

Linear Layout





Nội dung bài học

- ❖ Khái niệm Linear Layout
- ❖ Các thuộc tính thường dùng

Linear Layout

- Sử dụng trong các trường hợp xây dựng bố cục tổ chức hiển thị các đối tượng theo một chiều duy nhất (chiều dọc hoặc ngang).
- Đối tượng mặc định vị trí top left trên LinearLayout , có thể sử dụng thuộc tính Gravity để thiết lập lại vị trí.
- Ví dụ khai báo:

<LinearLayout

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

`android:layout_width="match_parent"`

`android:layout_height="match_parent"`

`android:orientation="vertical" >`

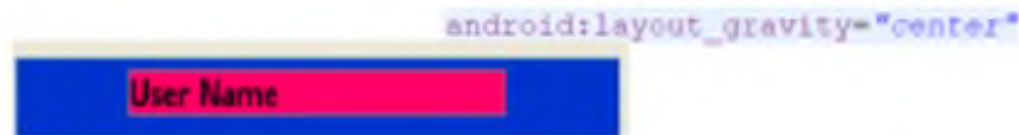
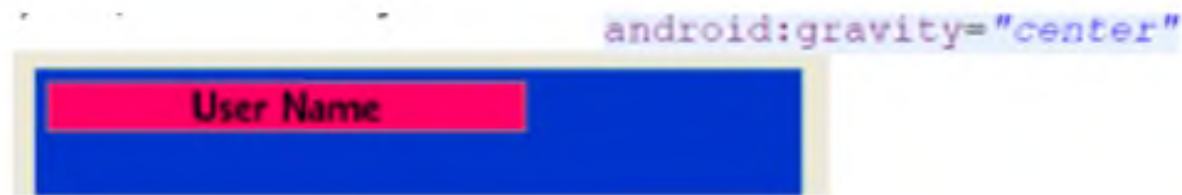
</LinearLayout>

Linear Layout



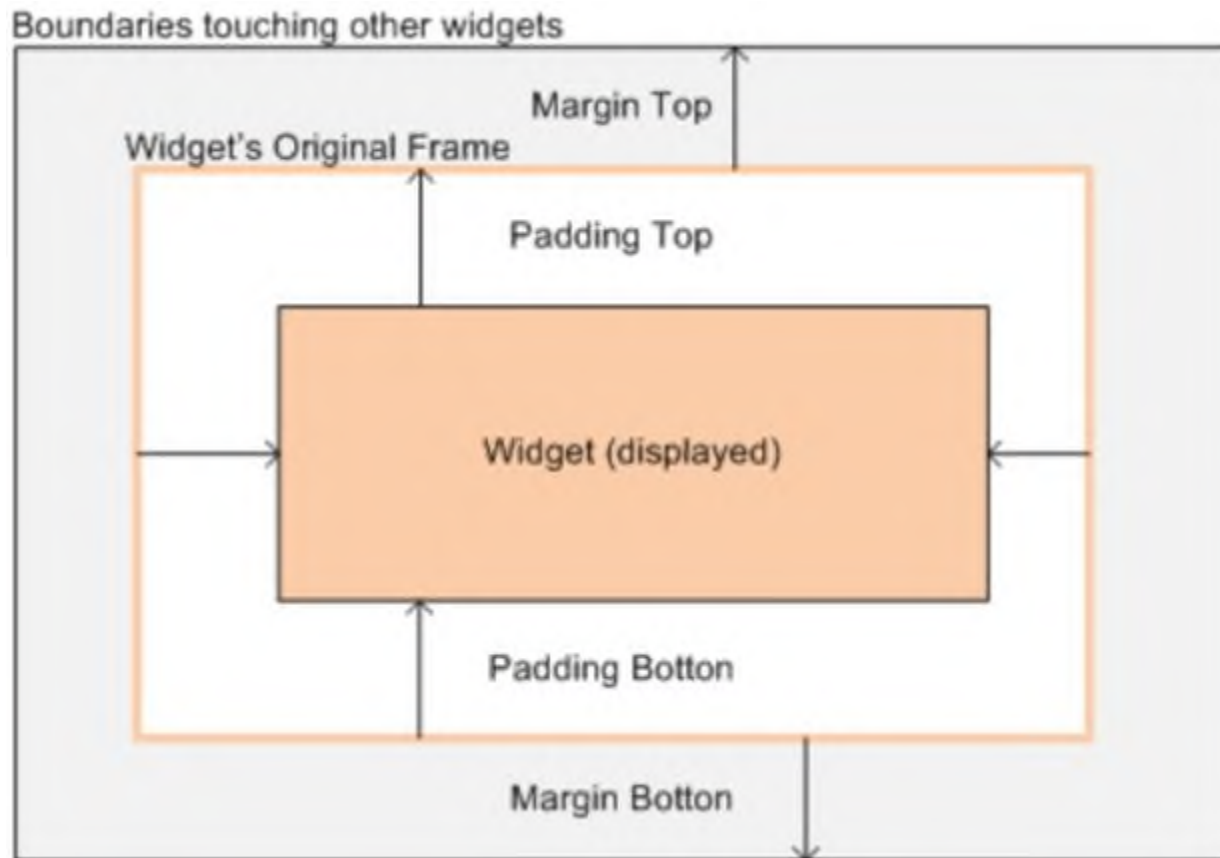
Linear Layout

- Bạn có thể dùng margin, gravity, weight để hỗ trợ cho việc thiết kế.
- Layout Gravity và Text Gravity



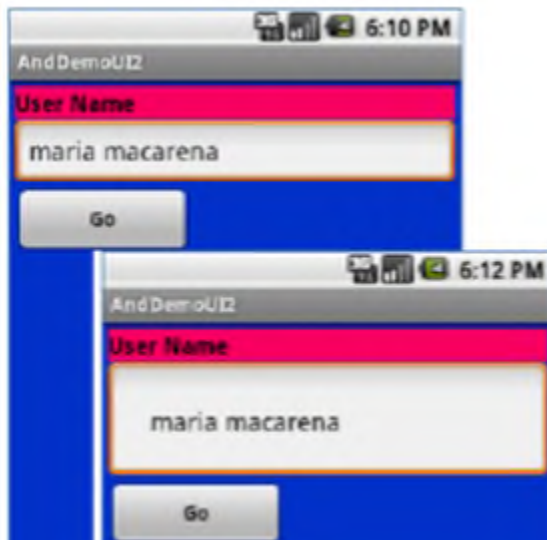
Linear Layout

Bạn cũng phải biết so sánh sự khác biệt giữa **Padding** và **Margin**:



Linear Layout

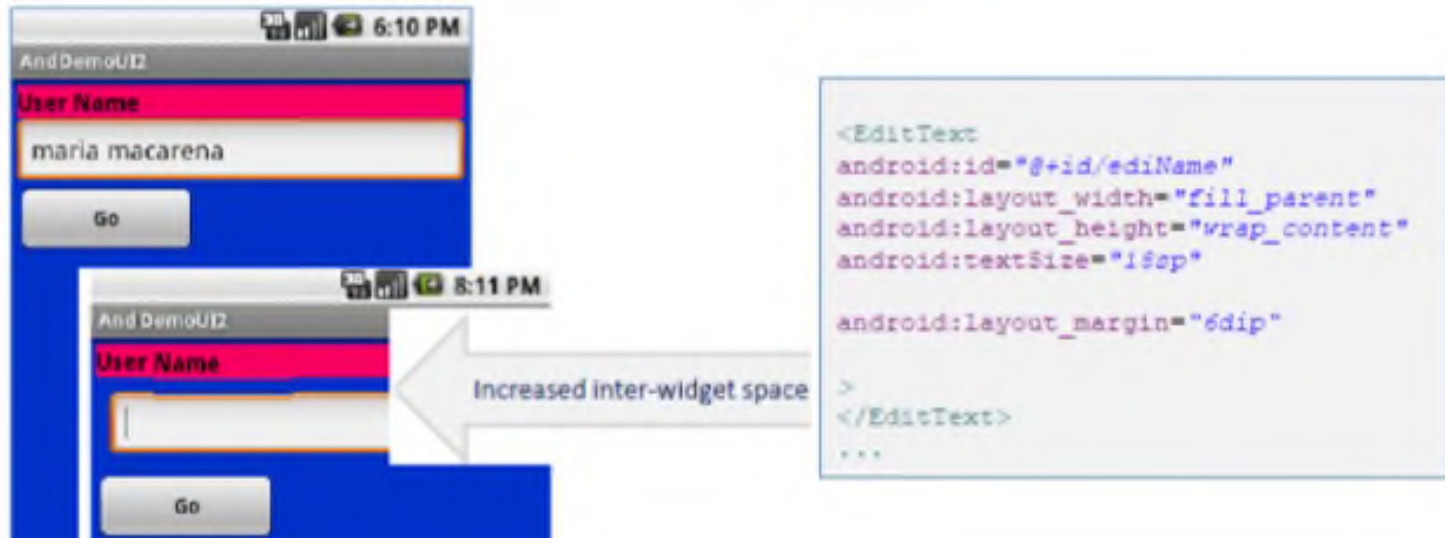
- Ví dụ thay đổi Padding (internal spacing – khoảng cách giữa nội dung bên trong so với đường viền của control):



```
<EditText
    android:id="@+id/ediName"
    android:layout_width="fill_parent"
    android:layout_height="wrap_content"
    android:textSize="18sp"
    android:padding="30dip"
/>
</EditText>
...
```

Linear Layout

- Ví dụ về đổi Margin (external spacing – khoảng cách giữa control này với control khác):



Ý nghĩa của hàm `findViewById`





Nội dung bài học

- ❖ Quy tắc đặt Id và ý nghĩa của Id
- ❖ Cách khởi tạo View trong bộ nhớ

Quy tắc đặt Id và ý nghĩa của Id

- ❖ Gọi là “android naming conventions” và cũng tùy thuộc vào tư tưởng của Lập trình viên
- ❖ Button → btn
- ❖ TextView → txt
- ❖ EditText → txt, edt
- ❖ RadioButton → rad
- ❖ CheckBox → chk
- ❖ ListView → lv

Cần đặt Id mang tính gợi nhớ. Id được định nghĩa bằng kiểu int, dùng cho Lập Trình Viên tương tác tới tài nguyên phần mềm. Nhưng Id này sẽ tự động được lưu trong R.java

Đặt Id phải giống như đặt tên biến, nếu không sẽ bị lỗi, R.java không thể phát sinh được.

Nội dung bài học

```
<Button  
    android:id="@+id/btnOk"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:text="@string/my_button_text"/  
>
```

```
Button btnOk = (Button)findViewById(R.id.btnOk);
```

findViewById truy suất tới Control/view trong bộ nhớ dựa vào id.

TextView, EditText, Button





Nội dung bài học

- ❖ TextView
- ❖ EditText
- ❖ Button

TextView

Widgets

TextView

Button

ToggleButton

CheckBox

RadioButton

CheckedTextView

Spinner

ProgressBar (Large)

ProgressBar

ProgressBar (Small)

ProgressBar (Horizontal)

SeekBar

SeekBar (Discrete)

QuickContactBadge

Component Tree

activity_main (LinearLayout)

textView - "TextView"

Nexus 4

24

AppTheme

Language

25%

0

100

200

300

0

100

200

300

textView

textView

Properties

ID

textView

layout_width

match_parent

layout_height

wrap_content

TextView

text

textView

text

contentDescription

textAppearance

Material.Small

fontFamily

sans-serif

typeface

none

textSize

14sp

lineSpacingExtra

none

textColor

?attr/textColorTertiary

textStyle

B I T

textAlignment

| || || || ||

View all properties

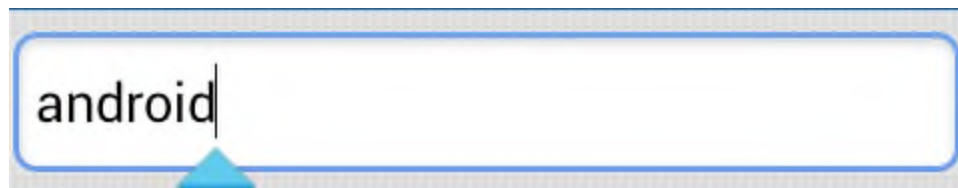


TextView

- TextView dùng để hiển thị dữ liệu và không cho phép chỉnh sửa, prefix **txt**
- Thiết lập nội dung hiển thị:
 - Trong Java code (Activity):
 - `textView.setText("Đối tượng TextView");`
 - Trong XML:
 - `android:text="Đối tượng TextView"`

EditText

- ❑ EditText là lớp kế thừa từ TextView, được dùng thay đổi nội dung text, chứa tất cả thuộc tính của TextView. Prefix là **edt** hoặc **txt**
- ❑ Được dùng để hiện thị và cho phép thay đổi dữ liệu
 - Hình minh họa:



EditText

Có rất nhiều loại EditText

activity_main.xml x MainActivity.java x

Palette

Space

Text Fields (EditText)

Plain Text

Password

Password (Numeric)

E-mail

Phone

Postal Address

Multiline Text

Time

Date

Number

Number (Signed)

Number (Decimal)

AutoCompleteTextView

MultiAutoCompleteTextVie

Layouts

Component Tree

activity_main (LinearLayout)

textView - "TextView"

editText - "Name"

editText2

editText3

25%

6

0 100 200 300

0 100 200 300

TextView

Name

LinearLayout

activity_main

textView

editText

editText2

editText3

editText4

Properties

ID

editText

layout_width

match_parent

layout_height

wrap_content

EditText

inputType

textPersonName

hint

style

editTextStyle

singleLine

-

selectAllOnFocus

-

TextView

text

Name

text

contentDescription

textAppearance

Material.Medium.Inverse

fontFamily

sans-serif

typeface

none

textSize

18sp

lineSpacingExtra

none

Button

Đối tượng Button được xây dựng từ TextView, cho phép thể hiện các nội dung văn bản, hình ảnh – nhận và phản hồi tương tác nhấn từ người dùng. Prefix **btn**

- Lắng nghe sự kiện nhấn trong Java-Code:

```
button.setOnClickListener(new OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Log.i("Obama", "onClick");  
    }  
});
```

- Lắng nghe sự kiện nhấn trong XML:

```
android:onClick="tenPhuongThuc"
```

Button

activity_main.xml x MainActivity.java x

Widgets

- TextView
- Button**
- ToggleButton
- CheckBox
- RadioButton
- CheckedTextView
- Spinner
- ProgressBar (Large)
- ProgressBar
- ProgressBar (Small)
- ProgressBar (Horizontal)
- SeekBar
- SeekBar (Discrete)
- QuickContactBadge
- RatingBar
- Switch

Component Tree

- activity_main (LinearLayout
 - textView - "TextView"
 - editText - "Name"
 - editText2
 - editText3

Nexus 4 Android 24 AppTheme Language

33% 7

HelloWorld

TextView

Name

OK OK OK

LinearLayout activity_main

Properties

ID

button

layout_width

match_parent

layout_height

wrap_content

Button

style

buttonStyle

background

drawable/btn_default_material

backgroundTint

stateListAnimator

state_list_anim_material

elevation

visibility

none

onClick

none

TextView

text

OK OK OK

text

contentDescription

textAppearance

ppCompat.Widget.Button

fontFamily

sans-serif-medium

typeface

none

onClick XML



1. Onclick in XML

Using onClick view property of view (android:onClick) in xml

```
<Button
    android:id="@+id/btnOK"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignParentRight="true"
    android:layout_below="@+id/editUserName"
    android:onClick="btnOk_click"
    android:text="@string/idstrok" />
```

```
public void btnOk_click(View v)
{
    Toast.makeText(this,
        "You click ok",
        Toast.LENGTH_LONG)
        .show();
}
```

1. Onclick in XML

Using onClick view property of view (android:onClick) in xml

`setContentView(R.layout.main)`

