

# **MODUL PRAKTIKUM PEMROGRAMAN MOBILE I**

## **3**

### **Android User Interface (UI) – XML Layout**



**STMIK SINAR NUSANTARA  
SURAKARTA  
2016**

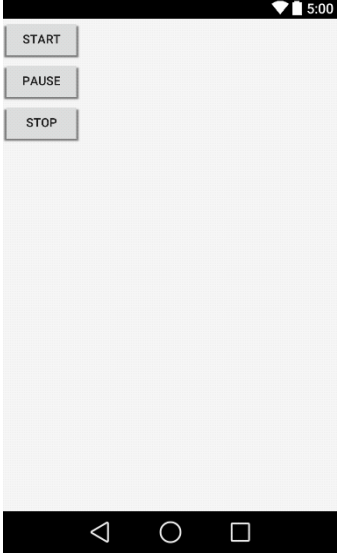
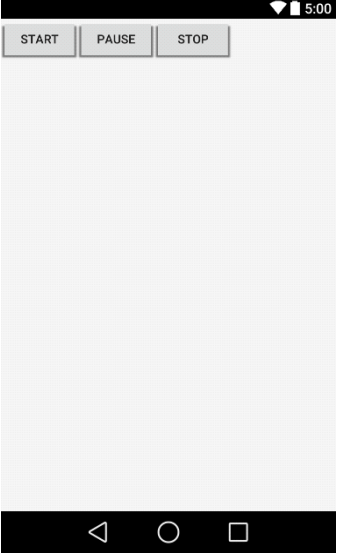
## Screen Layout

Untuk mengatur peletakan komponen-komponen pada layar (User Interface), android menggunakan layout manager yang berbasis XML. File XML ini dapat dimodifikasi langsung atau melalui Graphical Layout (Visual Editor).

Terdapat empat jenis layout manager yang memiliki karakteristik masing-masing, yaitu diantaranya LinearLayout, RelativeLayout, TableLayout, FrameLayout dan ScrollView. Berikut akan dibahas lebih rinci untuk setiap layout.

## Linear Layout

Linear layout menyimpan widget dalam satu baris atau satu kolom (tergantung orientasinya). Orientasi pada linear layout ada 2 macam yaitu vertical dan horizontal. Berikut adalah gambar linear layout.

| Orientation Vertical   | Orientation Horizontal  |
|--|---|
| <code>android:orientation="vertical"</code>  | <code>android:orientation="horizontal"</code>                                       |
|  |  |

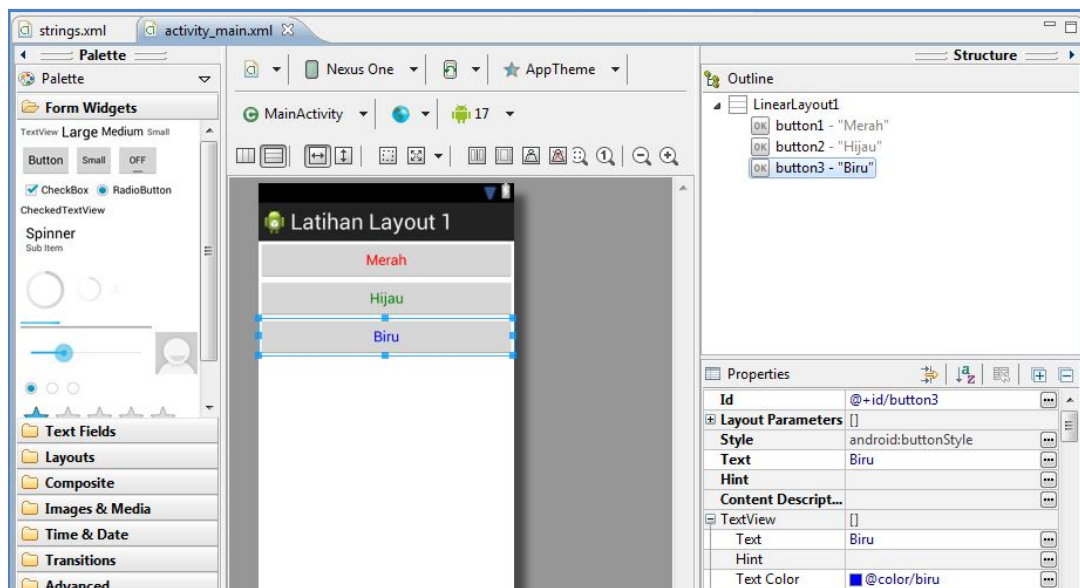
## Praktikum 1

1. Buat Project dengan nama **Latihan Layout 1**
2. Pada folder res/values edit file strings.xml tambahkan nilai berikut (lalu save):

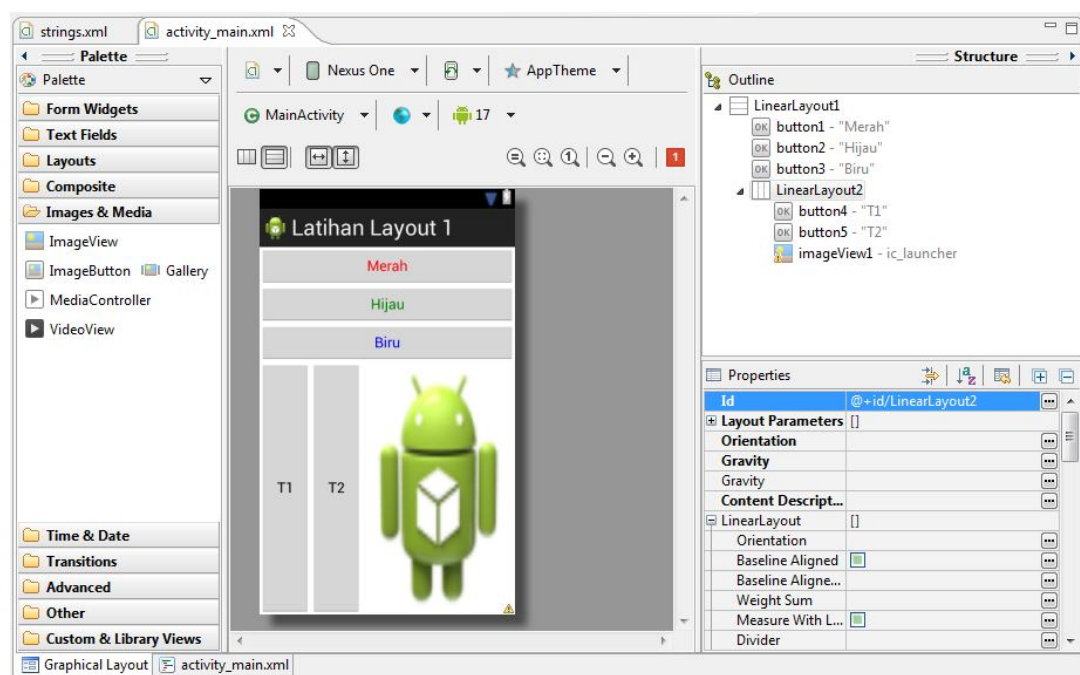
```
strings.xml
1 <?xml version="1.0" encoding="utf-8"?>
2 <resources>
3
4     <string name="app_name">Latihan Layout 1</string>
5     <string name="hello_world">Hello world!</string>
6     <string name="menu_settings">Settings</string>
7     <color name="merah">#FFFF0000</color>
8     <color name="hijau">#FF008000</color>
9     <color name="biru">#FF0000FF</color>
10
11 </resources>
```

3. Pada folder res/layout Double Klik file activity\_main.xml lalu pilih Graphical Layout, hapus TextView pada layar yang bertuliskan Hello World.
4. Pada Windows - Outline Klik Kanan RelativeLayout lalu pilih ChangeLayout
5. Pilih LinearLayout (Vertical) pada Pilihan New Layout Type
6. Letakkan 3 buah Button pada Layar, lalu buat tombol tersebut memenuhi lebar layar dengan memilih Toggle Fill Width.
7. Ubah tulisan pada masing-masing button dengan Text : Merah, Hijau, Biru

8. Ubah warna tulisan pada masing-masing button dengan Text Color : @color/merah, @color/hijau dan @color/biru , tampilan layout akan menjadi seperti berikut :



9. Pilih Kategori Layout pada Palette Layout, lalu pilih LinearLayout Horizontal, letakkan dibawah Button Biru, lalu pilih pada toolbar Toggle Fill Height.
10. Letakkan dua buah Button pada LinearLayout(Horizontal) lalu set Toggle Fill Height untuk kedua Button tersebut. Beri nama kedua Button tersebut berturut-turut T1 dan T2
11. Letakkan ImageView (Kategori Image & Media) disamping tombol T2. Lalu set Fill Height dan Width. Kemudian pada properties Scale Type pilih fitXY.
12. Tampilan akhir layout adalah sebagai berikut. Lalu jalankan Program tersebut di emulator.



## RelativeLayout

RelativeLayout mengatur posisi secara relatif berdasarkan posisi widget yang lain atau parentnya. Untuk *layout\_width* dan *layout\_height* ada dua pilihan yaitu *match\_parent* (view berusaha sebesar parent) dan *wrap\_content* (view hanya berusaha sebesar content yang berada di dalamnya). Sedangkan padding menyatakan jarak antara pinggir layar dengan komponen.

Perhatikan penggunaan atribut *layout\_toRightOf* yang menyatakan bahwa komponen kedua berada disebelah kanan komponen pertama. Sedangkan atribut *layout\_baseLine* digunakan agar kedua komponen aligned.

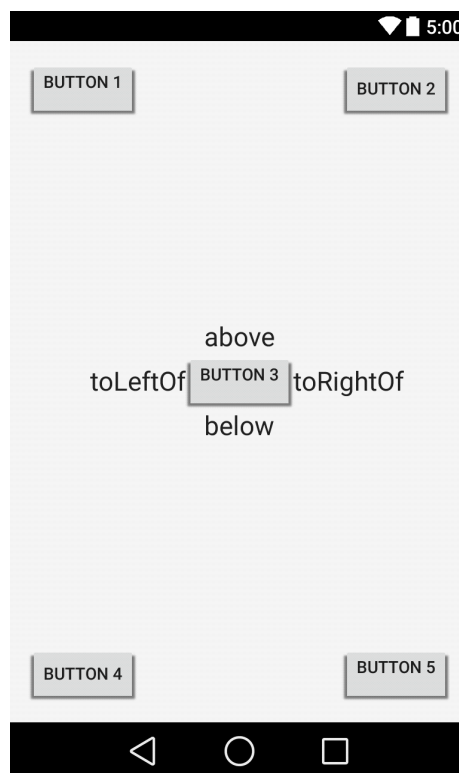
Beberapa atribut lain yang mengatur posisi widget dibandingkan dengan widget lain adalah:

- `android:layout_above`
- `android:layout_below`
- `android:layout_toLeftOf`
- `android:layout_toRightOf`

Untuk mengatur posisi ada tambahan lima atribut:

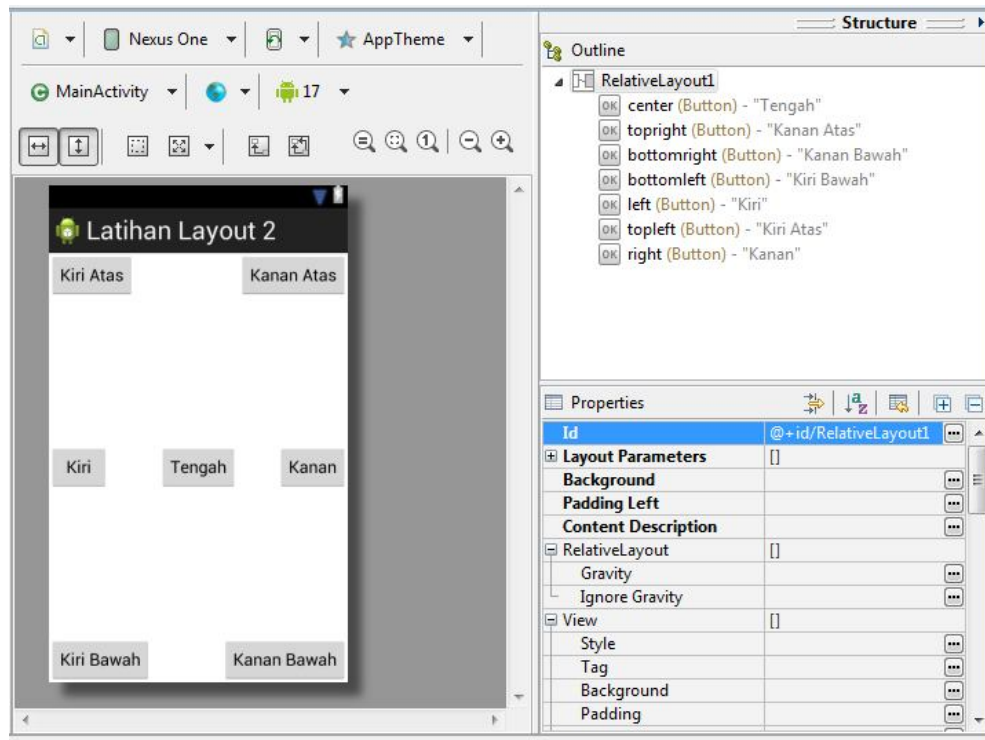
- `android:layout_alignTop`
- `android:layout_alignBottom`
- `android:layout_alignLeft`
- `android:layout_alignRight`
- `android:layout_alignBaseline`

Widget juga dapat diatur berdasarkan gravity android : *layout\_gravity*. Ada beberapa pilihan untuk gravity ini: *left*, *right*, *center\_horizontal*, *center\_vertical*, *bottom*, *top*. Gravity ini juga berlaku untuk layout manager yang lain.



## Praktikum 2

1. Buat Project dengan nama Latihan Layout 2
2. Buat Tampilan Layout seperti berikut :



3. Cek kode XML nya isinya adalah sebagai berikut :

```
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/RelativeLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    tools:context=".MainActivity" >

    <Button
        android:id="@+id/center"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerHorizontal="true"
        android:layout_centerVertical="true"
        android:text="Tengah" />

    <Button
        android:id="@+id/topright"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_alignParentTop="true"
        android:text="Kanan Atas" />

    <Button
        android:id="@+id/bottomright"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentRight="true"
        android:text="Kanan Bawah" />

    <Button
        android:id="@+id/bottomleft"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentLeft="true"
        android:text="Kiri Bawah" />

    <Button
        android:id="@+id/left"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_centerVertical="true"
        android:text="Kiri" />

    <Button
        android:id="@+id/topleft"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentLeft="true"
        android:layout_alignParentTop="true"
        android:text="Kiri Atas" />

    <Button
        android:id="@+id/right"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentRight="true"
        android:layout_centerVertical="true"
        android:text="Kanan" />

</RelativeLayout>
```

4. Simpan dan jalankan programnya.

## TableLayout

TableLayout meletakkan komponen-komponen dalam bentuk tabel (baris dan kolom), konsepnya mirip seperti Table yang digunakan pada HTML. Pengguna mendefinisikan baris dengan tag <TableRow> dan sistem secara otomatis mendefinisikan kolomnya.

Contoh screen yang menggunakan Table Layout adalah sebagai berikut. Bayangkan sebagai sebuah tabel dengan tiga kolom dan tiga baris.

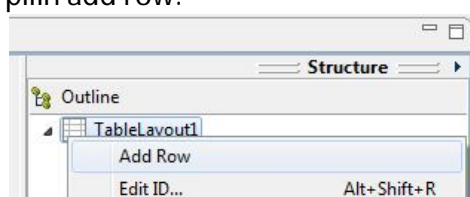


Banyaknya baris ditentukan oleh developer dengan memasukan widget (button, textview dan sebagainya) sebagai child element dari elemen <TableRow>. Dengan demikian, developer mengontrol secara langsung banyaknya baris yang diinginkan. Banyaknya kolom, akan ditentukan oleh Androidnya sendiri, dan para developer mengontrol banyaknya kolom secara tidak langsung. Paling sedikit, satu kolom untuk setiap widget.

Sebagai contoh, jika kita punya 3 baris (TableRow), baris 1 memiliki tiga widget; baris 2 memiliki dua widget; dan baris 3 memiliki empat widget, maka, paling sedikit akan ada empat kolom disana. Tetapi sebuah widget juga bisa memakai lebih dari satu kolom dengan menggunakan atribut layout\_span.

## Praktikum 3

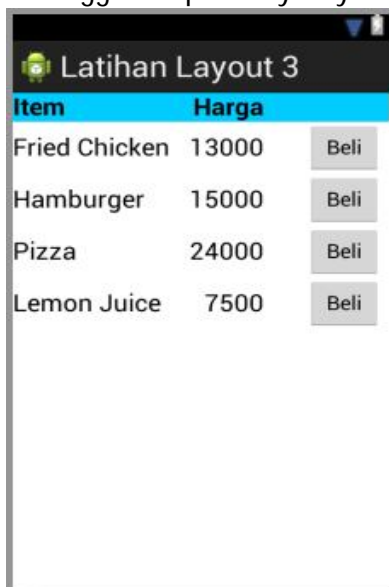
1. Buat Project dengan nama Latihan Layout 3
2. Pada Windows - Outline Klik Kanan RelativeLayout lalu pilih ChangeLayout kemudian pilih TableLayout.
3. Buat lima baris (TableRow) dengan cara klik kanan TableLayout pada window outline lalu pilih add row.



- Buat komponen-komponen berikut pada setiap baris (row) pada tabel.



- Sehingga tampilan layarnya sebagai berikut :



- Cek XML Layoutnya sebagai berikut:

```
<TableLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/TableLayout1"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:textAlignment="center"
    tools:context=".MainActivity" >

    <TableRow
        android:id="@+id/tableRow1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:background="#00CCFF" >

        <TextView
            android:id="@+id/textView1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Item"
            android:textAppearance="?android:attr/textAppearanceLarge"
            android:textStyle="bold" />
```

```

<TextView
    android:id="@+id/textView2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingLeft="20dp"
    android:text="Harga"
    android:textAppearance="?android:attr/textAppearanceLarge"
    android:textStyle="bold" />

<TextView
    android:id="@+id/textView3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingLeft="100dp"
    android:textAppearance="?android:attr/textAppearanceLarge" />
</TableRow>

<TableRow
    android:id="@+id/tableRow2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="bottom"
    android:gravity="center_vertical" >

    <TextView
        android:id="@+id/textView4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Fried Chicken"
        android:textAppearance="?android:attr/textAppearanceLarge" />

    <TextView
        android:id="@+id/textView5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:gravity="right"
        android:text="13000"
        android:textAppearance="?android:attr/textAppearanceLarge" />

    <Button
        android:id="@+id/button1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="right"
        android:text="Bel i" />
</TableRow>
....
....
Untuk baris berikutnya formatnya sama.

```

7. Simpan programnya dan jalankan pada Emulator.

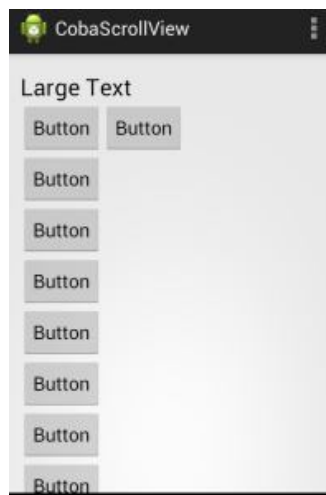
## ScrollView

ScrollView memungkinkan developer untuk membuat tampilan layout yang lebih panjang dari space layar yang ada pada device. Dengan menggunakan ScrollView, sebagian informasi akan muncul dalam satu waktu, sisanya akan muncul jika pengguna melakukan scrolling ke bawah.

Dengan scrollview, layout seolah-olah memiliki area layar yang lebih luas. Pengguna dapat melakukan scroll. Scrollview hanya mempunyai satu child dan umumnya adalah LinearLayout. Scrollview hanya dapat melakukan vertical scrolling.



Berikut adalah contoh scrollView yang menghasilkan layout seperti ini yang dapat discroll. Dua button teratas menggunakan RelativeLayout yang disisipkan dalam ScrollView. Ini adalah contoh penggabungan antara dua layout.



Kode XML Layout dari tampilan diatas adalah sebagai berikut :

```
<ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/scroll"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:fillViewport="true"
    android:paddingBottom="@dimen/activity_vertical_margin"
    android:paddingLeft="@dimen/activity_horizontal_margin"
    android:paddingRight="@dimen/activity_horizontal_margin"
    android:paddingTop="@dimen/activity_vertical_margin"
    tools:context=".MainActivity" >
```

```
<LinearLayout
```

```
    android:id="@+id/linearlayout"
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="vertical">
```

```
<TextView
```

```
    android:id="@+id/textView1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Large Text"
    android:textAppearance="?android:attr/textAppearanceLarge" />
```

```
<RelativeLayout
```

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

```
<Button
```

```
    android:id="@+id/button1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button" />
```

RelativeLayout didalam LinearLayout.  
Didalamnya terdapat dua button.

```

<Button
    android:id="@+id/button2"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_toRightOf="@+id/button1"
    android:text="Button" />
</RelativeLayout>

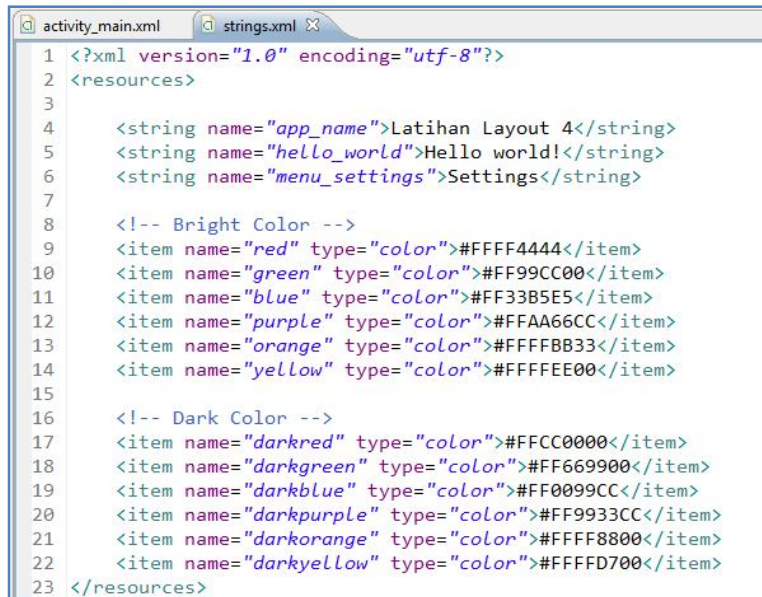
<Button
    android:id="@+id/button4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button" />
<Button
    android:id="@+id/button5"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button" />
<Button
    android:id="@+id/button6"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button" />
<Button
    android:id="@+id/button7"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button" />
<Button
    android:id="@+id/button8"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button" />
<Button
    android:id="@+id/button9"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button" />
<Button
    android:id="@+id/button10"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Button" />
</LinearLayout>
</ScrollView>

```

Dibuat beberapa button  
agar scroll terlihat.

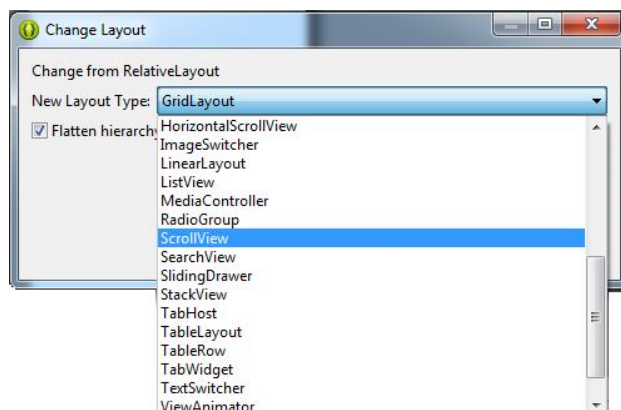
## Praktikum 4

1. Buat Project dengan nama Latihan Layout 4
2. Tambahkan setting warna (color) pada resource strings.xml sebagai berikut :

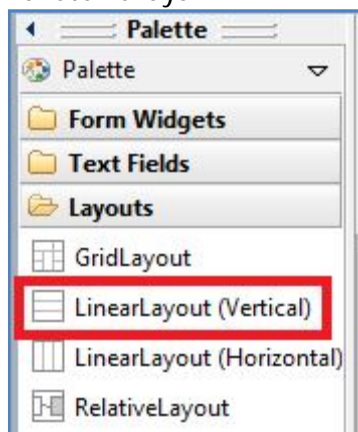


```
1 <?xml version="1.0" encoding="utf-8"?>
2 <resources>
3
4     <string name="app_name">Latihan Layout 4</string>
5     <string name="hello_world">Hello world!</string>
6     <string name="menu_settings">Settings</string>
7
8     <!-- Bright Color -->
9     <item name="red" type="color">#FFFF4444</item>
10    <item name="green" type="color">#FF99CC00</item>
11    <item name="blue" type="color">#FF33B5E5</item>
12    <item name="purple" type="color">#FFAA66CC</item>
13    <item name="orange" type="color">#FFFFBB33</item>
14    <item name="yellow" type="color">#FFFFEE00</item>
15
16    <!-- Dark Color -->
17    <item name="darkred" type="color">#FFCC0000</item>
18    <item name="darkgreen" type="color">#FF669900</item>
19    <item name="darkblue" type="color">#FF0099CC</item>
20    <item name="darkpurple" type="color">#FF9933CC</item>
21    <item name="darkorange" type="color">#FFFF8800</item>
22    <item name="darkyellow" type="color">#FFFFD700</item>
23 </resources>
```

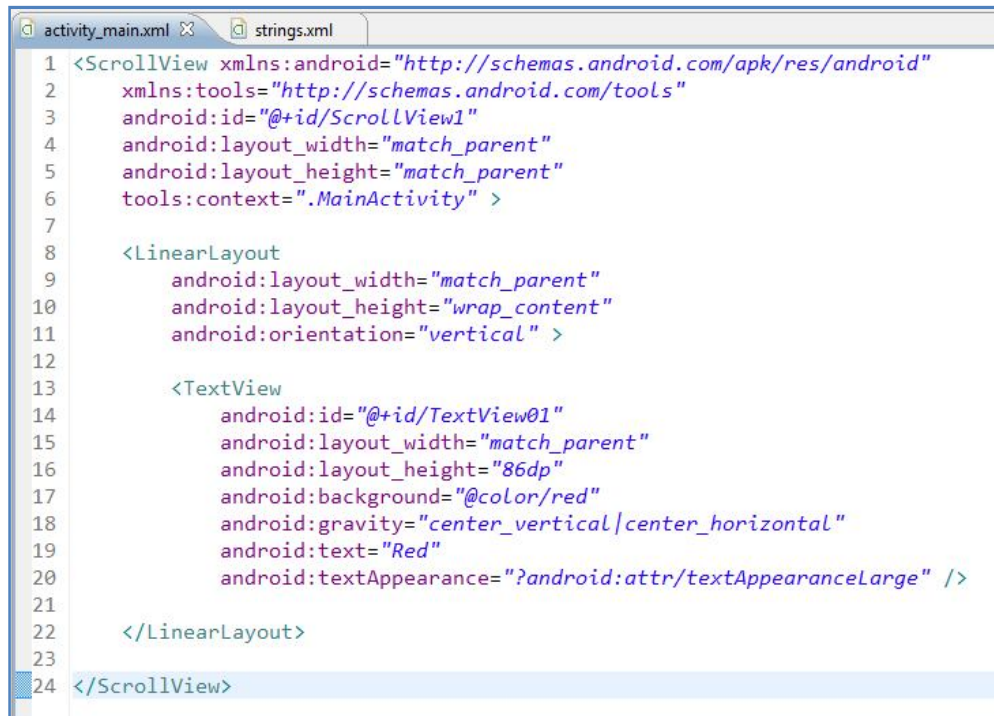
3. Pada Window Outline – klik kanan RelativeLayout, pilih Change Layout lalu pada New Layout Type: pilih ScrollView.



4. Tempatkan Linear Layout pada ScrollView - drag and drop LinearLayout(Vertical) dari Palette ke layar.

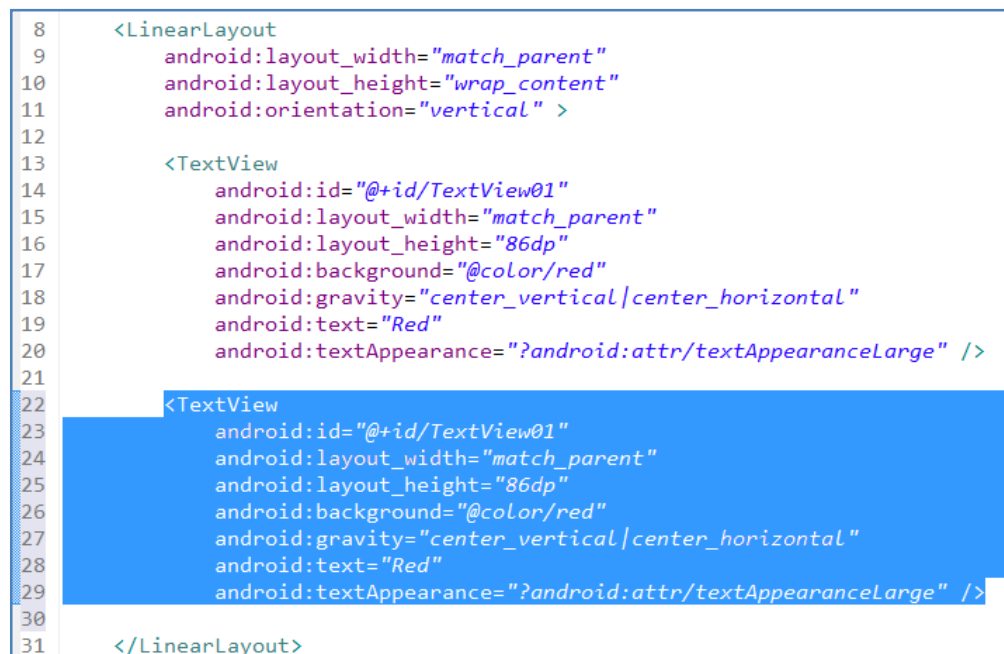


5. Buat Textview pada Linear Layout dengan properties sebagai berikut:



```
1 <ScrollView xmlns:android="http://schemas.android.com/apk/res/android"
2   xmlns:tools="http://schemas.android.com/tools"
3   android:id="@+id/ScrollView1"
4   android:layout_width="match_parent"
5   android:layout_height="match_parent"
6   tools:context=".MainActivity" >
7
8   <LinearLayout
9     android:layout_width="match_parent"
10    android:layout_height="wrap_content"
11    android:orientation="vertical" >
12
13    <TextView
14      android:id="@+id/TextView01"
15      android:layout_width="match_parent"
16      android:layout_height="86dp"
17      android:background="@color/red"
18      android:gravity="center_vertical|center_horizontal"
19      android:text="Red"
20      android:textAppearance="?android:attr/textAppearanceLarge" />
21
22  </LinearLayout>
23
24 </ScrollView>
```

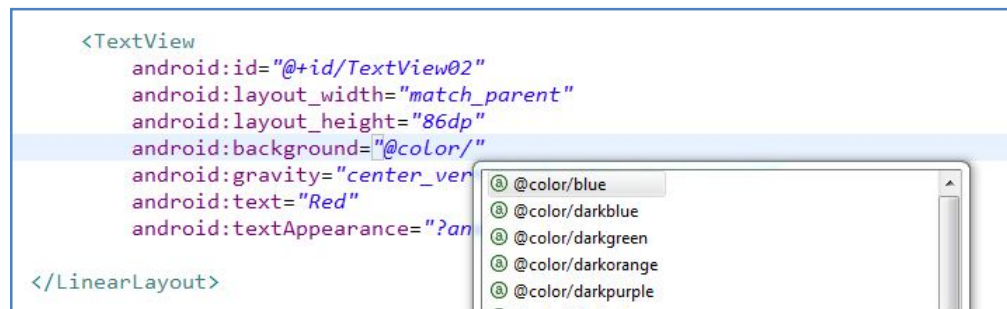
6. Copy Paste komponen TextView pada xml-file-layout (activity\_main.xml)



```
8   <LinearLayout
9     android:layout_width="match_parent"
10    android:layout_height="wrap_content"
11    android:orientation="vertical" >
12
13    <TextView
14      android:id="@+id/TextView01"
15      android:layout_width="match_parent"
16      android:layout_height="86dp"
17      android:background="@color/red"
18      android:gravity="center_vertical|center_horizontal"
19      android:text="Red"
20      android:textAppearance="?android:attr/textAppearanceLarge" />
21
22    <TextView
23      android:id="@+id/TextView01"
24      android:layout_width="match_parent"
25      android:layout_height="86dp"
26      android:background="@color/red"
27      android:gravity="center_vertical|center_horizontal"
28      android:text="Red"
29      android:textAppearance="?android:attr/textAppearanceLarge" />
30
31  </LinearLayout>
```

7. Ubah properties TextView kedua (hasil copy-paste) sebagai berikut :

```
android:id="@+id/TextView02",  
android:background="@color/green"  
android:text="Green"
```

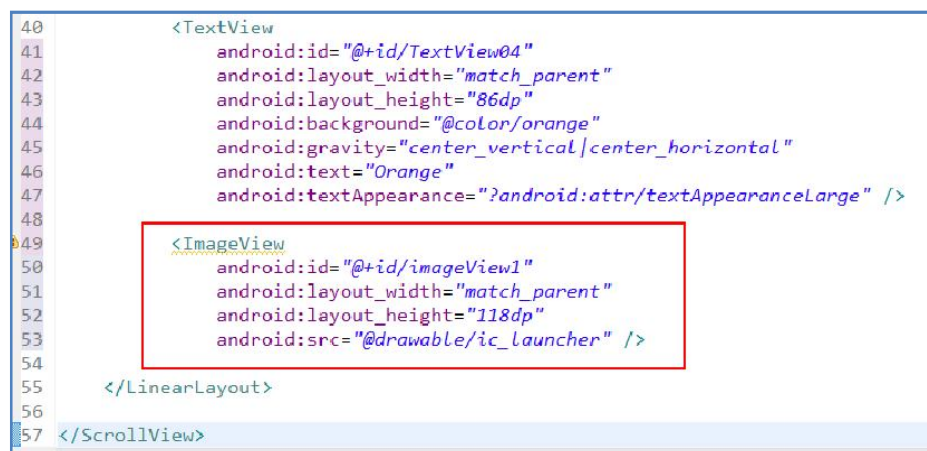


8. Lakukan hal serupa untuk dua TextView berikutnya dengan properties sebagai berikut:

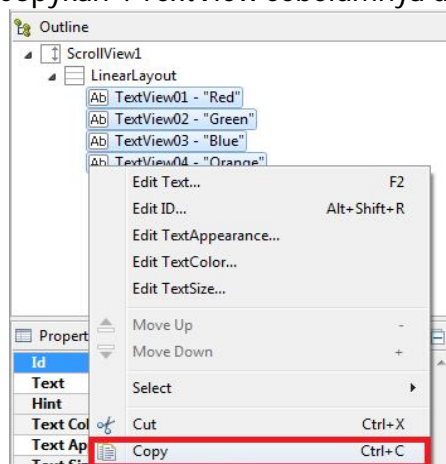
```
android:id="@+id/TextView03",  
android:background="@color/blue"  
android:text="Blue"
```

```
android:id="@+id/TextView04",  
android:background="@color/orange"  
android:text="Orange"
```

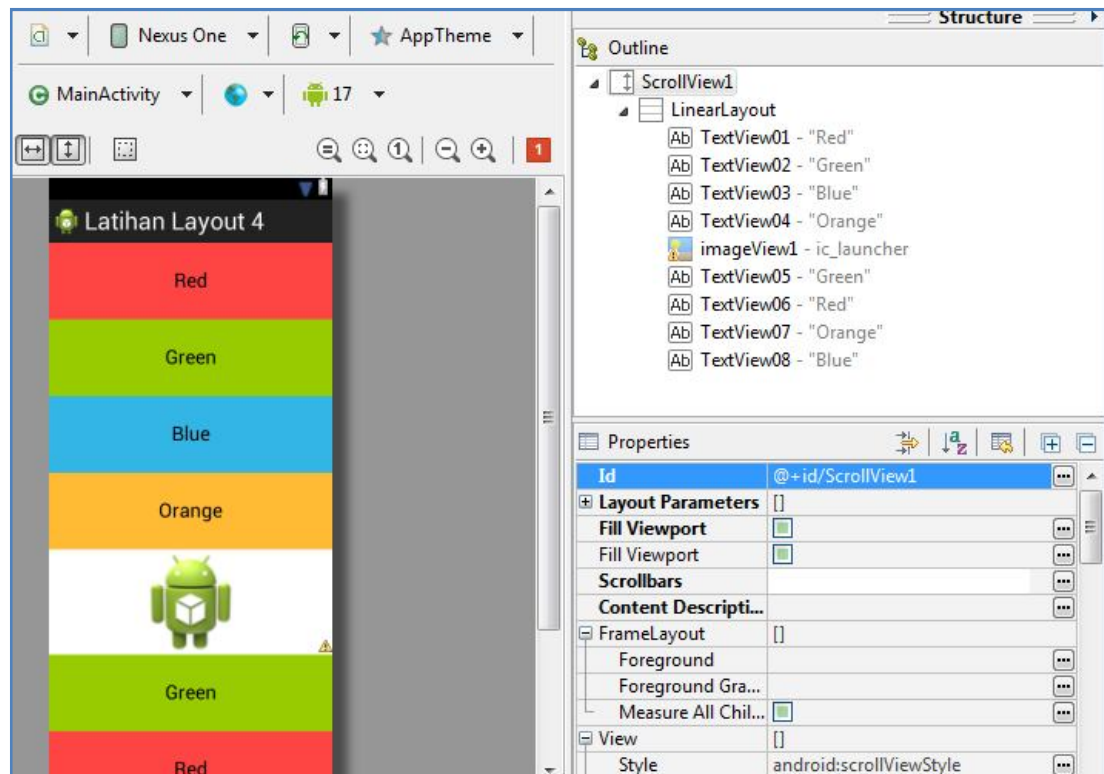
9. Tempatkan ImageView dibawah TextView04 dengan properties sebagai berikut :



10. Copykan 4 TextView sebelumnya dibawah ImageView.



11. Hasil pada Outlinenya seperti berikut ini :



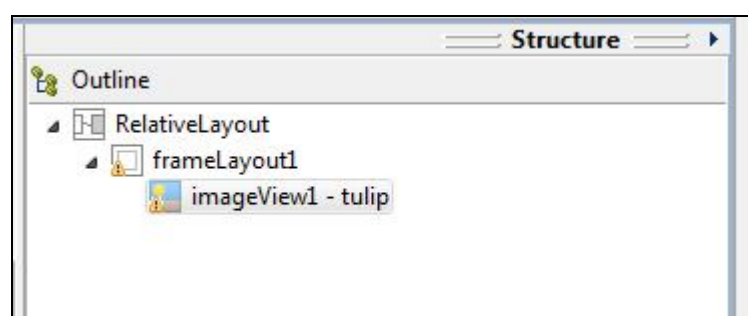
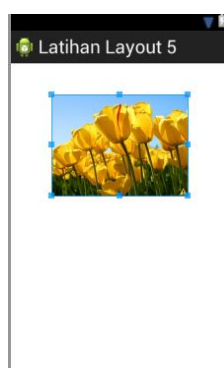
12. Jalankan program pada emulator, lalu lakukan scrolling pada layar.

## Frame Layout

Framelayout digunakan untuk menampilkan satu komponen dalam satu layar. Jika ada lebih dari satu widget, maka akan ditampilkan secara bertumpuk, walaupun posisi widget dapat diatur dengan gravity. Kita bisa menambahkan layout lain sebagai child di dalam frame layout.

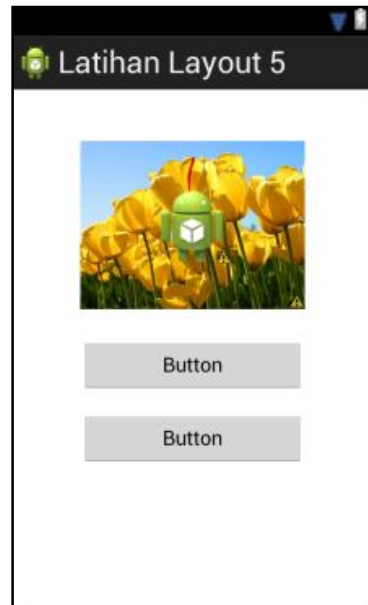
## Praktikum 5

1. Buat Project dengan nama Latihan Layout 5
2. Siapkan gambar tulip.jpg (bisa di copy dari Windows Sample Picture) . Buka dengan menggunakan program Paint lalu ubah ukuran gambar ke ukuran 300x225 pixel.
3. Tempatkan Frame layout di bawah RelativeLayout lalu tempatkan Imageview di Frame tersebut.





4. Tempatkan ImageView2 dengan gambar logo android diatas imageView1, lalu set gravity center.
5. Tempatkan dua buah Button di bawah framelayout1 buat ukurannya selebar framelayout1.
6. Hasil akhir tampilan sebagai berikut



## Menghubungkan Komponen pada Layout dengan kode java (Event Handling)

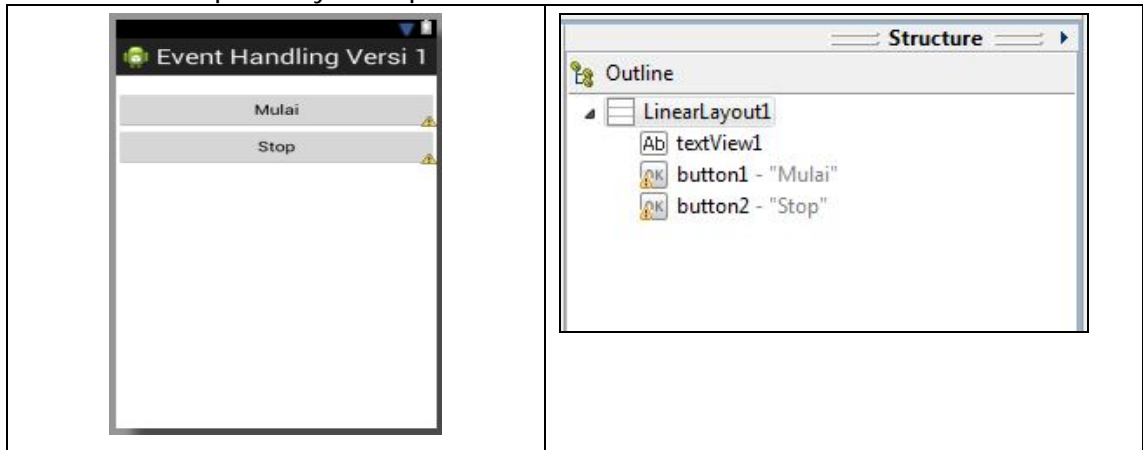
Setelah developer selesai merancang layout langkah berikutnya yang dilakukan adalah membuat program Javanya. Komponen-komponen yang sudah diatur peletakannya pada layout bisa diakses pada program java dengan terlebih dahulu dibuat objectnya. Dalam menerapkan event handling untuk komponen (misal Button) terdapat 3 cara yang bisa digunakan.

1. Menerapkan Interface Listener pada Main Activity  
cara ini lebih cocok apabila dalam aplikasi hanya ada single control pada activity, atau pada activity jumlah buttonnya hanya sedikit.
2. Menggunakan Anonymous Inner Class  
cara ini lebih fleksibel dibandingkan cara pertama, tetapi baris perintah atau statementnya yang dibuat lebih banyak.
3. Menggunakan properties On Click pada komponen untuk menjalankan method action untuk komponen tersebut.

Berikut ini contoh tiga project untuk perbandingan ketiga Event Handling diatas.

## Praktikum 6

1. Buat Project dengan nama Event Handling Versi 1
2. Tempatkan sebuah TextView dengan size Medium pada Canvas (Layar) lalu hapus properties Textnya agar menjadi kosong.
3. Tempatkan dua buah Button lalu ubah Textnya menjadi Mulai dan stop.
4. Hasil akhir tampilan Layout seperti berikut :



5. Ketikkan Program Java-nya (MainActivity.java) sebagai berikut :

```
activity_main.xml | MainActivity.java X
package com.example.eventhandlingversi1;

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

public class MainActivity extends Activity implements OnClickListener {

    TextView textView1;
    Button button1;
    Button button2;

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

        textView1 = (TextView) findViewById(R.id.textView1);
        button1 = (Button) findViewById(R.id.button1);
        button2 = (Button) findViewById(R.id.button2);

        button1.setOnClickListener(this);
        button2.setOnClickListener(this);
    }

    @Override
    public void onClick(View v) {

        if (v.getId() == button1.getId()) {
            textView1.setText("Program Dimulai ... !");
        }

        if (v.getId() == button2.getId()) {
            textView1.setText("Selesai, Terima kasih!..");
        }
    }
}
```



6. Jalankan programnya pada Emulator lalu klik tombol pada layar.
7. Berikan penjelasan mengenai Coding Java diatas menurut pemahaman Anda!

## Praktikum 7

1. Buat Project dengan nama Event Handling Versi 2
2. Buat layout yang sama dengan Layout pada Praktikum 6.
3. Ketikkan Coding Java berikut ini pada MainActivity.java

A screenshot of an IDE window showing the MainActivity.java file. The code is for an Android application named 'Event Handling Versi 2'. It imports necessary Android classes and defines the MainActivity class which extends Activity. The onCreate method initializes a TextView and two Buttons, and sets their click listeners to display specific messages.

```
package com.example.eventhandlingversi2;

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

public class MainActivity extends Activity {

    TextView textView1;
    Button button1;
    Button button2;

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

        textView1 = (TextView) findViewById(R.id.textView1);
        button1 = (Button) findViewById(R.id.button1);
        button2 = (Button) findViewById(R.id.button2);

        button1.setOnClickListener(new OnClickListener() {

            @Override
            public void onClick(View arg0) {
                textView1.setText("Program Dimulai ... !");
            }
        });

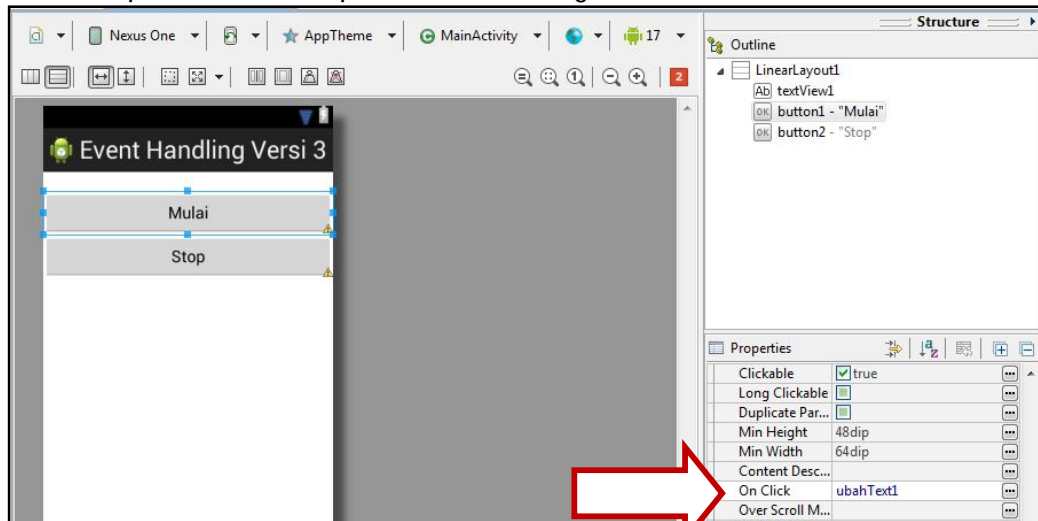
        button2.setOnClickListener(new OnClickListener() {

            @Override
            public void onClick(View arg0) {
                textView1.setText("Selesai, Terima kasih!..");
            }
        });
    }
}
```

4. Berikan penjelasan perbedaan dengan program sebelumnya!

## Praktikum 8

1. Buat Project dengan nama Event Handling Versi 3
2. Buat layout yang sama dengan Layout pada Praktikum 6.
3. Ubah Properties **On Click** pada Button1 dengan nilai **ubahText1**



4. Ubah Properties **On Click** pada Button2 dengan nilai **ubahText2**
5. Ketikkan program java-nya sebagai berikut :

```
activity_main.xml MainActivity.java X
package com.example.eventhandlingversi3;

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

public class MainActivity extends Activity {

    TextView textView1;
    Button button1;
    Button button2;

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

        textView1 = (TextView) findViewById(R.id.textView1);
    }

    public void ubahText1(View v) {
        textView1.setText("test AAAA");
    }

    public void ubahText2(View v) {
        textView1.setText("test BBBB");
    }
}
```

6. Berikan penjelasan mengenai program diatas sesuai pemahaman Anda!

## TUGAS 1

1. Buat aplikasi kalkulator sederhana dengan layout (tampilan) sebagai berikut :



2. Komponen yang digunakan pada layar diantaranya sebagai berikut :
  - a. Relative Layout
  - b. Dua Buah EditText
  - c. Dua Buah Button
  - d. Sebuah TextView
3. Buat Event Handling pada Button1 (jumlahkan) yang akan menjumlahkan dua buah bilangan yang diinputkan pada EditText1 dan EditText2. Hasil dari penjumlahan ditampilkan pada TextView1 (Result).
4. Buat Event Handling pada Button2 (kalikan) yang akan mengalikan dua buah bilangan yang diinputkan pada EditText1 dan EditText2. Hasil dari perkalian ditampilkan pada TextView1 (Result).
5. Gunakan fungsi `parseInt` pada Class Integer untuk mengubah input EditText menjadi integer, contoh :

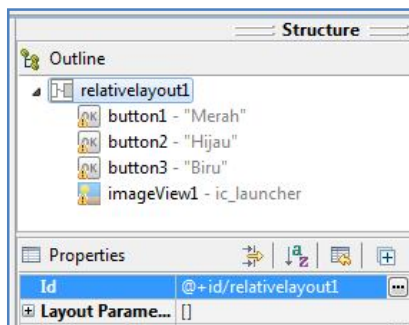
```
int angka1 = Integer.parseInt(et1.getText().toString());
```

## TUGAS 2

1. Buat aplikasi dengan tampilan sebagai berikut :



2. Komponen yang digunakan pada layar diantaranya sebagai berikut :
  - a. Relative Layout
  - b. Tiga buah Button
  - c. Sebuah ImageView
3. Set property id pada RelativeLayout dengan id : @+id/relativeLayout1



Set Property Id pada  
RelativeLayout

Lalu deklarasikan RelativeLayout seperti halnya Button

```
public class MainActivity extends Activity {  
  
    Button b1,b2,b3;  
    RelativeLayout r1;  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_main);  
  
        b1 = (Button) findViewById(R.id.button1);  
        r1 = (RelativeLayout) findViewById(R.id.relativeLayout1);  
    }  
}
```

4. Buat Event Handling pada Button Merah, Hijau dan Biru yang apabila masing – masing Button di klik akan mengubah warna background RelativeLayout menjadi Merah, Hijau dan Biru.  
Gunakan Method **setBackgroundCol or (Col or .Warna)** pada Cl ass Rel ati veLayout.