

一、下拉刷新介绍

大多数安卓应用都需要刷新数据，而在谷歌推荐的安卓设计标准 *Material Design* 中推荐的刷新方式为下拉刷新，本文介绍在天气项目中实现下拉刷新的方法。

二、下拉刷新的实现效果

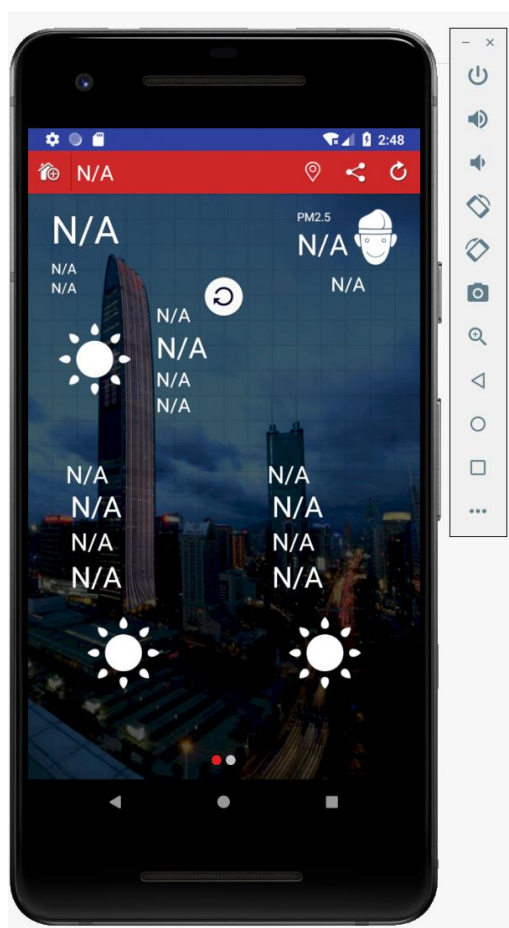


图 1 下拉刷新中



图 2 下拉刷新完成

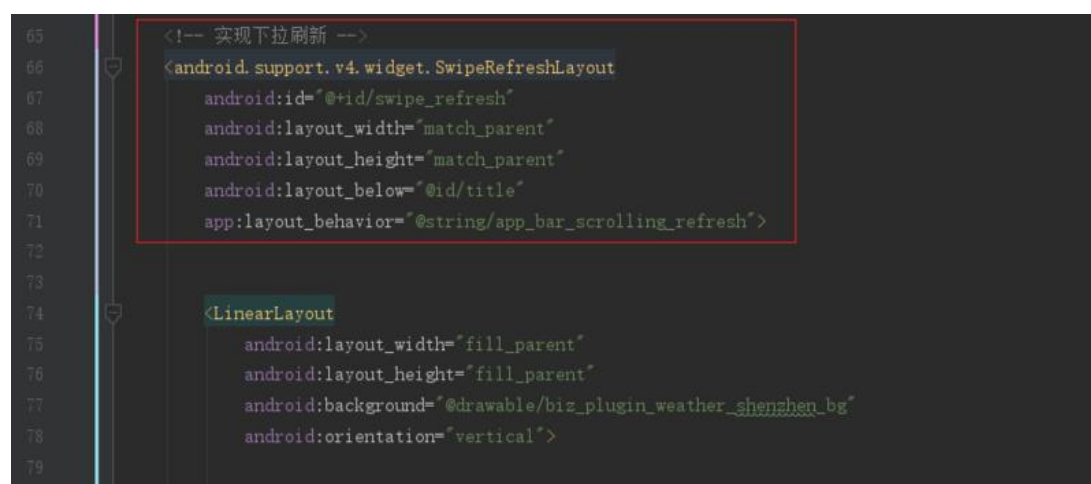
三、SwipeRefreshLayout 控件介绍

SwipeRefreshLayout 控件只接受一个子组件：即需要刷新的那个组件。它使用一个侦听机制来通知拥有该组件的监听器有刷新事件发生，我们的 Activity 必须实现通知的接口。该 Activity 负责处理事件刷新和刷新相应的视图。一旦监听者接收到该事件，就决定了刷新过程中应处理的地方。如果要展示一个“刷新动画”，它必须调用 `setRefreshing (true)`，否则取消动画就调用 `setRefreshing (false)`。

四、具体代码实现

1.修改 weather_info.xml 文件

为了实现 SwipeRefreshLayout 控件，首先我们需要在 weather_info.xml 文件中，修改标题栏以下的 LinearLayout 控件为 SwipeRefreshLayout 控件的子控件：



```
65 <!-- 实现下拉刷新 -->
66 <android.support.v4.widget.SwipeRefreshLayout
67     android:id="@+id/swipe_refresh"
68     android:layout_width="match_parent"
69     android:layout_height="match_parent"
70     android:layout_below="@id/title"
71     app:layout_behavior="@string/app_bar_scrolling_refresh">
72
73
74     <LinearLayout
75         android:layout_width="fill_parent"
76         android:layout_height="fill_parent"
77         android:background="@drawable/biz_plugin_weather_shenzhen_bg"
78         android:orientation="vertical">
```

图 3 在 LinearLayout 的开头修改

```

275         android:src="@drawable/point_enable"/>
276         <ImageView
277             android:layout_width="10dip"
278             android:layout_height="wrap_content"
279             android:layout_toRightOf="@id/page1Img"
280             android:layout_marginLeft="5dip"
281             android:id="@+id/page2Img"
282             android:src="@drawable/point_disable"/>
283     </RelativeLayout>
284 </RelativeLayout>
285 <!-- 六日天气信息 结束-->
286
287 <!-- 广告等信息 -->
288
289 </LinearLayout>
290
291 </android.support.v4.widget.SwipeRefreshLayout>
292 <!-- 实现下拉刷新 -->
293 </RelativeLayout>

```

图 4 在 LinearLayout 的结尾修改

修改之后，标题栏以下的线性布局就成为了 SwipeRefreshLayout 控件的子控件。

2.修改 MainActivity.java 文件

```

10     import android.support.v4.view.ViewPager;
11     import android.support.v4.widget.SwipeRefreshLayout;
12     import android.util.Log;
13     import android.view.LayoutInflater;

```

图 5 在开头引入 SwipeRefreshLayout 控件

```

59     private ImageView page1Img, page2Img;
60
61     private SwipeRefreshLayout Refresh;
62
63     private String updatedCityCode;

```

图 6 定义 SwipeRefreshLayout 类的变量

```

83 //初始化界面
84 void initView() [...]
163
164 //实现下拉刷新的线程
165 @SuppressWarnings("ResourceAsColor")
166 void PulltoRefresh() {
167     new Thread((Runnable) () -> {
168         //根据ID找到控件
169         Refresh = (SwipeRefreshLayout) findViewById(R.id.swipe_refresh);
170         //设置颜色
171         Refresh.setColorSchemeColors(R.color.colorPrimaryDark);
172         //创建监听器
173         Refresh.setOnRefreshListener(() -> {
174             //查询之前展示动画
175             Refresh.setRefreshing(true);
176             queryWeatherCode(updatedCityCode);
177             //查询之后关闭动画
178             Refresh.setRefreshing(false);
179         });
180     }).start();
181 }
182
183 //根据得到的数据更新界面
184 void updateTodayWeather(TodayWeather todayWeather) [...]
185
186
187
188
189
190
191

```

图 7 单独设置一个线程，实现下拉刷新的函数

```

577 mcitySelect = (ImageView) findViewById(R.id.title_city_manager);
578 mcitySelect.setOnClickListener(this);
579
580 initView();
581
582 PulltoRefresh();
583 }

```

图 8 在 onCreate 方法中调用 PulltoRefresh 方法

1801220002

陈国强