**Оценка 3:**

* Реализованы все 4 операции: сложение, вычитание, умножение, деление.
* Интерфейс соответствует скриншоту.
* Компоненты имеют осмысленные ID (например, txtA, buttonAdd).

**Оценка 4:**

* Текстовые надписи вынесены в strings.xml.
* Обработка ошибок:
  + Пустые строки: сообщение "Поля не могут быть пустыми!".
  + Нечисловые данные: сообщение "Введите корректные числа!".
  + Деление на 0: сообщение "Деление на ноль!".
* Сообщения об ошибках выводятся в поле результата (labC).

**Оценка 5:**

* Добавлены тригонометрические функции sin, cos, tan:
  + Используется только поле A.
  + Для tan проверяется корректность (например, при 90° — ошибка).
* Добавлены константы PI и e:
  + При нажатии на кнопки значения записываются в поле A.
* Ошибки выводятся в поле результата.

#### Дополнительные функции:

* Переключение между полями A и B с помощью кнопки ✔.
* Кнопка \_ очищает все поля.
* Кнопка <- удаляет последний символ.
* Кнопка - меняет знак числа.

Переходим к кодам, их просто втсавить и будет счастье.

Activity\_main.xaml:

<?xml version="1.0" encoding="utf-8"?>  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:padding="16dp">  
  
 <!-- Поле A -->  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal">  
 <TextView  
 android:id="@+id/textViewA"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/infoA"  
 android:textSize="16sp"  
 android:contentDescription="@string/infoA" />  
 <EditText  
 android:id="@+id/txtA"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:hint="@string/hintNumber"  
 android:inputType="numberDecimal|numberSigned"  
 android:textSize="16sp"  
 android:focusable="true"  
 android:focusableInTouchMode="true"  
 android:contentDescription="@string/hintNumberA" />  
 </LinearLayout>  
  
 <!-- Поле B -->  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="8dp"  
 android:orientation="horizontal">  
 <TextView  
 android:id="@+id/textViewB"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/infoB"  
 android:textSize="16sp"  
 android:contentDescription="@string/infoB" />  
 <EditText  
 android:id="@+id/txtB"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:hint="@string/hintNumber"  
 android:inputType="numberDecimal|numberSigned"  
 android:textSize="16sp"  
 android:focusable="true"  
 android:focusableInTouchMode="true"  
 android:contentDescription="@string/hintNumberB" />  
 </LinearLayout>  
  
 <!-- Поле результата -->  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="8dp"  
 android:orientation="horizontal">  
 <TextView  
 android:id="@+id/textViewResult"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/infoResult"  
 android:textSize="16sp"  
 android:contentDescription="@string/infoResult" />  
 <TextView  
 android:id="@+id/labC"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text=""  
 android:textSize="16sp"  
 android:contentDescription="@string/resultValue" />  
 </LinearLayout>  
  
 <!-- Кнопки операций -->  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="16dp"  
 android:orientation="horizontal">  
 <Button  
 android:id="@+id/buttonAdd"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonAdd"  
 android:contentDescription="@string/buttonAdd" />  
 <Button  
 android:id="@+id/buttonSubtract"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonSubtract"  
 android:contentDescription="@string/buttonSubtract" />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="8dp"  
 android:orientation="horizontal">  
 <Button  
 android:id="@+id/buttonMultiply"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonMultiply"  
 android:contentDescription="@string/buttonMultiply" />  
 <Button  
 android:id="@+id/buttonDivide"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonDivide"  
 android:contentDescription="@string/buttonDivide" />  
 </LinearLayout>  
  
 <!-- Кнопки тригонометрических функций -->  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="16dp"  
 android:orientation="horizontal">  
 <Button  
 android:id="@+id/buttonSin"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonSin"  
 android:contentDescription="@string/buttonSin" />  
 <Button  
 android:id="@+id/buttonCos"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonCos"  
 android:contentDescription="@string/buttonCos" />  
 <Button  
 android:id="@+id/buttonTan"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonTan"  
 android:contentDescription="@string/buttonTan" />  
 </LinearLayout>  
  
 <!-- Кнопки констант -->  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="8dp"  
 android:orientation="horizontal">  
 <Button  
 android:id="@+id/buttonPi"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonPi"  
 android:contentDescription="@string/buttonPi" />  
 <Button  
 android:id="@+id/buttonE"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonE"  
 android:contentDescription="@string/buttonE" />  
 </LinearLayout>  
  
 <!-- Кнопки цифр и операций -->  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="16dp"  
 android:orientation="horizontal">  
 <Button  
 android:id="@+id/button1"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="1"  
 android:contentDescription="1" />  
 <Button  
 android:id="@+id/button2"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="2"  
 android:contentDescription="2" />  
 <Button  
 android:id="@+id/button3"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="3"  
 android:contentDescription="3" />  
 <Button  
 android:id="@+id/buttonMinus"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonMinus"  
 android:contentDescription="@string/buttonMinus" />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="8dp"  
 android:orientation="horizontal">  
 <Button  
 android:id="@+id/button4"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="4"  
 android:contentDescription="4" />  
 <Button  
 android:id="@+id/button5"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="5"  
 android:contentDescription="5" />  
 <Button  
 android:id="@+id/button6"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="6"  
 android:contentDescription="6" />  
 <Button  
 android:id="@+id/buttonClear"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonClear"  
 android:contentDescription="@string/buttonClear" />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="8dp"  
 android:orientation="horizontal">  
 <Button  
 android:id="@+id/button7"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="7"  
 android:contentDescription="7" />  
 <Button  
 android:id="@+id/button8"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="8"  
 android:contentDescription="8" />  
 <Button  
 android:id="@+id/button9"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="9"  
 android:contentDescription="9" />  
 <Button  
 android:id="@+id/buttonBackspace"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonBackspace"  
 android:contentDescription="@string/buttonBackspace" />  
 </LinearLayout>  
  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="8dp"  
 android:orientation="horizontal">  
 <Button  
 android:id="@+id/button0"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="0"  
 android:contentDescription="0" />  
 <Button  
 android:id="@+id/buttonDot"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonDot"  
 android:contentDescription="@string/buttonDot" />  
 <Button  
 android:id="@+id/buttonOk"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:backgroundTint="@android:color/holo\_blue\_light"  
 android:text="@string/buttonOk"  
 android:contentDescription="@string/buttonOk" />  
 </LinearLayout>  
  
</LinearLayout>

String.xaml:

<resources>  
 <string name="app\_name">Lab 01 Calculator</string>  
 <string name="infoA">A:</string>  
 <string name="infoB">B:</string>  
 <string name="infoResult">Результат:</string>  
 <string name="buttonAdd">Add</string>  
 <string name="buttonSubtract">Subtract</string>  
 <string name="buttonMultiply">Multiply</string>  
 <string name="buttonDivide">Divide</string>  
 <string name="buttonSin">sin</string>  
 <string name="buttonCos">cos</string>  
 <string name="buttonTan">tan</string>  
 <string name="buttonPi">PI</string>  
 <string name="buttonE">e</string>  
 <string name="buttonMinus">-</string>  
 <string name="buttonClear">C</string>  
 <string name="buttonBackspace">⌫</string>  
 <string name="buttonDot">.</string>  
 <string name="buttonOk">✔</string>  
 <string name="hintNumber">Введите число</string>  
 <string name="hintNumberA">Введите первое число</string>  
 <string name="hintNumberB">Введите второе число</string>  
 <string name="resultValue">Значение результата</string>  
 <string name="errorEmpty">Поля не могут быть пустыми!</string>  
 <string name="errorInvalid">Введите корректные числа!</string>  
 <string name="errorDivideByZero">Деление на ноль!</string>  
 <string name="errorTan">Недопустимое значение для tan!</string>  
 <string name="errorOverflow">Результат слишком большой!</string>  
</resources>

MainActivity.java

package com.example.engineeringcalculator;  
  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
import androidx.appcompat.app.AppCompatActivity;  
  
import java.text.DecimalFormat;  
import java.text.DecimalFormatSymbols;  
import java.util.Locale;  
  
public class MainActivity extends AppCompatActivity {  
  
 private EditText txtA, txtB;  
 private TextView labC;  
 private boolean isFieldA = true; // Флаг для выбора поля (A или B)  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 // Инициализация компонентов  
 txtA = findViewById(R.id.*txtA*);  
 txtB = findViewById(R.id.*txtB*);  
 labC = findViewById(R.id.*labC*);  
  
 // Установка начального фокуса на поле A  
 txtA.requestFocus();  
  
 // Обработка смены фокуса  
 txtA.setOnFocusChangeListener((v, hasFocus) -> {  
 if (hasFocus) isFieldA = true;  
 });  
 txtB.setOnFocusChangeListener((v, hasFocus) -> {  
 if (hasFocus) isFieldA = false;  
 });  
  
 // Кнопки операций  
 Button buttonAdd = findViewById(R.id.*buttonAdd*);  
 Button buttonSubtract = findViewById(R.id.*buttonSubtract*);  
 Button buttonMultiply = findViewById(R.id.*buttonMultiply*);  
 Button buttonDivide = findViewById(R.id.*buttonDivide*);  
  
 // Кнопки тригонометрических функций  
 Button buttonSin = findViewById(R.id.*buttonSin*);  
 Button buttonCos = findViewById(R.id.*buttonCos*);  
 Button buttonTan = findViewById(R.id.*buttonTan*);  
  
 // Кнопки констант  
 Button buttonPi = findViewById(R.id.*buttonPi*);  
 Button buttonE = findViewById(R.id.*buttonE*);  
  
 // Кнопки цифр и операций  
 int[] numberButtonIds = {  
 R.id.*button0*, R.id.*button1*, R.id.*button2*, R.id.*button3*,  
 R.id.*button4*, R.id.*button5*, R.id.*button6*,  
 R.id.*button7*, R.id.*button8*, R.id.*button9* };  
 Button buttonDot = findViewById(R.id.*buttonDot*);  
 Button buttonMinus = findViewById(R.id.*buttonMinus*);  
 Button buttonClear = findViewById(R.id.*buttonClear*);  
 Button buttonBackspace = findViewById(R.id.*buttonBackspace*);  
 Button buttonOk = findViewById(R.id.*buttonOk*);  
  
 // Обработчик для кнопок цифр  
 View.OnClickListener numberClickListener = v -> {  
 Button button = (Button) v;  
 String number = button.getText().toString();  
 appendToField(number);  
 };  
  
 // Привязка обработчиков для кнопок цифр  
 for (int id : numberButtonIds) {  
 findViewById(id).setOnClickListener(numberClickListener);  
 }  
  
 // Десятичная точка  
 buttonDot.setOnClickListener(v -> appendToField("."));  
  
 // Смена знака  
 buttonMinus.setOnClickListener(v -> {  
 EditText activeField = isFieldA ? txtA : txtB;  
 String currentText = activeField.getText().toString();  
 if (!currentText.isEmpty()) {  
 if (currentText.startsWith("-")) {  
 activeField.setText(currentText.substring(1));  
 } else {  
 activeField.setText("-" + currentText);  
 }  
 }  
 });  
  
 // Очистка полей  
 buttonClear.setOnClickListener(v -> {  
 txtA.setText("");  
 txtB.setText("");  
 labC.setText("--");  
 isFieldA = true;  
 txtA.requestFocus();  
 });  
  
 // Удаление последнего символа  
 buttonBackspace.setOnClickListener(v -> {  
 EditText activeField = isFieldA ? txtA : txtB;  
 String currentText = activeField.getText().toString();  
 if (!currentText.isEmpty()) {  
 activeField.setText(currentText.substring(0, currentText.length() - 1));  
 }  
 labC.setText("--");  
 });  
  
 // Переключение между полями по кнопке OK  
 buttonOk.setOnClickListener(v -> {  
 if (isFieldA) {  
 isFieldA = false;  
 txtB.requestFocus();  
 } else {  
 isFieldA = true;  
 txtA.requestFocus();  
 }  
 });  
  
 // Операции  
 buttonAdd.setOnClickListener(v -> performOperation("add"));  
 buttonSubtract.setOnClickListener(v -> performOperation("subtract"));  
 buttonMultiply.setOnClickListener(v -> performOperation("multiply"));  
 buttonDivide.setOnClickListener(v -> performOperation("divide"));  
  
 // Тригонометрические функции  
 buttonSin.setOnClickListener(v -> performTrigFunction("sin"));  
 buttonCos.setOnClickListener(v -> performTrigFunction("cos"));  
 buttonTan.setOnClickListener(v -> performTrigFunction("tan"));  
  
 // Константы (записываются только в txtA)  
 buttonPi.setOnClickListener(v -> {  
 txtA.setText(String.*valueOf*(Math.*PI*));  
 labC.setText("--");  
 });  
 buttonE.setOnClickListener(v -> {  
 txtA.setText(String.*valueOf*(Math.*E*));  
 labC.setText("--");  
 });  
 }  
  
 // Метод для добавления символа в активное поле  
 private void appendToField(String value) {  
 EditText activeField = isFieldA ? txtA : txtB;  
 String currentText = activeField.getText().toString();  
  
 // Проверка на дублирование точки  
 if (value.equals(".")) {  
 if (currentText.contains(".")) {  
 return;  
 }  
 }  
  
 activeField.setText(currentText + value);  
 labC.setText("--"); // Сброс ошибки  
 }  
  
 // Выполнение арифметических операций  
 private void performOperation(String operation) {  
 String aText = txtA.getText().toString();  
 String bText = txtB.getText().toString();  
  
 // Проверка на пустые поля  
 if (aText.isEmpty() || bText.isEmpty()) {  
 labC.setText(getString(R.string.*errorEmpty*));  
 return;  
 }  
  
 try {  
 // Учет локального разделителя  
 DecimalFormatSymbols symbols = new DecimalFormatSymbols(Locale.*getDefault*());  
 String decimalSeparator = String.*valueOf*(symbols.getDecimalSeparator());  
 double a = Double.*parseDouble*(aText.replace(decimalSeparator, "."));  
 double b = Double.*parseDouble*(bText.replace(decimalSeparator, "."));  
  
 double result = 0;  
 switch (operation) {  
 case "add":  
 result = a + b;  
 break;  
 case "subtract":  
 result = a - b;  
 break;  
 case "multiply":  
 result = a \* b;  
 break;  
 case "divide":  
 if (b == 0) {  
 labC.setText(getString(R.string.*errorDivideByZero*));  
 return;  
 }  
 result = a / b;  
 break;  
 }  
  
 // Проверка на переполнение  
 if (Double.*isInfinite*(result) || Double.*isNaN*(result)) {  
 labC.setText(getString(R.string.*errorOverflow*));  
 return;  
 }  
  
 // Форматирование результата  
 DecimalFormat df = new DecimalFormat("#.######", new DecimalFormatSymbols(Locale.*US*));  
 labC.setText(df.format(result));  
 } catch (NumberFormatException e) {  
 labC.setText(getString(R.string.*errorInvalid*));  
 }  
 }  
  
 // Выполнение тригонометрических функций  
 private void performTrigFunction(String function) {  
 String aText = txtA.getText().toString();  
  
 // Проверка на пустое поле  
 if (aText.isEmpty()) {  
 labC.setText(getString(R.string.*errorEmpty*));  
 return;  
 }  
  
 try {  
 // Учет локального разделителя  
 DecimalFormatSymbols symbols = new DecimalFormatSymbols(Locale.*getDefault*());  
 String decimalSeparator = String.*valueOf*(symbols.getDecimalSeparator());  
 double a = Double.*parseDouble*(aText.replace(decimalSeparator, "."));  
  
 double result = 0;  
 switch (function) {  
 case "sin":  
 result = Math.*sin*(Math.*toRadians*(a));  
 break;  
 case "cos":  
 result = Math.*cos*(Math.*toRadians*(a));  
 break;  
 case "tan":  
 result = Math.*tan*(Math.*toRadians*(a));  
 if (Double.*isInfinite*(result) || Double.*isNaN*(result)) {  
 labC.setText(getString(R.string.*errorTan*));  
 return;  
 }  
 break;  
 }  
  
 // Проверка на переполнение  
 if (Double.*isInfinite*(result) || Double.*isNaN*(result)) {  
 labC.setText(getString(R.string.*errorOverflow*));  
 return;  
 }  
  
 // Форматирование результата  
 DecimalFormat df = new DecimalFormat("#.######", new DecimalFormatSymbols(Locale.*US*));  
 labC.setText(df.format(result));  
 } catch (NumberFormatException e) {  
 labC.setText(getString(R.string.*errorInvalid*));  
 }  
 }  
}