实验一 基础控件使用——计算器APP实现

——陈洁怡+201906061703

一、基础实验——基础控件使用

（一）实验目的

1. 熟悉Android工程结构，掌握使用Android Studio开发Android应用的基本步骤；
2. 掌握Android基本控件的使用方法；
3. 掌握Activity布局运用、控件事件编写；
4. 掌握Android界面优化方法。

（二）基本知识与原理

1. Android应用采用MVC界面设计模式，通过布局文件、资源文件与程序逻辑代码的分离，实现界面设计与逻辑开发的并行进行。
2. 分析计算器APP界面特点，选择合理的布局方式，设计并实现计算器APP基本功能。

（三）实验内容及步骤

1、在build：gradle中的dependencies中加入如下语句，使得项目能够支持JDK 底层调用 javaScript 的运算公式 :

|  |
| --- |
| implementation 'io.apisense:rhino-android:1.1.1' |

同时在mainActivity.java中加入如下语句 :

|  |
| --- |
| import javax.script.ScriptEngine;  import javax.script.ScriptEngineManager;  import javax.script.ScriptException; |

2、设置计算器的前端界面，在activity\_main.xml 中加入如下语句

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  xmlns:grid="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"  tools:context=".MainActivity"  tools:ignore="ExtraText">     <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content">   <EditText  android:id="@+id/et\_m"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:background="@null"  android:textColor="@color/black"  android:lines="1"  android:textSize="25sp" />  </LinearLayout>   <LinearLayout  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content">   <EditText  android:id="@+id/et"  android:layout\_width="match\_parent"  android:layout\_height="wrap\_content"  android:background="@null"  android:textColor="@color/black"  android:lines="2"  android:textSize="65sp" />  </LinearLayout>   <GridLayout  android:layout\_width="match\_parent"  android:layout\_height="match\_parent"  android:columnCount="4"  android:orientation="horizontal"  android:rowCount="9"  android:useDefaultMargins="true">   <Button  android:id="@+id/bt\_mc"  android:layout\_row="0"  android:layout\_rowWeight="1"  android:layout\_column="0"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/mc"  android:textColor="@color/black"  android:textSize="25dp" />   <Button  android:id="@+id/bt\_mPlus"  android:layout\_width="wrap\_content"  android:layout\_row="0"  android:layout\_rowWeight="1"  android:layout\_column="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/m"  android:textColor="@color/black"  android:textSize="25dp" />   <Button  android:id="@+id/bt\_mMinus"  android:layout\_row="0"  android:layout\_rowWeight="1"  android:layout\_column="2"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/m2"  android:textColor="@color/black"  android:textSize="25dp" />   <Button  android:id="@+id/bt\_mr"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/mr"  android:textColor="@color/black"  android:textSize="25dp" />   <Button  android:id="@+id/bt\_c"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_blue1"  android:text="@string/c"  android:textColor="#3F51B5"  android:textSize="35dp" />   <Button  android:id="@+id/bt\_division"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_blue1"  android:text="@string/division"  android:textColor="#3F51B5"  android:textSize="45dp" />   <Button  android:id="@+id/bt\_product"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_blue1"  android:text="@string/product"  android:textColor="#3F51B5"  android:textSize="45dp" />   <ImageButton  android:id="@+id/bt\_back"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_blue1"  android:gravity="center"  android:scaleType="fitCenter"  android:paddingStart="25dp"  android:paddingEnd="25dp"  android:src="@drawable/backspace"  />   <Button  android:id="@+id/bt\_7"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/\_7"  android:textColor="@color/black"  android:textSize="30dp" />   <Button  android:id="@+id/bt\_8"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/\_8"  android:textColor="@color/black"  android:textSize="30dp" />   <Button  android:id="@+id/bt\_9"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/\_9"  android:textColor="@color/black"  android:textSize="30dp" />   <Button  android:id="@+id/bt\_minus"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_blue1"  android:text="@string/minus"  android:textColor="#3F51B5"  android:textSize="35dp" />   <Button  android:id="@+id/bt\_4"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/\_4"  android:textColor="@color/black"  android:textSize="30dp" />   <Button  android:id="@+id/bt\_5"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/\_5"  android:textColor="@color/black"  android:textSize="30dp" />   <Button  android:id="@+id/bt\_6"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/\_6"  android:textColor="@color/black"  android:textSize="30dp" />   <Button  android:id="@+id/bt\_plus"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_blue1"  android:text="@string/plus"  android:textColor="#3F51B5"  android:textSize="35dp" />   <Button  android:id="@+id/bt\_1"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/\_1"  android:textColor="@color/black"  android:textSize="30dp" />   <Button  android:id="@+id/bt\_2"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/\_2"  android:textColor="@color/black"  android:textSize="30dp" />   <Button  android:id="@+id/bt\_3"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/\_3"  android:textColor="@color/black"  android:textSize="30dp" />   <Button  android:id="@+id/bt\_equals"  android:layout\_rowSpan="2"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_blue2"  android:text="@string/equals"  android:textColor="@color/white"  android:textSize="50dp" />   <Button  android:id="@+id/bt\_percent"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/percent"  android:textColor="@color/black"  android:textSize="33dp" />   <Button  android:id="@+id/bt\_0"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/\_0"  android:textColor="@color/black"  android:textSize="30dp" />   <Button  android:id="@+id/bt\_dot"  android:layout\_rowWeight="1"  android:layout\_columnWeight="1"  android:layout\_gravity="fill"  android:background="@drawable/selector\_gray"  android:text="@string/dot"  android:textColor="@color/black"  android:textSize="35dp" />   </GridLayout> </LinearLayout> |

3、对标华为p30的计算机样式，创建三种按钮样式，灰色按钮，黑色字体，按下时背景为更深的灰色；淡蓝色按钮，深蓝色字体，按下时背景为更深的蓝色；深蓝色按钮，白色字体，按下时为更深的蓝色

在res->drawable new->drawable resourse file , 为灰色按钮不按下时的背景样式light\_gray.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?>  <shape xmlns:android="http://schemas.android.com/apk/res/android">  <corners  android:radius="15dp"/>  <solid  android:color="#E1E1E1"/>  </shape> |

在res->drawable new->drawable resourse file , 为灰色按钮按下时的背景样式drak\_gray.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <shape xmlns:android="http://schemas.android.com/apk/res/android">  <corners  android:radius="15dp"/>  <solid  android:color="#A3A3A3"/> </shape> |

在res->drawable new->drawable resourse file ,设置按钮不按时和按下时的样式selector\_gray.xml

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <selector xmlns:android="http://schemas.android.com/apk/res/android">  <item android:drawable="@drawable/dark\_gray"  android:state\_pressed="true"/>  <item android:drawable="@drawable/light\_gray"/> </selector> |

同理创建light\_blue1.xml为淡蓝色按钮不按下时的样式

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <shape xmlns:android="http://schemas.android.com/apk/res/android">  <corners  android:radius="15dp"/>  <solid  android:color="#B8D0F4"/> </shape> |

创建drak\_blue1.xml为淡蓝色按钮按下时的样式

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <shape xmlns:android="http://schemas.android.com/apk/res/android">  <corners  android:radius="15dp"/>  <solid  android:color="#7FA8E6"/> </shape> |

创建selector\_blue1.xml为淡蓝色按钮按钮不按时和按下时的样式

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <selector xmlns:android="http://schemas.android.com/apk/res/android">  <item android:drawable="@drawable/dark\_blue1"  android:state\_pressed="true"/>  <item android:drawable="@drawable/light\_blue1"/> </selector> |

同理创建light\_blue2.xml为深蓝色按钮不按下时的样式

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <shape xmlns:android="http://schemas.android.com/apk/res/android">  <corners  android:radius="15dp"/>  <solid  android:color="#3F51B5"/> </shape> |

创建drak\_blue1.xml为淡蓝色按钮按下时的样式

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <shape xmlns:android="http://schemas.android.com/apk/res/android">  <corners  android:radius="15dp"/>  <solid  android:color="#144CA1"/> </shape> |

创建selector\_blue2.xml为淡蓝色按钮按钮不按时和按下时的样式

|  |
| --- |
| <?xml version="1.0" encoding="utf-8"?> <selector xmlns:android="http://schemas.android.com/apk/res/android">  <item android:drawable="@drawable/dark\_blue2"  android:state\_pressed="true"/>  <item android:drawable="@drawable/light\_blue2"/> </selector> |

4、在计算值的时候直接使用了现成的函数，但该函数只能计算表达式格式正确时表达式的值，同时考虑到一些实际情况（手机自带计算器的使用情况），最后基本逻辑如下 ：输入显示；可以退格更改；只有当按下“=”键或与“M+”“M-”等时才计算当前显示出来的表达式的值；不可以使用软键盘，只能按屏幕上的按钮（限制输入）；限制表达式长度为18，对于会使表达式长度超过该数值的操作作忽略处理；对于符号，连续的若干个运算符会构成非法表达式，在这里仿照手机自带计算器的业务逻辑，对连续运算符，退格更改；减号具有二义性，特判；不可以有前导零，不可以有连续的小数点；以上情况直接在输入时处理，对于其他的非法情况（比如百分号在两个数字中间，一个数中有连续的小数点等情况），显示“error”。

|  |
| --- |
| package com.example.androidtest2;  import androidx.appcompat.app.AppCompatActivity;  import android.annotation.SuppressLint; import android.os.Bundle; import android.util.Log; import android.view.View; import android.view.WindowManager; import android.widget.Button; import android.widget.EditText; import android.widget.ImageButton;  import javax.script.ScriptEngine; import javax.script.ScriptEngineManager; import javax.script.ScriptException;   public class MainActivity extends AppCompatActivity implements View.OnClickListener {   ScriptEngine jse = new ScriptEngineManager().getEngineByName("javascript");   Button bt\_mc, bt\_mPlus, bt\_mMinus, bt\_mr;  Button bt\_division, bt\_product, bt\_minus, bt\_plus, bt\_equals;  Button bt\_0, bt\_1, bt\_2, bt\_3, bt\_4, bt\_5, bt\_6, bt\_7, bt\_8, bt\_9;  Button bt\_percent, bt\_dot, bt\_c;  ImageButton bt\_back;  EditText et\_m, et;   boolean is\_m = false;   public void init() {  bt\_mc = findViewById(R.id.bt\_mc);  bt\_mPlus = findViewById(R.id.bt\_mPlus);  bt\_mMinus = findViewById(R.id.bt\_mMinus);  bt\_mr = findViewById(R.id.bt\_mr);   bt\_division = findViewById(R.id.bt\_division);  bt\_product = findViewById(R.id.bt\_product);  bt\_minus = findViewById(R.id.bt\_minus);  bt\_plus = findViewById(R.id.bt\_plus);  bt\_equals = findViewById(R.id.bt\_equals);   bt\_0 = findViewById(R.id.bt\_0);  bt\_1 = findViewById(R.id.bt\_1);  bt\_2 = findViewById(R.id.bt\_2);  bt\_3 = findViewById(R.id.bt\_3);  bt\_4 = findViewById(R.id.bt\_4);  bt\_5 = findViewById(R.id.bt\_5);  bt\_6 = findViewById(R.id.bt\_6);  bt\_7 = findViewById(R.id.bt\_7);  bt\_8 = findViewById(R.id.bt\_8);  bt\_9 = findViewById(R.id.bt\_9);   bt\_percent = findViewById(R.id.bt\_percent);  bt\_dot = findViewById(R.id.bt\_dot);  bt\_c = findViewById(R.id.bt\_c);   bt\_back = findViewById(R.id.bt\_back);   et\_m = findViewById(R.id.et\_m);  et = findViewById(R.id.et);   // set onClickListener  bt\_division.setOnClickListener(this);  bt\_product.setOnClickListener(this);  bt\_minus.setOnClickListener(this);  bt\_plus.setOnClickListener(this); // bt\_equals.setOnClickListener(this); //!!   bt\_0.setOnClickListener(this);  bt\_1.setOnClickListener(this);  bt\_2.setOnClickListener(this);  bt\_3.setOnClickListener(this);  bt\_4.setOnClickListener(this);  bt\_5.setOnClickListener(this);  bt\_6.setOnClickListener(this);  bt\_7.setOnClickListener(this);  bt\_8.setOnClickListener(this);  bt\_9.setOnClickListener(this);   bt\_percent.setOnClickListener(this);  bt\_dot.setOnClickListener(this);  bt\_c.setOnClickListener(this);   bt\_back.setOnClickListener(this);  }    @SuppressLint({"SetTextI18n", "NonConstantResourceId"})  @Override  public void onClick(View v) {  String textContent = et.getText().toString();  if (textContent.equals("error")) {  et.setText("");  textContent = "";  }  int len = textContent.length();  char lastChar = len == 0 ? '#' : textContent.charAt(len - 1);  char last2Char = len <= 1 ? '#' : textContent.charAt(len - 2);  switch (v.getId()) {  case R.id.bt\_0:  if (len == 18) return;  if (len == 1 && lastChar == '0') return;  case R.id.bt\_1:  case R.id.bt\_2:  case R.id.bt\_3:  case R.id.bt\_4:  case R.id.bt\_5:  case R.id.bt\_6:  case R.id.bt\_7:  case R.id.bt\_8:  case R.id.bt\_9:  if (len == 18) return;  if (len == 1 && lastChar == '0') et.setText(((Button) v).getText());  else et.setText(textContent + ((Button) v).getText());  break;  case R.id.bt\_division:  if (len == 18) return;  if (lastChar == '÷') return;  if (lastChar == '×' || lastChar == '-' || lastChar == '+') {  if (!(last2Char == '+' || last2Char == '-' || last2Char == '×' || last2Char == '÷')) {  String tmp = textContent.substring(0, len - 1);  et.setText(tmp + '÷');  }  return;  }  et.setText(textContent + "÷");  break;  case R.id.bt\_product:  if (len == 18) return;  if (lastChar == '×') return;  if (lastChar == '÷' || lastChar == '-' || lastChar == '+') {  if (!(last2Char == '+' || last2Char == '-' || last2Char == '×' || last2Char == '÷')) {  String tmp = textContent.substring(0, len - 1);  et.setText(tmp + '×');  }  return;  }  et.setText(textContent + "×");  break;  case R.id.bt\_minus:  if (len == 18) return;  if (lastChar == '-') return;  if (lastChar == '+') {  String tmp = textContent.substring(0, len - 1);  et.setText(tmp + '-');  return;  }  if (!(last2Char == '+' || last2Char == '-' || last2Char == '×' || last2Char == '÷')) {  et.setText(textContent + "-");  }  break;  case R.id.bt\_plus:  if (len == 18) return;  if (lastChar == '+') return;  if (lastChar == '÷' || lastChar == '-' || lastChar == '×') {  if (!(last2Char == '+' || last2Char == '-' || last2Char == '×' || last2Char == '÷')) {  String tmp = textContent.substring(0, len - 1);  et.setText(tmp + '+');  }  return;  }  et.setText(textContent + "+");  break;  case R.id.bt\_percent:  if (len == 18) return;  et.setText(textContent + "%");  break;  case R.id.bt\_dot:  if (len == 18) return;  if (lastChar == '.') return;  et.setText(textContent + ".");  break;  case R.id.bt\_back:  int tmplen = textContent.length() - 1;  if (tmplen <= 0) {  et.setText("");  } else {  String tmp = textContent.substring(0, tmplen);  et.setText(tmp);  }  break;  case R.id.bt\_c:  et.setText("");  break;  }  }   private static final String debug = "debug";   boolean isExpReasonable(String exp) {  String[] tmp = exp.split("\\+|-|×|÷");  for (String s : tmp) {  Log.d(debug, s);  int cntDot = 0;  int len = s.length();  if (s.charAt(len - 1) == '%') len--;  for (int i = 0; i < len; ++i) {  if (s.charAt(i) == '%') return false;  if (s.charAt(i) == '.') cntDot++;  }  if (cntDot > 1) return false;  }  return true;  }   double changeNumber() {  String s = et.getText().toString();  if(!isExpReasonable(s)) {  et.setText("error");  return -1;  }  s.replace('÷', '/');  s.replace('×', '\*');  String t = "";  for (int i = 0; i < s.length(); ++i) {  if (s.charAt(i) == '÷') t = t + '/';  else if (s.charAt(i) == '×') t = t + '\*';  else t = t + s.charAt(i);  }  String ans = "0";  Log.d(debug, s);  Log.d(debug, t);  s = t;   try {  ans = jse.eval(s).toString();  Log.d(debug, ans);  et.setText(ans);  } catch (ScriptException e) {  e.printStackTrace();  }  return Double.parseDouble(ans);  }   double number = 0;   @Override  protected void onCreate(Bundle savedInstanceState) {  getWindow().setSoftInputMode(WindowManager.LayoutParams.SOFT\_INPUT\_STATE\_ALWAYS\_HIDDEN);  super.onCreate(savedInstanceState);   setContentView(R.layout.activity\_main);   init();    et.setEnabled(false);  et\_m.setEnabled(false);   bt\_mPlus.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View view) {  if (!is\_m) {  is\_m = true;  number = 0;  et\_m.setText("M");  }   double tmpNum = changeNumber();  if (!et.getText().toString().equals("error")) number = number + tmpNum;  else number = 0;  }  });   bt\_mMinus.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View view) {  if (!is\_m) {  is\_m = true;  number = 0;  et\_m.setText("M");  }  double tmpNum = changeNumber();  if (!et.getText().toString().equals("error")) number = number - tmpNum;  else number = 0;  }  });   bt\_mc.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View view) {  is\_m = false;  number = 0;  et\_m.setText("");  }  });   bt\_mr.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View view) {  if (!is\_m) return;  String s = "" + number;  String context = et.getText().toString();  context = context + s;  if (context.length() <= 18) et.setText(context);  }  });   bt\_equals.setOnClickListener(new View.OnClickListener() {  @Override  public void onClick(View view) {  changeNumber();  }  });  } } |

1. 运行结果截图

1、页面布局情况

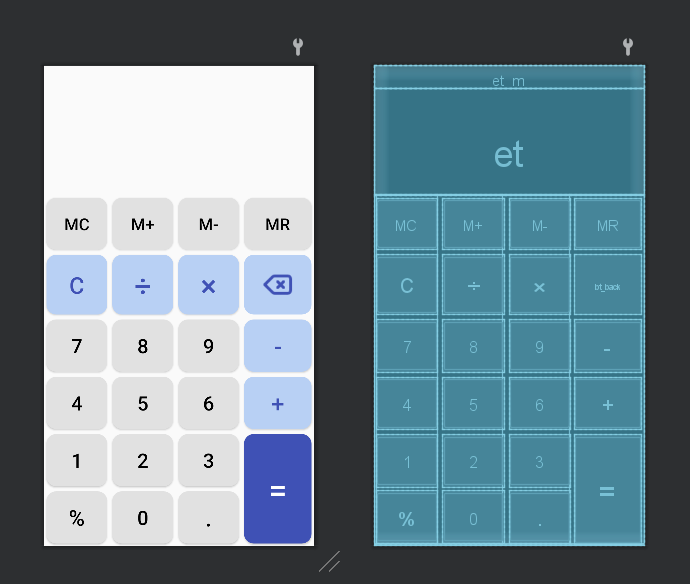


图4-1 页面布局情况

2、运行结果截图



图4-2 运行结果截图

3、输入表达式



图4-3 输入表达式

1. 4、刚才表达式运算结果（loacat debug显示截图）

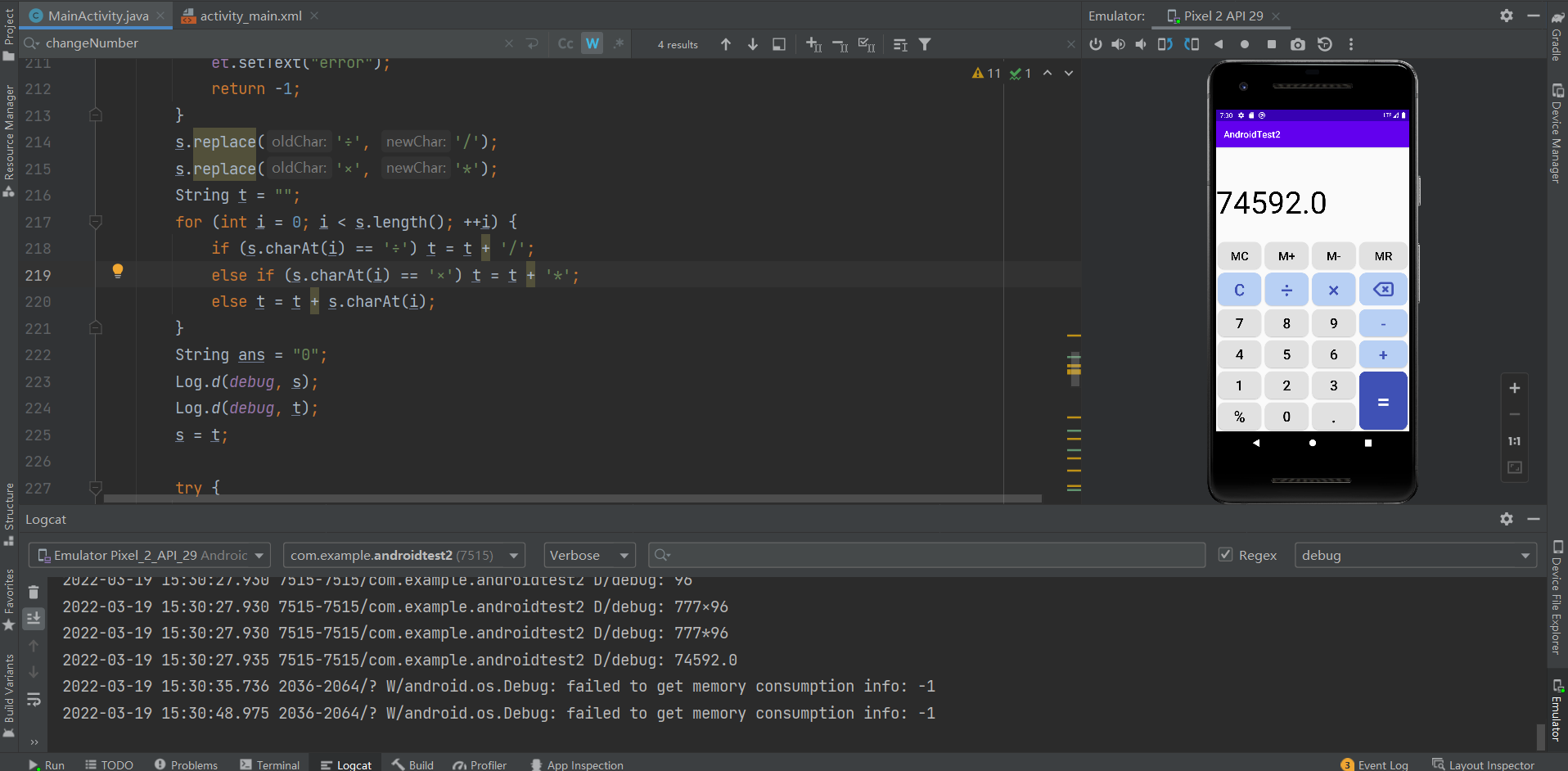


图4-4 表达式运算结果

5、使用M+ 和MR的结果



图4-5 使用M+和MR的结果

1. 非法表达式的情况

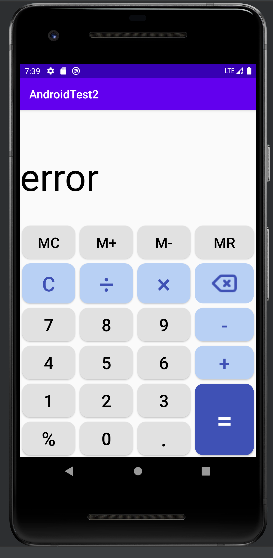
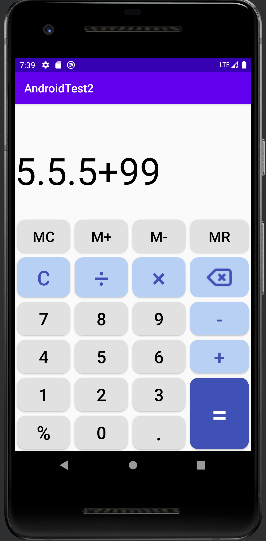


图4-6 非法表达式的情况

1. 连按运算符的情况（只有-可以和别的共存，但也最多只有一个减号；不然连按会变成最后按下的符号）



图4-7 连按运算符的情况

（五）实验收获及总结

刚开始尝试只使用gridlayout仿照手机自带的计算器做这次实验，但发现布局并不能全屏显示，经过与网上代码比对并没有发现明显问题，与老师沟通交流并更正查阅资料的方式，发现问题是由于gridlayout需要配合属性设置权重，同时只设置一部分控件的权重属性时并不显示布局变化，对layout\_width等属性有了更深刻的理解，同时深刻感到直接使用gridlayout的不妥，将布局更正为linearlayout和gridlayout的结合，通过该次实验对布局设计与具体实现有了更深刻的理解。

另外在业务逻辑上，因为是仿照自己手机（华为P30）自带计算器做的，主要思想是通过输入字符显示，按下等于键的时候计算显示在屏幕上的这串字符串的运算值，计算值直接套用了JDK 底层调用 javaScript 的运算公式，但这个公式只能处理正确的输入的情况，因此在判断输入正确与否上写了比较多的代码，同时这里尽量仿照手机自带的计算器去实现一些例如连续输入运算符时的替换、“-”可以减号和负号二义性的处理，连续输入0时前导零的处理等，考虑多种不同的输入顺序情况，同时在遇到不合法的输入时尽量直接对字符串进行更正，最后计算时先判断字符串的合法性再交由javaScript 的运算公式计算，同时在这里显示和计算用了不同的字符（×和\*，÷和/）。这部分内容锻炼了我的观察设计能力和业务逻辑代码实现能力。

同时使用了java自带的split函数减少代码量，过程中了解到+在正则中有不同的意义，需要转义，并学会了使用log.d()结合catlog进行debug。