



《移动终端开发技术》 电子教案

第二单元 使用 Intent 在活动之间穿梭

所属专业（教研室）： 计算机软件技术

制定人： 陈媛媛

合作人：

制定时间： 2018年2月

日照职业技术学院



单元标题	使用 Intent 在活动之间穿梭	单元教学学时	4 课时
		在整体设计中的位置	第 10 次
授课班级		上课地点	一体化教室
上课时间	周 月 日第 节		
教学目标	能力目标	知识目标	素质目标
	1、能够熟练使用显式意图和隐式意图； 2、能够熟练使用 Intent 传递数据。	1、了解 Intent 的含义； 2、掌握显式意图和隐式意图的用法； 3、掌握使用 Intent 传递数据的方法。	1、养成积极主动学习意识； 2、养成勤于动手的习惯。
教学重点、难点	教学重点：显式意图和隐式意图的用法 教学难点：Intent 传递数据的方法		
教学方法	采用反转课堂教学模式，课前学生学习微课了解知识点，课上采用教师引导、演示，学生分组练习、讨论等教学方法。 运用多媒体、AndroidStudio 开发环境、实训助手、教学平台等辅助授课。		
课前需掌握的知识点	简单的意图使用方法： <code>Intent intent=new Intent(FirstActivity.this,SecondActivity.class);</code> <code>startActivity(intent);</code>		
教学任务分解	任务一、认识 Intent 任务二、显式 Intent 和隐式 Intent 任务三、使用 Intent 传递数据		
教学总结			



一、情景导入

1、引入 Intent

教师引导，很久以前通信技术不发达时，人们只能通过信件的方式互相通信，这中间一定会用到“邮递员”，这个邮递员就起到了传递信息的作用。那么大家思考一下，在 Android 系统中，组件之间怎么进行数据传递呢？

2、引出本单元相关学习内容

本节课我们将一起学习 Intent 的用途、显式 Intent 和隐式 Intent 的使用方法、使用 Intent 传递数据的方法。

二、课前测试

在网络课程平台上做以下练习：

1、Android 中下列属于 Intent 的作用的是（）

- A. 实现应用程序间的数据共享
- B. 是一段长的生命周期，没有用户界面程序，可以保持应用程序在后台运行，而不会因为切换页面而消失
- C. 可以实现界面间的切换，可以包含动作和动作数据，连接四大组件的纽带
- D. 处理一个应用程序整体性的工作

2、下列组件中，不能使用 Intent 启动的是（）。

- A. Activity
- B. 服务
- C. 广播
- D. 内容提供者

答案 D

3、Intent intent=new Intent (FirstActivity. this, SecondActivity. class); startActivity(intent); 是用来启动隐式意图的。（）

答案 错

4、Android 中 Intent 寻找目标组件的方式分为两种，一种是显式意图和一种是隐式意图。（）

答案 对

三、知识讲解

任务一、认识 Intent

很久以前通信技术不发达时，人们只能通过信件的方式互相通信，这中间一定会用到“邮递员”，这个邮递员就起到了传递信息的作用。那么大家思考一下，在 Android 系统中，组件之间怎么进行数据传递呢？

在 Android 系统中，使用的是 Intent 进行数据传递。Intent 中文翻译为“意图”，它是 Android 程序中各个组件进行信息传递的一种重要方式，它不仅指定当前组件要执行的动作，还可以在不同组件之间进行数据传递。

Intent 一般用于启动 Activity、启动服务、发送广播等，承担了 Android 应用程序三大核心组件相互间的通信功能。Android 中 Intent 寻找目标组件的方式分为两种，一种是显式意图和一种是隐式意图。

任务二、显式意图和隐式意图

(1) 显式意图

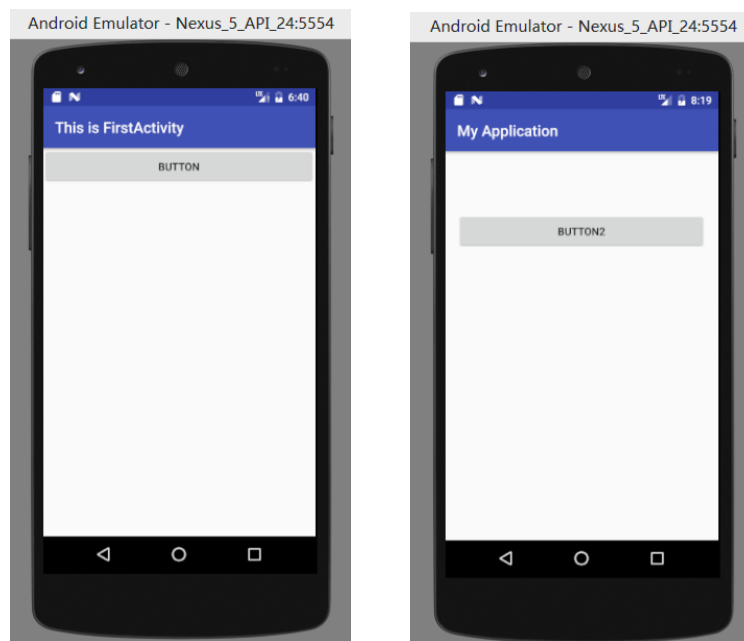
通过 Intent 启动 Activity 时，需要明确指定激活组件的名称。

新建 Activity 名为 secondActivity

在第一课时的基础上进行修改

修改 FirstActivity 中按钮的点击事件：

```
Button button1=(Button) findViewById(R.id.button);
button1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new Intent(FirstActivity.this,SecondActivity.class);
        startActivity(intent);
    }
});
```



运行效果图

(2) 隐式意图

没有明确指定组件名的 Intent 称为隐式意图。Android 系统会根据隐式意图中设置的动作 (action)、类别 (category)、数据 (Uri 和数据类型) 找到最合适组件。

```
<activity android:name="com.itcast.intent.Activity02">
    <intent-filter>
        <action android:name="cn.itscast.xxx"/>
        <category
android:name="android.intent.category.DEFAULT"/>
    </intent-filter>
</activity>
```

使用隐式意图开启 Activity 的示例代码如下所示：

```
Intent intent = new Intent();
```

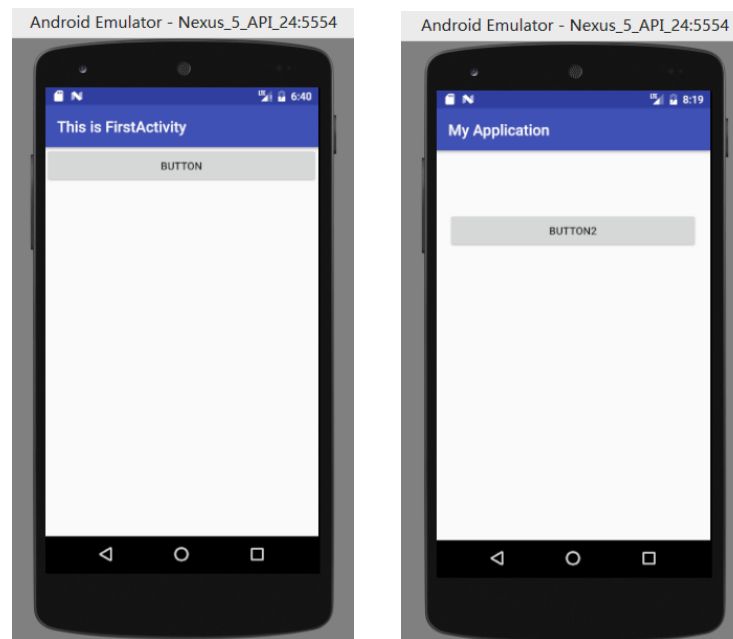
```
// 设置动作和清单文件一样  
intent.setAction("cn.itscast.xxx");  
startActivity(intent);
```

案例一、AndroidManifest.xml 中代码

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

FirstActivity 中代码

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



运行效果图

案例二、在 FirstActivity 中加入一个 Category 后，再运行程序

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

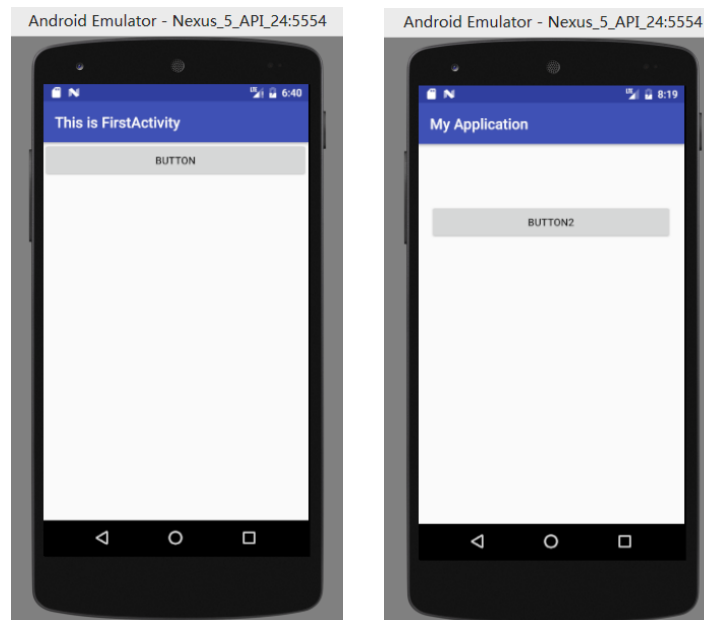
出现错误:

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

需要在 AndroidManifest.xml 中加入代码

```
<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>
```

再运行程序查看结果

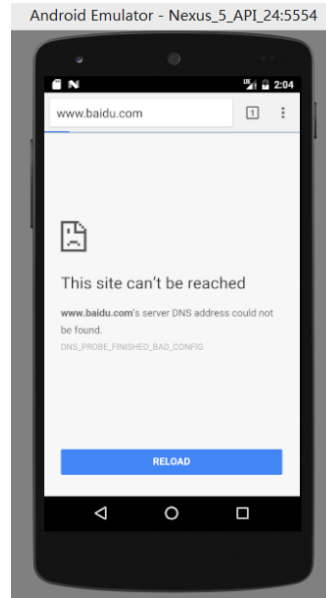


运行效果图

案例三、打开浏览器

修改 FirstActivity 中按钮点击事件的代码

```
button1.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent=new Intent(Intent.ACTION_VIEW);
        intent.setData(Uri.parse("http://www.baidu.com"));
        startActivity(intent);
    }
});
```



运行效果图

任务三、数据的传递

(1) 向下一个活动传递数据

在 Android 开发中，经常要在 Activity 之间传递数据。使用 Intent 传递数据只需调用 `putExtra()` 方法将想要存储的数据存在 Intent 中即可。当启动了另一个 Activity 后，再把这些数据从 Intent 中取出即可。

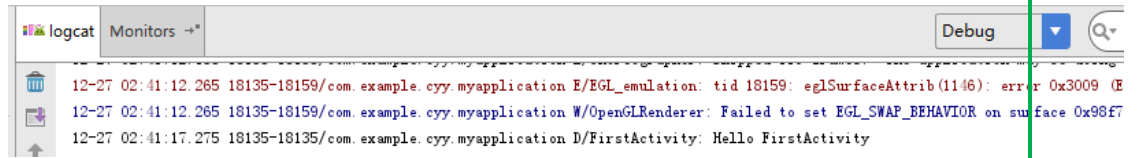
案例：在 FirstActivity 中将数据存储存在 Intent 中

```
button1.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View view) {  
        String data="Hello SecondActivity";  
        Intent intent=new Intent(FirstActivity.this,SecondActivity.class);  
        intent.putExtra("extra_data",data) ;  
        startActivity(intent);  
    }  
});
```

从 SecondActivity 中取出传递过来的数据

```
protected void onCreate(Bundle savedInstanceState) {  
    super.onCreate(savedInstanceState);  
    setContentView(R.layout.second_layout);  
    Intent intent=getIntent();  
    String data=intent.getStringExtra("extra_data");  
    Log.d("SecondActivity",data);  
}
```

运行效果：



(2) 返回数据给上一个活动

FirstActivity

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.first_layout);
    Button button1=(Button) findViewById(R.id.button);
    button1.setOnClickListener(new View.OnClickListener() {
```

```
    @Override
```

```
    public void onClick(View view) {
        String data="Hello SecondActivity";
        Intent intent=new Intent(FirstActivity.this,SecondActivity.class);
        intent.putExtra("extra_data", data) ;
        startActivityForResult(intent,1);
    }
});
```

```
}
protected void onActivityResult(int requestCode,int resultCode,Intent data){
    switch (requestCode){
        case 1:
            if(resultCode==RESULT_OK){
                String returnData=data.getStringExtra("data_return");
                Log.d("FirstActivity",returnData);
            }
    }
}
```

SecondActivity

```
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.second_layout);
    Button button2=(Button) findViewById(R.id.button2);
    button2.setOnClickListener(new View.OnClickListener() {
```

```
    @Override
```

```
    public void onClick(View view) {
        Intent intent=new Intent();
        intent.putExtra("data_return","Hello FirstActivity");
        setResult(RESULT_OK,intent);
        finish();
    }
}
```




```
});}  
public void onBackPressed() {  
    Intent intent=new Intent();  
    intent.putExtra("data_return","Hello FirstActivity");  
    setResult(RESULT_OK,intent);  
    finish();  
}
```

运行效果:

```
12-27 02:13:43.925 28248-28343/com.example.cyy.myapplication D/OpenGLESRenderer: Swap behavior 1  
12-27 02:13:44.009 28248-28343/com.example.cyy.myapplication E/EGL_emulation: tid 28343: eglSurfaceAttrib(1146): error 0x3009 (EGL_BAD  
12-27 02:13:44.009 28248-28343/com.example.cyy.myapplication W/OpenGLESRenderer: Failed to set EGL_SWAP_BEHAVIOR on surface 0x98f24100,  
12-27 02:13:50.764 28248-28248/com.example.cyy.myapplication D/SecondActivity: Hello SecondActivity
```

五、作品展示

各小组展示作品，教师提出作品优点和存在的问题，各小组课后完善作品。

六、知识巩固

- 1、总结知识点，使用教学平台中的随堂练习题巩固本所学知识。
- 2、使用教学平台中的测试题给学生布置作业。

拓展
作业

- 1、教学平台中的测试题；
- 2、搜索更多 Intent 的应用。

教学
后记