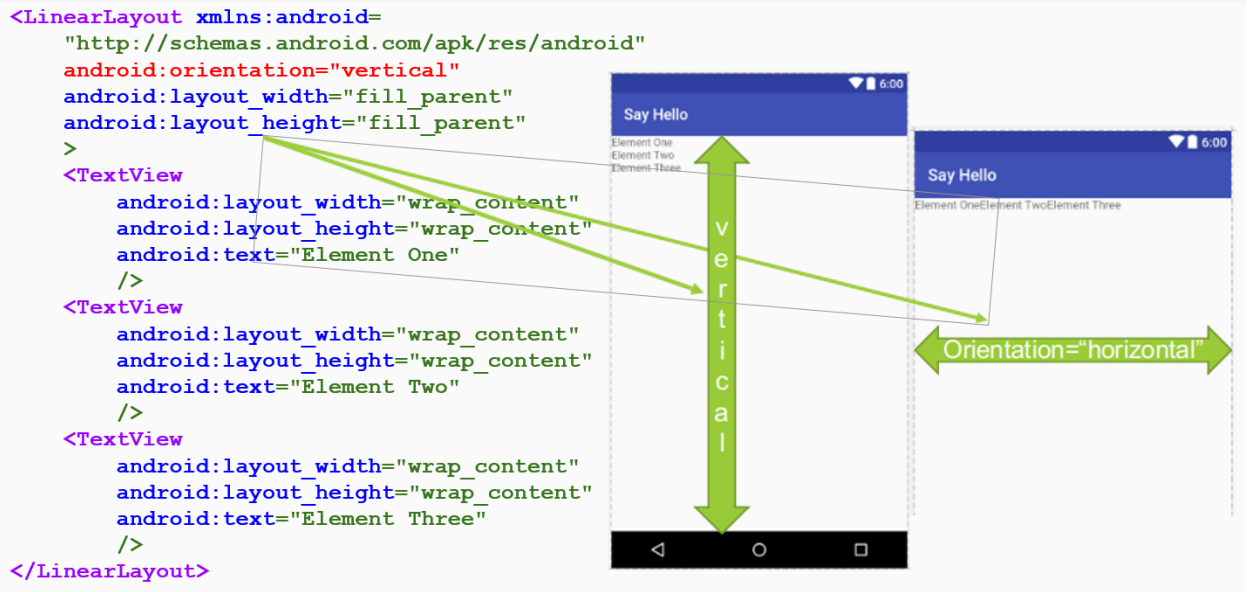


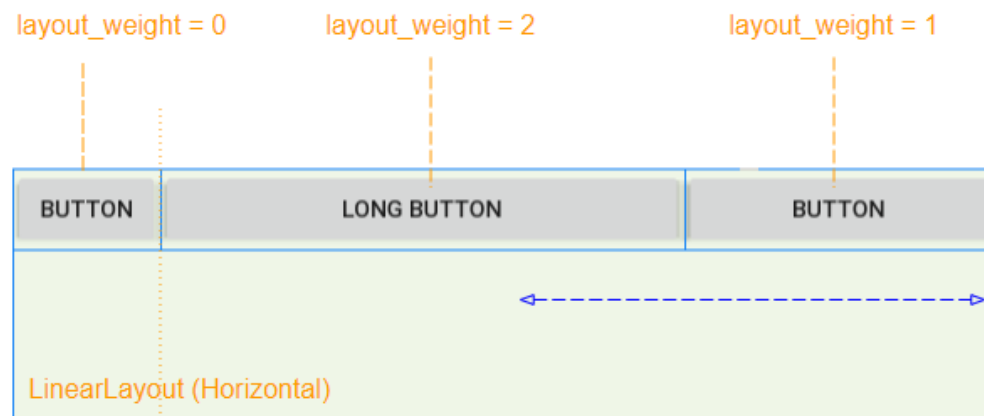
Week3: Android Views & Layouts

1. Linear Layout

- Một GroupView cho phép / sắp xếp các child View elements ngang hoặc dọc



- Thiết lập thuộc tính **android:orientation="..."** thành trong file layout xml hoặc `setOrientation()` bằng mã khi chương trình đang chạy để thay đổi hướng.
- Kích thước của các element:
 - o Giá trị cụ thể: 200 dip
 - o Vừa đủ nội dung bên trong: `wrap_content`
 - o Bằng độ lớn của đối tượng cha: `fill_parent` hoặc `match_parent`
- Weight
 - o `Android:layout_weight` – trọng số để xác định tỉ lệ tương ứng
 - o Giá trị có thể là 1, 2, 3... mặc định là 0



- Gravity: Xác định cách căn lề, các giá trị có thể là left, center, right, top, bottom,

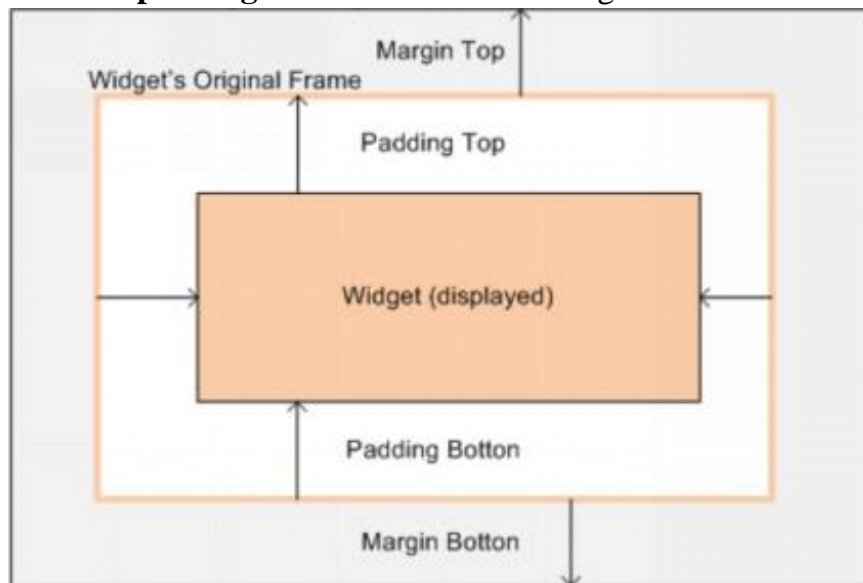
`android:gravity="center"`



`android:layout_gravity="center"`



- Margin & Padding
 - `android:layout_margin="..."` : Cách lề bên ngoài
 - `android:padding="..."` : Cách lề bên trong



2. Relative Layout

- Các thành phần của nó có vị trí tương đối so với thành phần cha hoặc anh chị em
- Một số thuộc tính

Tên thuộc tính	Mô tả
<u>android:layout_above</u>	Đặt phần tử hiện tại nằm kế sau phần tử có id được chỉ ra
<u>android:layout_alignBaseline</u>	Đặt phần tử này lên cùng dòng với phần tử có id được chỉ ra
<u>android:layout_alignBottom</u>	Canh sao cho đáy của phần tử hiện thời trùng với đáy của phần tử có id được chỉ ra
<u>android:layout_alignLeft</u>	Đặt cạnh trái của phần tử hiện thời trùng với cạnh trái của phần tử có id được chỉ ra
<u>android:layout_alignParentBottom</u>	Nếu thiết lập là true thì phần tử hiện thời sẽ được canh xuống đáy của phần tử chứa nó
<u>android:layout_alignParentLeft</u>	Nếu được thiết lập là true thì phần tử hiện thời sẽ canh trái so với phần tử chứa nó
<u>android:layout_alignParentRight</u>	Nếu được thiết lập là true thì phần tử hiện thời sẽ canh phải so với phần tử chứa nó
<u>android:layout_alignParentTop</u>	Nếu được thiết lập là true thì phần tử hiện thời sẽ canh lên đỉnh phần tử chứa nó
<u>android:layout_alignRight</u>	Canh cạnh phải của phần tử hiện thời trùng với cạnh phải của phần tử có id được chỉ ra
<u>android:layout_alignTop</u>	Canh đỉnh của phần tử hiện thời trùng với đỉnh của phần tử có id được chỉ ra
<u>android:layout_alignWithParentIfMissing</u>	Nếu thiết lập là true, thì phần tử sẽ được canh theo phần tử chứa nó nếu các thuộc tính canh của phần tử không có.
<u>android:layout_below</u>	Đặt phần tử hiện thời ngay sau phần tử có id được chỉ ra.
<u>android:layout_centerHorizontal</u>	Nếu thiết lập là true thì phần tử hiện thời sẽ được canh giữa theo chiều ngang phần tử chứa nó.
<u>android:layout_centerInParent</u>	Nếu thiết lập là true thì phần tử hiện thời sẽ được canh chính giữa theo chiều phải trái và trên dưới so với phần tử chứa nó.
<u>android:layout_centerVertical</u>	Nếu thiết lập là true thì phần tử hiện thời sẽ được canh chính giữa theo chiều dọc phần tử chứa nó.
<u>android:layout_toLeftOf</u>	Đặt cạnh phải của phần tử hiện thời trùng với cạnh trái của phần tử có id được chỉ ra.
<u>android:layout_toRightOf</u>	Đặt cạnh trái của phần tử hiện thời trùng với cạnh phải của phần tử có id được chỉ ra.

3. Absolution Layout

- Chỉ định vị trí các view con thông qua tọa độ (x, y)

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

    <RelativeLayout
        android:id="@+id/x1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content">

        <TextView
            android:id="@+id/txt1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Text 1" />

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_centerVertical="true"
            android:layout_toRightOf="@+id/txt1"
            android:text="Text 2" />

    </RelativeLayout>

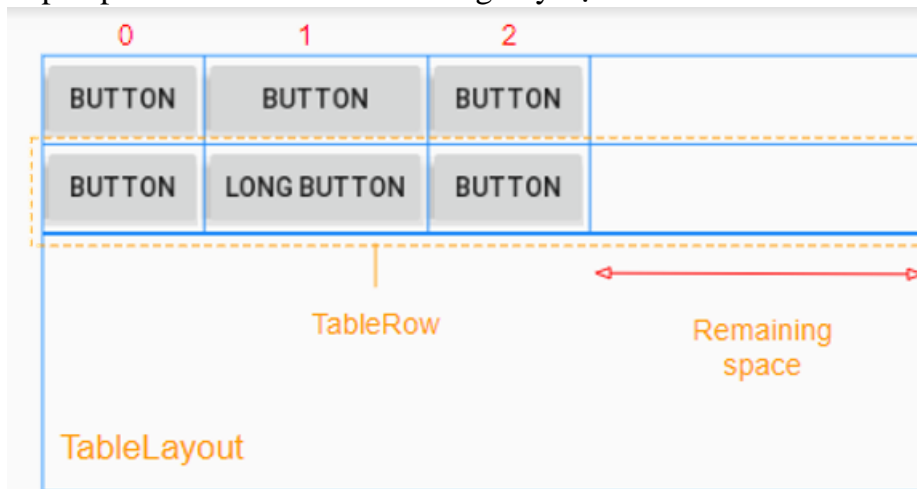
    <Button
        android:id="@+id/btn1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/x1"
        android:layout_centerHorizontal="true"
        android:padding="10dp"
        android:text="Button 1" />

    <Button
        android:id="@+id/btn2"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/btn1"
        android:layout_centerHorizontal="true"
        android:text="Button 2" />

</RelativeLayout>
```

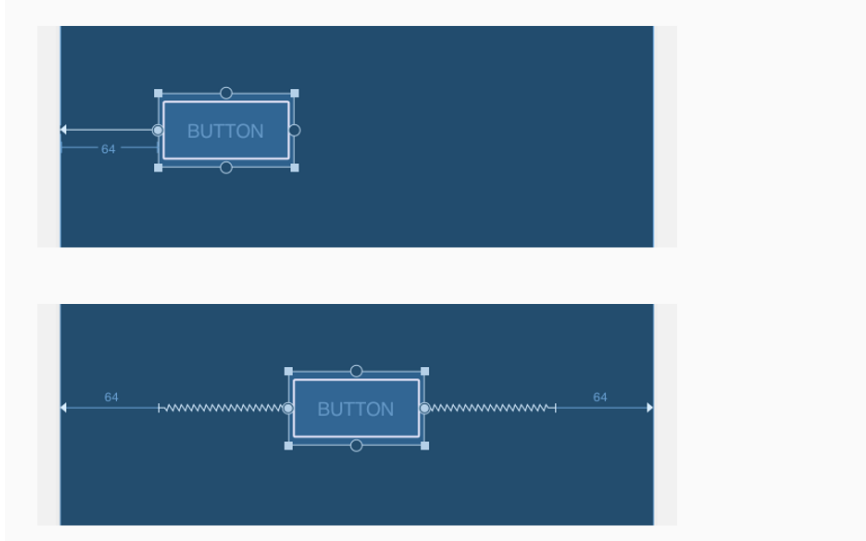
4. Table Layout

- Sắp xếp các element con theo hàng hay cột



5. ConstraintLayout

- Tương tự RelativeLayout nhưng hỗ trợ kéo thả mạnh mẽ



- Các giá trị xác định kích thước:
 - Fixed: chỉ định cứng
 - Match_constraint: tương tự match_parent trong RelativeLayout
 - Wrap_content