

## Android 第三次实验报告

### 一、实验目的

1. 熟练掌握 RadioGroup 控件的使用，实现其与 Fragment 的绑定及 Fragment 切换功能，理解组件间关联逻辑。
2. 深入理解 Bundle 数据传输机制，能够独立完成 Activity 与 Activity、Activity 与 Fragment、Fragment 与 Fragment 三种场景下的数据传递。
3. 明确 Activity 生命周期中 onSaveInstanceState 方法的调用时机，掌握屏幕旋转时数据保存与恢复的实现方式，解决状态丢失问题。

### 二、实验步骤与结果记录

#### 第一部分： RadioGroup 控制 Fragment 切换

要求：

- 创建包含 4 个 RadioButton 的 RadioGroup
- 每个 RadioButton 对应一个不同的 Fragment
- 点击 RadioButton 时切换到对应的 Fragment
- 每个 Fragment 要有独特的 UI 和功能区分
- res/drawable/radio\_button\_selector.xml 定义背景选择器, 定义 radiobutton 是否 check 的状态
- 在 style 中定义 radiobutton 统一样式

```
<resources>
    <style name="RadioButtonStyle" parent="Widget.AppCompat.CompoundButton.RadioButton">
        <item name="android:background">@drawable/radio_button_selector</item>
        <item name="android:padding">12dp</item>
        <item name="android:textColor">@android:color/white</item>
        <item name="android:layout_margin">4dp</item>
    </style>
</resources>
```



## 第二部分：Bundle 数据传输

要求实现三种场景：

**场景 A: Activity → Activity** 从 MainActivity 传递用户数据到 DetailActivity，包含：用户名、年龄、是否学生等信息

**场景 B: Activity ↔ Fragment** MainActivity 向 Fragment 传递初始数据，Fragment 向 Activity 返回处理结果

**场景 C: Fragment → Fragment** 在两个 Fragment 之间通过 Activity 中转传递数据

## DetailActivity-接收来自MainActivity数据

用户名: lzd

年龄: 18

是否学生: 是

返回MAINACTIVITY

## Fragment B - Fragment → Fragment 数据传递

7777

发送到FRAGMENT C



消息已发送到FragmentC

## 第三部分：屏幕旋转与状态保存 onSaveInstanceState

- log onSaveInstanceState 的在生命周期的哪个阶段出现
- 在 onSaveInstanceState 中保存 edittext 中内容，并在屏幕旋转后实现在 textview 中进行展示

