

MODUL PEMBELAJARAN

APLIKASI KOMPUTASI BERGERAK



DOSEN PENGAMPU :
Sopian Alviana, S.Kom., M.Kom

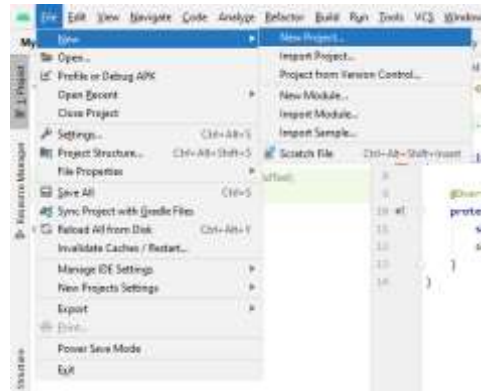
PROGRAM STUDI TEKNIK INFORMATIKA
FAKULTAS TEKNIK DAN ILMU KOMPUTER
UNIVERSITAS KOMPUTER INDONESIA
2021

1. Layout Antar Muka Bagian I

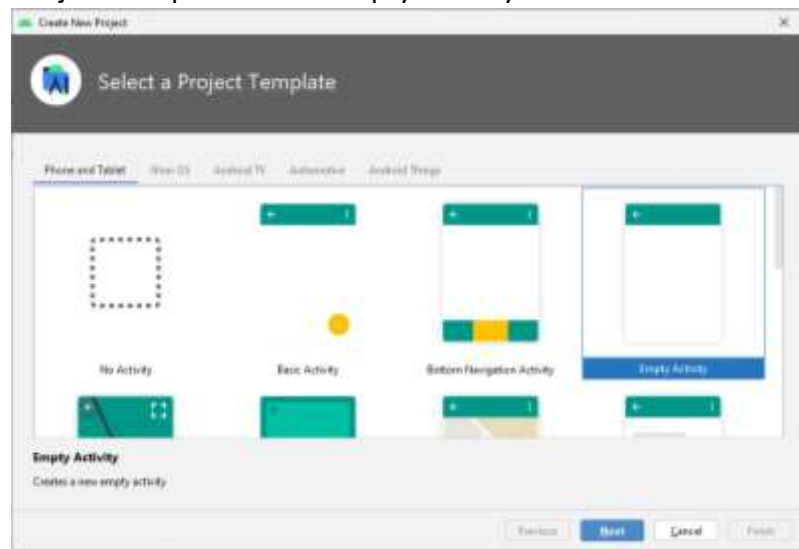
Layout merupakan tampilan dari aplikasi android yang memungkinkan untuk menata komponen – komponen pada aplikasi seperti Foto, Teks, Video maupun komponen lainnya. Fungsi layout mirip dengan kertas, yaitu sebagai media dalam menggambar. Pada layout kita bisa menambahkan komponen aplikasi bahkan, mengatur tampilan menjadi lebih indah dan nyaman di mata pengguna aplikasi.

Tahapan Membuat Project Baru

❖ File > New > New Project



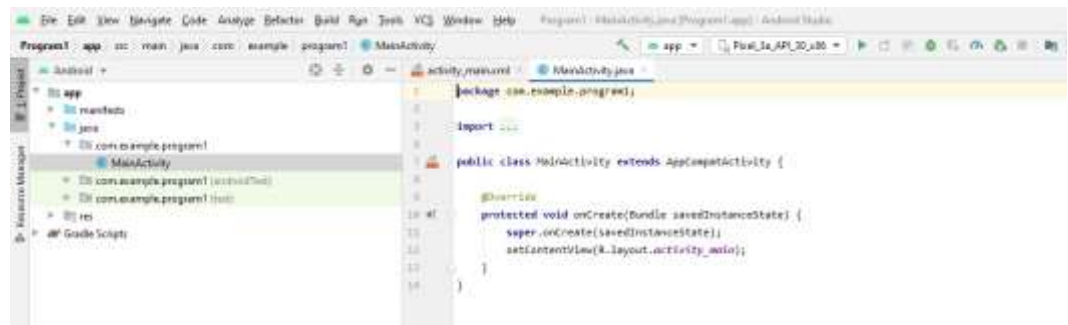
❖ Select a Project Template > Pilih Empty Activity > Next



❖ Configure Your Project

- Name : Isi Nama Project Aplikasi
- Package Name : Mengikuti Nama Project
- Save Location : Lokasi Project yang dibuat
- Language : Bahasa Pemrograman yang digunakan (Java / Kotlin)
- Minimum SDK : Minimum SDK Android yang digunakan untuk aplikasi

Jika sudah diisi > Klik Finish

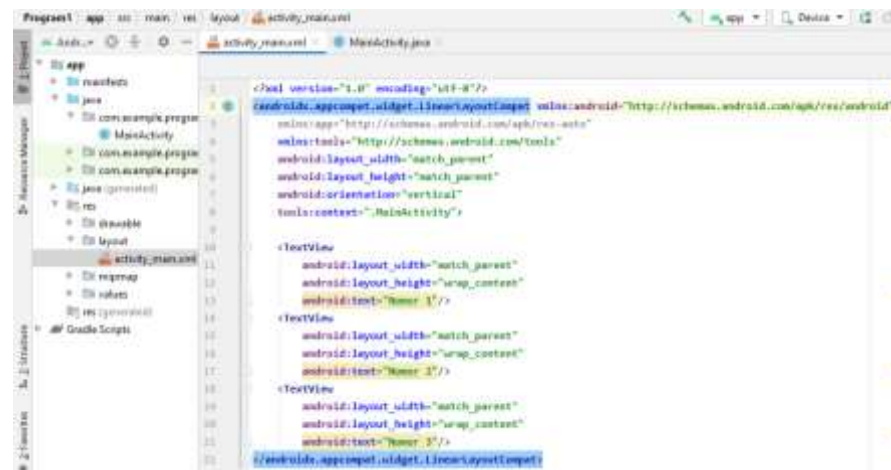


➤ Linear Layout

Merupakan layout yang berfungsi untuk menampilkan komponen – komponen aplikasi secara vertical atau horizontal.

Linear Layout pada android studio diawali dengan Tag pembuka <LinearLayout> dan diakhiri dengan tag </LinearLayout> pada file Activity.XML

Kode XML :



Hasil Program :



Silahkan coba dengan mengganti orientasi dari vertical menjadi horizontal. Dan modifikasi Teks dari program yang dibuat.

➤ Relative Layout

Merupakan layout yang berfungsi untuk menampilkan komponen – komponen aplikasi secara bebas tidak hanya vertical atau horizontal.

Relative Layout pada android studio diawali dengan Tag pembuka <RelativeLayout> dan diakhiri dengan tag </RelativeLayout> pada file Activity.XML

Kode XML :

```
<?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"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_alignParentLeft="true"
        android:text="Nomor 1"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:text="Nomor 2"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_alignParentRight="true"
        android:text="Nomor 3"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerVertical="true"
        android:layout_alignParentLeft="true"
        android:text="Nomor 4"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:text="Center"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_centerVertical="true"
        android:layout_alignParentRight="true"
        android:text="Nomor 5"/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_alignParentLeft="true"
        android:text="Nomor 6"/>
    <TextView
```

```

        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentBottom="true"
        android:layout_centerHorizontal="true"
        android:text="Nomor 7"/>
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentBottom="true"
    android:layout_alignParentRight="true"
    android:text="Nomor 8"/>
</RelativeLayout>

```

Hasil Program :



➤ Frame Layout

Merupakan layout yang berfungsi untuk menampilkan komponen – komponen aplikasi secara tumpang tindih.

Frame Layout pada android studio diawali dengan Tag pembuka <FrameLayout> dan diakhiri dengan tag </FrameLayout> pada file Activity.XML

Kode XML :

```

<?xml version="1.0" encoding="utf-8"?>
<FrameLayout 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"
    tools:context=".MainActivity">

    <ImageView
        android:layout_width="300dp"
        android:layout_height="300dp"
        android:layout_gravity="center"
        android:background="@color/design_default_color_primary"/>

    <Button
        android:layout_width="300dp"
        android:layout_height="100dp"
        android:layout_gravity="center"
        android:text="Contoh Button"/>

```

Hasil Program :



➤ **TextView**

Komponen untuk menampilkan teks pada layar. TextView berisi property dari ukuran, style, dan warna. Berikut property untuk TextView

Kode XML :

```
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Contoh"
    android:textSize="12dp"
    android:textColor="@color/design_default_color_primary"
    android:textStyle="bold"
    android:id="@+id/TextView1"/>
```

TextView"

Hasil Program :



➤ **EditText**

EditText merupakan kustomisasi dari TextView yang dapat dikonfigurasi atau disunting sendiri.

Kode XML :

```
<EditText
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Contoh TextView"
    android:textSize="12dp"
    android:textColor="@color/design_default_color_primary"
    android:textStyle="bold"
    android:id="@+id/TextView1"/>
```

Hasil Program :



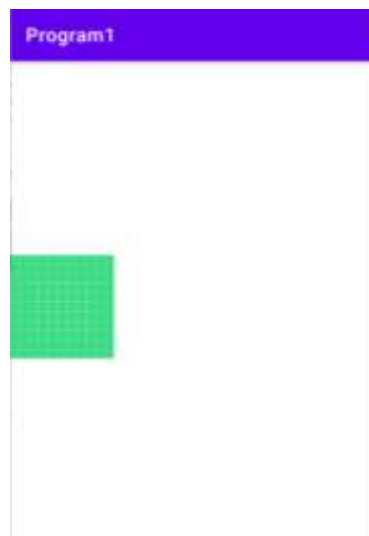
➤ **ImageView**

ImageView merupakan komponen yang digunakan untuk menampilkan gambar atau Ikon.

Kode XML :

```
<ImageView
    android:layout_width="wrap_content"
    android:layout_height="match_parent"
    android:src="@drawable/ic_launcher_background"/>
```

Hasil Program :



➤ **Button**

Button merupakan komponen yang digunakan untuk Tombol.

Kode XML :

```
<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Contoh Button"/>
```

Hasil Program :



➤ Event Handling

Event Listener merupakan antar muka pada kelas View yang berisi metode pemanggilan Kembali. Metode ini akan dipanggil android Ketika view yang terdaftar dengan pendengar dipicu oleh interaksi pengguna.

| Method | Description |
|-----------------------|---|
| onClick() | This method is called when the user touches or focuses on the item using navigation-keys or trackball or presses on the "enter" key or presses down on the trackball. |
| onLongClick() | This method is called when the user touches and holds the item or focuses on the item using navigation-keys or trackball and presses and holds "enter" key or presses and holds down on the trackball (for one second). |
| onFocusChange() | This method is called when the user navigates onto or away from the item, using the navigation-keys or trackball. |
| onKey() | This method is called when the user is focused on the item and presses or releases a hardware key on the device. |
| onTouch() | This method is called when the user performs a touch event, including a press, a release, or any movement gesture on the screen. |
| onCreateContextMenu() | This method is called when a Context Menu is being built. |

Contoh Kode XML Event On Click :

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.appcompat.widget.LinearLayoutCompat
    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="wrap_content"
    android:layout_height="wrap_content"
    android:orientation="vertical"
    tools:context=".MainActivity">

    <TextView
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:text="Contoh Event On Click"
        android:textSize="15dp"
```



```

        android:textAlignment="center"/>
<Button
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:text="Tampilkan Data"/>
<TextView
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:id="@+id/hasilButton"/>
</androidx.appcompat.widget.LinearLayoutCompat>

```

Hasil Program :



Agar Button Tampilkan Data dapat menampilkan hasil data maka lakukan modifikasi pada MainActivity.Java, sebagai berikut :

```

public class MainActivity extends AppCompatActivity {
    Button button1; //Deklarasi Button
    TextView textView; //Deklarasi TextView

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        button1 = (Button) findViewById(R.id.buton) //Memanggil Button
        textView = (TextView) findViewById(R.id.hasilButton) // Menampilkan Hasil

        //Event On Click
        button1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                textView.setText("Anda Melakukan Klik Tombol");
            }
        });
    }
}

```

Hasil Program :



➤ Gaya dan Tema Aplikasi

Gaya dan tema di Android memungkinkan Anda memisahkan detail desain aplikasi dari struktur dan perilaku UI, mirip dengan stylesheet di desain web. Gaya adalah kumpulan atribut yang menentukan tampilan untuk view Tunggal. Gaya dapat menentukan atribut seperti warna font, ukuran font, warna latar belakang, dan banyak lagi.

Tema adalah jenis gaya yang diterapkan ke seluruh aplikasi, aktivitas, atau hierarki tampilan — bukan hanya tampilan individual. Saat Anda menerapkan gaya sebagai tema, setiap tampilan di aplikasi atau aktivitas akan menerapkan setiap atribut gaya yang didukungnya. Tema juga dapat menerapkan gaya ke elemen non-tampilan, seperti status bar dan latar belakang jendela. Gaya dan tema dideklarasikan di `res/values/` bernama `Styles.XML`

❖ Gaya dan Tema Terang (Light)

Mengatur gaya dan tema dengan mode terang dengan melakukan modifikasi pada bagian `Res > Values > Styles` untuk Android Studio versi lama. Sedangkan untuk versi android terbaru dapat dimodifikasi pada `Res > Themes > themes.xml` (untuk terang) dan `themes.xml (night)` untuk mode gelap. Untuk versi android studio terbaru sudah mendukung dua versi tema tidak perlu dibuat ulang.

❖ Gaya dan Tema Gelap (Dark)

Mengatur gaya dan tema dengan mode terang dengan melakukan modifikasi pada bagian `Res > Values > Styles` untuk Android Studio versi lama. Sedangkan untuk versi android terbaru dapat dimodifikasi pada `Res > Themes > themes.xml` (untuk terang) dan `themes.xml (night)` untuk mode gelap. Untuk versi android studio terbaru sudah mendukung dua versi tema tidak perlu dibuat ulang.

Latihan :

Dari beberapa tutorial diatas dan contoh program. Silahkan anda berlatih dengan memodifikasi atau membuat program baru dengan beberapa jenis Layout, berbeda komponen dan tema. Serta mencoba membuat aplikasi dengan kombinasi berbagai komponen dan layout yang telah dijelaskan diatas. Good Luck selamat belajar