ITE-5333-0NB-Android

Instructor: Ammar Al-Qaraghuli

Midterm - F19

Student Name: Kien Tran

Student ID: N01260692

APP SCREEN SHOOTS

|  |  |  |
| --- | --- | --- |
|  |  |  |
|  |  |  |
|  |  |  |

**SOURCE CODE**

**MainActivity.java**

package com.kientran.week04;  
  
import androidx.appcompat.app.AppCompatActivity;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.TextView;  
  
public class MainActivity extends AppCompatActivity {  
  
 private TextView txtCalcHistory;  
 private TextView txtResult;  
  
 private CalculatorEngine calcEngine;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 try  
 {  
 this.getSupportActionBar().hide();  
 }  
 catch (NullPointerException e){}  
 setContentView(R.layout.*activity\_main*);  
  
 txtCalcHistory = findViewById(R.id.*txt\_all*);  
 txtResult = findViewById(R.id.*txt\_result*);  
  
 calcEngine = new CalculatorEngine();  
 }  
  
 public void buttonPressed(View view){  
  
 calcEngine.calc(String.*valueOf*(((Button) view).getText()));  
 txtCalcHistory.setText(calcEngine.getCalcHistory());  
  
 String resultLine = calcEngine.getCalcResult();  
 if(resultLine.length() > 8){  
 txtResult.setTextSize(50);  
 } else {  
 txtResult.setTextSize(90);  
 }  
 txtResult.setText(calcEngine.getCalcResult());  
 }  
}

**CalculatorEngine.java**

package com.kientran.week04;  
  
  
public class CalculatorEngine {  
  
 final static String *ZERO* = "0";  
 final static String *PLUS\_SIGN* = "+";  
 final static String *MINUS\_SIGN* = "\u2212";  
 final static String *MULTIPLY\_SIGN* = "\u00D7";  
 final static String *DEVIDE\_SIGN* = "\u00F7";  
 final static String *PERCENT\_SIGN* = "%";  
 final static String *PLUS\_OR\_MINUS\_SIGN* = "+/-";  
 final static String *CLEAR\_SIGN* = "AC";  
 final static String *EQUAL\_SIGN* = "=";  
 final static String *DOT\_SIGN* = ".";  
 final static String *SQUARE\_ROOT\_SIGN* = "\u221a";  
 final static String *MEMORY\_ADD* = "M+";  
 final static String *MEMORY\_SUBTRACT* = "M-";  
 final static String *MEMORY\_RECALL* = "MR";  
 final static String *MEMORY\_CLEAR* = "MC";  
  
  
 private String calcResult = "";  
 private String calcHistory = "";  
 private String storedOperator = "";  
 private String lastInputtedKey = "";  
  
 private Double result = new Double(0);  
 private Double memory = new Double(0);  
  
  
 private Boolean isCalculated = false;  
 private int historyLineLength;  
 private int maxInputLength;  
  
 public CalculatorEngine() {  
 historyLineLength = 80;  
 maxInputLength = 28;  
 }  
  
 public CalculatorEngine(int historyLineLength, int maxInputLength) {  
 this.historyLineLength = historyLineLength;  
 this.maxInputLength = maxInputLength;  
 }  
  
 public void calc(String value) {  
  
 switch (value) {  
 case *CLEAR\_SIGN*:  
 calcHistory = "";  
 calcResult = "";  
 result = new Double(0);  
 storedOperator = "";  
 isCalculated = false;  
 break;  
 case *PERCENT\_SIGN*:  
 calculatePercentage();  
 break;  
 case *SQUARE\_ROOT\_SIGN*:  
 calculateSquareRoot();  
 break;  
 case *MEMORY\_CLEAR*:  
 memory = new Double(0);  
 break;  
 case *MEMORY\_RECALL*:  
 if (lastInputtedKey.equals(*EQUAL\_SIGN*)) {  
 calcHistory = " | MR: " + getDisplayText(memory) + " | ";  
 calcResult = getDisplayText(memory);  
 storedOperator = "";  
 result = memory;  
 isCalculated = false;  
 } else {  
 if (isCalculated == true) {  
 isCalculated = false;  
   
 }  
 calcResult = getDisplayText(memory);  
 calcHistory += " | MR: " + getDisplayText(memory) + " | ";  
 }  
  
  
 break;  
 case *MEMORY\_ADD*:  
 memoryOperation(true);  
 break;  
 case *MEMORY\_SUBTRACT*:  
 memoryOperation(false);  
 break;  
 case *DEVIDE\_SIGN*:  
 case *MULTIPLY\_SIGN*:  
 case *PLUS\_SIGN*:  
 case *MINUS\_SIGN*:  
 calculateOnOperator(value);  
 break;  
 case *EQUAL\_SIGN*:  
 calculateOnEqual();  
 break;  
 case *PLUS\_OR\_MINUS\_SIGN*:  
 if (!calcResult.isEmpty()) {  
  
 if (calcResult.indexOf("-") == -1) {  
 calcResult = "-" + calcResult;  
 } else {  
 calcResult = calcResult.substring(1);  
 }  
 }  
 break;  
 case *DOT\_SIGN*:  
 if (!calcResult.isEmpty() && isCalculated == false && calcResult.indexOf(".") == -1 && calcResult.length() < maxInputLength) {  
 calcHistory += value;  
 calcResult += value;  
 }  
 break;  
 default:  
 if (lastInputtedKey.equals(*EQUAL\_SIGN*)) {  
 calcHistory = value;  
 calcResult = value;  
 storedOperator = "";  
 result = new Double(0);  
 isCalculated = false;  
 } else if (isCalculated == true) {  
 isCalculated = false;  
 calcResult = value;  
 calcHistory += value;  
 } else {  
 if (calcResult.length() < maxInputLength && (!value.equals(*ZERO*) || (value.equals(*ZERO*) && !calcResult.equals(*ZERO*)))) {  
 calcResult += value;  
 calcHistory += value;  
 }  
 }  
  
 break;  
 }  
  
 lastInputtedKey = value;  
  
 }  
  
 private void memoryOperation(boolean isAdd) {  
 if (!calcResult.isEmpty()) {  
 String value = calcResult;  
 if (!isAdd) {  
 value = "-" + value;  
 calcHistory += " | M-: " + calcResult + " | ";  
 } else {  
 calcHistory += " | M+: " + calcResult + " | ";  
 }  
  
 memory = memory + Double.*valueOf*(value);  
 }  
 }  
  
 private void calculatePercentage() {  
 if (!calcResult.isEmpty()) {  
 result = (new Double(calcResult)) \* 0.01;  
  
 calcResult = getDisplayText(result);  
 calcHistory = removeLastOperator(calcHistory) + *PERCENT\_SIGN* + *EQUAL\_SIGN* + getDisplayText(result);  
 isCalculated = true;  
 storedOperator = "";  
 }  
 }  
  
 private void calculateSquareRoot() {  
 if (!calcResult.isEmpty()) {  
 Double num = new Double((calcResult));  
 if (num > 0) {  
 result = Math.*sqrt*(num);  
  
 }  
 calcResult = getDisplayText(result);  
 calcHistory = removeLastOperator(calcHistory) + *SQUARE\_ROOT\_SIGN* + *EQUAL\_SIGN* + getDisplayText(result);  
 isCalculated = true;  
 storedOperator = "";  
 }  
 }  
  
 private void calculateOnEqual() {  
 if (!calcResult.isEmpty() && !storedOperator.isEmpty()) {  
  
 if (isCalculated) {  
  
 Double second = new Double(result);  
  
 result = calculate(result, second, storedOperator);  
  
 calcHistory = removeLastOperator(calcHistory) + storedOperator + getDisplayText(second) + *EQUAL\_SIGN* + getDisplayText(result);  
 } else {  
  
 result = calculate(result, new Double(calcResult), storedOperator);  
  
  
 calcHistory = calcHistory + *EQUAL\_SIGN* + getDisplayText(result);  
  
 }  
 calcResult = getDisplayText(result);  
 isCalculated = true;  
  
 }  
 }  
  
 private void calculateOnOperator(String operator) {  
 if (!calcResult.isEmpty()) {  
 if (storedOperator.isEmpty()) {  
 result = new Double(calcResult);  
 calcHistory = calcHistory + operator;  
 isCalculated = true;  
 } else {  
 if (isCalculated) {  
 calcHistory = removeLastOperator(calcHistory) + operator;  
 } else {  
 result = calculate(result, new Double(calcResult), storedOperator);  
 isCalculated = true;  
 calcHistory = calcHistory + "=" + getDisplayText(result) + operator;  
 calcResult = getDisplayText(result);  
 }  
 }  
  
 storedOperator = operator;  
 }  
  
 }  
  
 private String removeLastOperator(String value) {  
 if (value.length() > 0) {  
 String lastChar = value.substring(value.length() - 1);  
 try {  
 Integer.*valueOf*(lastChar);  
 } catch (NumberFormatException e) {  
 return value.substring(0, value.length() - 1);  
 }  
 }  
 return value;  
  
 }  
  
 private Double calculate(Double first, Double second, String operator) {  
 Double result;  
  
 result = new Double(0);  
  
 switch (operator) {  
 case *PLUS\_SIGN*:  
 result = first + second;  
 break;  
 case *MINUS\_SIGN*:  
 result = first - second;  
 break;  
 case *MULTIPLY\_SIGN*:  
 result = first \* second;  
 break;  
 case *DEVIDE\_SIGN*:  
 result = first / second;  
 break;  
 default:  
 break;  
 }  
  
 return result;  
 }  
  
 public String getDisplayText(Double num) {  
  
 long iPart;  
 double fPart;  
  
 iPart = (long) num.doubleValue();  
 fPart = num - iPart;  
  
 if (fPart == 0.0) {  
 return String.*valueOf*(num.longValue());  
 } else {  
  
 return num.toString();  
 }  
  
 }  
  
 public String getCalcResult() {  
 return calcResult;  
 }  
  
 public String getCalcHistory() {  
 if (calcHistory.length() <= historyLineLength) {  
 return calcHistory;  
 } else {  
  
 return calcHistory.substring(calcHistory.length() - historyLineLength);  
 }  
 }  
  
  
}

**CustomizedLinearLayout.java**

package com.kientran.week04;  
  
import android.content.Context;  
import android.util.AttributeSet;  
import android.widget.LinearLayout;  
  
import androidx.annotation.Nullable;  
  
public class CustomizedLinearLayout extends LinearLayout {  
  
 public CustomizedLinearLayout(Context context) {  
 super(context);  
 }  
  
 public CustomizedLinearLayout(Context context, @Nullable AttributeSet attrs) {  
 super(context, attrs);  
 }  
  
 public CustomizedLinearLayout(Context context, @Nullable AttributeSet attrs, int defStyleAttr) {  
 super(context, attrs, defStyleAttr);  
 }  
  
 public CustomizedLinearLayout(Context context, AttributeSet attrs, int defStyleAttr, int defStyleRes) {  
 super(context, attrs, defStyleAttr, defStyleRes);  
 }  
  
 @Override  
 protected void onMeasure(int widthMeasureSpec, int heightMeasureSpec) {  
 //h = w/4\*5 (AR) = 4.0/5  
  
 double aspectR = 4.0/5;  
 int w = MeasureSpec.*getSize*(widthMeasureSpec);  
 int h = MeasureSpec.*getSize*(heightMeasureSpec);  
  
 int newHeight = (int) ((w-getPaddingLeft()-getPaddingRight())\*1.0/aspectR);  
  
 super.onMeasure(widthMeasureSpec, MeasureSpec.*makeMeasureSpec*(newHeight, MeasureSpec.*EXACTLY*));  
 }  
}

**main\_activity.xml**

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout 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"  
 tools:context=".MainActivity"  
 >  
  
 <LinearLayout  
 android:id="@+id/vll1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:orientation="vertical"  
 app:layout\_constraintBottom\_toTopOf="@+id/vll2"  
 app:layout\_constraintTop\_toTopOf="parent"  
 android:padding="5px"  
 android:background="@drawable/bodered\_linear">  
  
 <TextView  
 android:id="@+id/txt\_all"  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:layout\_weight="3"  
 android:background="@drawable/bodered\_textview"  
 android:textColor="@color/result\_text"  
 android:textSize="@dimen/calculator\_accumulate\_line\_text\_size"  
 android:gravity="end"  
 android:fontFamily="sans-serif-light"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <TextView  
 android:id="@+id/txt\_result"  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:layout\_weight="7"  
 android:background="@drawable/bodered\_textview"  
 android:textColor="@color/result\_text"  
 android:gravity="bottom|end"  
 android:textSize="@dimen/calculator\_line\_text\_size"  
 android:fontFamily="sans-serif-light"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent" />  
 </LinearLayout>  
  
 <com.kientran.week04.CustomizedLinearLayout  
 android:id="@+id/vll2"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:background="@drawable/bodered\_linear"  
 android:orientation="vertical"  
 android:paddingLeft="15px"  
 android:paddingRight="15px"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/vll1">  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:layout\_weight="1">  
  
 <Button  
 android:id="@+id/btn\_mc"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="MC"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_m\_plus"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="M+"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_m\_minus"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="M-"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_mr"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:textColor="#FFF"  
 android:text="MR"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_orange"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:layout\_weight="1"  
 >  
  
 <LinearLayout  
 android:layout\_width="0dp"  
 android:layout\_weight="3"  
 android:layout\_height="match\_parent">  
 <Button  
 android:id="@+id/btn\_ac"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:fontFamily="sans-serif-light"  
 android:onClick="buttonPressed"  
 android:text="AC"  
 android:textSize="@dimen/btn\_text\_size"  
 android:layout\_margin="0px"  
 android:background="@drawable/button\_background\_gray"  
 />  
  
 <Button  
 android:id="@+id/btn\_square"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:fontFamily="sans-serif-light"  
 android:onClick="buttonPressed"  
 android:text="\u221a"  
 android:textSize="@dimen/btn\_text\_size"  
 android:layout\_margin="0px"  
 android:background="@drawable/button\_background\_gray"  
 />  
  
 <Button  
 android:id="@+id/btn\_plus\_flash\_minus"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="+/-"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"  
 />  
  
 <Button  
 android:id="@+id/btn\_percent"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="%"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"  
 />  
 </LinearLayout>  
  
  
  
 <Button  
 android:id="@+id/btn\_divide"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:textColor="#FFF"  
 android:text="\u00F7"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_orange"  
 />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:layout\_weight="1">  
  
 <Button  
 android:id="@+id/btn\_7"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="7"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_8"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="8"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_9"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="9"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_multiply"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:textColor="#FFF"  
 android:text="\u00D7"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_orange"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:layout\_weight="1">  
  
 <Button  
 android:id="@+id/btn\_4"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="4"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_5"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="5"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_6"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="6"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_minus"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:textColor="#FFF"  
 android:text="\u2212"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_orange"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:layout\_weight="1">  
  
 <Button  
 android:id="@+id/btn\_1"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:fontFamily="sans-serif-light"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:text="1"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_2"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="2"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_3"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="3"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_plus"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:textColor="#FFF"  
 android:text="+"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_orange"/>  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:layout\_weight="1">  
  
 <Button  
 android:id="@+id/btn\_zero"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="2"  
 android:onClick="buttonPressed"  
 android:fontFamily="sans-serif-light"  
 android:text="0"  
 android:textSize="@dimen/btn\_text\_size"  
 android:background="@drawable/button\_background\_gray"/>  
  
  
 <Button  
 android:id="@+id/btn\_decimal\_point"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:text="."  
 android:textSize="@dimen/btn\_text\_size"  
 android:fontFamily="sans-serif-light"  
 android:background="@drawable/button\_background\_gray"/>  
  
 <Button  
 android:id="@+id/btn\_equal"  
 android:layout\_width="0dp"  
 android:layout\_height="match\_parent"  
 android:layout\_weight="1"  
 android:onClick="buttonPressed"  
 android:textColor="#FFF"  
 android:text="="  
 android:textSize="@dimen/btn\_text\_size"  
 android:fontFamily="sans-serif-light"  
 android:background="@drawable/button\_background\_orange"/>  
 </LinearLayout>  
 </com.kientran.week04.CustomizedLinearLayout>  
  
  
</androidx.constraintlayout.widget.ConstraintLayout>

**bordered\_linear.xml**

<?xml version="1.0" encoding="utf-8"?>  
<shape xmlns:android="http://schemas.android.com/apk/res/android"  
 android:shape="rectangle">  
  
 <stroke android:width="20px" android:color="@color/main\_border\_color" />  
</shape>

**bodered\_textview.xml**

<?xml version="1.0" encoding="utf-8"?>  
<shape xmlns:android="http://schemas.android.com/apk/res/android"  
 android:shape="rectangle">  
 <solid android:color="@color/result\_background" />  
  
 <stroke android:width="0px" android:color="@color/main\_border\_color" />  
</shape>

**button\_background\_gray.xml**

<?xml version="1.0" encoding="utf-8"?>  
<ripple xmlns:android="http://schemas.android.com/apk/res/android"  
 android:color="@color/button\_gray\_ripple">  
 <item android:id="@android:id/background">  
 <shape android:shape="rectangle">  
 <gradient  
 android:endColor="@color/button\_gray\_end"  
 android:startColor="@color/button\_gray\_start"  
 android:type="radial"  
 android:gradientRadius="1000"/>  
  
 <stroke android:width="1px" android:color="@color/main\_border\_color" />  
 </shape>  
 </item>  
</ripple>

**button\_background\_orange.xml**

<?xml version="1.0" encoding="utf-8"?>  
<ripple xmlns:android="http://schemas.android.com/apk/res/android"  
 android:color="@color/button\_orange\_ripple">  
 <item android:id="@android:id/background">  
 <shape android:shape="rectangle">  
 <gradient  
 android:endColor="@color/button\_orange\_end"  
 android:startColor="@color/button\_orange\_start"  
 android:type="radial"  
 android:gradientRadius="1000"/>  
 <stroke android:width="1px" android:color="@color/main\_border\_color" />  
 </shape>  
 </item>  
</ripple>

**Color.xml**

<?xml version="1.0" encoding="utf-8"?>  
<resources>  
 <color name="colorPrimary">#008577</color>  
 <color name="colorPrimaryDark">#00574B</color>  
 <color name="colorPrimaryLight">#B3E5FC</color>  
 <color name="colorAccent">#D81B60</color>  
 <color name="main\_border\_color">#000000</color>  
 <color name="keypad\_main\_color">#000000</color>  
 <color name="keypad\_right\_color">#fff</color>  
 <color name="result\_text">#fff</color>  
 <color name="result\_background">#000000</color>  
 <color name="button\_gray\_start">#c5c6c8</color>  
 <color name="button\_gray\_end">#cbccce</color>  
 <color name="button\_gray\_ripple">#BDBDBD</color>  
 <color name="button\_orange\_start">#d97f24</color>  
 <color name="button\_orange\_end">#f78929</color>  
 <color name="button\_orange\_ripple">#FF9800</color>  
  
</resources>

**demens.xml**

<?xml version="1.0" encoding="utf-8"?>  
<resources>  
 <dimen name="btn\_text\_size">45dp</dimen>  
 <dimen name="calculator\_line\_text\_size">90dp</dimen>  
 <dimen name="calculator\_accumulate\_line\_text\_size">20dp</dimen>  
</resources>

**Style.xml**

<resources>  
  
 <!-- Base application theme. -->  
 <style name="AppTheme" parent="Theme.AppCompat.Light.DarkActionBar">  
 <!-- Customize your theme here. -->  
 <item name="colorPrimary">@color/colorPrimary</item>  
 <item name="colorPrimaryDark">@color/colorPrimaryDark</item>  
 <item name="colorAccent">@color/colorAccent</item>  
 </style>  
  
 <style name="Theme.AppCompat.Light.NoActionBar.FullScreen" parent="@style/Theme.AppCompat.Light.NoActionBar">  
 <item name="android:windowNoTitle">true</item>  
 <item name="android:windowActionBar">false</item>  
 <item name="android:windowFullscreen">true</item>  
 <item name="android:windowContentOverlay">@null</item>  
 </style>  
  
</resources>

**AndroidManifest.xml**

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 package="com.kientran.week04" >  
  
 <application  
 android:allowBackup="true"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/AppTheme" >  
 <activity android:name=".MainActivity"  
 android:theme="@style/Theme.AppCompat.Light.NoActionBar.FullScreen"  
 android:screenOrientation="portrait"  
 >  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 </application>  
  
</manifest>