Android Note

I. Layout: Gồm các loại layout sau

- Framelayout , LinearLayout, RelativeLayout, TableLayout, GridLayout, ConstraintLayout

1. Framelayout: Luôn giữ các view bên trong ở góc trái màn hình và không thể thay đổi location hiển thị, Các view được thiết lập sau sẽ đè lên các View thiết lập trước.

<FrameLayout 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"

tools:context=".MainActivity" >

<TextView

android:id="@+id/textView"

android:layout\_width="300dp"

android:layout\_height="300dp"

android:background="@color/colorAccent"

android:text="TextView" />

<TextView

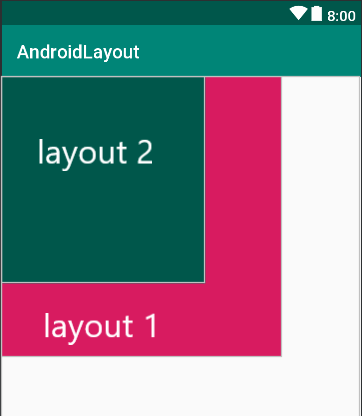
android:id="@+id/textView2"

android:layout\_width="218dp"

android:layout\_height="221dp"

android:background="@color/colorPrimaryDark" />

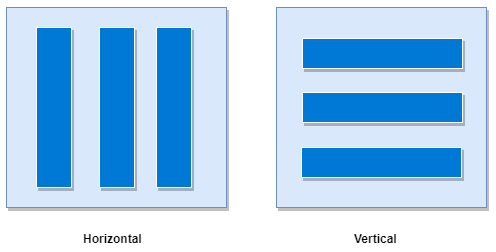
</FrameLayout>



2. LinearLayout: Bố trí các view theo chiều ngang hoặc chiều dọc, mặc định sẽ là chiều ngang.

Orientation = Vertical : Các view bên trong sẽ được xếp từ trên xuống.

Orientation = Horizontal: Các view bên trong sẽ được xếp trái qua phải.



<LinearLayout

xmlns:android1="http://schemas.android.com/apk/res/android"

android1:layout\_width="match\_parent"

android1:layout\_height="match\_parent"

android1:orientation="horizontal" >

<TextView

android1:layout\_width="80dp"

android1:layout\_height="80dp"

android1:text="1"

android1:textColor="#fff"

android1:textSize="15pt"

android1:textAlignment="center"

android1:textStyle="bold"

android1:background="@color/colorAccent" />

<TextView

android1:layout\_width="80dp"

android1:layout\_height="80dp"

android1:text="1"

android1:textColor="#fff"

android1:textSize="15pt"

android1:textAlignment="center"

android1:textStyle="bold"

android1:background="@color/colorPrimary" />

<TextView

android1:layout\_width="80dp"

android1:layout\_height="80dp"

android1:text="3"

android1:textColor="#fff"

android1:textSize="15pt"

android1:textAlignment="center"

android1:textStyle="bold"

android1:background="#8c0520" />

<TextView

android1:layout\_width="80dp"

android1:layout\_height="80dp"

android1:background="#efcd21"

android1:text="4"

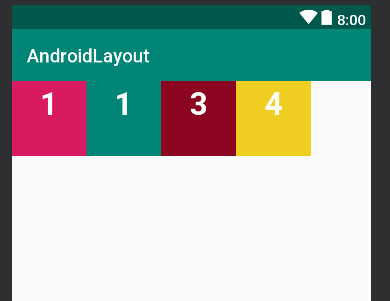
android1:textAlignment="center"

android1:textColor="#fff"

android1:textSize="15pt"

android1:textStyle="bold" />

</LinearLayout>



- android:weightSum="2": Setting tổng bố cục của item cha

- android:layout\_weight="{number}": Setting ở child item để chia bố cục dựa theo weightSum của item cha.

**Chú ý:** Cần setting android:layout\_height="0dp" nếu đang xếp từ trên xuống   
hoặc android:layout\_width="0dp" nếu đang xếp từ trái sang phải để có thể bố cục theo giá trị layout\_weight

3. RelativeLayout: Vị trí của mỗi view bên trong sẽ được xác định so với view khác hoặc so với thành phần cha của nó thông qua ID.

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

android:layout\_width="match\_parent"

android:layout\_height="match\_parent">

<TextView

android:layout\_width="150dp"

android:layout\_height="50dp"

android:text="1"

android:textColor="#fff"

android:id="@+id/hop1"

android:textSize="10pt"

android:textAlignment="center"

android:background="@color/colorAccent" />

<TextView

android:layout\_width="150dp"

android:layout\_height="50dp"

android:text="2"

android:textColor="#fff"

android:id="@+id/hop2"

android:textSize="10pt"

android:textAlignment="center"

android:layout\_toRightOf="@+id/hop1"

android:background="@color/colorPrimary" />

<TextView

android:layout\_width="150dp"

android:layout\_height="50dp"

android:text="3"

android:textColor="#fff"

android:id="@+id/hop3"

android:background="#8e0917"

android:textSize="10pt"

android:textAlignment="center"

android:layout\_below="@+id/hop1" />

<TextView

android:layout\_width="150dp"

android:layout\_height="50dp"

android:text="4"

android:textColor="#fff"

android:id="@+id/hop4"

android:textSize="10pt"

android:textAlignment="center"

android:background="#edcb25"

android:layout\_below="@+id/hop2"

android:layout\_toRightOf="@+id/hop3" />

</RelativeLayout>

4. TableLayout: Các view bên trong được xếp theo dạng table. Mỗi row là 1 TableRow , bên trong TableRow sẽ chứa các View.

<TableLayout android:layout\_height="match\_parent"

android:layout\_width="match\_parent"

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

<TableRow>

<TextView

android:text="TableLayout Example"

android:background="#c98282"

android:gravity="center"/>

</TableRow>

<TableRow>

<Button android:text="B1 B1" />

<Button android:text="B2"/>

</TableRow>

<TableRow>

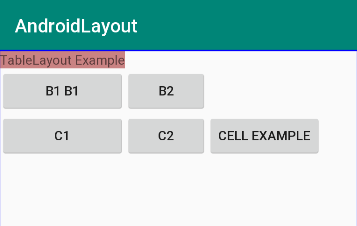
<Button android:text="C1" />

<Button android:text="C2" />

<Button android:text="Cell example" />

</TableRow>

</TableLayout>



5. GridLayout: Layout dạng lưới với số dòng và số cột ta quy định. Mỗi View bên trong sẽ nằm ở mỗi ô trong lưới này.

<GridLayout

xmlns:android1="http://schemas.android.com/apk/res/android"

android1:layout\_width="match\_parent"

android1:layout\_height="match\_parent"

android1:columnCount="2"

android1:rowCount="2"

>

<TextView

android1:layout\_width="80dp"

android1:layout\_height="80dp"

android1:text="1"

android1:textColor="#fff"

android1:textSize="15pt"

android1:textAlignment="center"

android1:textStyle="bold"

android1:background="@color/colorAccent" />

<TextView

android1:layout\_width="80dp"

android1:layout\_height="80dp"

android1:text="2"

android1:textColor="#fff"

android1:textSize="15pt"

android1:textAlignment="center"

android1:textStyle="bold"

android1:background="@color/colorPrimary" />

<TextView

android1:layout\_width="80dp"

android1:layout\_height="80dp"

android1:text="3"

android1:textColor="#fff"

android1:textSize="15pt"

android1:textAlignment="center"

android1:textStyle="bold"

android1:background="#8c0520" />

<TextView

android1:layout\_width="80dp"

android1:layout\_height="80dp"

android1:background="#efcd21"

android1:text="4"

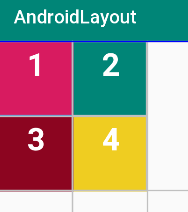
android1:textAlignment="center"

android1:textColor="#fff"

android1:textSize="15pt"

android1:textStyle="bold" />

</GridLayout>



6. ConstraintLayout:

II. Items:

- Button:

android:textAllCaps="true/false" : Setting tất cả chữ hoa(true)/chữ thường(false)

- EditText:

android:hint="{text}": Hiển thị gợi ý

android:drawableLeft="{image}", android:drawableRight="{image}" : Set image trong EditText

- ImageView:

android:scaleType="{setting}" : Set tỉ lệ của image so với khung của image (khung của image không phải là khung activity)

- ImageButton:

android:background="@null" : Bỏ đi phần background của button, chỉ giữ lại image

- RadioButton: Setting các RadioButton cùng 1 Group: kéo thả RadioGroup -> kéo thả RadioButton

<RadioGroup>

<RadioButton/>

<RadioButton/>

</RadioGroup>

**Inflate:**

Instantiates a layout XML file into its corresponding {**@link** android.view.View} objects.

Nó dùng để inflate (bơm, nạp…) cái phần XML (Phần giao diện) vào View.

Giả sử bạn muốn tạo một cái view, thì để biết cái view đó đang chịu trách nhiệm phần giao diện nào (XML) thì bạn phải inflate cái XML đó vào view.

Một trường hợp rất thường gặp là khi bạn hiện thực CustomAdapter cho ListView. Trong method getView(), bạn cần phải inflate cái XML (ở đây là layout cho item) vào biến convertView thì ListView mới biết là phần giao diện nào của item cần được thể hiện ra màn hình.

Lớp LayoutInflater này rất hữu ích nếu bạn dùng để tạo giao diện linh hoạt (có thể thêm bớt các view vào root view) chỉ bằng code thuần .

Drawable:

1. shape: hình dạng  
Trong thẻ shape cần có thuộc tính android:shape để khai báo hình dạng

- solid: mang giá trị color

- size: kích thước(height, width)

- corners: bo góc (radius)

- padding: padding item

- stroke: tạo viền (color: màu viền, width: độ rộng của viền, dashGap(chiều dài nét đứt),dashWidth(chiều dài nét liền): tạo nét đứt của viền)

- grandient: tạo độ đậm nhạt theo góc nghiêng(angle: góc(tối thiểu 45), startColor: màu bắt đầu, centerColor: màu ở giữa, endColor: màu kết thúc)

2. layer-list: xếp chồng các layer lên nhau

- item: là layer(top: cách trên, left: cách trái,…)

+ bitmap: thẻ hình ảnh(src: đường dẫn hình ảnh, gravity: trọng tâm hình ảnh)

3. selector: xử lý trạng thái select

- item: là item(drawable: thẻ drawable, state\_pressed: trạng thái nhấn vào, color: màu chữ)

Note: vì selector duyệt item từ trên xuống nên item mang giá trị default phải để cuối cùng.

4. clip: cắt hình (drawable: hình, clipOrientation: hướng cắt”horizontal/vertical”, gravity: cắt từ vị trí nào)

Note: code behind cần xử lý setImageLevel để có thể load được clip

Date time:

- Giá trị month được tính từ 0-11 tháng