

使用Intent在活动之间穿梭

2018年3月29日 20:07

只有一个活动太简单枯燥没面子？下面尝试一下多活动怎么样。

使用显示Intent

仍然还是右击 `com.example.activitytest` 包→New→Activity→Empty Activity，会弹出一个创建活动的对话框，我们这次将活动命名为 `SecondActivity`，并勾选 `Generate Layout File`，给布局文件起名为 `second_layout`，但不要勾选 `Launcher Activity` 选项，如图 2.14 所示。

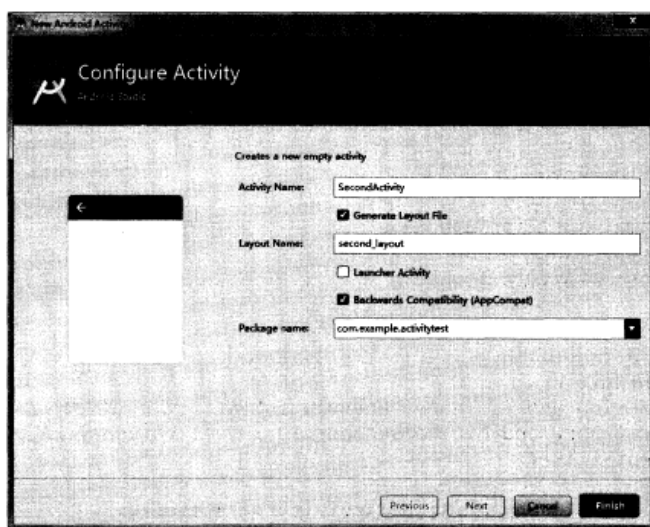


图 2.14 创建 SecondActivity

点击 `Finish` 完成创建，Android Studio 会为我们自动生成 `SecondActivity.java` 和 `second_layout.xml` 这两个文件。不过自动生成的布局代码目前对你来说可能有些复杂，这里我们仍然还是使用最熟悉的 `LinearLayout`，编辑 `second_layout.xml`，将里面的代码替换成如下内容：

```
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent">

    <Button
        android:id="@+id/button_2"
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="Button 2"
    />

</LinearLayout>
```

我们还是定义了一个按钮，按钮上显示 `Button 2`。

然后 `SecondActivity` 中的代码已经自动生成了一部分，我们保持默认不变就好，如下所示：

```
public class SecondActivity extends AppCompatActivity {

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

}
```

Intent有两种**显式Intent**和**隐式Intent**

显式Intent

Intent的构造函数有多个重载，其中一个是Intent (Context packageContext, class<?> cls)。这个构造函数接收两个参数，第一个参数Context要求提供一个启动活动的上下文，第二个参数Class则指定想要启动的活动。然后Activity类提供了一个startActivity的方法，这个方法是专门用于启动活动的，它接收一个intent参数，这里我们将构建好的intent传进去就可以启动活动了。

修改第一个活动中的Button的点击事件，代码：

```
button1.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View view) {  
        Toast.makeText(MainActivity.this,"Finish",Toast.LENGTH_SHORT).show();  
        finish();  
        Intent intent = new Intent( packageContext: MainActivity.this,SecondActivity.class);  
        startActivity(intent);  
    }  
});
```

我们的代码的意图：

intent构造时传入当前活动作为上下文，再将SecondActivity作为目标活动。这样做就是在当前活动的基础上打开SecondActivity这个活动。然后通过startActivity去执行这个Intent。

这种Intent的意图非常明显，我们称之为显式Intent

使用隐式Intent

相比于显式Intent，隐式Intent较为含蓄，它不明确之指出启动哪一个活动，而是制定了一系列更为抽象的action和category等信息，让系统去分析启动哪一个活动比较合适。

什么是合适的活动呢，简单地说就是可以响应我们这个隐式Intent的活动。目前的SecondActivity可以响应什么样的隐式Intent的呢？额，现在好像什么也响应不了。。。。



通过在<activity>标签下配置<intent-filter>的内容可以指定当前活动能够响应的action和category,打开AndroidManifest.xml添加如下代码:

```
<activity android:name=".SecondActivity" >  
    <intent-filter>  
        <action android:name="com.example.activitytest.ACTION_START" />  
        <category android:name="android.intent.category.DEFAULT" />  
    </intent-filter>  
</activity>
```

在action标签中指明了当前活动可以响应com.example.activitytest.ACTION_START这个action，而category标签还包含了一些附加信息，更精确地指明了当前的活动能响应的Intent中还可能带有的category。只有<action>和<category>中的内容同时能够匹配上Intent中指定的action和category时，这个活动才可以响应Intent。

有关Intent的深入探讨可以参考https://blog.csdn.net/todo_/article/details/50737227

修改第一个活动中的按钮的点击事件，代码：

```

        button1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent("com.example.activitytest.ACTION_START");
                startActivity(intent);
            }
        });

```

category的匹配规则

category是一个字符串，系统预定义了一些category，同时我们也可以在应用中定义自己的category。

category的匹配规则是：Intent可以没有category，但是如果有category，不管是几个category，则这些category必须每个都要与IntentFilter中的某个category能够匹配上，否则匹配失败。如果Intent没有category也能够匹配成功，原因是系统在调用startActivity或者startActivityForResult的时候会默认为Intent加上“android.intent.category.DEFAULT”这个category，所以如果使用隐式Intent则必须在要启动的组件的intent-filter中指定“android.intent.category.DEFAULT”这个category

所以在代码中虽然我们没有写有关category的东西，但系统还是自动帮我们完成了category的配对

现在在主活动的按钮的点击事件中添加粗体代码：

```

        button1.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent("com.example.activitytest.ACTION_START");
                intent.addCategory("com.example.activitytest.MY_CATEGORY");
                startActivity(intent);
            }
        });

```

再次运行程序发现点击Button程序崩了。

在Logcat里看到错误信息：

```

Process: com.example.activitytest, PID: 24027
android.content.ActivityNotFoundException: No Activity found to handle Intent {
  act=com.example.activitytest.ACTION_START cat=[com.example.activitytest.MY_CATEGORY] }

```

图 2.16 错误信息

错误信息中说没有一个活动可以相应我们的Intent，原因是我们新添加了一个category，而SecondActivity的<intent-filter>标签中并没有声明可以响应这一个category，所以就出现了没有活动可以响应该Intent的情况。我们可以在SecondActivity的<intent-filter>中添加对我们刚才添加的category的声明；

```

<activity android:name=".SecondActivity" >
    <intent-filter>
        <action android:name="com.example.activitytest.ACTION_START" />
        <category android:name="android.intent.category.DEFAULT" />
        <category android:name="com.example.activitytest.MY_CATEGORY"/>
    </intent-filter>
</activity>

```

然后就可以响应了。