9	<code>android:layout_width="wrap_content"</code>
10	<code>android:layout_height="wrap_content"</code>
11	<code>android:text="Username"</code>
12	<code>android:textAppearance="?android:attr/textAppearanceMedium"</code> <code>/&gt;</code>
13	
14	<code>&lt;EditText</code>
15	<code>android:id="@+id/editText1"</code>
16	<code>android:layout_width="match_parent"</code>
17	<code>android:layout_height="wrap_content"</code>

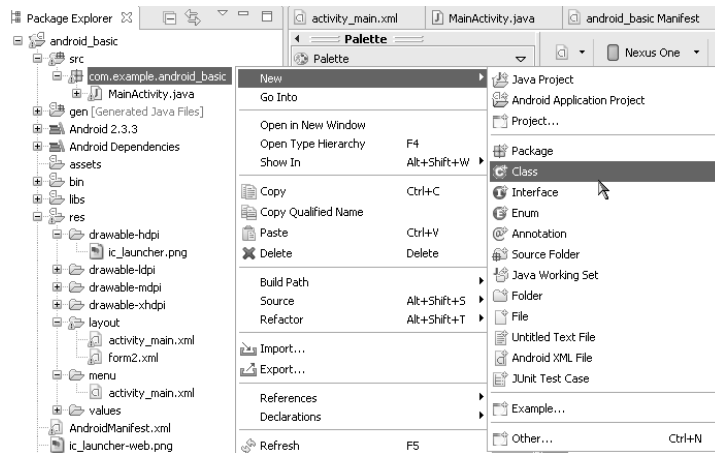
18	android:ems="10" >
19	<requestFocus />
20	</EditText>
21	
22	<TextView
23	android:id="@+id/textView2"
24	android:layout_width="wrap_content"
25	android:layout_height="wrap_content"
26	android:text="Password"
27	android:textAppearance="?android:attr/textAppearanceMedium" />
28	
29	<EditText
30	android:id="@+id/editText2"
31	android:layout_width="match_parent"
32	android:layout_height="wrap_content"
33	android:ems="10"
34	android:inputType="textPassword" />
35	
36	<Button
37	android:id="@+id/button1"
38	android:layout_width="match_parent"
39	android:layout_height="wrap_content"
40	android:text="Do Login" />
41	
42	</LinearLayout>

Pada source code tersebut, terlihat jelas bahwa pada form login ada 2 TextView (yang dinamai *textView1* dan *textView2*). 2 EditText (yang dinamai *editText1* dan *editText2*). Dan 1 Button (yang dinamai *button1*).

**STEP 3.** Pada Step 1 (Subbab 1.2) telah disiapkan file Graphic Layout. Pada Step 2 (Subbab 1.2) telah dibuatkan desainnya. Maka, sesuai inti teknik Android Programming yang telah dibuat pada awal bab, langkah selanjutnya adalah siapkan file class untuknya (berupa source code *\*.java* yang ada di dalam folder *src/com.example.android\_basic*).

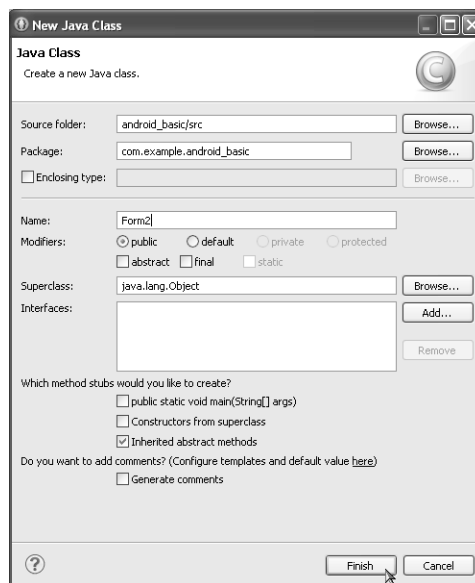
Oleh karena itu, sekarang siapkan file class dengan nama *Form2.java* untuk menampilkan form yang telah dibuat di Step 2 (Subbab 1.2). Cara merealisasinya adalah, klik kanan pada folder *src/com.example.android\_basic* sampai muncul menunya seperti gambar berikut.





**Gambar 1.21** Klik kanan src/com.example.android\_basic

Pada menunya yang muncul tersebut, klik **New / Class** sampai muncul **Java Class** seperti gambar berikut.

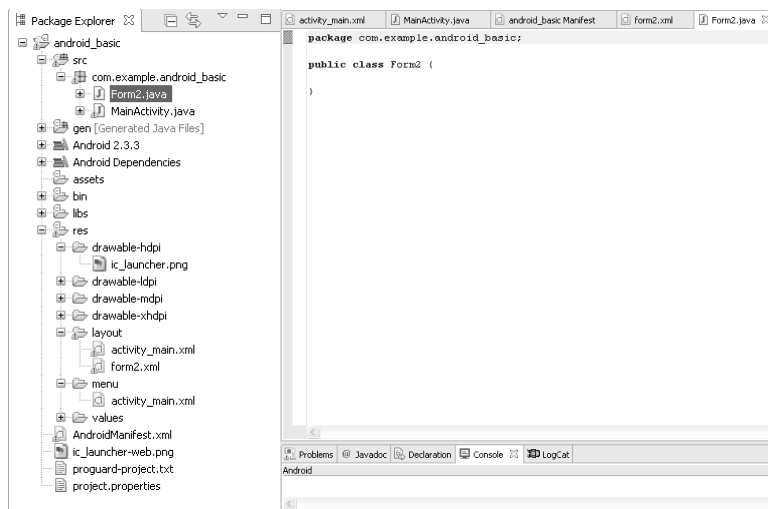


**Gambar 1.22** Setting New Java Class

Biarkan saja setting default seperti pada Gambar 1.22, bila gambar kurang jelas, setting pada gambar tersebut adalah seperti berikut:

**Source Folder:** android\_basic/src  
**Package:** com.example.android\_basic  
**Name:** Form2  
 Contreng radio **public**, checkbox **inherited abstract methods**

Setelah diatur seperti demikian, klik tombol **Finish** yang ada padanya, maka pada **Package Explorer** muncul *Form2.java* di dalam folder *src/com.example.android\_basic*, dan di sebelah kanannya muncul source code darinya yang masih kosong (seperti gambar berikut).



**Gambar 1.23** Akibat menambahkan *Form2.java*

Untuk menghemat waktu pembahasan, ganti semua source code yang ada padanya dengan berikut.

1	<code>package com.example.android_basic;</code>
2	<code>import android.os.Bundle;</code>
3	<code>import android.app.Activity;</code>
4	<code>import android.content.Intent;</code>
5	<code>import android.view.Menu;</code>
6	<code>import android.view.MenuItem;</code>
7	<code>import android.view.View;</code>
8	<code>import android.widget.Toast;</code>
9	
10	<code>public class Form2 extends Activity</code>
11	<code>{</code>
12	<code>    @Override</code>

13	protected void onCreate(Bundle savedInstanceState)
14	{
15	super.onCreate(savedInstanceState);
16	//set tampilan form dengan res/layout/form2.xml
17	setContentView(R.layout.form2);
18	}
19	
20	@Override
21	public boolean onCreateOptionsMenu(Menu menu)
22	{
23	// set menu form dengan res/menu/activity_main.xml
24	getMenuInflater().inflate(R.menu.activity_main, menu);
25	return true;
26	}
27	@Override
28	public boolean onOptionsItemSelected(MenuItem item)
29	{
30	//jika klik menu form
31	
32	}
33	}
34	
35	}

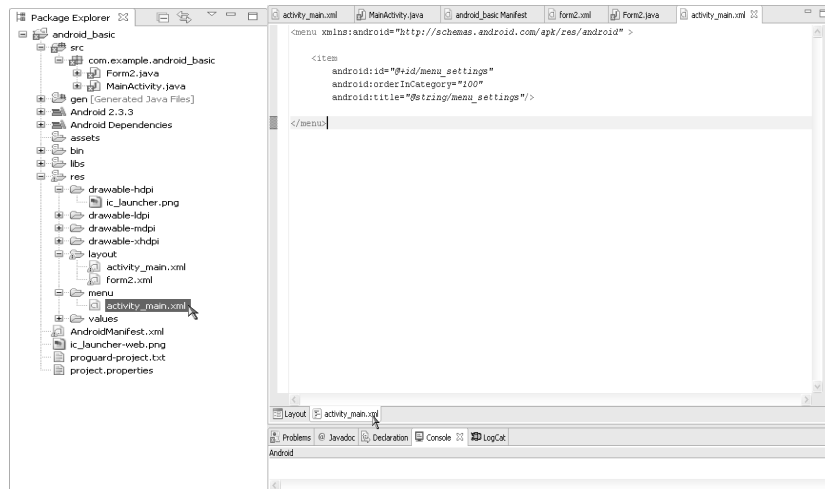
Pada source code tersebut, terlihat jelas bahwa file class Form2.java menampilkan form login yang telah dibuat di Step2 (Subbab 1.2) via source code berikut.

```
setContentView(R.layout.form2);
```

Pada potongan source code tersebut, *R.layout.form2* sama dengan direktori *res/layout/form2.xml*. Yang mana file *form2.xml* adalah nama file layout form login (yang telah dibuat di Step 2 Subbab 1.2).

**STEP 4.** Pada Step 2 telah disiapkan file Graphic Layout. Pada Step 3 telah disiapkan file classnya. Maka, sesuai inti teknik Android Programming yang telah dibuat pada awal bab, langkah selanjutnya adalah update file *menu*-nya dalam rangka menyediakan 2 buah menu. Menu satu untuk membuka (me-redirect ke) form login, menu dua untuk membuka (me-redirect ke) form registrasi.

Cara meng-update-nya adalah, pada **Package Explorer**, klik 2x pada file *activity\_main.xml* yang ada di dalam folder *res/menu* sampai di sebelah kanannya muncul source code-nya (seperti gambar berikut).



**Gambar 1.24** Source code menu (*res/menu/activity\_main.xml*)

Untuk menghemat waktu pembahasan, ganti semua source code tersebut dengan berikut.

1	<menu xmlns:android="http://schemas.android.com/apk/res/android" >
2	
3	<item
4	android:id="@+id/menu1"
5	android:orderInCategory="100"
6	android:title="Ke Form1"/>
7	<item
8	android:id="@+id/menu2"
9	android:orderInCategory="100"
10	android:title="Ke Form2"/>
11	
12	</menu>

Pada source code menu tersebut, terlihat bahwa ada 2 buah menu dengan nama (identifikasi) **menu1**, **menu2**. Yang terlihat dari *android:id*-nya (baris 4 dan baris 8).

Setelah meng-update file *menu*-nya, maka jangan lupa untuk juga meng-update source code file class-nya dalam rangka menugaskan masing-masing menu yang telah dideklarasikan pada awal Step 4, Subbab 1.2. Dalam contoh bab ini, file class yang dimaksud ada 2, yaitu:

- **File class yang menampilkan form registrasi (*MainActivity.java*).**  
Pada file class ini, **menu2** digunakan untuk membuka (me-redirect ke) form login.
- **File class yang menampilkan form login (*Form2.java*).**  
Pada file class ini, **menu1** digunakan untuk membuka (me-redirect ke) form registrasi.

Adapun bagian source code file class *MainActivity.java* yang di-update adalah baris 42 sampai dengan baris 47 (dari Step 6 Subbab 1.1) dengan source code berikut.

1	<code>//jika klik menu form</code>
2	<code>switch (item.getItemId())</code>
3	<code>{</code>
4	<code>case R.id.menu1:</code>
5	<code>Toast.makeText(getApplicationContext(), "Go To Form1", Toast.LENGTH_SHORT).show();</code>
6	<code>return true;</code>
7	<code>case R.id.menu2:</code>
8	<code>Toast.makeText(getApplicationContext(), "Go To Form2", Toast.LENGTH_SHORT).show();</code>
9	<code>Intent i = new Intent(MainActivity.this, Form2.class);</code>
10	<code>startActivity(i);</code>
11	<code>return true;</code>
12	<code>default:</code>
13	<code>return super.onOptionsItemSelected(item);</code>
14	<code>}</code>

Pada bagian source code file class *MainActivity.java* yang di-update tersebut, dapat disimpulkan bahwa:

1. **menu1** yang ada pada form registrasi digunakan untuk memunculkan konfirmasi 'Go To Form1' via source code berikut.  
`Toast.makeText(getApplicationContext(), "Go To Form1", Toast.LENGTH_SHORT).show();`
2. **menu2** yang ada pada form registrasi digunakan untuk membuka (me-redirect ke) form login via source code berikut.

```
Intent i = new Intent(MainActivity.this, Form2.class);
startActivity(i);
```

Adapun bagian source code file class *Form2.java* yang di-update adalah baris 30 sampai dengan baris 31 (dari Step 3 Subbab 1.2) dengan source code berikut.

1	<code>switch (item.getItemId())</code>
2	<code>{</code>
3	<code>case R.id.menu1:</code>
4	<code>Toast.makeText(getApplicationContext(), "Go To Form1", Toast.LENGTH_SHORT).show();</code>
5	<code>Intent i = new Intent(Form2.this, MainActivity.class);</code>
6	<code>startActivity(i);</code>
7	<code>return true;</code>
8	<code>case R.id.menu2:</code>
9	<code>Toast.makeText(getApplicationContext(), "Go To Form2", Toast.LENGTH_SHORT).show();</code>
10	<code>return true;</code>
11	<code>default:</code>
12	<code>return super.onOptionsItemSelected(item);</code>
13	<code>}</code>

Pada bagian source code file class *Form2.java* yang di-update tersebut, dapat disimpulkan bahwa:

1. **menu1** yang ada pada form login digunakan untuk membuka (me-redirect ke) form registrasi via source code berikut.

```
Intent i = new Intent(Form2.this, MainActivity.class);
startActivity(i);
```

2. **menu2** yang ada pada form login digunakan untuk memunculkan konfirmasi 'Go To Form2' via source code berikut.

```
Toast.makeText(getApplicationContext(), "Go To Form2",
Toast.LENGTH_SHORT).show();
```

**STEP 5.** Pada Step 2 telah disiapkan file Graphic Layout. Pada Step 3 telah disiapkan file class-nya. Pada Step 4 telah di-update file menu-nya. Maka, sesuai inti teknik Android Programming yang telah dibuat pada awal bab, langkah selanjutnya adalah update file *AndroidManifest*-nya (*AndroidManifest.xml*) dalam rangka supaya apk-nya bisa digunakan untuk membuka form-form yang telah dibuat tanpa ada error.

Sehingga pada akhirnya source code *AndroidManifest*-nya akan menjadi seperti berikut.

1	<?xml version="1.0" encoding="utf-8"?>
2	<manifest
3	xmlns:android="http://schemas.android.com/apk/res/android"
4	package="com.example.android.basic"
5	android:versionCode="1"
6	android:versionName="1.0" >
7	<uses-sdk
8	android:minSdkVersion="8"
9	android:targetSdkVersion="10" />
10	<application
11	android:allowBackup="true"
12	android:icon="@drawable/ic_launcher"
13	android:label="@string/app_name"
14	android:theme="@style/AppTheme" >
15	<activity
16	android:name="com.example.android.basic.MainActivity"
17	android:label="@string/app_name" >
18	<intent-filter>
19	<action android:name="android.intent.action.MAIN" />
20	<category android:name="android.intent.category.LAUNCHER"
21	/>
22	</intent-filter>
23	</activity>
24	<activity
25	android:name=".Form2"
26	android:label="Form2" >
27	<intent-filter>
28	<action android:name="android.intent.action.VIEW" />
29	<category android:name="android.intent.category.DEFAULT"
30	/>
31	</intent-filter>
32	</activity>
33	</application>
34	</manifest>

Pada source code *AndroidManifest* yang terakhir tersebut, terlihat di antara tag `<application>...</application>` ada 2 tag `<activity>`.

Tag `<activity>` yang pertama menampilkan file class *MainActivity.java* via source code berikut.

```
<activity android:name="com.example.android.basic.MainActivity"
```

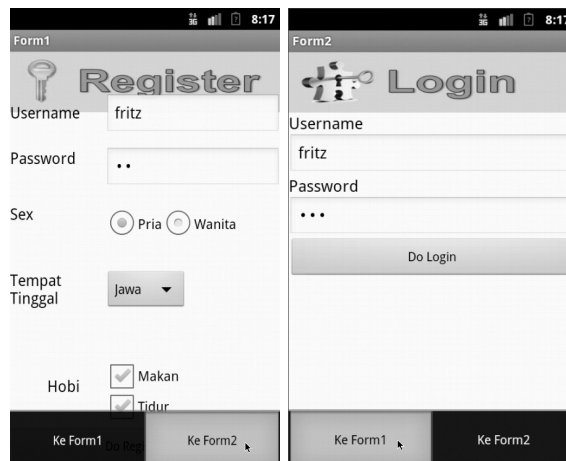
Tag `<activity>` yang kedua menampilkan file class *Form2.java* via source code berikut.

```
<activity android:name=".Form2"
```


Seperti yang telah dibuat sebelumnya, bahwa file class *MainActivity.java* digunakan untuk menampilkan form registrasi. Sedangkan, file class *Form2.java* digunakan untuk menampilkan form login.

**STEP 7.** Pada Step 2 telah disiapkan file Graphic Layout. Pada Step 3 telah disiapkan file class-nya. Pada Step 4 telah di-update file menunya. Pada Step 5 telah di-update *AndroidManifest*-nya. Maka, sesuai inti teknik Android Programming yang telah dibuat pada awal bab, langkah selanjutnya adalah Build Proyek dan Upload apk-nya ke device.

Adapun langkah-langkahnya sama seperti pada Step 8 (Subbab 1.1). Pastikan hasil yang muncul di emulatoanya adalah seperti gambar berikut.



**Gambar 1.25 Hasil (di emulator) form 1 dan form 2**

 <b>CATATAN</b>	<p>Konsep dasar membuat banyak form (lebih dari 1) pada Android Programming adalah:</p> <ol style="list-style-type: none"><li>1. Tambah file layoutnya (source code xml)</li><li>2. Tambah file classnya (source class java)</li><li>3. Update file menunya</li><li>4. Update file <i>AndroidManifest</i>nya</li><li>5. Terakhir, Build Proyeknya dan Upload apk-nya ke device.</li></ol>
-------------------------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------



## 1.3 Teknik Dasar Database Programming pada Android

Pada Subbab 1.1 telah dibuat form registrasi. Pada Subbab 1.2 telah dibuat form login. Maka, sesuai dengan rencana pembahasan yang ada di awal bab, pada Subbab ini tinggal menghubungkan kedua form tersebut dengan database.

Sebagai informasi, ada 2 tipe koneksi ke database yang dapat digunakan, yaitu:

- **Offline**, Contoh database yang men-support tipe koneksi ini adalah SQLite.
- **Online**, Contoh database yang men-support tipe koneksi ini adalah MySQL.

Untuk memprogram dengan tipe online, maka Android Programming memakai library *HttpClient* dan *HttpPost* untuk menghubungkan device dengan pihak ke-3 yang untuk selanjutnya pihak ke-3 tersebutlah yang memanipulasi database-nya.

Manipulasi tersebut bisa berupa meng-input data ke table, membaca data dari table, menghapus data yang ada di table, dan masih banyak contoh manipulasi table lainnya yang bisa dilakukan. Adapun pihak ke-3 yang dimaksud adalah bahasa pemrograman web (contohnya: *PHP*).

Untuk pembahasan Subbab 1.3 ini, kita akan menggunakan yang tipe offline (dengan database SQLite). Untuk tipe online, Anda bisa mengeksplornya sendiri. Pembahasan serupa juga dapat dijumpai pada Smartphone Programming lainnya (Blackberry, Windows Phone, iPhone, J2ME).

Adapun yang akan dikerjakan pada Subbab 1.3 ini, adalah seperti berikut:

**STEP1.** Update file class *MainActivity.java* dalam rangka apabila klik Button '*Do Registration*' yang ada pada form registrasi maka data-data yang terisi pada setiap field (yang ada pada form registrasi) akan ter-input ke field yang sesuai pada table tuser.

Adapun struktur table tuser dibuat seperti berikut.

<i>Kolom</i>	<i>Tipe</i>
username	TEXT
password	TEXT
sex	TEXT
tempat_tinggal	TEXT
hobi	TEXT

Adapun bagian source code file class *MainActivity.java* yang di-update adalah antara baris 29 dan baris 30 (dari Step 6, Subbab 1.1) dengan berikut.

1	//objek apa saja yang ada di form
2	txtUsername = (EditText) findViewById(R.id.editText1);
3	txtPassword = (EditText) findViewById(R.id.editText2);
4	RdJenisKelamin=(RadioGroup) findViewById(R.id.radioGroup1);
5	chk1=(CheckBox) findViewById(R.id.checkBox1);
6	chk2=(CheckBox) findViewById(R.id.checkBox2);
7	btnRegister = (Button) findViewById(R.id.button1);
8	
9	//jika klik tombol send, maka entry data ke tabel tuser
10	btnRegister.setOnClickListener(new OnClickListener()
11	{
12	@Override
13	public void onClick(View v)
14	{
15	//get textbox data
16	String
17	vusername=txtUsername.getText().toString();
18	String
19	vpassword=txtPassword.getText().toString();
20	//get radiobutton data
21	String vjeniskelamin="";
22	int
23	radio_index=RdJenisKelamin.getCheckedRadioButtonId();
24	if(radio_index==R.id.radio0){vjeniskelamin="Pria";}
25	if(radio_index==R.id.radio1){vjeniskelamin="Wanita";}
26	//get combobox data
27	String
28	vtempat tinggal=CBTempatTinggal.getSelectedItem().toString();
29	//get checkbox data
30	String vhobi="";
31	if(chk1.isChecked()){vhobi="Makan";}
32	if(chk1.isChecked() &&

	chk2.isChecked() {vhobi+=",";}
29	if(chk2.isChecked()) {vhobi+="Tidur";}
30	
31	//create or open database
32	db=openOrCreateDatabase("MYDB.db", SQLiteDatabase.CREATE_IF NECESSARY,null);
33	//create table
34	try{
35	db.execSQL("CREATE TABLE IF NOT EXISTS tuser(username TEXT,password TEXT,sex TEXT,tempat tinggal TEXT,hobi TEXT)");
36	}
37	catch(Exception e){
38	Toast.makeText(getApplicationContext(), "create table tuser failed", Toast.LENGTH_SHORT).show();
39	}
40	//insert data into table
41	try{
42	db.execSQL("INSERT INTO tuser(username,password,sex,tempat tinggal,hobi)VALUES('"+ vusername+"','"+vpassword+"','"+vjeniskelamin+"','"+vtempa t tinggal+"','"+vhobi+"')");
43	}
44	catch(Exception e){
45	Toast.makeText(getApplicationContext(), "insert data into tuser failed", Toast.LENGTH_SHORT).show();
46	}
47	//close database
48	db.close();
49	}
50	});

Pada source code tersebut, terlihat bahwa:

1. Objek editText1 yang ada pada form registrasi diwakilkan dengan nama variable txtUsername, dan apa yang diisikan kepadanya ditampung ke dalam variable vusername via source code berikut.

```
txtUsername = (EditText) findViewById(R.id.editText1);
vusername=txtUsername.getText().toString();
```

2. Event klik objek button1 (button 'Do Registration') yang ada pada form registrasi dideteksi via source code berikut.

```
btnRegister = (Button) findViewById(R.id.button1);
btnRegister.setOnClickListener(new OnClickListener() {
```

3. Apabila klik button 'Do Registration', maka data-data yang diisikan pada form registrasi akan ter-input ke field yang sesuai pada table tuser via source code berikut.

```
String vusername=txtUsername.getText().toString();
String vpassword=txtPassword.getText().toString();
...
...
db.execSQL("INSERT INTO
tuser(username,password,sex,tempat_tinggal,hobi)VALUES('"+
+vusername+"','"+vpassword+"','"+vjeniskelamin+"','"+vtem
pat_tinggal+"','"+vhobi+"')");
```

**STEP 2.** Update file class *Form2.java* dalam rangka apabila klik Button 'Do Login', akan dicek apakah pada table tuser ada data username dan password yang sama dengan data username dan password yang terisi pada form login. Jika ada, maka munculkan konfirmasi 'User Account Available'. Sebaliknya, jika tidak ada, maka munculkan konfirmasi 'User Account Not Available'.

Adapun bagian source code file class *Form2.java* yang di-update adalah antara baris 17 dan baris 18 (dari Step 3 Subbab 1.2) dengan source code berikut.

1	//objek apa saja yang ada di form
2	txtUsername = (EditText) findViewById(R.id.editText1);
3	txtPassword = (EditText) findViewById(R.id.editText2);
4	btnLogin = (Button) findViewById(R.id.button1);
5	
6	//jika klik tombol send, maka entry data ke tabel tuser
7	btnLogin.setOnClickListener(new OnClickListener()
8	{
9	@Override
10	public void onClick(View v)
11	{
12	//get textbox data
13	String
	vusername=txtUsername.getText().toString();
14	String
	vpassword=txtPassword.getText().toString();
15	
16	int i=0;
17	int ada=0;
18	//open or create database
19	db=openOrCreateDatabase("MYDB.db", SQLiteDatabase.CREATE_IF
	_NECESSARY,null);
20	//select * from tuser
21	Cursor c = db.query("tuser", null, null,
	null, null, null, null);

22	<code>c.moveToFirst();</code>
23	<code>//where username=... &amp; password=...</code>
24	<code>while(c.isAfterLast()==false)</code>
25	<code>{</code>
26	<code>if(c.getString(0).equals(vusername) &amp;&amp;</code> <code>c.getString(1).equals(vpassword))</code>
27	<code>{ada=1;}</code>
28	<code>c.moveToNext();</code>
29	<code>}</code>
30	<code>//close database</code>
31	<code>db.close();</code>
32	
33	<code>//show confirmation</code>
34	<code>if(ada==0){Toast.makeText(getApplicationContext(), "User</code> <code>Account Not Available", Toast.LENGTH_SHORT).show();}</code>
35	<code>else{Toast.makeText(getApplicationContext(), "User Account</code> <code>Available", Toast.LENGTH_SHORT).show();}</code>
36	<code>}</code>
37	<code>});</code>

Pada source code tersebut, terlihat bahwa isi table tuser di-enumerasikan semuanya via source code berikut.

```
while(c.isAfterLast()==false)
```

Kemudian dicek apakah pada setiap barisnya di kolom pertama (`c.getString(0)`) ada data yang sama dengan yang diketikkan oleh user pada kotak Username (di form login) dan di kolom kedua (`c.getString(1)`) ada data yang sama dengan yang diketikkan oleh user pada kotak Password (di form login) via source code berikut.

```
if(c.getString(0).equals(vusername) &&
c.getString(1).equals(vpassword))
```

Jika ada, maka akan ditampilkan konfirmasi 'User Account Available' via potongan source code berikut.

```
Toast.makeText(getApplicationContext(), "User Account
Available", Toast.LENGTH_SHORT).show();
```

Jika tidak ada, maka akan ditampilkan konfirmasi 'User Account Not Available' via potongan source code berikut.

```
Toast.makeText(getApplicationContext(), "User Account Not
Available", Toast.LENGTH_SHORT).show();
```

**STEP 3.** Build Proyeknya dan upload apk-nya ke device HP Android Anda untuk di-testing.

Bisa karena terbiasa. Dengan mengetahui inti tekniknya dan terbiasa melihat source code (Bab 1 dan Proyek 1 s.d. Proyek 5), mencoba memodifikasinya sesuai kreasi sendiri atau mengikuti tips development yang diberikan, maka akan membuat Anda semakin hebat dalam pemrograman Android. Untuk mencapainya, mungkin butuh ribuan mil. Tapi, ingatlah ribuan mil itu adalah hasil dari 1 langkah di setiap harinya. Karena itu, mulailah melangkah! Hal serupa juga berlaku pada Bahasa Pemrograman Apapun yang Anda ketahui.



#### CATATAN

Konsep teknik dasar database Programming pada Android sebenarnya tidak jauh berbeda dengan konsep teknik dasar database programming pada bahasa pemrograman lainnya.

Yakni terkait penggunaan query (seperti: ALTER, SELECT, INSERT, UPDATE, DELETE, DROP, TRUNCATE, dan QUERY yang lainnya) dalam rangka memanipulasi database sesuai keperluan.