PRAKTEK MANDIRI#2
ANDROID LAYOUT

Creating Layout

To create UI layout for your activity you can do in two ways.

1. Creating view object from code programatically

   1.1.Write these code in your MainActivity.java

```java
public class MainActivity extends ActionBarActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        RelativeLayout rl = new RelativeLayout(this);
        TextView tv = new TextView(this);
        RelativeLayout.LayoutParams tvParams = new RelativeLayout.LayoutParams(
                                        RelativeLayout.LayoutParams.WRAP_CONTENT,
                                        RelativeLayout.LayoutParams.WRAP_CONTENT);
        tvParams.addRule(RelativeLayout.CENTER_HORIZONTAL);
        tv.setText("Assalamualaikum");
        rl.addView(tv,tvParams);
        setContentView(rl);
    }
}
```

   1.2.Run and see the result

2. Creating with xml

   2.1.Write these code in activity_main.xml

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    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="com.example.belajarlayout.MainActivity" >

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:text="@string/hello_world" />

</RelativeLayout>
```

2.2.Write these code in your MainActivity.java

```java
public class MainActivity extends ActionBarActivity {

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

}
```

2.3.Run and see the result

Linear Layout
1. Layout Horizontal
    1.1. Create new android project, name it with "Linear Layout Demo".
    1.2. Create layout_horizontal.xml under res/layout/ folder

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    tools:context=".MainActivity"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <TextView
        android:id="@+id/a"
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:background="#00FF00"
        android:text="satu" />

    <TextView
        android:id="@+id/b"
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:background="#FF0000"
        android:text="dua" />

    <TextView
        android:id="@+id/c"
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:background="#0000FF"
        android:text="tiga" />

</LinearLayout>
```

    1.3. Use layout_horizontal.xml as layout for MainActivity
    1.4. Run and see the result

2. Layout Horizontal

2.1. Copy layout_horizontal.xml to layout_vertical.xml, modify these following codes:

```xml
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
xmlns:tools="http://schemas.android.com/tools"
tools:context=".MainActivity"
android:layout_width="match_parent"
android:layout_height="match_parent"
android:orientation="vertical"
>
```

2.2. Use layout_vertical.xml as layout for MainActivity
2.3. Run and see the result

3. Modify Height

3.1. Create form_layout.xml

```xml
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="16dp"
    android:paddingRight="16dp"
    android:orientation="vertical" >
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="to" />
    <EditText
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="subject" />
    <EditText
        android:layout_width="match_parent"
        android:layout_height="0dp"
        android:layout_weight="1"
        android:gravity="top"
        android:hint="message" />
    <Button
        android:layout_width="100dp"
        android:layout_height="wrap_content"
        android:layout_gravity="right"
        android:text="send" />
</LinearLayout>
```

3.2. Use form_layout.xml as layout for MainActivity
3.3. Run and see the result

Relative Layout

1. Create new android project, name it with "Relative Layout Demo".
2. Create relative_layout.xml and write these code

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:paddingLeft="16dp"
    android:paddingRight="16dp" >
    <EditText
        android:id="@+id/name"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:hint="reminder" />
    <Spinner
        android:id="@+id/dates"
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_below="@id/name"
        android:layout_alignParentLeft="true"
        android:layout_toLeftOf="@+id/times" />
    <Spinner
        android:id="@id/times"
        android:layout_width="96dp"
        android:layout_height="wrap_content"
        android:layout_below="@id/name"
        android:layout_alignParentRight="true" />
    <Button
        android:layout_width="96dp"
        android:layout_height="wrap_content"
        android:layout_below="@id/times"
        android:layout_alignParentRight="true"
        android:text="done" />
</RelativeLayout>
```

3. Use relative_layout.xml as layout for MainActivity
4. Run and see the result

List View

1. Static ListView item from xml
    1.1. Create new android project, name it with "List View Demo".
    1.2. Create listview_layout.xml and write these code

```xml
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="#FFFFFF">
    <ListView
        android:id="@+id/listView1"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        android:entries="@array/data_list"/>
</LinearLayout>
```

    1.3. Add string array at res/values/strings.xml

```xml
<string-array name="data_list">
    <item>Isi 1</item>
    <item>Isi 2</item>
    <item>Isi 3</item>
    <item>Isi 4</item>
    <item>Isi 5</item>
    <item>Isi 6</item>
    <item>Isi 7</item>
    <item>Isi 8</item>
    <item>Isi 9</item>
    <item>Isi 10</item>
    <item>Isi 11</item>
    <item>Isi 12</item>
    <item>Isi 13</item>
    <item>Isi 14</item>
    <item>Isi 15</item>
</string-array>
```

    1.4. Use listview_layout.xml as layout for MainActivity
    1.5. Run and see the result

2. Static ListView using array adapter
    2.1. Create new activity ListArrayActivity.java, set as main activity, modify onCreate() method

```java
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.listarray_layout);
    ListView lv = (ListView) findViewById(R.id.listView1);
    List<String> data = new ArrayList<String>();
    data.add("satu");
    data.add("dua");
    data.add("tiga");
    data.add("empat");
    data.add("lima");
    ArrayAdapter<String> aa = new ArrayAdapter<String>(this,
            android.R.layout.simple_list_item_1, data);
    lv.setAdapter(aa);
}
```

2.2. Use layout listarray_layout.xml

```xml
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:background="#FFFFFF">
    <ListView
        android:id="@+id/listView1"
        android:layout_width="fill_parent"
        android:layout_height="fill_parent"
        />
</LinearLayout>
```

2.3. Run and see the result

Grid View

1. Modify activity_main.xml

```xml
<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    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="com.example.gridviewdemo.MainActivity" >

    <TextView
        android:id="@+id/textView1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_alignParentTop="true"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="18dp"
        android:text="Demo GridView" />

    <GridView
        android:id="@+id/gridView1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@+id/textView1"
        android:layout_centerHorizontal="true"
        android:layout_marginTop="18dp"
        android:numColumns="3" >
    </GridView>

</RelativeLayout>
```

2. Add image to drawwable folder (example:andro1.png,andro2.png,andro3.png, …)
3. Create class ImageAdapter.java
4. Set adapter at MainActivity.java, change onCreate() method to:

```java
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);
    GridView gv = (GridView) findViewById(R.id.gridView1);
    gv.setAdapter(new ImageAdapter(this));

}
```

```java
import android.content.Context;
import android.view.View;
import android.view.ViewGroup;
import android.widget.BaseAdapter;
import android.widget.GridView;
import android.widget.ImageView;

public class ImageAdapter extends BaseAdapter {
    private Context mContext;

    public Integer[] gambar = {
            R.drawable.andro1,R.drawable.andro2,R.drawable.andro3,
            R.drawable.andro3,R.drawable.andro1,R.drawable.andro2
    };

    public ImageAdapter(Context c){
        mContext = c;
    }

    @Override
    public int getCount() {
        return gambar.length;
    }

    @Override
    public Object getItem(int position) {
        return gambar[position];
    }

    @Override
    public long getItemId(int position) {
        return 0;
    }

    @Override
    public View getView(int position, View convertView, ViewGroup parent) {
        ImageView imageView = new ImageView(mContext);
        imageView.setImageResource(gambar[position]);
        imageView.setScaleType(ImageView.ScaleType.CENTER_CROP);
        imageView.setLayoutParams(new GridView.LayoutParams(160, 160));
        return imageView;
    }


}
```