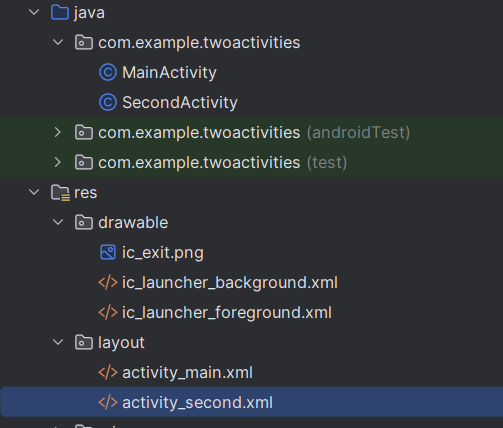
Сначала:

1. Щелкни правой кнопкой мыши на app/java/com.example.twoactivities → **New → Activity → Empty Activity**.
2. Назови вторую Activity SecondActivity.
3. Убедись, что в AndroidManifest.xml автоматически добавилась запись для SecondActivity.
4.  Нарисуй простую иконку в Paint (например, значок выхода).
5.  Сохрани её как PNG-файл, например, ic\_exit.png.
6.  Помести файл в папку app/src/main/res/drawable.

+ Создать Вторая Activity (activity\_second.xml)



#### Оценка 3:

* Реализован ввод текста в MainActivity и передача его в SecondActivity через Intent.
* В SecondActivity текст отображается в текстовом поле.
* Интерфейс соответствует скриншоту, компоненты имеют осмысленные ID: txtData, buttonOpenDialog, buttonExitApp, и т.д.

#### Оценка 4:

* Текстовые надписи вынесены в strings.xml.
* Возврат текста из SecondActivity в MainActivity происходит только при нажатии кнопки OK.
* После возврата текст отображается в текстовом поле MainActivity, и показывается всплывающее сообщение Toast.

#### Оценка 5:

* При нажатии кнопки EXIT APP отображается диалоговое окно AlertDialog с подтверждением.
* Установлен заголовок "Confirm Exit" и пользовательская иконка (ic\_exit).

Переходим к кодам, их просто вставить и будет счастье.

AndroidManifest.xml:

<?xml version="1.0" encoding="utf-8"?>  
<manifest xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools">  
  
 <application  
 android:allowBackup="true"  
 android:dataExtractionRules="@xml/data\_extraction\_rules"  
 android:fullBackupContent="@xml/backup\_rules"  
 android:icon="@mipmap/ic\_launcher"  
 android:label="@string/app\_name"  
 android:roundIcon="@mipmap/ic\_launcher\_round"  
 android:supportsRtl="true"  
 android:theme="@style/Theme.TwoActivities"  
 tools:targetApi="31">  
 <activity android:name=".SecondActivity" />  
 <activity  
 android:name=".MainActivity"  
 android:exported="true">  
 <intent-filter>  
 <action android:name="android.intent.action.MAIN" />  
  
 <category android:name="android.intent.category.LAUNCHER" />  
 </intent-filter>  
 </activity>  
 </application>  
  
</manifest>

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">  
  
 <!-- Текстовое поле -->  
 <EditText  
 android:id="@+id/txtData"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="@string/hintData" />  
  
 <!-- Кнопки -->  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal">  
 <Button  
 android:id="@+id/buttonOpenDialog"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonOpenDialog" />  
 <Button  
 android:id="@+id/buttonExitApp"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonExitApp" />  
 </LinearLayout>  
  
</LinearLayout>

Activity\_second.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">  
  
 <!-- Текстовое поле -->  
 <EditText  
 android:id="@+id/txtData"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:hint="@string/hintData" />  
  
 <!-- Кнопки -->  
 <LinearLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:orientation="horizontal">  
 <Button  
 android:id="@+id/buttonOk"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonOk" />  
 <Button  
 android:id="@+id/buttonCancel"  
 android:layout\_width="0dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_weight="1"  
 android:text="@string/buttonCancel" />  
 </LinearLayout>  
  
</LinearLayout>

String.xaml:

<resources>  
 <string name="app\_name">TwoActivities</string>  
 <string name="hintData">Name</string>  
 <string name="buttonOpenDialog">OPEN DIALOG</string>  
 <string name="buttonExitApp">EXIT APP</string>  
 <string name="buttonOk">OK</string>  
 <string name="buttonCancel">CANCEL</string>  
 <string name="dialogExitTitle">Confirm Exit</string>  
 <string name="dialogExitMessage">Are you sure you want to exit?</string>  
 <string name="dialogExitButtonYes">Yes</string>  
 <string name="dialogExitButtonNo">No</string>  
 <string name="toastMessage">Returned: %s</string>  
 <string name="emptyInputError">Please enter some text</string> </resources>

SecondActivity.java

package com.example.twoactivities;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class SecondActivity extends AppCompatActivity {  
  
 private EditText txtData;  
 private Button buttonOk, buttonCancel;  
 private static final String *KEY\_TEXT\_DATA* = "text\_data";  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_second*);  
  
 // Инициализация компонентов  
 txtData = findViewById(R.id.*txtData*);  
 buttonOk = findViewById(R.id.*buttonOk*);  
 buttonCancel = findViewById(R.id.*buttonCancel*);  
  
 // Восстановление текста при повороте экрана  
 if (savedInstanceState != null) {  
 String savedText = savedInstanceState.getString(*KEY\_TEXT\_DATA*);  
 if (savedText != null) {  
 txtData.setText(savedText);  
 }  
 } else {  
 // Получение данных из MainActivity  
 Intent intent = getIntent();  
 String data = intent.getStringExtra("data");  
 if (data != null) {  
 txtData.setText(data);  
 }  
 }  
  
 // Обработчик кнопки OK  
 buttonOk.setOnClickListener(v -> {  
 String result = txtData.getText().toString().trim();  
 if (result.isEmpty()) {  
 Toast.*makeText*(this, R.string.*emptyInputError*, Toast.*LENGTH\_SHORT*).show();  
 return;  
 }  
 Intent returnIntent = new Intent();  
 returnIntent.putExtra("result", result);  
 setResult(*RESULT\_OK*, returnIntent);  
 finish();  
 });  
  
 // Обработчик кнопки CANCEL  
 buttonCancel.setOnClickListener(v -> {  
 setResult(*RESULT\_CANCELED*);  
 finish();  
 });  
 }  
  
 // Сохранение текста при повороте экрана  
 @Override  
 protected void onSaveInstanceState(Bundle outState) {  
 super.onSaveInstanceState(outState);  
 outState.putString(*KEY\_TEXT\_DATA*, txtData.getText().toString());  
 }  
}

MainActivity.java

package com.example.twoactivities;  
  
import android.content.Intent;  
import android.os.Bundle;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.Toast;  
import androidx.appcompat.app.AlertDialog;  
import androidx.appcompat.app.AppCompatActivity;  
  
public class MainActivity extends AppCompatActivity {  
  
 private static final int *REQUEST\_CODE* = 1;  
 private EditText txtData;  
 private Button buttonOpenDialog, buttonExitApp;  
 private static final String *KEY\_TEXT\_DATA* = "text\_data";  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
  
 // Инициализация компонентов  
 txtData = findViewById(R.id.*txtData*);  
 buttonOpenDialog = findViewById(R.id.*buttonOpenDialog*);  
 buttonExitApp = findViewById(R.id.*buttonExitApp*);  
  
 // Восстановление текста при повороте экрана  
 if (savedInstanceState != null) {  
 String savedText = savedInstanceState.getString(*KEY\_TEXT\_DATA*);  
 if (savedText != null) {  
 txtData.setText(savedText);  
 }  
 }  
  
 // Обработчик кнопки OPEN DIALOG  
 buttonOpenDialog.setOnClickListener(v -> {  
 String data = txtData.getText().toString().trim();  
 if (data.isEmpty()) {  
 Toast.*makeText*(this, R.string.*emptyInputError*, Toast.*LENGTH\_SHORT*).show();  
 return;  
 }  
 Intent intent = new Intent(MainActivity.this, SecondActivity.class);  
 intent.putExtra("data", data);  
 startActivityForResult(intent, *REQUEST\_CODE*);  
 });  
  
 // Обработчик кнопки EXIT APP  
 buttonExitApp.setOnClickListener(v -> showExitDialog());  
 }  
  
 // Сохранение текста при повороте экрана  
 @Override  
 protected void onSaveInstanceState(Bundle outState) {  
 super.onSaveInstanceState(outState);  
 outState.putString(*KEY\_TEXT\_DATA*, txtData.getText().toString());  
 }  
  
 // Получение результата от SecondActivity  
 @Override  
 protected void onActivityResult(int requestCode, int resultCode, Intent data) {  
 super.onActivityResult(requestCode, resultCode, data);  
 if (requestCode == *REQUEST\_CODE* && resultCode == *RESULT\_OK* && data != null) {  
 String returnedData = data.getStringExtra("result");  
 if (returnedData != null) {  
 txtData.setText(returnedData);  
 Toast.*makeText*(this, String.*format*(getString(R.string.*toastMessage*), returnedData), Toast.*LENGTH\_SHORT*).show();  
 }  
 }  
 }  
  
 // Диалог подтверждения выхода  
 private void showExitDialog() {  
 new AlertDialog.Builder(this)  
 .setTitle(R.string.*dialogExitTitle*)  
 .setIcon(R.drawable.*ic\_exit*)  
 .setMessage(R.string.*dialogExitMessage*)  
 .setPositiveButton(R.string.*dialogExitButtonYes*, (dialog, which) -> finish())  
 .setNegativeButton(R.string.*dialogExitButtonNo*, (dialog, which) -> dialog.cancel())  
 .show();  
 }  
}