

6 апреля в 03:17

Новый SwipeRefreshLayout из библиотеки Android support — «потяните чтобы обновить» tutorial

 Разработка под Android*

Тихо и незаметно вместе с обновлением библиотеки Android support до версии 19.1 в нее [прокрался](#) новый компонент, который часто бывает нужным и полезным в быту. Сей совсем крохотный росчерк я посвящаю ему, [SwipeRefreshLayout](#).

Он добавляет функционал, который один мой коллега называет «резинка от трусов», а остальные знают этот UI-паттерн под названием «потяни, чтобы обновить». Он бывает нужен, когда есть какой-то контент, который пользователь хочет часто обновлять, и может это сделать, просто потянув контент жестом вниз, а потом отпустив.

Очень похоже, что компонент был создан для [GoogleNow](#), а потом перекочевал в support lib (очень похожа анимация).

Алгоритм простой.

1. Компонент «оборачивается» вокруг View или Layout, для которого нужно сделать обновление и может содержать только **одного** потомка (похоже на ScrollView).
2. Устанавливается [OnRefreshListener](#) с одним единственным методом [onRefresh\(\)](#). Как на него реагировать — решаем сами.
3. Когда нужно показать, что идет обновление (и запустить анимацию), вызываем метод [setRefreshing\(boolean\)](#) и передаем ему *true*. Наиболее логичным кажется вызов метода внутри [onRefresh\(\)](#).
4. Когда обновление закончено, вызываем метод [setRefreshing\(boolean\)](#) и передаем ему *false*.



Давайте попробуем набросать пример использования.

Наш XML (если нужно, можно и динамически создать компонент и добавить в него потомка, компонент наследуется от [ViewGroup](#)).

```
<android.support.v4.widget.SwipeRefreshLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools"
    android:id="@+id/refresh"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context="com.shppandroid1.app.MainActivity">
    <ListView
        android:id="@android:id/list"
        android:layout_width="match_parent"
        android:layout_height="match_parent" />
</android.support.v4.widget.SwipeRefreshLayout>
```

Теперь наша Activity, а именно метод onCreate:

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    mSwipeRefreshLayout = (SwipeRefreshLayout) findViewById(R.id.refresh);
    mSwipeRefreshLayout.setOnRefreshListener(this);
    // делаем повеселее
    mSwipeRefreshLayout.setColorScheme(R.color.blue, R.color.green, R.color.yellow, R.color.red);

    ListView listView = (ListView) findViewById(android.R.id.list);
    // классический адаптер
    listView.setAdapter(new ArrayAdapter<>(this, android.R.layout.simple_list_item_1, ITEMS));
}
```

В данном случае я сделал нашу Activity реализующей интерфейс OnRefreshListener

```
public class MainActivity extends ActionBarActivity implements SwipeRefreshLayout.OnRefreshListener
```

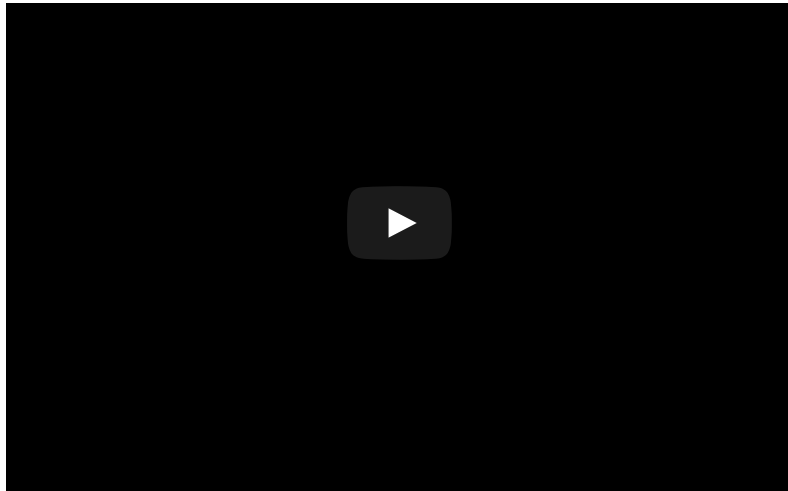
Обратите внимание на метод [setColorScheme](#), он как раз и раскрашивает в разные цвета прогресс во время обновления. В видео ниже видно, о чем я говорю. Первый цвет из 4-х — цвет прогресса во время оттягивания контента вниз перед обновлением. Остальные — сменяющиеся цвета прогресса.

А теперь сам onRefresh();

```
@Override
public void onRefresh() {
    // говорим о том, что собираемся начать
    Toast.makeText(this, R.string.refresh_started, Toast.LENGTH_SHORT).show();
    // начинаем показывать прогресс
    mSwipeRefreshLayout.setRefreshing(true);
}
```

```
// ждем 3 секунды и прячем прогресс
mSwipeRefreshLayout.postDelayed(new Runnable() {
    @Override
    public void run() {
        mSwipeRefreshLayout.setRefreshing(false);
        // говорим о том, что собираемся закончить
        Toast.makeText(MainActivity.this, R.string.refresh_finished, Toast.LENGTH_SHORT).show();
    }
}, 3000);
}
```

Демонстрация:



UPD:

Для желающих добавил саму [APK](#)

Ссылка на APK в виде кода, чтобы было жить легче:



android, pulltorefresh, view, support library, виды вышивки

+37 9549 135 rimmer 9,5

Похожие публикации

Android L, Nexus 5, Google Search и все-все-все вчера в 18:23

Android NDK: работа с OpenGL ES 8 сентября в 10:57

Новый Evernote 6 для Android: редизайн, веб-клиппер и многое другое 5 сентября в 12:51

Google и Bungie создали интерактивный мир игры Destiny с использованием Street View 3 сентября в 12:01

Новинки в Support-Library 29 июля в 12:20

ActionBar на Android 2.1+ с помощью Support Library. Часть 3 — Полезные функции 14 января в 19:56

Соединение SlidingMenu с Support Library и обход возможных проблем 17 сентября 2013 в 12:47

ActionBar на Android 2.1+ с помощью Support Library. Часть 2 — Навигация 20 августа 2013 в 16:02

ActionBar на Android 2.1+ с помощью SupportLibrary 12 августа 2013 в 17:03

ActionBar на Android 2.1 с помощью Android Support Library 26 июля 2013 в 20:30

Комментарии (7) отслеживать новые: ☐ в почте ☐ в трекере

AntonRychagov, 6 апреля 2014 в 11:13 #

0

Очень полезно, спасибо за статью!

zoo, 6 апреля 2014 в 13:24 #

+1

Давно пора, а то приходилось пользоваться вот такими [велосипедами](#)

-  **rimmer**, 6 апреля 2014 в 13:59 # ☆ ↻ ↑

+5 ↑ ↓

К сожалению, должен заметить, что библиотека [ActionBar-PullToRefresh](#) пока более совершенна, так как может еще показывать сообщения в ActionBar на разные события компонента и более кастомизируема.
-  **jaleel**, 7 апреля 2014 в 03:35 # ☆ ↻ ↑

0 ↑ ↓

И у которой есть например такой вот не очень приятный баг, который подпортил мне нервы :(github.com/chrisbanes/ActionBar-PullToRefresh/issues/163
-  **Shultc**, 6 апреля 2014 в 13:42 (комментарий был изменён) # ☆

+5 ↑ ↓

Это не всегда интуитивно, но довольно удобно.

Когда у вас действительно есть контент, который добавляется сверху вниз как в стек, то вы этим действием как бы заглядываете за последний элемент. Ну, или оттягиваете чтобы дать место новому элементу. Как в gmail, к примеру.

Однако, обновлять таким образом вещи, вроде уведомлений в facebook мне кажется очень нелогичным.
-  **BRATiK**, 9 апреля 2014 в 22:06 # ☆

0 ↑ ↓

Спасибо, наконец таки избавлюсь от ActionBar-PullToRefresh.
-  **sapl**, 20 мая 2014 в 08:41 # ☆

0 ↑ ↓

Как то можно к ней прикрутить прогресс как в gmail и др (без этой радуги)?

Brainstorage

Эксперт по нагрузочному тестированию

Системный администратор Unix

Младший инженер доступности сервисов (Site Reliability Operatio...

Site Reliability Engineer

Технический менеджер поисковых проектов

Frontend Developer

Менеджер по продажам IT-решений

Java Architect в Департамент систем автоматизации ИТ

Java Developer в Департамент систем автоматизации ИТ

Senior Java Developer в Департамент систем автоматизации ИТ

все вакансии

ФРИЛАНСИМ

Нужен front-end и back-end нового сайта компании

Разработка User eXperience, UI

Требуется натяжка шаблона магазина oscommerce

Отрисовка баннера 760 x 90

Ряд задач по JS

Разработка модуля для Magento

Разработка сложной системы под wordpress (много backend`а)

Дорисовать страницу чекаута в магазине

Этикетка для банки черной икры

UI для Leap Motion

все заказы