

## 第三讲 Activity 和 Fragment

### 使用说明

【源代码根目录】： 工程目录 ActivityTaskFragment

【记号】:

(@编程练习): 表明该实验是需要编写工程文件和实验报告一起提交。

本次实验的编程部分，请建立在名为 KH3 的 project 中。

【实验报告的要求】

- 1、 文件名规范： 学号+姓名+实验名称.doc
- 2、 内容格式见 实验报告格式.doc

## 目录

第三讲 Activity 和 Fragment .....	1
【课堂实验 KT3.1】:Activity 的启动 .....	3
【课堂实验 KT3.2】:Activity 间数据传递 .....	4
【课堂实验 KT3.3】:startActivityForResult .....	4
【课堂实验 KT3.4】:Activity 生命周期 .....	5
【课堂实验 KT3.5】: Fragment 基本用法 .....	6

## 【课堂实验 KT3.1】:Activity 的启动

【工程模块】: ActivityBaseSample

【工程描述】: 该程序展示了 activity 通过显式或隐式的启动方法。

【目标要求】:

- 1、理解显式或隐式启动 activity 的机制
- 2、掌握 activity 切换的基本方法
- 3、掌握声明权限的方法

【实验步骤】:

- 1、运行程序，点击按钮或者文字观察现象。

【问题】

1. 如下代码为什么要使用 try{}catch{}结构?

```
textView.setOnClickListener(new View.OnClickListener() {  
    @Override  
    public void onClick(View v) {  
        Intent intent = new Intent(Intent.ACTION_CALL, Uri.parse("tel:134"));  
        try {  
            startActivity(intent);  
        } catch (Exception e) {  
            e.printStackTrace();  
        }  
    }  
});
```

2. （@编程练习）实现如下功能：在自己编写的 activity 中传递一个 URL，然后隐式启动

一个 activity 来浏览该网址？

3. 思考如下代码

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

为何不直接写 SecondActivity，而要写 SecondActivity.class。我们可以用显式方法启动非本 app 中的 activity 吗？

## 【课堂实验 KT3.2】:Activity 间数据传递

【工程模块】: ActivityDataExSample

【工程描述】: 该程序展示了 activity 之间传递数据的基本方法。

【目标要求】:

理解 Intent 的 Extras 属性的作用

掌握使用 Extras 和 Bundle 传递键值对数据的方法

掌握利用 Application 设置公共变量进行数据传递的方法

【实验步骤】:

- 1、运行程序，在上下两个文本框中分别输入文字和数字，点击第一个按钮，观察现象。
- 2、理解分析代码

【问题】

### 1、(@编程练习)

该程序仅用了 Intent 的 putExtras 和 getExtras 系列函数来直接传送键值对数据，请实现余下的两个按钮的事件，分别用另外两种方法来实现数据传递。

- 2、使用 application 公共变量的方法，需要注意什么？

## 【课堂实验 KT3.3】:startActivityForResult

【工程模块】: StartActivityForResultSample

【工程描述】: 该程序展示了 activity 之间传递数据的另一种方法。

【目标要求】:

掌握使用 startActivityForResult 进行 activity 间数据传递的方法

【实验步骤】:

- 1、运行程序，分别点击按钮，启动 activityresult1 和 activityresult2 后按 back 键后观察提示文字的变化。
- 2、分析理解代码

【问题】

- 1、如下回调函数中的参数都有什么作用？

onActivityResult(**int** requestCode, **int** resultCode, Intent data)

2、本例代码中用了很多常量，你觉得这样做的优点有哪些？

## 【课堂实验 KT3.4】:Activity 生命周期

【工程模块】: ActivityLifecycleSample

【工程描述】: 展示了一个 activity 从创建到结束的完整生命周期中，回调函数的调用顺序以及两个 activity 互相切换其回调函数交错的调用顺序。

【目标要求】:

理解 Activity 的各个生命周期状态，及其发生条件

理解 Activity 生命周期包含的三种周期。

理解两个 Activity 相互影响，它们回调函数的调用顺序

理解掌握 OnPause 状态的作用和使用时机。

【实验步骤】:

- 1、运行程序，点击按钮，多进行几次 activity 间的切换。
- 2、打开 IDE 下方的 Android monitor 面板，选中 Logcat,查看程序运行信息
- 3、为方便过滤信息，可以选中面板中的 log level,选择 info。
- 4、点击按钮或者按 back 键，记录代码中 log.i 的输出信息。

【问题与思考】

- 1、分析 log.i 输出信息，总结 activity 创建和撤销过程中回调函数的执行顺序。
- 2、按 UI 中的按钮或者 back 键均可能返回到上一个 activity，但它们有什么不同效果吗？

【知识点注释】:

- 注意，OnPause 是活动撤出前可以确保执行的回调函数，而其他 onStop，onDestroy 都不保证能被执行。
- 除了上面提及的生命周期中的回调函数外，还有一个 onSaveInstanceState() 也是 activity 的一个回调函数，它的作用是当 Activity 状态需要保存，比如输入的用户名，如果我们通过 home 按键暂时离开应用，等再次回到该应用时，希望这些输入能如离开时一样。但是，由于系统可能会因为内存不足等原因销毁暂停的应用，因此这些状态要保留就需要在回调函数 onSaveInstanceState() 中通过一个 bundle 对象交给系统保存，当这个应用再次被启动时，系统会用这个 bundle 对象重新恢复 Activity 的状态（请注意 onCreate 函数的参数）。这个函数的调用时机由系统根据情况调用，并不一定会在 activity onPause 时调用，甚至可能就根本不调用。

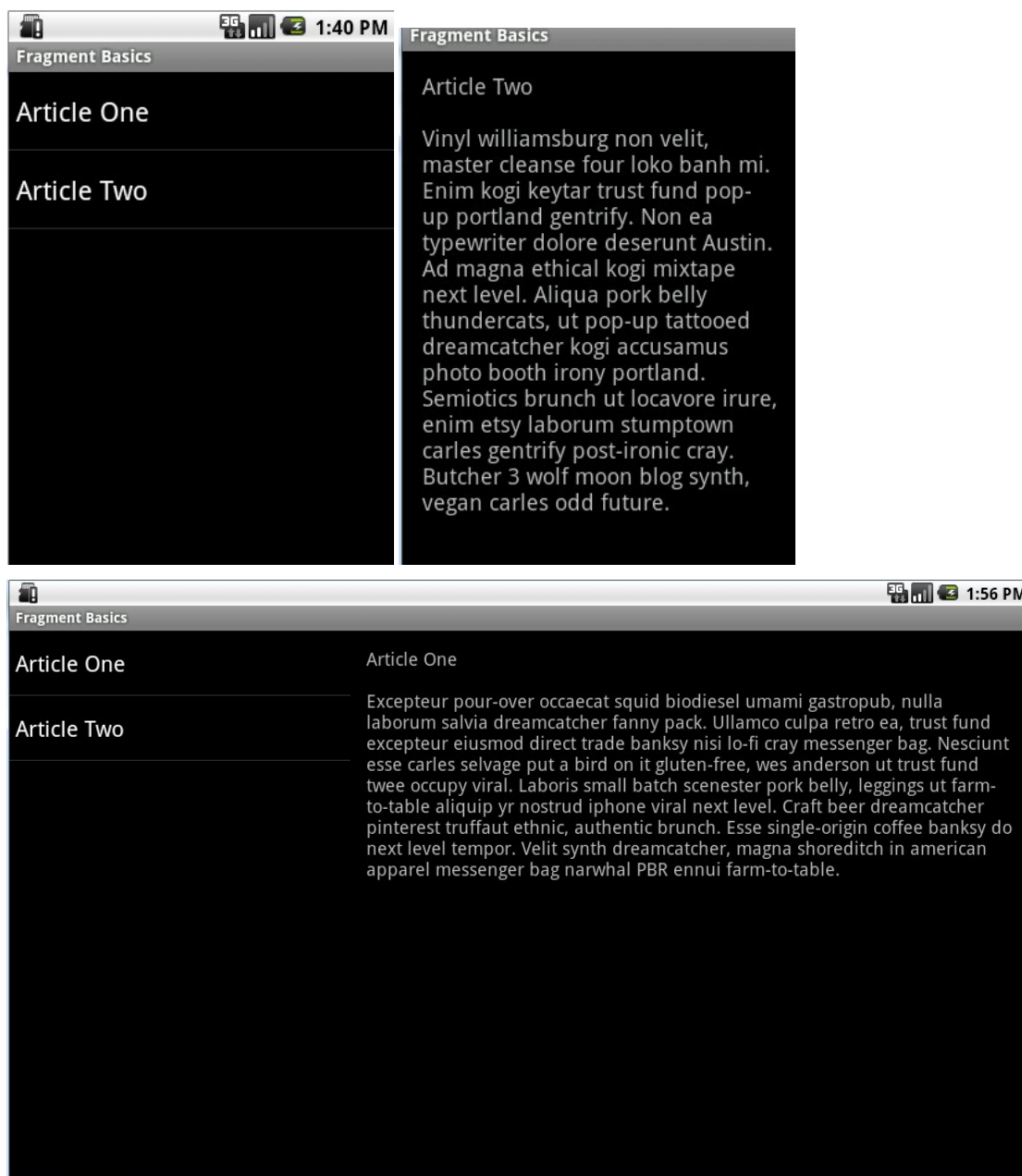
这个回调函数还是很有用的，比如当我们将屏幕有横向变成纵向时，是不希望输入的数据丢失的，但实际上这个操作，系统为了更新资源（比如横屏布局换成竖屏布局）需要先销毁 activity 然后在 create。如果我们通过这个方法保存了中间数据，就可以让系统重建时还原数据。但真实情况是我们发现及时我们没有实现该回调函数，UI 也能保存中间数据，这是因为这些 UI 控件已经在自己的类中实现了相同的回调函数。此部分请参阅阅读材料。

## 【课堂实验 KT3.5】：Fragment 基本用法

【工程模块】： FragmentBasicSample

【工程描述】：

展示了使用 Fragment 的通用方法，以及 fragment 和 activity 之间交互的方法



【知识点和注释】：

- 1、在 res/layout-large 下放置和 res/layout 下同名的 xml，在大尺寸下回自动选择
- 2、注意 fragment 间传递数据的方法

```
ArticleFragment newFragment = new ArticleFragment();
```

```
Bundle args = new Bundle();
```

```
args.putInt(ArticleFragment.ARG_POSITION, position);
```

```
newFragment.setArguments(args);
```

```
Bundle args = getArguments();
```

3、有两种方式在 activity 中加入 fragment

- Xml 中定义

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="horizontal"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
    <fragment android:name="com.example.news.ArticleListFragment"
        android:id="@+id/list"
        android:layout_weight="1"
        android:layout_width="0dp"
        android:layout_height="match_parent" />
    <fragment android:name="com.example.news.ArticleReaderFragment"
        android:id="@+id/viewer"
        android:layout_weight="2"
        android:layout_width="0dp"
        android:layout_height="match_parent" />
</LinearLayout>
```

- 程序中加入

```
FragmentManager fragmentManager = getFragmentManager\(\)
FragmentManager fragmentManager = fragmentManager.beginTransaction();
ExampleFragment fragment = new ExampleFragment();
fragmentTransaction.add(R.id.fragment_container, fragment);
fragmentTransaction.commit();
```

## 【问题】

1、（@编程练习）分析学习本例代码，实现如下程序。

设计一个 Activity，嵌入两个 fragment，其中一个的布局中包含有一个 Textview。另一个的布局中含有一个 EditText 和一个 Button。它们上下均等地排列在 Activity 的主布局中。当点击按钮时，将 edittext 中输入的文字显示在上方 Fragment 的 Textview 中。

2、理解如下代码，为何主 Activity 要实现定义在 fragment 中的 OnHeadlineSelectedListener 接口，Fragment 如何同其宿主 Activity 通信？

```
public class MainActivity extends AppCompatActivity implements
HeadlinesFragment.OnHeadlineSelectedListener
```

本次实验的编程作业，请建立在名为 KH3 的 project 中。