**Скрыть закрыть панель через Animator**

В [примере](http://developer.alexanderklimov.ru/android/animation/panel.php) использовалась традиционная анимация. Реализуем анимацию сворачивая/разворачивания блока через новый класс **android.animation.Animator** (API 11). Кода будет меньше.

Создадим разметку:

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

xmlns:tools="http://schemas.android.com/tools"

android:id="@+id/container"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical" >

<LinearLayout

android:id="@+id/header"

android:layout\_width="match\_parent"

android:layout\_height="64dp"

android:layout\_marginLeft="16dp"

android:layout\_marginRight="16dp"

android:layout\_marginTop="16dp"

android:background="#FCF"

android:orientation="horizontal" >

<TextView

android:id="@+id/color"

android:layout\_width="48dp"

android:layout\_height="48dp"

android:layout\_marginBottom="8dp"

android:layout\_marginLeft="8dp"

android:layout\_marginTop="8dp"

android:background="@drawable/rectangle"

android:fontFamily="sans-serif-light"

android:gravity="center\_vertical"

android:text=""

android:textAlignment="center"

android:textStyle="bold" />

<TextView

android:id="@+id/clickme"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="4dp"

android:fontFamily="sans-serif-light"

android:gravity="center\_vertical"

android:text="Тронь меня"

android:textStyle="bold" />

</LinearLayout>

<LinearLayout

android:id="@+id/expandable"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:layout\_marginLeft="16dp"

android:layout\_marginRight="16dp"

android:background="#FFF"

android:orientation="vertical"

android:paddingLeft="4dp" >

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Имя: Дорофей" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Семейное положение: Холост" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Слуга: Дмитрий Медведев" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Национальность: Невский маскарадный" />

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Характеристика: Склонен к побегу" />

</LinearLayout>

</LinearLayout>

Переходи к коду активности:

package ru.alexanderklimov.test;

import android.os.Bundle;

import android.animation.Animator;

import android.animation.ValueAnimator;

import android.app.Activity;

import android.view.View;

import android.view.ViewGroup;

import android.view.ViewTreeObserver;

import android.widget.LinearLayout;

public class MainActivity extends Activity {

LinearLayout mLinearLayout;

LinearLayout mLinearLayoutHeader;

ValueAnimator mAnimator;

@Override

public void onCreate(Bundle savedInstanceState) {

super.onCreate(savedInstanceState);

setContentView(R.layout.activity\_main);

setTitle("Карточка");

mLinearLayout = (LinearLayout) findViewById(R.id.expandable);

// mLinearLayout.setVisibility(View.GONE);

mLinearLayoutHeader = (LinearLayout) findViewById(R.id.header);

// Add onPreDrawListener

mLinearLayout.getViewTreeObserver().addOnPreDrawListener(

new ViewTreeObserver.OnPreDrawListener() {

@Override

public boolean onPreDraw() {

mLinearLayout.getViewTreeObserver()

.removeOnPreDrawListener(this);

mLinearLayout.setVisibility(View.GONE);

final int widthSpec = View.MeasureSpec.makeMeasureSpec(

0, View.MeasureSpec.UNSPECIFIED);

final int heightSpec = View.MeasureSpec

.makeMeasureSpec(0,

View.MeasureSpec.UNSPECIFIED);

mLinearLayout.measure(widthSpec, heightSpec);

mAnimator = slideAnimator(0,

mLinearLayout.getMeasuredHeight());

return true;

}

});

mLinearLayoutHeader.setOnClickListener(new View.OnClickListener() {

@Override

public void onClick(View v) {

if (mLinearLayout.getVisibility() == View.GONE) {

expand();

} else {

collapse();

}

}

});

}

private void expand() {

// set Visible

mLinearLayout.setVisibility(View.VISIBLE);

/\*

\* Remove and used in preDrawListener final int widthSpec =

\* View.MeasureSpec.makeMeasureSpec(0, View.MeasureSpec.UNSPECIFIED);

\* final int heightSpec = View.MeasureSpec.makeMeasureSpec(0,

\* View.MeasureSpec.UNSPECIFIED); mLinearLayout.measure(widthSpec,

\* heightSpec);

\*

\* mAnimator = slideAnimator(0, mLinearLayout.getMeasuredHeight());

\*/

mAnimator.start();

}

private void collapse() {

int finalHeight = mLinearLayout.getHeight();

ValueAnimator mAnimator = slideAnimator(finalHeight, 0);

mAnimator.addListener(new Animator.AnimatorListener() {

@Override

public void onAnimationEnd(Animator animator) {

// Height=0, but it set visibility to GONE

mLinearLayout.setVisibility(View.GONE);

}

@Override

public void onAnimationStart(Animator animator) {

}

@Override

public void onAnimationCancel(Animator animator) {

}

@Override

public void onAnimationRepeat(Animator animator) {

}

});

mAnimator.start();

}

private ValueAnimator slideAnimator(int start, int end) {

ValueAnimator animator = ValueAnimator.ofInt(start, end);

animator.addUpdateListener(new ValueAnimator.AnimatorUpdateListener() {

@Override

public void onAnimationUpdate(ValueAnimator valueAnimator) {

// Update Height

int value = (Integer) valueAnimator.getAnimatedValue();

ViewGroup.LayoutParams layoutParams = mLinearLayout

.getLayoutParams();

layoutParams.height = value;

mLinearLayout.setLayoutParams(layoutParams);

}

});

return animator;

}

}

Вспомогательный файл для отрисовки прямоугольника. Он нужен для красоты по задумке автора и не обязателен.

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

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

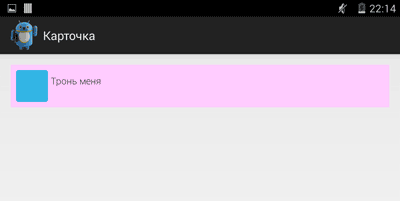
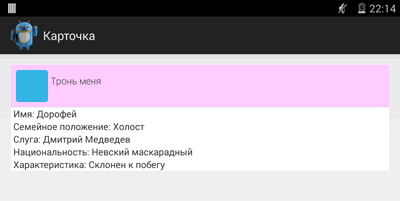
android:shape="rectangle" >

<solid android:color="#33b5e5" />

<corners android:radius="8px" />

</shape>

По мотивам статьи [Expand and collapse animation](http://gmariotti.blogspot.ru/2013/09/expand-and-collapse-animation.html)

**Р**