

intent和intent-filter通信

Intent

包括Component、Action、Category、Data、Type、Extra和Flag这7种属性;

1.1 Component

显示调用:

```
ComponentName comp = new ComponentName(MainActivity.this,SecondActivity.class);
Intent intent = new Intent();
//为intent设置Component
intent.setComponent(comp);
startActivity(intent);
```

一般简化为:

```
Intent intent = new Intent(MainActivity.this,SecondActivity.class);
startActivity(intent);
```

SencondActivity中

```
//获取该Activity对应的Intent的Component属性
ComponentName comp = getIntent().getComponent();
//显示该CompontName对象的包名, 类名
comp.getPackageName();    //SencondActivity的包名
comp.getClassName();     //SencondActivity的类名
```

1.2 Action、Category属性与intent-filter配置 (隐式调用)

Action、Category属性值都是一个普通字符串;

Action: Intent所要完成的一个抽象动作;

Category: 用于为Action增加额外的附加类别信息;

通常Action和Category属性结合使用;

示例Intent启动代码:

```
//创建Intent对象
Intent intent = new Intent();
//为Intent设置Action属性 (属性值就是一个普通字符串)
intent.setAction("I want to go to the SecondActivity");
startActivity(intent);
```

AndroidManifest.xml中<activity.../>的<intent-filter.../>子元素用于配置该Activity所能“响应”的Intent。

<intent-filter.../>包含如下子元素：

- 0~N个<action.../>子元素；
- 0~N个<category.../>子元素；
- 0~1个<data.../>子元素；

【注意】：Intent最多只能有一个Action，但是可以有多个Category；

Intent在未指定Category时，Intent默认启动Category时有一个`android.intent.category.DEFAULT`的属性值；

指定了Category，AndroidManifest中一定包含`android.intent.category.DEFAULT`；

通过Action、Category可以查看系统的Activity

Data、Type属性与intent-filter配置

Data属性：通常用于向Action属性提供操作的数据。Data属性接受一个Uri对象，该Uri对象通常通过如下形式的字符串来表示。

```
content://com.android.contacts/contacts/1
tel:123
```

Uri字符串满足如下格式：

```
scheme://host:port/path
```

Type属性：用于指定该Data属性所指定Uri对应的MIME（Multipurpose Internet Mail Extensions）类型，这种MIME类型可以是任何自定义的MIME类型，只要符合abc/xyz格式的字符串即可；

Data属性与Type属性会相互覆盖，例如：

- 如果为Intent先设置Data属性，后设置Type属性，那么Type属性将会覆盖Data属性。
- 如果为Intent先设置Type属性，后设置Data属性，那么Data属性将会覆盖Type属性。
- 如果希望Intent既有Data属性，也有Type属性，则应该调用Intent的setDataAndType()方法。

在AndroidManifest.xml文件中为组件声明Data、Type属性都是通过<data.../>元素，<data.../>元素格式如下：

```
<data android:mimeType=""
      android:scheme=""
      android:host=""
      android:port=""
      android:path=""
      android:pathPrefix=""
      android:pathPattern="" />
```

上面的<data.../>元素支持如下属性。

- mimeType：用于声明该组件所能匹配的Intent的Type属性。
- scheme：用于声明该组件所能匹配的Intent的Data属性的scheme部分。

- host：用于声明该组件所能匹配的Intent的Data属性的host部分。
- port：用于声明该组件所能匹配的Intent的Data属性的port部分。
- path：用于声明该组件所能匹配的Intent的Data属性的path部分。
- pathPrefix：用于声明该组件所能匹配的Intent的Data属性的path前缀。
- pathPattern：用于声明该组件所能匹配的Intent的Data属性的path字符串模板。

【注意】：Intent的Type属性要完全一致才可以启动此组件；而Data属性并不要求完全一致，如下：

- 目标组件的<data.../>子元素只指定了android:scheme属性，那么只要Intent的Data属性的scheme部分与android:scheme属性值相同，即可启动该组件。
- 目标组件的<data.../>子元素只指定了android:scheme、android:host属性，那么只要Intent的Data属性的scheme、host部分与android:scheme、android:host属性值相同，即可启动该组件。
- 目标组件的<data.../>子元素只指定了android:scheme、android:host、android:port属性，那么只要Intent的Data属性的scheme、host、port部分与android:scheme、android:host、android:port属性值相同，即可启动该组件。

Extra属性

Intent的Extra属性通常用于在多个Action之间进行数据交换，Intent的Extra属性值应该是一个Bundle对象，Bundle对象就像一个Map对象，它可以存入多个key-value对，这样就可以同过Intent在不同Activity之间进行数据交换了。

Flag属性

Intent的Flag属性用于为该Intent添加一些额外的控制旗标，Intent可调用addFlags方法来添加控制旗标。

如：Intent.FLAG_ACTIVITY_CLEAR_TOP旗标可用于清除当前activity栈中的Activity；

小结

本章介绍了Android中的Intent的功能和用法，而且四大组件都是用Intent来启动的，讲述了显示启动，和隐式启动；