



## 获取用户信息

### 1. 实训目的

- (1) 熟悉常用布局类型；
- (2) 掌握常用控件 EditText、TextView、Radio Button、CheckBox 等的用法；
- (3) 运用所学知识设计用户界面并实现用户交互。

### 2. 实训要求

- (1) 创建项目并选择合适的布局类型；
- (2) 创建布局和设计布局文件；
- (3) 完成交互代码，获取用户选择的信息；
- (4) 完成实验报告。

### 3. 实训指导

RadioButton、CheckBox 这两个控件都存在一个选中和没有选中的状态。都继承自 android.widget.CompoundButton 类，而 CompoundButton 又继承自 Button 类，在这个类中封装了一个 checked 的属性，用于判断是否选中。

Android:checked 指定初始时是否被选中。

RadioButton 通常要和 RadioGroup 一起使用，用于定义一组单选按钮。

#### (1) RadioButton 和 CheckBox 的区别：

单个 RadioButton 在选中后，通过点击无法变为未选中；

单个 CheckBox 在选中后，通过点击可以变为未选中；

RadioButton 在大部分 UI 框架中默认都以圆形表示；

CheckBox 在大部分 UI 框架中默认都以矩形表示；

#### (2) RadioButton 和 RadioGroup 的关系：

RadioButton 表示单个圆形单选框，而 RadioGroup 是可以容纳多个 RadioButton 的容器。

每个 RadioGroup 中的 RadioButton 同时只能有一个被选中。

不同的 RadioGroup 中的 RadioButton 互不相干，即如果组 A 中有一个选中了，组 B 中依然可以有一个被选中。

大部分场合下，一个 RadioGroup 中至少有 2 个 RadioButton。

大部分场合下，一个 RadioGroup 中的 RadioButton 默认会有一个被选



中，并建议您将它放在 RadioGroup 中的起始位置。

(3) 当 RadioGroup 中的选中项变更或 CheckBox 选项变更时，您可能需要做一些相应，为其注册监听器 setOnCheckedChangeListener()。

获取用户信息源码：

```
<TableLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="match_parent"
    android:layout_height="match_parent">
<TableRow>
<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:text="性别："/>
<!--定义一组单选按钮-->
<RadioGroup
    android:id="@+id/rg"
    android:orientation="horizontal"
    android:layout_gravity="center_horizontal">
<!--定义两个单选按钮-->
<RadioButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/male"
    android:text="男"
    android:checked="true"/>
<RadioButton
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/female"
```



```
        android:text="女" />
</RadioGroup>
</TableRow>
<TableRow>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="你喜欢的颜色: " />
    <!--定义一个垂直的线性布局-->
    <LinearLayout
        android:layout_gravity="center_horizontal"
        android:orientation="vertical"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content">
        <!--定义三个复选按钮-->
        <CheckBox
            android:id="@+id/ck1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="红色"
            android:checked="true" />
        <CheckBox
            android:id="@+id/ck2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="蓝色" />
        <CheckBox
            android:id="@+id/ck3"
            android:layout_width="wrap_content"
```



```
        android:layout_height="wrap_content"
        android:text="黑色" />
    </LinearLayout>
</TableRow>
<TextView
    android:id="@+id/show"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>
<TextView
    android:id="@+id/show1"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"/>
</TableLayout>
MainActivity.java
public class MainActivity extends AppCompatActivity {
    String tip1="你喜欢: ";
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.radiocheckdemo);
        //获取 rg、show 两个组件
        RadioGroup rg=(RadioGroup) findViewById(R.id.rg);
        final TextView tw=(TextView) findViewById(R.id.show);
        final TextView tw1=(TextView) findViewById(R.id.show1);
        final CheckBox ck1=(CheckBox) findViewById(R.id.ck1);
        final CheckBox ck2=(CheckBox) findViewById(R.id.ck2);
        final CheckBox ck3=(CheckBox) findViewById(R.id.ck3);
        rg.setOnCheckedChangeListener (new
RadioGroup.OnCheckedChangeListener () {
```



```
@Override
public void onCheckedChanged(RadioGroup radioGroup, int i)
{
    String tip=i==R.id.male?"您的性别为男":"您的性别为女";
    tw.setText(tip);
}
});
ck1.setOnCheckedChangeListener(new
CheckBox.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(CompoundButton
compoundButton, boolean b) {
        if(b==true)
            tip1+=" 红色 ";
        tw1.setText(tip1);
    }
});
ck2.setOnCheckedChangeListener(new
CheckBox.OnCheckedChangeListener() {
    @Override
    public void onCheckedChanged(CompoundButton
compoundButton, boolean b) {
        if(b==true)
            tip1+=" 蓝色 ";
        tw1.setText(tip1);
    }
});
ck3.setOnCheckedChangeListener(new
CheckBox.OnCheckedChangeListener() {
```

```
@Override
public void onCheckedChanged(CompoundButton
compoundButton, boolean b) {
    if(b==true)
        tip1+=" 黑色 ";
    tw1.setText(tip1);
}
});
}
```

运行效果图：

