



BK Mobile

Bài 4

Lập trình Android - User Interfaces

Building UI

Nội dung

- Android logging system
- Relative layout
- Toggle buttons và Switchs
- Seekbar và Rating bars
- Bài tập

Android logging system

- Hệ thống log của Android cung cấp một cơ chế cho việc thu thập và hiển thị debug output từ hệ thống, emulator và ứng dụng của bạn.
- *LogCat* – một tiện ích cho phép hiển thị và lọc log messages
 - Log messages từ system, emulator và ứng dụng của bạn
 - Sẵn có trong Android Studio và có thể dùng từ command-line

Sử dụng Log Class

- Lớp *Log* định nghĩa một số phương thức tĩnh có chức năng logging
- Phương thức logging:
 - `Log.e(tag, message)` error
 - `Log.w(tag, message)` warning
 - `Log.i(tag, message)` information
 - `Log.d(tag, message)` debug
 - `Log.v(tag, message)` verbose

<http://developer.android.com/reference/android/util/Log.html>

Qui ước Logging

- *tag* xác định nguồn của 1 log message
 - Nghĩa là class hay activity nơi log được gọi xảy ra
- Khai báo hằng **TAG** trong class của bạn

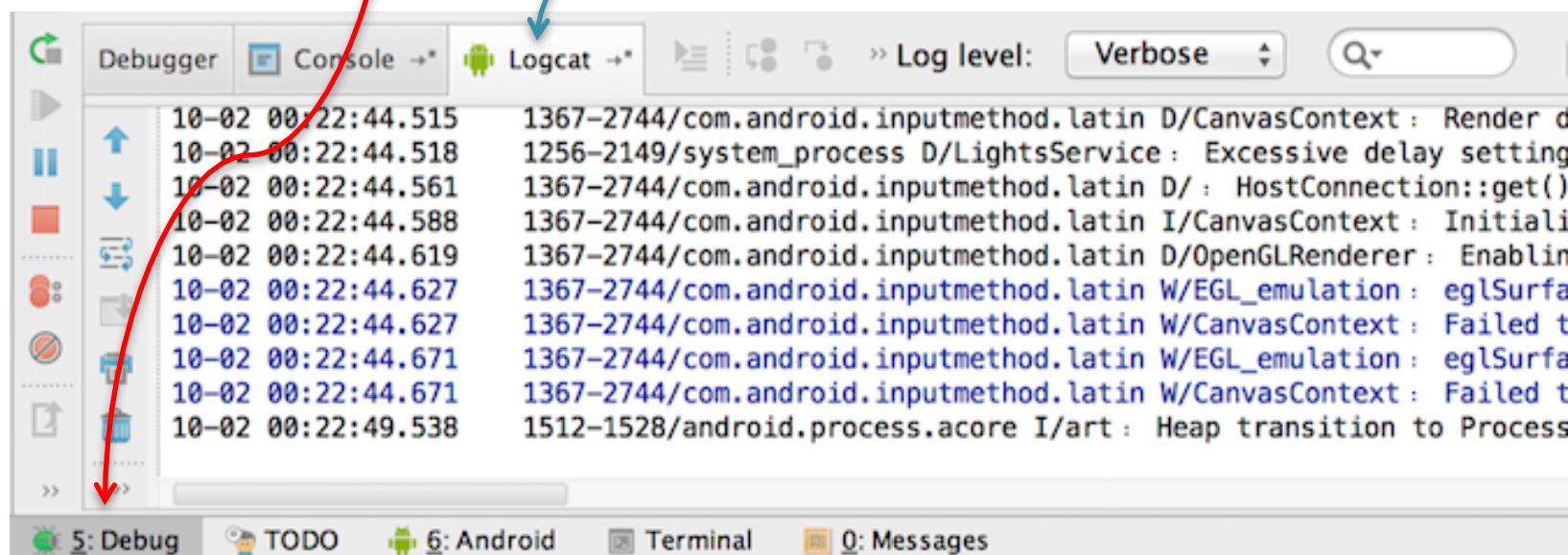
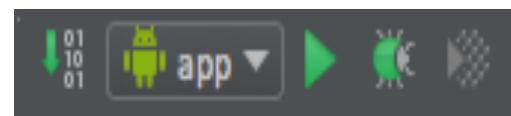
```
private static final String TAG = "MainActivity";
```

- Gọi phương thức log

```
Log.d(TAG, "A log message");
```

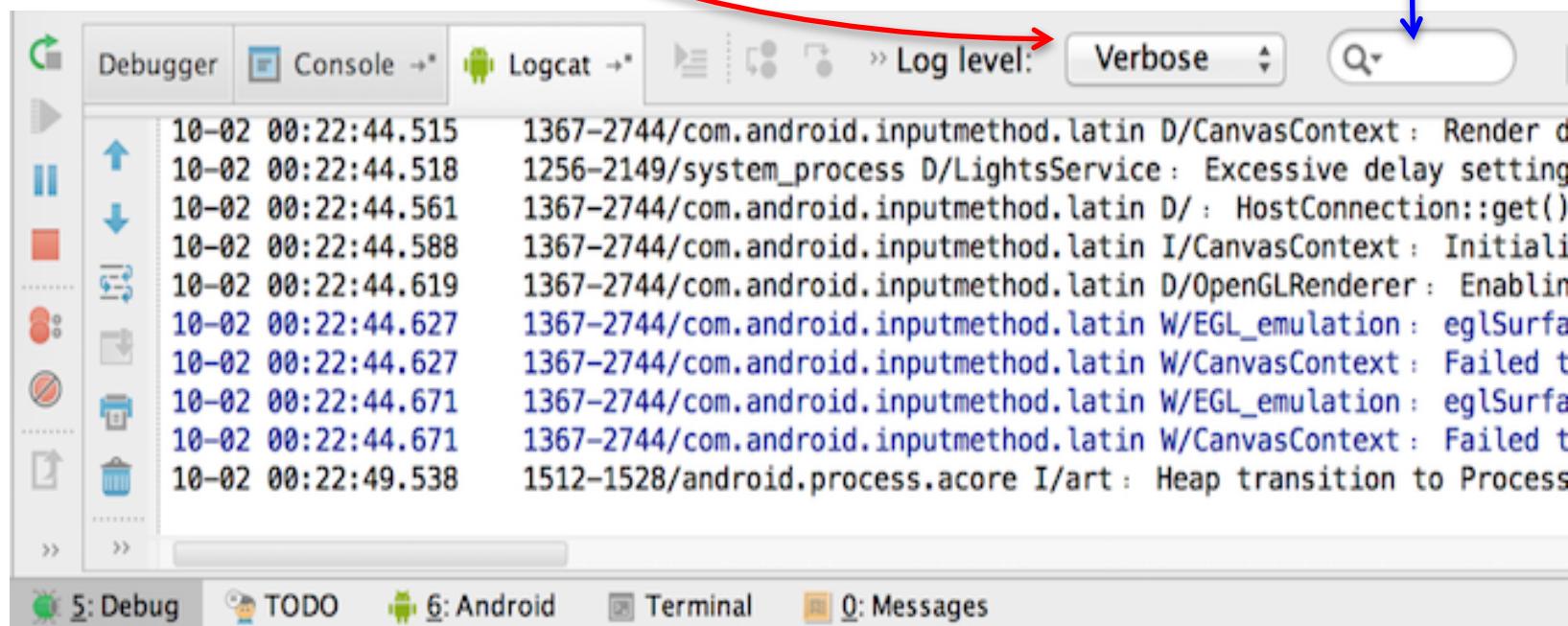
View Log Message (1)

- Chạy ứng dụng của bạn ở chế độ Debug
- Mở “Debug” tabs
- Click vào “LogCat” tab



View Log Message (2)

- Để lọc log message theo log level
 - Chọn level ở *Log Level*
- Để đọc log message với một chuỗi xác định
 - Nhập chuỗi vào trong search box và nhấp **Enter**



Relative Layout

- Là một view group sắp xếp các con của nó dựa trên **quan hệ vị trí** giữa chúng.
- Vị trí của mỗi view có thể được quan hệ tới:
 - anh chị em của nó, hoặc
 - cha nó
- Không cần **nested layouts**
 - Tăng hiệu suất



Ràng buộc trong Relative Layout

- Mặc định, tất cả con được đặt ở góc top-left của cha
- Ràng buộc có thể được khai báo ở bất kì thứ tự nào
 - View1 là trên/dưới của View2
 - View1 là trái/phải của View2
 - View1 được căn chỉnh với View2 ở trái/phải/trên/dưới/ngang, hoặc bắt đầu/kết thúc dựa trên hướng dòng chảy văn bản
 - Căn chỉnh View với cha của nó ở trái/phải/trên/dưới
 - Căn giữa View ở chiều ngang/chiều dọc của cha
- **Không thể có phụ thuộc vòng tròn**

Tham số trong Relative Layout

- Một số tham số cho Relative Layout

`layout_alignParentTop` [`true` | `false`]

Căn chỉnh top của con ở cạnh top của cha

`layout_centerVertical` [`true` | `false`]

Căn giữa con theo chiều dọc so với cha

`layout_below` *view ID*

Đặt con phía dưới anh chị em chỉ định của nó

`layout_toRightOf` *view ID*

Đặt con bên phải anh chị em chỉ định của nó

Ví dụ - The Message App



To: _____

Subject: _____

Message:

CANCEL SEND

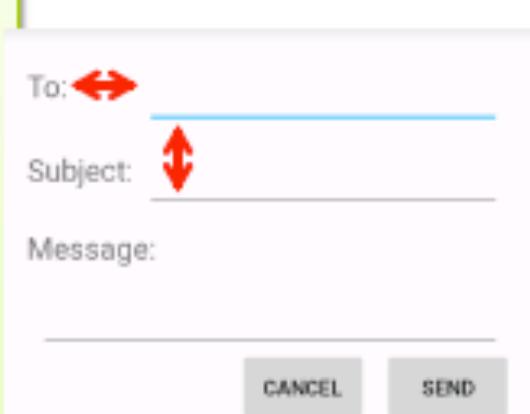
Ví dụ - Hàng đầu tiên

```
<RelativeLayout ...  
    android:layout_width="match_parent"  
    android:layout_height="match_parent" >  
    <TextView ...  
        android:id="@+id/to_label"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:layout_alignBaseline="@+id/to" />  
    <EditText ...  
        android:id="@+id/to"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:layout_alignParentEnd="true"  
        android:layout_alignStart="@+id/subject" />  
    ...  
    </RelativeLayout>
```



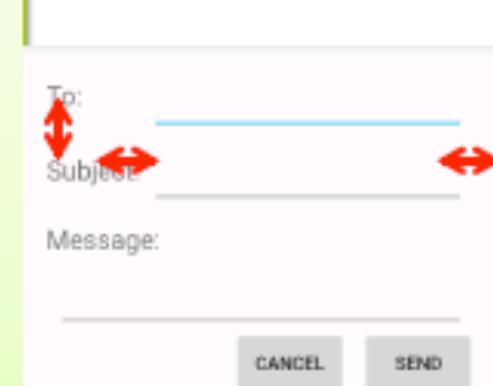
Ví dụ - Hàng đầu tiên (thay thế)

```
<RelativeLayout ...  
    android:layout_width="match_parent"  
    android:layout_height="match_parent" >  
    <TextView ...  
        android:id="@+id/to_label"  
        android:layout_width="wrap_content"  
        android:layout_height="wrap_content"  
        android:layout_alignBaseline="@+id/to" />  
    <EditText ...  
        android:id="@+id/to"  
        android:layout_width="match_parent"  
        android:layout_height="wrap_content"  
        android:layout_alignStart="@+id/subject" />  
    ...  
    </RelativeLayout>
```



Ví dụ - Hàng thứ hai

```
<RelativeLayout ...>
...
<TextView ...
    android:id="@+id/subject_label"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_alignBaseline="@+id/subject" >
<EditText
    android:id="@+id/subject"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_below="@+id/to"
    android:layout_alignParentEnd="true"
    android:layout_toEndOf="@+id/subject_label" >
...
</RelativeLayout>
```



Ví dụ - Hàng thứ ba và bốn

```
<RelativeLayout ...>
    ...
    <TextView
        android:id="@+id/label"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/subject" />
    <EditText ...
        android:id="@+id/entry"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_below="@id/label" />
    ...
</RelativeLayout>
```

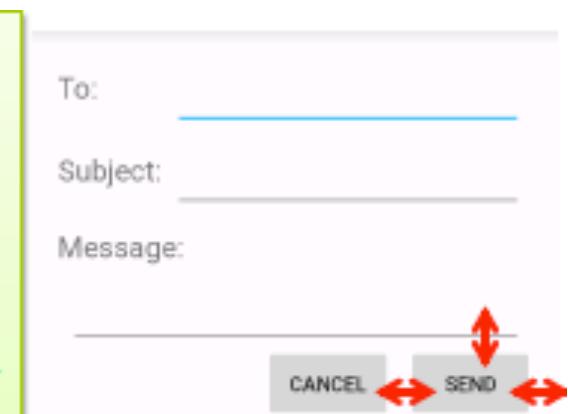


CANCEL

SEND

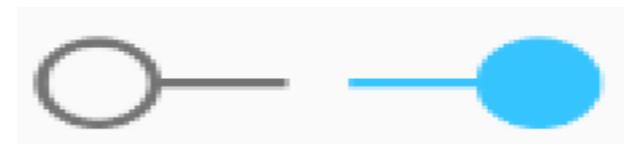
Ví dụ - Hàng thứ năm

```
<RelativeLayout ...>
    ...
    <Button
        android:id="@+id/ok"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_below="@+id/entry"
        android:layout_alignParentEnd="true" />
    <Button
        android:id="@+id/cancel"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_toStartOf="@+id/ok"
        android:layout_alignTop="@+id/ok" />
</RelativeLayout>
```



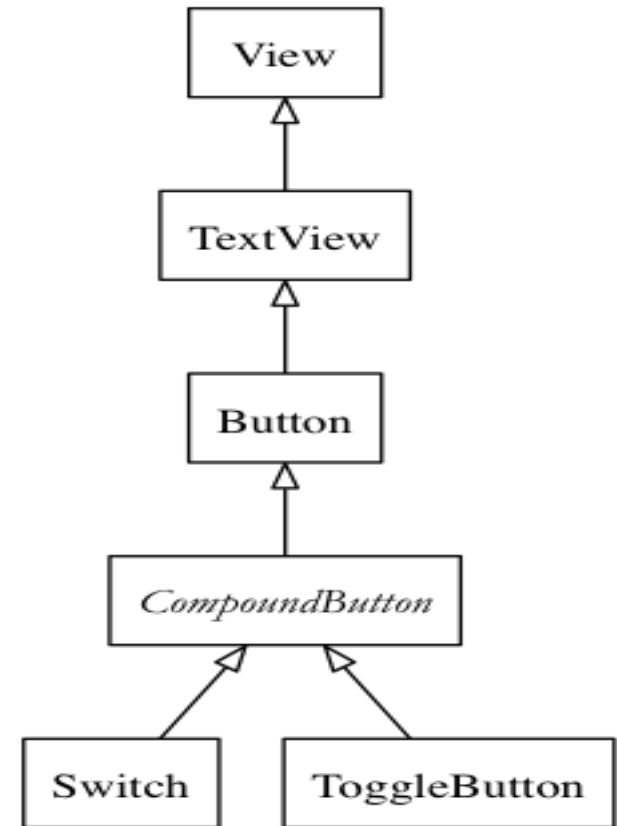
Toggle Buttons và Switches (1)

- Button hai trạng thái
- Toggle Button
 - Một nút on/off
 - Nút + chỉ thị on/off
- Switch (4.0 hoặc mới hơn)
 - 4.0 – 4.4 (API level 14 – 20)
 - Click on, click of
 - 5.0+ (API level 21 hoặc mới hơn)
 - Click on, click of
 - Hành động **lướt/kéo**



Toggle Buttons và Switches (2)

- Class:
`android.widget.ToggleButton`
`android.widget.Switch`
- XML element
`<ToggleButton>`
`<Switch>`
 - Tất cả thuộc tính của *Button* đều có trong *Toggle Button* và *Switch*
 - Thuộc tính luận lý: `checked`



<http://developer.android.com/reference/android/widget/ToggleButton.html>

<http://developer.android.com/reference/android/widget/Switch.html>

Ví dụ - Toggle Buttons và Switches demo

- Toggle Buttons
 - Cơ bản
 - Tùy chỉnh text cho trạng thái on và off
- Switches
 - Cơ bản
 - Tùy chỉnh text cho trạng thái on và off



Ví dụ - phiên bản Android

- Ở phiên bản KitKat (4.x)
 - Switches trông có sự khác biệt

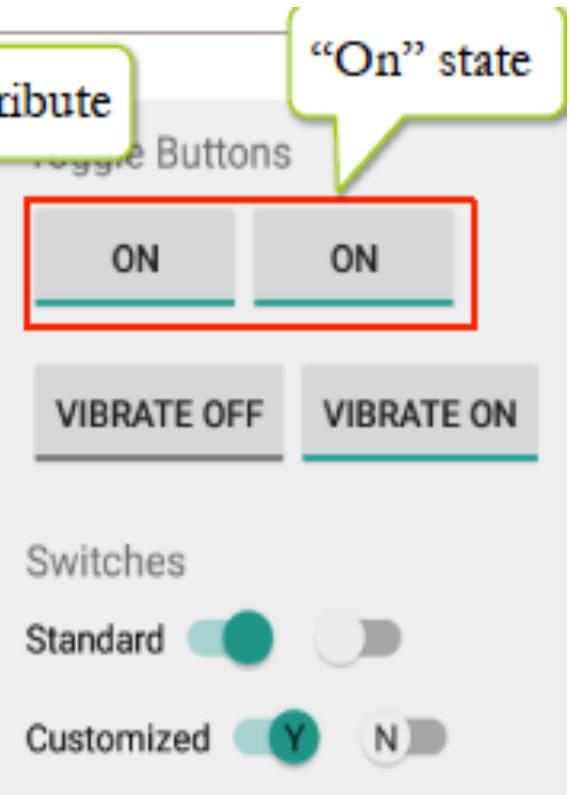


Ví dụ - Định nghĩa Toggle Buttons

```
<LinearLayout ...>
    <ToggleButton
        android:id="@+id/toggle1"
        android:tag="toggle1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="onToggleClicked" />
    <ToggleButton
        android:id="@+id/toggle2"
        android:tag="toggle2"
        android:checked="true"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:onClick="onToggleClicked" />
</LinearLayout>
```

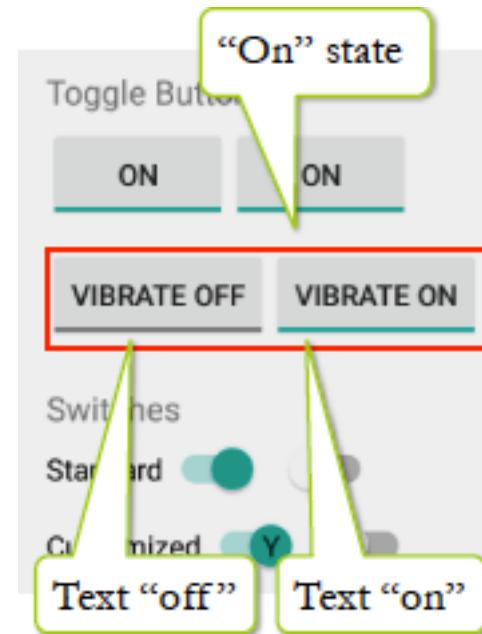
The tag attribute

“On” state



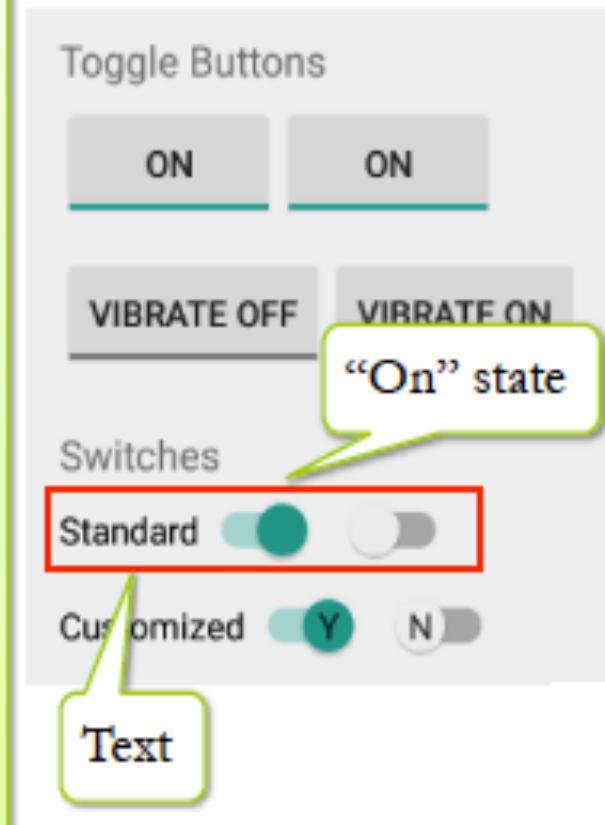
Ví dụ - Tuỳ chỉnh Text của Toggle Buttons

```
<ToggleButton
    android:id="@+id/toggle3"
    android:tag="toggle3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textOn="Vibrate on"
    android:textOff="Vibrate off"
    android:onClick="onToggleClicked" />
<ToggleButton
    android:id="@+id/toggle4"
    android:tag="toggle4"
    android:checked="true"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:textOn="Vibrate on"
    android:textOff="Vibrate off"
    android:onClick="onToggleClicked" />
```



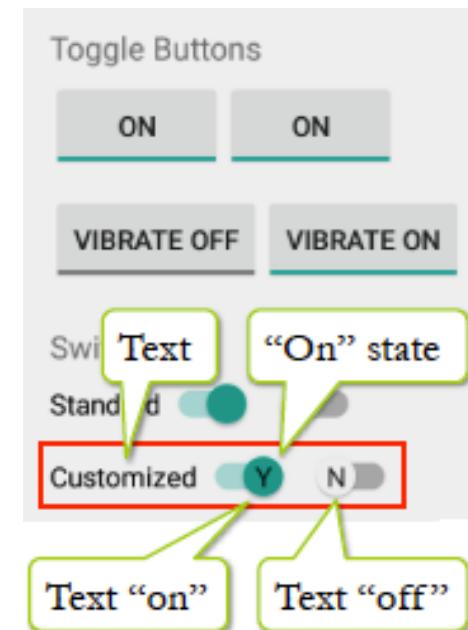
Ví dụ - Định nghĩa Switches

```
<Switch  
    android:id="@+id/switch1"  
    android:tag="switch1"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_marginBottom="10dp"  
    android:text="Standard"  
    android:onClick="onSwitchClicked" />  
  
<Switch  
    android:id="@+id/switch2"  
    android:tag="switch2"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_marginLeft="10dp"  
    android:checked="true"  
    android:onClick="onSwitchClicked" />
```



Ví dụ - Tuỳ chỉnh Text của Switches

```
<Switch
    android:id="@+id/switch3"
    android:tag="switch3"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="Customized"
    android:textOn="Y"
    android:textOff="N"
    android:onClick="onSwitchClicked" />
<Switch
    android:id="@+id/switch4"
    android:tag="switch4"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:paddingLeft="10dp"
    android:checked="true"
    android:textOn="Y"
    android:textOff="N"
    android:onClick="onSwitchClicked" />
```



Ví dụ - Hành động onClick

The view that triggered the event

Use the tag attribute

```
public void onToggleClicked(View view) {  
    ToggleButton toggleButton = (ToggleButton) view;  
    Log.d(TAG, "onToggleClicked() " + toggleButton.getTag() +  
        " " + (toggleButton.isChecked() ? "on" : "off"));  
}  
  
public void onSwitchClicked(View view) {  
    Switch sw = (Switch) view;  
    Log.d(TAG, "onSwitchClicked() " + sw.getTag() +  
        " " + (sw.isChecked() ? "on" : "off"));  
}
```

The state of the *ToggleButton*

Ví dụ - Hiện thực Action Listener

```
public class MainActivity extends Activity
    implements CompoundButton.OnCheckedChangeListener {
    ...
    @Override
    public void onCheckedChanged(CompoundButton button,
                                boolean isChecked) {
        Toast.makeText(this, button.getTag() + " is " +
        (isChecked ? "on" : "off"),
        Toast.LENGTH_SHORT).show();
    }
    ...
}
```

The listener interface

Use the tag attribute

The *Button* that triggered the event and its state

Ví dụ - Set Action Listener

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.activity_main);  
    ...  
    ToggleButton toggleButton1 =  
        (ToggleButton) findViewById(R.id.toggle1);  
    toggleButton1.setOnCheckedChangeListener(this);  
    ...  
    Switch switch1 =  
        (Switch) findViewById(R.id.switch1);  
    switch1.setOnCheckedChangeListener(this);  
    ...  
}
```

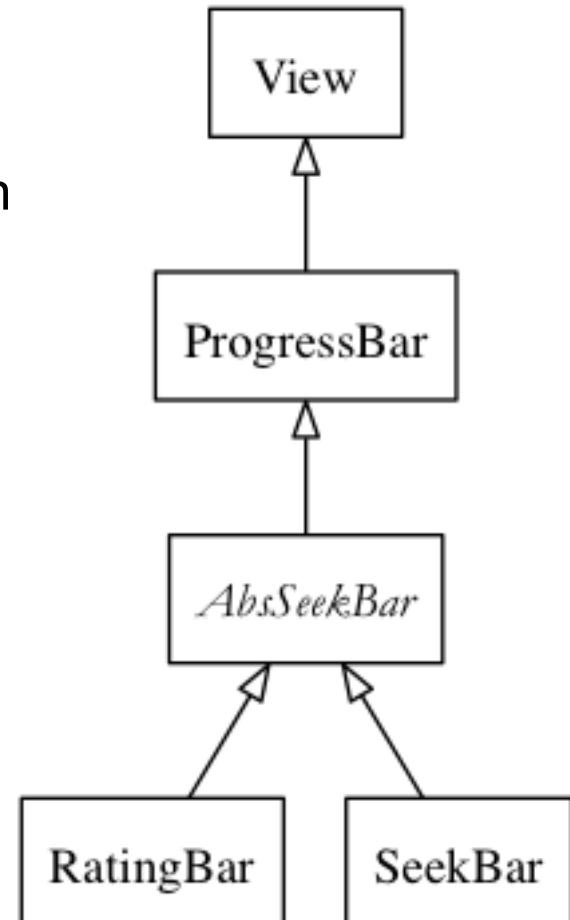
Seek Bars and Rating Bars

- *Seek Bar*

- Thanh trượt tương tác cho lựa chọn 1 giá trị từ một khoảng liên tục hoặc rời rạc các giá trị bằng dùng ngón tay cái di chuyển thanh trượt

- *Rating Bar*

- Hiển thị một đánh giá bằng các hình ngôi sao
- Có thể set rating bằng cách chạm hoặc kéo



Seek Bars

- Class
`android.widgetSeekBar`
- XML element
`<SeekBar>`
- Thuộc tính
 - `max` – giá trị lớn nhất (nhỏ nhất luôn là 0)
 - `progress` – giá trị hiện tại, giữa 0 và max

<http://developer.android.com/reference/android/widget/SeekBar.html>

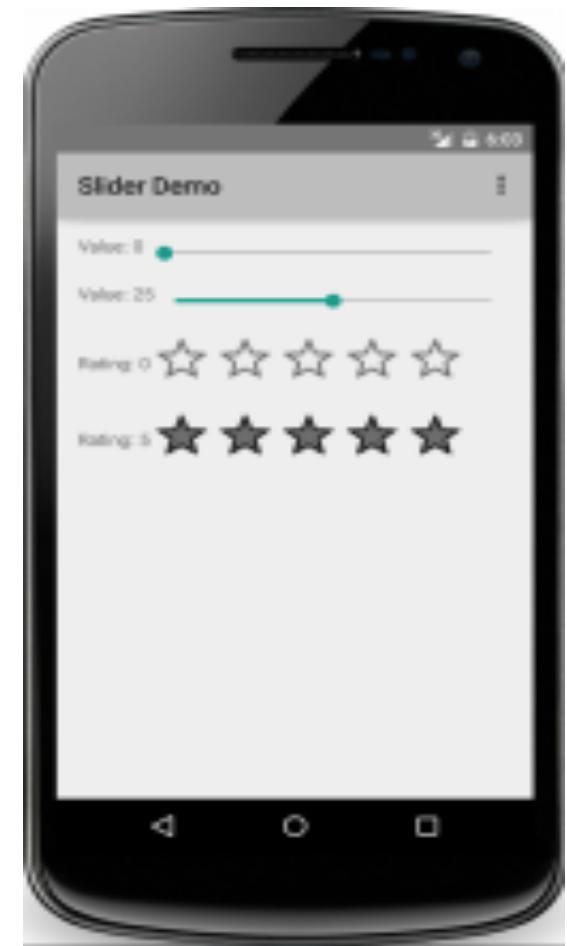
Rating Bars

- Class
`android.widget.RatingBar`
- XML element
`<RatingBar>`
- Thuộc tính
 - `numStars` – số ngôi sao
 - `rating` – giá trị rating hiện tại, giữa 0 và `numStars`
 - `stepSize` – kích thước bước nhảy của rating

<http://developer.android.com/reference/android/widget/RatingBar.html>

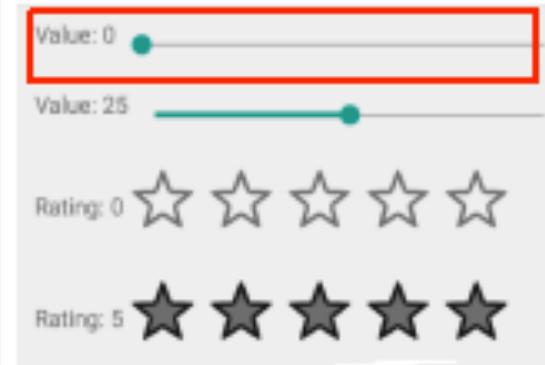
Ví dụ - Slider Demo App

- App chỉ có 1 screen
- *Seek Bars*
 - Thiết lập mặc định
 - Thiết lập tùy chỉnh
- *Rating Bars*
 - Thiết lập mặc định
 - Thiết lập tùy chỉnh



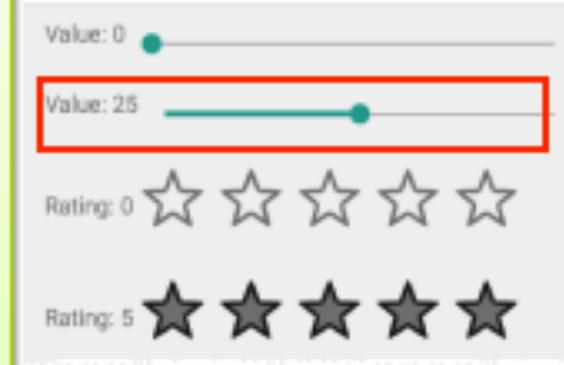
Định nghĩa Seek Bar – Thiết lập mặc định

```
<LinearLayout ...>
    <TextView
        android:id="@+id/value1"
        android:text="Value: 0"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
    <SeekBar
        android:id="@+id/seekbar1"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"/>
</LinearLayout>
```



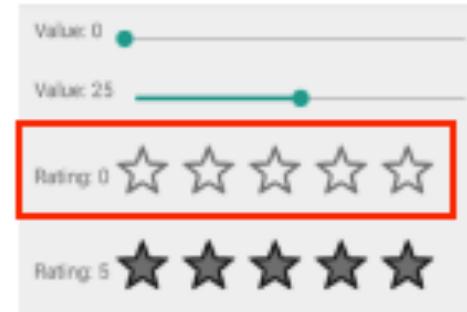
Định nghĩa Seek Bar – Thiết lập tùy chỉnh

```
<LinearLayout ...>
    <TextView
        android:id="@+id/value2"
        android:text="Value: 25"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content" />
    <SeekBar
        android:id="@+id/seekbar2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:max="50"
        android:progress="25" />
</LinearLayout>
```



Định nghĩa Rating Bar – Thiết lập mặc định

```
<LinearLayout ...>
    <TextView
        android:id="@+id/value3"
        android:text="Rating: 0"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"/>
    <RatingBar
        android:id="@+id/ratingkbar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:numStars="5" />
</LinearLayout>
```



Định nghĩa Rating Bar – Thiết lập tùy chỉnh

```
<LinearLayout ...>
    <TextView
        android:id="@+id/value3"
        android:text="Rating: 0"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_gravity="center_vertical"/>
    <RatingBar
        android:id="@+id/ratingkbar1"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:numStars="5" />
</LinearLayout>
```



Hiện thực Seek Bar Listener (1)

```
protected void onCreate(Bundle savedInstanceState) {  
    ...  
    final TextView value1 = (TextView) findViewById(R.id.value1);  
    SeekBar seekBar1 = (SeekBar) findViewById(R.id.seekbar1);  
    seekBar1.setOnSeekBarChangeListener(  
        new SeekBar.OnSeekBarChangeListener() {  
            public void onProgressChanged(SeekBar seekBar,  
                int i, boolean b) {  
                Log.d(TAG, "onProgressChanged");  
                value1.setText("Value: " + i);  
            }  
            public void onStartTrackingTouch(SeekBar seekBar) { }  
            public void onStopTrackingTouch(SeekBar seekBar) { }  
        });  
    ...  
}
```

The listener

Hiện thực Seek Bar Listener (2)

```
protected void onCreate(Bundle savedInstanceState) {  
    ...  
    final TextView value2 = (TextView) findViewById(R.id.value2);  
    SeekBar seekBar2 = (SeekBar) findViewById(R.id.seekbar2);  
    seekBar2.setOnSeekBarChangeListener(  
        new SeekBar.OnSeekBarChangeListener() {  
            public void onProgressChanged(SeekBar seekBar,  
                int i, boolean b) {  
                Log.d(TAG, "onProgressChanged");  
                value2.setText("Value: " + i);  
            }  
            public void onStartTrackingTouch(SeekBar seekBar) { }  
            public void onStopTrackingTouch(SeekBar seekBar) { }  
        });  
    ...  
}
```

Hiện thực Rating Bar Listener (1)

```
protected void onCreate(Bundle savedInstanceState) {  
    ...  
    final TextView value3 = (TextView) findViewById(R.id.value3);  
    RatingBar ratingBar1 = (RatingBar) findViewById(R.id.ratingkbar1);  
    ratingBar1.setOnRatingBarChangeListener(  
        new RatingBar.OnRatingBarChangeListener() {  
            public void onRatingChanged(RatingBar ratingBar,  
                float v, boolean b) {  
                Log.d(TAG, "onRatingChanged");  
                value3.setText("Rating: " + v);  
            }  
        });  
    ...  
}
```

Hiện thực Rating Bar Listener (2)

```
protected void onCreate(Bundle savedInstanceState) {  
    ...  
    final TextView value4 = (TextView) findViewById(R.id.value4);  
    RatingBar ratingBar2 = (RatingBar) findViewById(R.id.ratingkbar2);  
    ratingBar2.setOnRatingBarChangeListener(  
        new RatingBar.OnRatingBarChangeListener() {  
            public void onRatingChanged(RatingBar ratingBar,  
                float v, boolean b) {  
                Log.d(TAG, "onRatingChanged");  
                value4.setText("Rating: " + v);  
            }  
        });  
    ...  
}
```

Bài tập

