# Materi 2: Pembuatan View dan Layout

## Pokok Bahasan

1. Membuat interactive UI dengan view.
2. Membuat layout pada aplikasi dengan menggunakan RelativeLayout
3. Tugas: Membuat Textview dan ScrollView

## Tujuan

Di Akhir pertemuan ini, diharapkan mahasiswa dapat membuat tampilan berupa view disertai layout. Selain itu mahasiswa dapat membuat program sederhana dengan menerapkan TextView serta ScrollView.

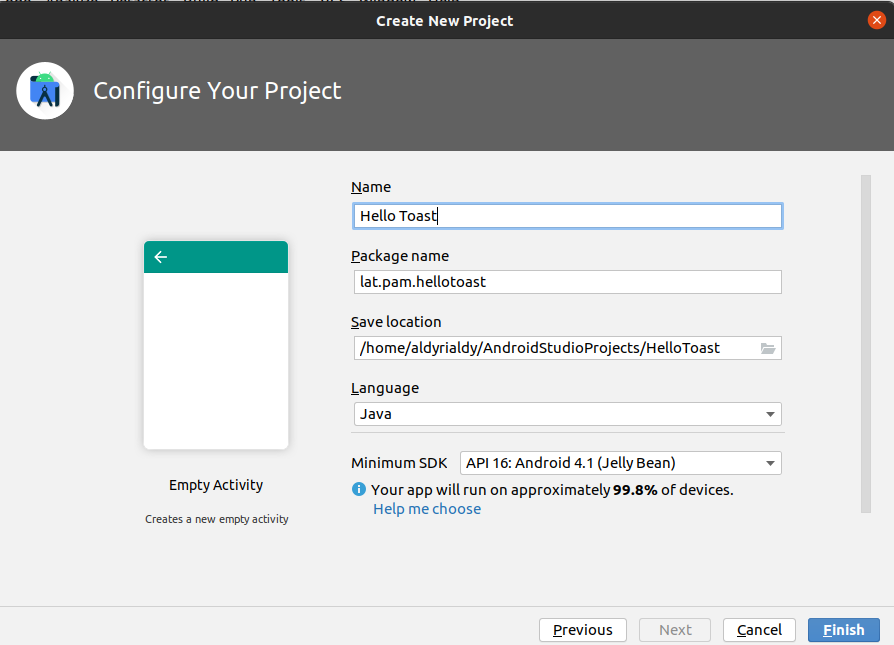
## Modul 1: Membuat Interactive UI dengan View

Seperti yang sudah dijelaskan sebelumnya bahwa komponen dari View dapat digunakan untuk membuat aplikasi Android. Komponen tersebut diantaranya sebagai berikut:

1. Komponen untuk menampilkan teks (TextView)
2. Komponen yang dapat digunakan untuk mengedit teks (EditText)
3. Komponen berupa tombol yang dapat diklik (Button) dan komponen interaktif lainnya
4. Komponen berisi teks yang dapat digulir (ScrollView) dan item yang dapat digulir (RecyclerView)
5. Komponen yang menampilkan gambar (ImageView)
6. Komponen layout yang menampung tampilan lain untuk kemudian memposisikannya pada aplikasi (LinearLayout)
7. Selain itu ada komponen lain seperti komponen untuk memunculkan menu dan komponen interaktif lainnya.

Pada modul ini anda akan membuat satu buah project dimana berisi tampilan Button, Text View dan Toast. Anda dapat mengikuti modul ini dengan mengikuti langkah-langkah berikut:

1. Buatlah satu project baru Android dengan nama **Hello Toast**
2. Pilih tampilan template project Empty Activity, kemudian sesuaikan atribut dibawah ini untuk project baru yang telah dibuat



1. Setelah project tersebut dibuat dan menyesuaikan dengan atribut yang ada diatas, anda dapat melanjutkannya dengan menekan tombol Finish
2. Selanjutnya beberapa komponen yang akan dibuat yaitu:
   1. TextView - Tampilan yang menampilkan teks.
   2. Button - Tombol dengan label sehingga terdapat event handler dari tombol yang diklik.
   3. LinearLayout - Tampilan yang berfungsi sebagai kontainer untuk mengatur tampilan lainnya. LinearLayout merupakan ViewGroup yang dapat mengatur layout dalam format horizontal atau vertikal.
3. Tahapan selanjutnya, anda dapat mengubah Layout yang ada kedalam bentuk LinearLayout. Secara default tampilan layout pertama kali project dibuat yaitu menggunakan ConstraintLayout. Layout tersebut dapat diakses pada file app> res > layout > activity\_main.xml
4. Kemudian tambahkan atribut orientation, padding vertical serta horizontal pada atribut dimen dan dapat kita set resource valuenya dengan ukuran 20dp, sehingga kode XML yang dituliskan menjadi berikut:

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

**<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"**

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

**android: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">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Hello World!"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

**</LinearLayout>**

1. Sehingga menghasilkan kode dimens.xml yang berada pada lokasi project app > res > values sebagai berikut.

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

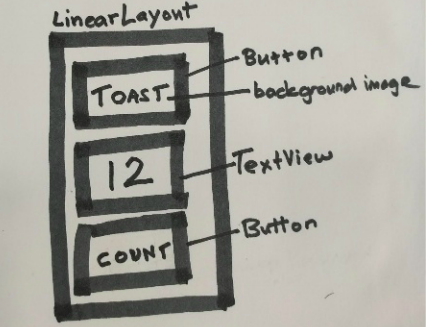
<resources>

<dimen name="activity\_vertical\_margin">20dp</dimen>

<dimen name="activity\_horizontal\_margin">20dp</dimen>

</resources>

1. Kemudian ubahlah layout didalamnya dimana didalamnya terdapat komponen TextView dan Button. Berikut ini dapat menjadi acuan untuk tampilan yang akan dibuat.



1. Anda dapat menambahkan kode XML pada bagian app > res > layout > activity\_main.xml dan pergunakan editor untuk menambahkan komponen-komponen tersebut. Komponen yang ditambahkan diantaranya Button, kemudian diikuti TextView dan dilanjutkan Button kembali

<Button

android:id="@+id/button\_toast"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Toast"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<TextView

android:id="@+id/show\_count"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="0"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<Button

android:id="@+id/button\_count"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="Count"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

1. Lakukan generate extract string resource dengan menekan Alt+Enter pada text tersebut dan buatlah label pada setiap textnya.

| **Jenis** | **Resource value** | **Resource name** |
| --- | --- | --- |
| Button | Toast | button\_label\_toast |
| TextView | 0 | count\_initial\_value |
| Button | Count | button\_label\_count |

1. Kemudian kita tambahkan string resource kembali untuk menampilkan Toast yang akan muncul.

<string name="toast\_message">Hello Toast !</string>

1. Kemudian anda dapat mengubah warna yang ada pada komponen tersebut, untuk Button (Toast dan Count) dapat anda tambahkan kode berikut:

android:backgroundTint="@color/purple\_200"

1. Sedangkan untuk TextView dapat anda tambahkan ukuran teks, warna background dan tambahkan pula atribut weight pada kode tersebut.

android:textSize="160sp"

android:background="#FFFF00"

android:layout\_weight="2"

1. Berikut ini merupakan keseluruhan kode XML yang dituliskan pada activity\_main.xml.

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

<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingBottom="@dimen/activity\_vertical\_margin"

tools:context=".MainActivity">

<Button

android:id="@+id/button\_toast"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="@string/button\_label\_toast"

android:background="@color/purple\_200"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<TextView

android:id="@+id/show\_count"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight="2"

android:text="@string/count\_initial\_value"

android:textSize="160sp"

android:background="#FFFF00"

android:gravity="center"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

<Button

android:id="@+id/button\_count"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:text="@string/button\_label\_count"

android:background="@color/purple\_200"

app:layout\_constraintBottom\_toBottomOf="parent"

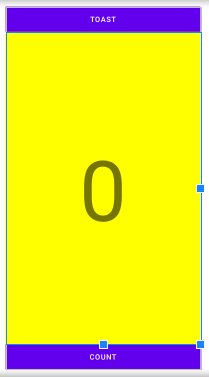
app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

</LinearLayout>

1. Berikut ini merupakan tampilan dari yang sudah dibuat.



1. Untuk menambahkan aksi pada saat tombol tersebut di klik, maka anda perlu menambahkan event handler pada Button yang hendak di klik.
2. Buka kembali file activity\_main.xml dan tambahkan properti berikut ini ke tombol button\_toast

android:onClick="showToast"

1. Kemudian tambahkan pula properti onClick pada button\_count

android:onClick="countUp"

1. Tambahkan method baru pada Activity. File ini anda dapat temukan pada app > java > [nama package] > MainActivity.java. Kemudian tambahkan beberapa baris program/kode berikut.

public void showToast(View view) {

Toast toast = Toast.*makeText*(this, R.string.*toast\_message*, Toast.*LENGTH\_LONG*);

toast.show();

}

1. Tambahkan kode program ini untuk menginisialisasi TextView terlebih dahulu dan letakkan di bawah tulisan public class MainActivity extends AppCompatActivity {

private int mCount = 0;

private TextView mShowCount;

1. Untuk memanggil TextView pada activity\_main.xml, tambahkan kode berikut ini pada bagian OnCreate

mShowCount = findViewById(R.id.*show\_count*);

1. Kemudian tambahkan kode program tambahan pada method countUp, berikut kode program tambahannya.

public void countUp(View view) {

mCount++;

if (mShowCount != null)

mShowCount.setText(Integer.toString(mCount));

}

1. Berikut ini kode program secara lengkap.

package lat.pam.hellotoast;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.TextView;

import android.widget.Toast;

public class MainActivity extends AppCompatActivity {

private int mCount = 0;

private TextView mShowCount;

@Override

protected void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.*activity\_main*);

mShowCount = findViewById(R.id.*show\_count*);

}

public void showToast(View view) {

Toast toast = Toast.*makeText*(this, R.string.*toast\_message*, Toast.*LENGTH\_LONG*);

toast.show();

}

public void countUp(View view) {

mCount++;

if (mShowCount != null)

mShowCount.setText(Integer.*toString*(mCount));

}

}

1. Jalankan program tersebut pada emulator ataupun pada device anda.

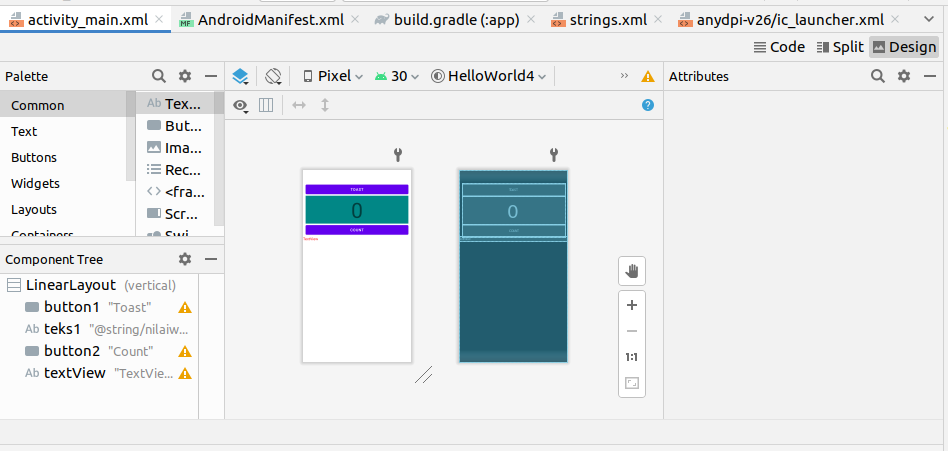
## 

## Modul 2: Membuat layout dengan menggunakan RelativeLayout pada aplikasi

Dengan menggunakan project pada modul 1, kita akan mengimplementasikan tampilan sebelumnya dengan menggunakan Relative Layout.

Relative Layout merupakan bagian dari ViewGroup dimana tampilan dapat diletakkan dan diluruskan relatif terhadap tampilan lainnya di dalam grup tampilan tersebut. Posisi tampilan dapat dikaitkan dengan satu sama lain atau dengan grup tampilan induk.

1. Anda dapat menggunakan Relative Layout ini untuk tampilan landscape dari aplikasi yang dibuat, dengan cara membuka file activity\_main.xml terlebih dahulu.
2. Kemudian arahkan untuk membuat mode landscape dengan cara seperti pada gambar dibawah ini. Pilih Create Landscape variation untuk membuat mode landscape dari project yang dibuat.



1. Ubahlah tampilan layout menjadi menggunakan Relative Layout.
2. Dan tambahkan properti berikut ini pada bagian TextView. Dan tambahkan margin untuk membuat jarak dengan komponen diatasnya.

android:layout\_below="@id/button\_toast"

android:layout\_marginTop="30dp"

1. Kemudian tambahkan properti untuk memposisikan Button count pada sebelah kanan dari Button Toast dengan kode berikut. Kemudian tambahkan pula marginnya.

android:layout\_marginLeft="50dp"

android:layout\_toRightOf="@id/button\_toast"

1. Berikut tampilan kode secara lengkap.

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

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:paddingLeft="@dimen/activity\_horizontal\_margin"

android:paddingTop="@dimen/activity\_vertical\_margin"

android:paddingRight="@dimen/activity\_horizontal\_margin"

android:paddingBottom="@dimen/activity\_vertical\_margin"

tools:context=".MainActivity">

<Button

android:id="@+id/button\_toast"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="@string/button\_label\_toast"

android:background="@color/purple\_200"

android:onClick="showToast"/>

<TextView

android:id="@+id/show\_count"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_weight="2"

android:text="@string/count\_initial\_value"

android:textSize="160sp"

android:background="#FFFF00"

android:gravity="center"

android:layout\_below="@id/button\_toast"

android:layout\_marginTop="30dp"/>

<Button

android:id="@+id/button\_count"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="50dp"

android:layout\_toRightOf="@id/button\_toast"

android:background="@color/purple\_200"

android:onClick="countUp"

android:text="@string/button\_label\_count" />

</RelativeLayout>

1. Jalankan program tersebut dan ubah kedalam bentuk landscape, lihat apakah ada perubahan pada tampilannya.

## Tugas : Membuat tampilan berita

Buatlah tampilan berikut, dapat anda gunakan RelativeLayout ataupun LinearLayout dan anda dapat menggunakan ScrollView.

