
南方科技大学
SOUTHERN UNIVERSITY OF SCIENCE AND TECHNOLOGY

# 《电子科学创新实验 I》

# 课程报告

题　　目：基于 Android Studio 的计算器 APP 的设计与开发

姓　　名：张皓东

学　　号：12113010

系　　别：电子与电气工程系

专　　业：通信工程

指导教师：徐琳琳

# 基于 Android Studio 的计算器 APP 开发

[摘要]：Android Studio 是谷歌推出的一个 Android 集成开发工具，基于 IntelliJ IDEA 用于 Android APP 的开发。Android Studio 提供了集成的 Android 开发工具用于开发和调试。本项目用 Android Studio 设计了一款计算器 APP，实现了基本的加减乘除功能、清零功能、删除功能。并在原有项目的基础上修复了一定的问题并进行了界面的美化和部分功能上的提升。

## 1. 与原项目的比较

### 1.1 界面（布局）改进

本项目对原有项目的界面进行了较大的改进和优化。首先在初始界面中，对原有的背景进行了更换，并对按钮的背景进行了重新设计，将按钮的背景变为透明色，同时增加了边框。并且为按钮的点击增加了点击效果，包括背景颜色改变、有声效产生。初始界面前后对比如下：



下面阐述一下界面改进的实现方式：

1. 为界面添加背景，选择一张美观的图片，在 drawable 文件夹中导入改图片，并在 activity 中将背景选择为该图片：

```
android:background="@drawable/bg"
```

2. 为按钮增添背景及点击效果，在 drawable 文件夹中新建 xml 文件“btnbg.xml”。在该文件中将按钮背景颜色改为透明色，并增加

边框，将边框的四角设为半圆形的弧度。同时也设置了点击背景颜色信息改变，即点击时将背景颜色变为白色，以上操作的实现方法如下：

btnbg.xml 文件代码：

```
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
<item android:state_pressed="false">
<shape xmlns:android="http://schemas.android.com/apk/res/android">
    <gradient android:startColor="#c0000000"
android:endColor="#c0000000" android:angle="90" /><!--背景颜色渐变 angle
为渐变角度-->
    <solid android:color="#00ffffff" /><!-- 背景填充颜色 -->
    <stroke android:width="1dp" android:color="@color/black" /><!-- 描
边，边框宽度、颜色 -->
    <corners android:radius="10dp" /><!-- 边角圆弧的半径 -->
    <padding android:left="3dp" android:top="3dp" android:right="3dp"
android:bottom="3dp" /><!-- 四周留出来的空白 -->
</shape>
</item>
    <item android:state_pressed="true">
        <shape
xmlns:android="http://schemas.android.com/apk/res/android">
            <gradient android:startColor="#c0000000"
android:endColor="#c0000000" android:angle="90" /><!--背景颜色渐变 angle
为渐变角度-->
            <solid android:color="#FFFFFF" /><!-- 背景填充颜色 -->
            <stroke android:width="1dp" android:color="@color/black"
/><!-- 描边，边框宽度、颜色 -->
            <corners android:radius="10dp" /><!-- 边角圆弧的半径 -->
            <padding android:left="3dp" android:top="3dp"
android:right="3dp" android:bottom="3dp" /><!-- 四周留出来的空白 -->
        </shape>
    </item>
</selector>
```

并在 activity 中将按钮背景设为上面的背景：

```
android:background="@drawable/btnbg"
```

除此之外，本项目还为按钮的点击添加了音效，首先创建一个文件夹 raw，并在该文件夹中导入选择的音乐资源“sound.mp3”。之后在 MainActivity.java 中写一个 playSound()方法，并在点击事件中调用

该方法。这样每次点击时都会有音效产生。

playSound()方法代码如下（其他代码详情见附录）：

```
private void playSound() {
    soundPool.play(
            soundID,
            0.1f,   //左耳道音量【0~1】
            0.5f,   //右耳道音量【0~1】
            0,     //播放优先级【0 表示最低优先级】
            0,     //循环模式【0 表示循环一次，-1 表示一直循环，其他表示数字+1 表示当前数字对应的循环次数】
            1     //播放速度【1 是正常，范围从 0~2】
    );
}
```

1.2 部分问题的修复

1.2.1 原项目中，对于输入“0”后还会再输入其他的数字，比如出现“00”“07”等情况，如下图所示：

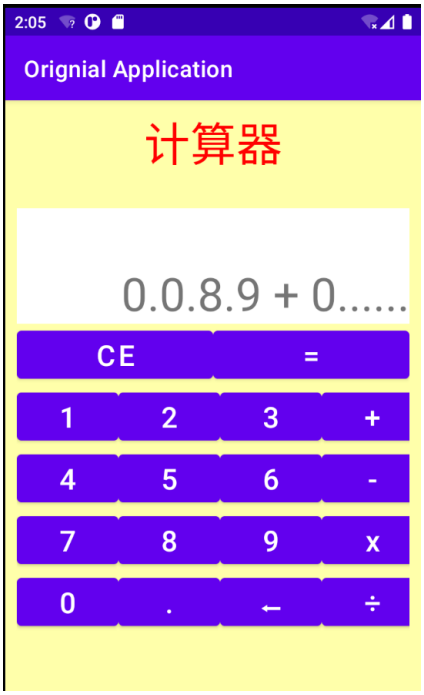

为了避免出现这样的情况，本项目对代码进行了修改，即在每个数字的点击响应中添加如下代码：

```
    if(ss.contains("0")) {
        if (ss.equals("0"))
```

```
            break;
        if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")+3).equals("0")))
            break;
    }
```

也就是在每次输入任意数字都进行检查，如果此时显示的数字（用字符串 ss 表示）在操作符之前或者操作符之后的数字只有一个“0”，那么将不可以输入数字。因此该问题得以解决。

1.2.2 关于小数点，原项目中也有若干问题，比如允许输入多个“.”，包括小数点可以多个连续输入，在同一个数中会出现多个小数点，表现如下图所示：




对于该问题，该项目对关于“.”的点击事件中加了若干判断，来避免上述情况的产生，修改后代码如下所示（其中黄色下划线区域为新项目所添加的代码）：

```
case R.id.dian: {
    if (ss.length() == 0) {
        break;
    } else if (!ss.contains(" ")&&ss.contains(".")){
```

```
        break;
        }
    else if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")).contains(".")||ss.indexOf(" ")==ss.length()-3)){
        break;
    } else{
        ss += ".";
        edit.setText(ss);
    }
}
break;
```

上述代码增添对了正要输入的数字（操作符前和操作符后）有无“.”的检查，如果有“.”，则新的“.”将不可再次输入，问题得以解决。

1.2.3 在调试的过程中发现，点“back”按钮时，如果此时显示框中没有任何数字，此时整个 app 程序会被关闭，这是不符合正常的计算器使用操作的。经过对代码的研读发现是因为“back”的点击响应程序中在显示框没有任何数字时仍然会有操作，因此程序将被退出，原代码如下：

```
case R.id.back: {
    if (ss.contains(" ")){
        if (ss.indexOf(" ") == ss.length() - 3) {
            ss = ss.substring(0, ss.length() - 3);
            edit.setText(ss);
            break;
        }
    }
    ss = ss.substring(0, ss.length() - 1);
    edit.setText(ss);
    break;
}
```

在此基础上检查一下为空时，将该操作 break 即可，修改后如下：

```
case R.id.back: {
    if (ss.length() == 0) {
        break;
```

```
    }
    if (ss.contains(" ")){
        if (ss.indexOf(" ") == ss.length() - 3) {
            ss = ss.substring(0, ss.length() - 3);
            edit.setText(ss);
            break;
        }
    }

    ss = ss.substring(0, ss.length() - 1);
    edit.setText(ss);
    break;
}
```

1.2.4 在原项目调试中发现会出现计算错误的问题，比如，输入“7.89+0.01”，得到的结果应当为“7.9”，但是原项目计算器显示的结果为“7.899999……”，如下图所示：


经过相关资料查询和推断，发现这个问题应该是 java 中 double 数据类型的精度问题。因为计算机是使用二进制存储数据的。而平常生活中，大多数情况下我们都是使用的十进制，因此计算机显示给我们

看的内容大多数也是十进制的，这就使得很多时候数据需要在二进制与十进制之间进行转换。对于整数来说，两种进制可以做到一一对应。而对于小数来讲就不一定是如此。而在用 java 中 double 数据类型计算时，此浮点数没有办法使用二进制进行精确表示，所以会产生一定的误差和精度丢失，为了使得计算器可以更精确的显示计算结果，经过相关资料查询后决定采用 java.math.BigDecimal 类来进行精确计算。也就是将原项目 getResult()方法中的 double 类型的变量之间的计算转化为 BigDecimal 之间的计算。代码前后变化如下：

原项目中 getResult()方法：

```
private void getResult() {
       double result = 0;
       if (ss == null || ss.equals("")) return;
       if (!ss.contains(" ")) return;
       String s1 = ss.substring(0, ss.indexOf(" "));
       String op = ss.substring(ss.indexOf(" ") + 1, ss.indexOf(" ") +
2);
       String s2 = ss.substring(ss.indexOf(" ") + 3);
       if (!s1.equals("") && !s2.equals("")) {
          double d1 = Double.parseDouble(s1);
          double d2 = Double.parseDouble(s2);
          switch (op) {
             case "+":
                result = d1 + d2;
                break;
             case "-":
                result = d1 - d2;
                break;
             case "×":
                result = d1 * d2;
                break;
             case "/": {
                if (d2 == 0) {
                   Toast.makeText(MainActivity.this, "不能除以零",
Toast.LENGTH_SHORT).show();
                   break;
```

```
                }
                result = d1 / d2 * 1.0;
            }
            break;
        }
        int r = (int) result;
        if (r == result) {
            edit.setText("" + r);
            ss = "" + r;
        } else {
            ss = "" + result; //将 double 类型的 result 转成 string 类型的
数据存储在 ss 中
            ss = ss.substring(0,ss.length()-1);
            edit.setText(ss);
        }
    }
}
```

修改后的 getResult()方法的代码如下：

```
private void getResult() {
    BigDecimal result = BigDecimal.ZERO;
    if (ss == null || ss.equals("")) return;
    if (!ss.contains(" ")) return;
    String s1 = ss.substring(0, ss.indexOf(" "));
    String op = ss.substring(ss.indexOf(" ") + 1, ss.indexOf(" ") +
2);
    String s2 = ss.substring(ss.indexOf(" ") + 3);
    ss_total.append(ss);
    if (!s1.equals("") && !s2.equals("")) {
        BigDecimal d1 = new BigDecimal(s1);
        BigDecimal d2 = new BigDecimal(s2);
        switch (op) {
            case "+":
                result = d1.add(d2);
                break;
            case "-":
                result = d1.subtract(d2);
                break;
            case "×":
                result = d1.multiply(d2);
                break;
            case "÷": {
                if (d2.compareTo(BigDecimal.ZERO)==0) {
                    Toast.makeText(MainActivity.this, "不能除以零",
```

```
Toast.LENGTH_SHORT).show();
                    break;
                }
                result = d1.divide(d2,20,BigDecimal.ROUND_HALF_UP);
            }
            break;
        }
            ss = result.stripTrailingZeros().toPlainString(); // 将
result 后面的 0 去掉
            edit.setText(ss);
      }
      ss_total.append("=");
      ss_total.append(ss);
      ss_total.append("\r\n");
   }
```

值得注意的是，为了确保精度需要直接将 String 类型数据和 BigDecimal 对象之间进行转化，而不能经过 double 类型数据。同时 BigDecimal 对象之间做除法运算如果出现除不尽的线下时也会出现一定的问题，因此除法需要指定保留位数，并给出进位规则，在这里选择有效位数 20 位并选择四舍五入的规则。相关代码为：

```
result = d1.divide(d2,20,BigDecimal.ROUND_HALF_UP);
```

这样，我们就可以解决了小数计算的问题。上面那个例子（7.89+0.01）的结果经修改为：



1.3 功能改进与扩展

本项目除了优化了界面，修复了错误和问题之外，还增添了新的功能：查询历史记录。通过该功能可以查询之前一段时间内的计算结果，方便用户查找之前的结果。实现方法如下：

首先在界面中增加一个按钮，利用 android studio 中自带的经典矢量图作为按钮背景，并添加点击事件：当点击该按钮时，会弹到另一个界面，在该界面中显示历史记录（第二界面的主界面代码及程序代码见附录）。现阐述历史记录功能的逻辑，首先创建一个 StringBuffer 对象 ss_total（因为 String 类型的数据是不可改变的）。

```
public static StringBuffer ss_total = new StringBuffer();
```

然后在每次调用 getResult()方法之前和之后都利用 append()方法将两字符串 ss_total 和 ss 进行拼接。当点击按钮时，便将 ss_total 字符串显示到第二界面中，为了能够在第二个 activity 中使用第一个 activity 的 ss_total，故将 ss_total 设为 static。（完整代码见附录）

实现效果如下：


Calculator
计算器
CE
=
1
2
3
+
4
5
6
-
7
8
9
x
0
.
←
÷


Calculator
历史记录
7 × 2=14
2 + 456=458
886 - 666=220

## 2. 附录

1）布局文件代码

1. 第一个布局文件（activity_main）代码：

```
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:padding="10dp"
    android:background="@drawable/bg"
    >

    <TextView
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:text="计算器"
        android:textSize="40sp"
        android:textColor="@color/black"
        android:layout_marginBottom="30dp"
        android:gravity="center"/>
    <TextView
        android:id="@+id/edit"
        android:background="@color/white"
        android:textSize="40dp"
        android:layout_width="match_parent"
        android:layout_height="100dp"
        android:gravity="end|bottom"
        />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <Button
            android:id="@+id/clear"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
```

```
            android:text="CE"
            android:textSize="25dp"
            android:textColor="@color/black"
            android:background="@drawable/btnbg"

            />

        <Button
            android:id="@+id/calculation"
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:text="="
            android:textSize="25dp"
            android:textColor="@color/black"
            android:background="@drawable/btnbg"
        />
    </LinearLayout>

    <GridLayout
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:rowCount="4"
        android:columnCount="4"
        android:layout_gravity="center"
        >



        <Button
            android:id="@+id/btn1"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="1"
            android:textSize="25dp"
            android:textColor="@color/black"
            android:background="@drawable/btnbg"
            />
        <Button
            android:id="@+id/btn2"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="2"
            android:textSize="25dp"
            android:textColor="@color/black"
```

```
        android:background="@drawable/btnbg"/>
    <Button
        android:id="@+id/btn3"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="3"
        android:textSize="25dp"
        android:textColor="@color/black"
        android:background="@drawable/btnbg"/>

    <Button
        android:id="@+id/jia"
        android:layout_width="76dp"
        android:layout_height="wrap_content"
        android:background="@drawable/btnbg"
        android:text="+"
        android:textColor="@color/black"
        android:textSize="25dp" />
    <Button
        android:id="@+id/btn4"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="4"
        android:textSize="25dp"
        android:textColor="@color/black"
        android:background="@drawable/btnbg"/>
    <Button
        android:id="@+id/btn5"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="5"
        android:textSize="25dp"
        android:textColor="@color/black"
        android:background="@drawable/btnbg"/>
    <Button
        android:id="@+id/btn6"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="6"
        android:textSize="25dp"
        android:textColor="@color/black"
        android:background="@drawable/btnbg"/>

    <Button
```

```
        android:id="@+id/jian"
        android:layout_width="76dp"
        android:layout_height="wrap_content"
        android:background="@drawable/btnbg"
        android:text="-"
        android:textColor="@color/black"
        android:textSize="25dp" />
    <Button
        android:id="@+id/btn7"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="7"
        android:textSize="25dp"
        android:textColor="@color/black"
        android:background="@drawable/btnbg"/>
    <Button
        android:id="@+id/btn8"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="8"
        android:textSize="25dp"
        android:textColor="@color/black"
        android:background="@drawable/btnbg"/>
    <Button
        android:id="@+id/btn9"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="9"
        android:textSize="25dp"
        android:textColor="@color/black"
        android:background="@drawable/btnbg"/>

    <Button
        android:id="@+id/cheng"
        android:layout_width="76dp"
        android:layout_height="wrap_content"
        android:background="@drawable/btnbg"
        android:text="x"
        android:textAllCaps="false"
        android:textColor="@color/black"
        android:textSize="25dp" />
    <Button
        android:id="@+id/btn0"
        android:layout_width="wrap_content"
```

```
            android:layout_height="wrap_content"
            android:text="0"
            android:textSize="25dp"
            android:textColor="@color/black"
            android:background="@drawable/btnbg"/>
        <Button
            android:id="@+id/dian"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="."
            android:textSize="25dp"
            android:textColor="@color/black"
            android:background="@drawable/btnbg"/>
        <Button
            android:id="@+id/back"
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="⬅"
            android:textSize="25dp"
            android:textColor="@color/black"
            android:background="@drawable/btnbg"/>
        <Button
            android:id="@+id/chu"
            android:layout_width="76dp"
            android:layout_height="wrap_content"
            android:text="÷"
            android:textSize="25dp"
            android:textColor="@color/black"
            android:background="@drawable/btnbg"/>
    </GridLayout>

    <Button
        android:id="@+id/history_recorder"
        android:layout_width="40dp"
        android:layout_height="40dp"
        android:layout_marginBottom="10dp"
        android:background="@drawable/history_recorder"
        />
</LinearLayout>
```

2. 第二个布局文件（activity_main2）代码

```
3.      <?xml version="1.0" encoding="utf-8"?>
    <LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
```

```
        xmlns:app="http://schemas.android.com/apk/res-auto"
        xmlns:tools="http://schemas.android.com/tools"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        android:orientation="vertical"
        android:padding="10dp"
        android:background="@drawable/bg"
        >
        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:text="历史记录"
            android:textSize="40sp"
            android:textColor="@color/black"
            android:layout_marginBottom="10dp"
            android:layout_marginTop="20dp"
            android:gravity="center"/>

        <TextView
            android:id="@+id/history"
            android:layout_width="match_parent"
            android:layout_height="500dp"
            android:background="@color/white"
            android:gravity="top|left"

            android:textSize="20dp" />
    </LinearLayout>
```

2）资源文件代码

1. 按钮背景代码

```
<?xml version="1.0" encoding="utf-8"?>
<selector xmlns:android="http://schemas.android.com/apk/res/android">
<item android:state_pressed="false">
<shape xmlns:android="http://schemas.android.com/apk/res/android">
    <gradient android:startColor="#c0000000"
android:endColor="#c0000000" android:angle="90" /><!--背景颜色渐变 angle
为渐变角度-->
    <solid android:color="#00ffffff" /><!-- 背景填充颜色 -->
    <stroke android:width="1dp" android:color="@color/black" /><!-- 描
边，边框宽度、颜色 -->
    <corners android:radius="10dp" /><!-- 边角圆弧的半径 -->
    <padding android:left="3dp" android:top="3dp" android:right="3dp"
android:bottom="3dp" /><!-- 四周留出来的空白 -->
```

```
</shape>
</item>
    <item android:state_pressed="true">
        <shape
xmlns:android="http://schemas.android.com/apk/res/android">
            <gradient android:startColor="#c0000000"
android:endColor="#c0000000" android:angle="90" /><!--背景颜色渐变 angle
为渐变角度-->
            <solid android:color="#FFFFFF" /><!-- 背景填充颜色 -->
            <stroke android:width="1dp" android:color="@color/black"
/><!-- 描边，边框宽度、颜色 -->
            <corners android:radius="10dp" /><!-- 边角圆弧的半径 -->
            <padding android:left="3dp" android:top="3dp"
android:right="3dp" android:bottom="3dp" /><!-- 四周留出来的空白 -->
        </shape>
    </item>
</selector>
```

2. 背景图片

2）界面逻辑程序代码

1. 第一界面逻辑（MainActivity.java）代码

```java
package com.example.calculator;


import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.media.SoundPool;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.example.calculator.R;

import java.math.BigDecimal;

public class MainActivity extends AppCompatActivity {

    private SoundPool soundPool;//声明一个 SoundPool
    private int soundID;//创建某个声音对应的音频 ID
    public static StringBuffer ss_total = new StringBuffer();

    Button btn1, btn2, btn3, btn4, btn5, btn6, btn7, btn8, btn9, btn0;
// 数字按钮
    Button jia, jian, cheng, chu, dian, sum, clear,back;// +号
    Button history_recorder; // 查看历史记录
    TextView edit; // 显示文本


    private String ss = "";//设置一个 String 类型的变量
    private String ss1 = "1 + ";

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
        // 获取页面上的控件
        btn1 = findViewById(R.id.btn1);
        btn2 = findViewById(R.id.btn2);
        btn3 = findViewById(R.id.btn3);
        btn4 = findViewById(R.id.btn4);
        btn5 = findViewById(R.id.btn5);
```

```
        btn6 = findViewById(R.id.btn6);
        btn7 = findViewById(R.id.btn7);
        btn8 = findViewById(R.id.btn8);
        btn9 = findViewById(R.id.btn9);
        btn0 = findViewById(R.id.btn0);
        jia = findViewById(R.id.jia);
        jian = findViewById(R.id.jian);
        cheng = findViewById(R.id.cheng);
        chu = findViewById(R.id.chu);
        sum = findViewById(R.id.calculation);
        dian = findViewById(R.id.dian);
        clear = findViewById(R.id.clear);
        edit = findViewById(R.id.edit);
        back = findViewById(R.id.back);
        history_recorder = findViewById(R.id.history_recorder);


        // 按钮的单击事件
        btn1.setOnClickListener(new Click());
        btn2.setOnClickListener(new Click());
        btn3.setOnClickListener(new Click());
        btn4.setOnClickListener(new Click());
        btn5.setOnClickListener(new Click());
        btn6.setOnClickListener(new Click());
        btn7.setOnClickListener(new Click());
        btn8.setOnClickListener(new Click());
        btn9.setOnClickListener(new Click());
        btn0.setOnClickListener(new Click());
        jia.setOnClickListener(new Click());
        jian.setOnClickListener(new Click());
        cheng.setOnClickListener(new Click());
        chu.setOnClickListener(new Click());
        sum.setOnClickListener(new Click());
        dian.setOnClickListener(new Click());
        clear.setOnClickListener(new Click());
        edit.setOnClickListener(new Click());
        back.setOnClickListener(new Click());
        history_recorder.setOnClickListener(new Click());

        initSound();
    }

    private void initSound() {
        soundPool = new SoundPool.Builder().build();
```

```
        soundID = soundPool.load(this, R.raw.sound, 1);
    }
    private void playSound() {
        soundPool.play(
                soundID,
                0.1f,   //左耳道音量【0~1】
                0.5f,   //右耳道音量【0~1】
                0,     //播放优先级【0 表示最低优先级】
                0,     //循环模式【0 表示循环一次，-1 表示一直循环，其他表示数字+1
表示当前数字对应的循环次数】
                1     //播放速度【1 是正常，范围从 0~2】
        );
    }

    // 设置按钮点击后的监听
    class Click implements View.OnClickListener {
        public void onClick(View v) {
            playSound();
            switch (v.getId()) {              //switch 循环获取点击按钮后的
值
                case R.id.clear: {
                    ss = "";
                    edit.setText(ss);//在 edittext 里面显示字符串 ss
                }
                break;

                case R.id.btn0: {
                    if(ss.contains("0")) {
                        if (ss.equals("0"))
                            break;
                        if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")+3).equals("0")))
                            break;
                    }
                    ss += "0";
                    edit.setText(ss);
                }
                break;
                case R.id.btn1: {
                    if(ss.contains("0")) {
                        if (ss.equals("0"))
                            break;
                        if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")+3).equals("0")))
```

```
                        break;
                }
                ss += "1"; // ss=ss+1
                edit.setText(ss);
            }
            break;
            case R.id.btn2: {
                if(ss.contains("0")) {
                    if (ss.equals("0"))
                        break;
                    if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")+3).equals("0")))
                        break;
                }
                ss += "2";
                edit.setText(ss);
            }
            break;

            case R.id.btn3: {
                if(ss.contains("0")) {
                    if (ss.equals("0"))
                        break;
                    if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")+3).equals("0")))
                        break;
                }
                ss += "3";
                edit.setText(ss);
            }
            break;
            case R.id.btn4: {
                if(ss.contains("0")) {
                    if (ss.equals("0"))
                        break;
                    if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")+3).equals("0")))
                        break;
                }
                ss += "4";
                edit.setText(ss);
            }
            break;
            case R.id.btn5: {
```

```
                if(ss.contains("0")) {
                    if (ss.equals("0"))
                        break;
                    if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")+3).equals("0")))
                        break;
                }
                ss += "5";
                edit.setText(ss);
            }
            break;

            case R.id.btn6: {
                if(ss.contains("0")) {
                    if (ss.equals("0"))
                        break;
                    if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")+3).equals("0")))
                        break;
                }
                ss += "6";
                edit.setText(ss);
            }
            break;
            case R.id.btn7: {
                if(ss.contains("0")) {
                    if (ss.equals("0"))
                        break;
                    if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")+3).equals("0")))
                        break;
                }
                ss += "7";
                edit.setText(ss);
            }
            break;
            case R.id.btn8: {
                if(ss.contains("0")) {
                    if (ss.equals("0"))
                        break;
                    if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")+3).equals("0")))
                        break;
                }
```

```
                ss += "8";
                edit.setText(ss);
            }
            break;
            case R.id.btn9: {
                if(ss.contains("0")) {
                    if (ss.equals("0"))
                        break;
                    if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")+3).equals("0")))
                        break;
                }
                ss += "9";
                edit.setText(ss);
            }
            break;
            case R.id.dian: {
                if (ss.length() == 0) {
                    break;
                } else if (!ss.contains(" ")&&ss.contains(".")){
                    break;
                    }
                else if(ss.contains(" ")&&(ss.substring(ss.indexOf("
")).contains(".")||ss.indexOf(" ")==ss.length()-3)){
                    break;
                } else{
                    ss += ".";
                    edit.setText(ss);
                }
            }
            break;
            case R.id.jia: {
                if (ss.length() == 0) {
//                    edit.setText("0 + ");
//                    ss = "0 + ";
                    break;
                }
                if (ss.contains(" ")) {
                    if (ss.indexOf(" ") == ss.length() - 3)
                        break;
                    getResult();
                }

                ss += " + ";
```

```
                edit.setText(ss);
            }
            break;
            case R.id.jian: {
                if (ss.length() == 0) {
                    edit.setText("0 - ");
                    ss = "0 - ";
                    break;
                }
                if (ss.contains(" ")) {
                    if (ss.indexOf(" ") == ss.length() - 3 )
                        break;
                    getResult();
                }

                ss += " - ";
                edit.setText(ss);
            }
            break;
            case R.id.cheng: {
                if (ss.length() == 0) {
                    edit.setText("0 × ");
                    ss = "0 × ";
                    break;
                }
                if (ss.contains(" ")) {
                    if (ss.indexOf(" ") == ss.length() - 3)
                        break;
                    getResult();
                }

                ss += " × ";
                edit.setText(ss);
            }
            break;
            case R.id.chu: {
                if (ss.length() == 0) {
                    edit.setText("0 ÷ ");
                    ss = "0 ÷ ";
                    break;
                }
                if (ss.contains(" ")) {
                    if (ss.indexOf(" ") == ss.length() - 3)
                        break;
```

```
                    getResult();
                }

                ss += " ÷ ";
                edit.setText(ss);
            }
            break;
            case R.id.calculation:

                getResult();

                break;
            case R.id.back: {
                if (ss.length() == 0) {
                    break;
                }
                if (ss.contains(" ")){
                    if (ss.indexOf(" ") == ss.length() - 3) {
                        ss = ss.substring(0, ss.length() - 3);
                        edit.setText(ss);
                        break;
                    }
                }

                ss = ss.substring(0, ss.length() - 1);
                edit.setText(ss);
                break;
            }
            case R.id.history_recorder: {
                Intent mIntent = new Intent(MainActivity.this,
MainActivity2.class);
                startActivity(mIntent);
                break;
            }
        }
    }


}


    private void getResult() {
        BigDecimal result = BigDecimal.ZERO;
        if (ss == null || ss.equals("")) return;
```

```
        if (!ss.contains(" ")) return;
        String s1 = ss.substring(0, ss.indexOf(" "));
        String op = ss.substring(ss.indexOf(" ") + 1, ss.indexOf(" ") +
2);
        String s2 = ss.substring(ss.indexOf(" ") + 3);
        ss_total.append(ss);
        if (!s1.equals("") && !s2.equals("")) {
//            float d1 = Float.parseFloat(s1);
//            float d2 = Float.parseFloat(s2);
            BigDecimal d1 = new BigDecimal(s1);
            BigDecimal d2 = new BigDecimal(s2);
            switch (op) {
                case "+":
                    result = d1.add(d2);
                    break;
                case "-":
                    result = d1.subtract(d2);
                    break;
                case "×":
                    result = d1.multiply(d2);
                    break;
                case "÷": {
                    if (d2.compareTo(BigDecimal.ZERO)==0) {
                        Toast.makeText(MainActivity.this, "不能除以零",
Toast.LENGTH_SHORT).show();
                        break;
                    }
                    result = d1.divide(d2,20,BigDecimal.ROUND_HALF_UP);
                }
                break;
            }
                ss = result.stripTrailingZeros().toPlainString(); // 将
result 后面的 0 去掉
                edit.setText(ss);
        }
        ss_total.append("=");
        ss_total.append(ss);
        ss_total.append("\r\n");
    }
}
```

2. 第二界面逻辑（MainActivity2.java）代码

```java
package com.example.calculator;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;
import android.widget.TextView;

public class MainActivity2 extends AppCompatActivity {
    TextView history; // 显示文本
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main2);
        history = findViewById(R.id.history);
        history.setText(MainActivity.ss_total);
    }
}
```