**Министерство науки и высшего образования Российской Федерации**

Федеральное государственное автономное образовательное учреждение

высшего образования «Уральский федеральный университет

имени первого Президента России Б.Н. Ельцина»

Институт радиоэлектроники и информационных технологий - РТФ

Центр ускоренного обучения

**ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ 16**

по дисциплине «Разработка мобильных приложений»

**Тема: Unit 2 Пользовательский интерфейс “Чертежи, стили и темы”**

Студент группы РИЗ-300028у: А.С. Севостьянов

Преподаватель: С.И.Тимошенко,

доц., к.т.н.

**Екатеринбург 2022**

содержание

[1 Постановка задачи 3](#_Toc118041527)

[2 Результаты проделанной работы 4](#_Toc118041528)

[3 Анализ полученных результатов 5](#_Toc118041529)

## 1 Постановка задачи

Создать новое приложение и добавить ButtonTextView элементы и в макет. Создать Drawebel ресурсы в формате XML и используйте их в качестве фона для Button элементов. Применить стили к элементам пользовательского интерфейса. Добавить пункт меню, который меняет тему приложения на низкоконтрастный "ночной режим".

.

## Результаты проделанной работы

Программный код проекта, файл MainActivity.

package com.example.android.scorekeeper;  
  
import android.os.Bundle;  
import android.support.v7.app.AppCompatActivity;  
import android.support.v7.app.AppCompatDelegate;  
import android.view.Menu;  
import android.view.MenuItem;  
import android.view.View;  
import android.widget.TextView;  
  
public class MainActivity extends AppCompatActivity {  
  
 // Переменные-члены для удержания оценки  
 private int mScore1;  
 private int mScore2;  
  
 // Переменные-члены для двух элементов score TextView  
 private TextView mScoreText1;  
 private TextView mScoreText2;  
  
 // Теги, которые будут использоваться в качестве ключей в Onsaveinstancestate  
 static final String STATE\_SCORE\_1 = "Team 1 Score";  
 static final String STATE\_SCORE\_2 = "Team 2 Score";  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
  
 //текстовое представление по идентификатору  
 mScoreText1 = findViewById(R.id.score\_1);  
 mScoreText2 = findViewById(R.id.score\_2);  
  
 // Восстанавливает результаты, если есть savedInstanceState.  
 if (savedInstanceState != null) {  
 mScore1 = savedInstanceState.getInt(STATE\_SCORE\_1);  
 mScore2 = savedInstanceState.getInt(STATE\_SCORE\_2);  
  
 //Установть текстовые представления  
 mScoreText1.setText(String.valueOf(mScore1));  
 mScoreText2.setText(String.valueOf(mScore2));  
 }  
 }  
  
 */\*\** Обрабатывает повторное нажатие обеих кнопок уменьшения. *\*/* public void decreaseScore(View view) {  
 // Получите идентификатор кнопки, на которую был нажат.  
 int viewID = view.getId();  
 switch (viewID) {  
 // Если бы это было в команде 1:  
 case R.id.decreaseTeam1:  
 // Уменьшите оценку и обновите TextView.  
 mScore1--;  
 mScoreText1.setText(String.valueOf(mScore1));  
 break;  
 //Если это было в команде 2:  
 case R.id.decreaseTeam2:  
 // Уменьшите оценку и обновите TextView.  
 mScore2--;  
 mScoreText2.setText(String.valueOf(mScore2));  
 }  
 }  
  
 public void increaseScore(View view) {  
 // Получите идентификатор кнопки, на которую был нажат.  
 int viewID = view.getId();  
 switch (viewID) {  
 // Если бы это было в команде 1:  
 case R.id.increaseTeam1:  
 //Увеличьте оценку и обновите текстовое представление.  
 mScore1++;  
 mScoreText1.setText(String.valueOf(mScore1));  
 break;  
 //Если бы это была команда 2:  
 case R.id.increaseTeam2:  
 // Увеличьте оценку и обновите текстовое представление.  
 mScore2++;  
 mScoreText2.setText(String.valueOf(mScore2));  
 }  
 }  
  
 /\*  
 Создает опцию меню ночного режима.  
 \*/  
 @Override  
 public boolean onCreateOptionsMenu(Menu menu) {  
 getMenuInflater().inflate(R.menu.main\_menu, menu);  
 // Измените метку меню в зависимости от состояния приложения.  
 int nightMode = AppCompatDelegate.getDefaultNightMode();  
 if(nightMode == AppCompatDelegate.MODE\_NIGHT\_YES){  
 menu.findItem(R.id.night\_mode).setTitle(R.string.day\_mode);  
 } else{  
 menu.findItem(R.id.night\_mode).setTitle(R.string.night\_mode);  
 }  
 return true;  
 }  
  
  
 /\*  
 \*Обрабатывает щелчки по элементам меню опций.  
 \*/  
  
 @Override  
 public boolean onOptionsItemSelected(MenuItem item) {  
 // Проверьте, был ли нажат правильный элемент.  
 if (item.getItemId() == R.id.night\_mode) {  
 // Получите состояние приложения в ночном режиме.  
 int nightMode = AppCompatDelegate.getDefaultNightMode();  
 // Установите режим темы для перезапущенного действия.  
 if (nightMode == AppCompatDelegate.MODE\_NIGHT\_YES) {  
 AppCompatDelegate.setDefaultNightMode  
 (AppCompatDelegate.MODE\_NIGHT\_NO);  
 } else {  
 AppCompatDelegate.setDefaultNightMode  
 (AppCompatDelegate.MODE\_NIGHT\_YES);  
 }  
 //Повторите действие, чтобы изменение темы вступило в силу.  
 recreate();  
 }  
 return true;  
 }  
  
 /\*  
 \* Метод, который вызывается при изменении конфигурации  
 \*/  
  
 @Override  
 protected void onSaveInstanceState(Bundle outState) {  
 //Сохраните результаты.  
 outState.putInt(STATE\_SCORE\_1, mScore1);  
 outState.putInt(STATE\_SCORE\_2, mScore2);  
 super.onSaveInstanceState(outState);  
 }  
}

Отображение проекта.

<?xml version="1.0" encoding="utf-8"?>  
  
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:orientation="vertical"  
 android:padding="16dp"  
 tools:context="com.example.android.scorekeeper.MainActivity">  
  
 <RelativeLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:layout\_weight="1">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentTop="true"  
 android:layout\_centerHorizontal="true"  
 android:text="@string/team\_1"  
 style="@style/TeamText" />  
  
 <ImageButton  
 android:id="@+id/decreaseTeam1"  
 android:layout\_width="@dimen/button\_size"  
 android:layout\_height="@dimen/button\_size"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_centerVertical="true"  
 style="@style/MinusButtons"  
 android:onClick="decreaseScore"/>  
  
 <TextView  
 android:id="@+id/score\_1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerHorizontal="true"  
 android:layout\_centerVertical="true"  
 android:text="@string/initial\_count"  
 style="@style/ScoreText" />  
  
 <ImageButton  
 android:id="@+id/increaseTeam1"  
 android:layout\_width="@dimen/button\_size"  
 android:layout\_height="@dimen/button\_size"  
 android:layout\_alignParentEnd="true"  
 android:layout\_alignParentRight="true"  
 android:layout\_centerVertical="true"  
 style="@style/PlusButtons"  
 android:onClick="increaseScore"/>  
  
 </RelativeLayout>  
  
 <RelativeLayout  
 android:layout\_width="match\_parent"  
 android:layout\_height="0dp"  
 android:layout\_weight="1">  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_alignParentTop="true"  
 android:layout\_centerHorizontal="true"  
 android:text="@string/team\_2"  
 style="@style/TeamText" />  
  
 <ImageButton  
 android:id="@+id/decreaseTeam2"  
 android:layout\_width="@dimen/button\_size"  
 android:layout\_height="@dimen/button\_size"  
 android:layout\_alignParentLeft="true"  
 android:layout\_alignParentStart="true"  
 android:layout\_centerVertical="true"  
 style="@style/MinusButtons"  
 android:onClick="decreaseScore"/>  
  
 <TextView  
 android:id="@+id/score\_2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_centerHorizontal="true"  
 android:layout\_centerVertical="true"  
 android:text="@string/initial\_count"  
 style="@style/ScoreText" />  
  
 <ImageButton  
 android:id="@+id/increaseTeam2"  
 android:layout\_width="@dimen/button\_size"  
 android:layout\_height="@dimen/button\_size"  
 android:layout\_alignParentEnd="true"  
 android:layout\_alignParentRight="true"  
 android:layout\_centerVertical="true"  
 style="@style/PlusButtons"  
 android:onClick="increaseScore"/>  
 </RelativeLayout>  
  
</LinearLayout>

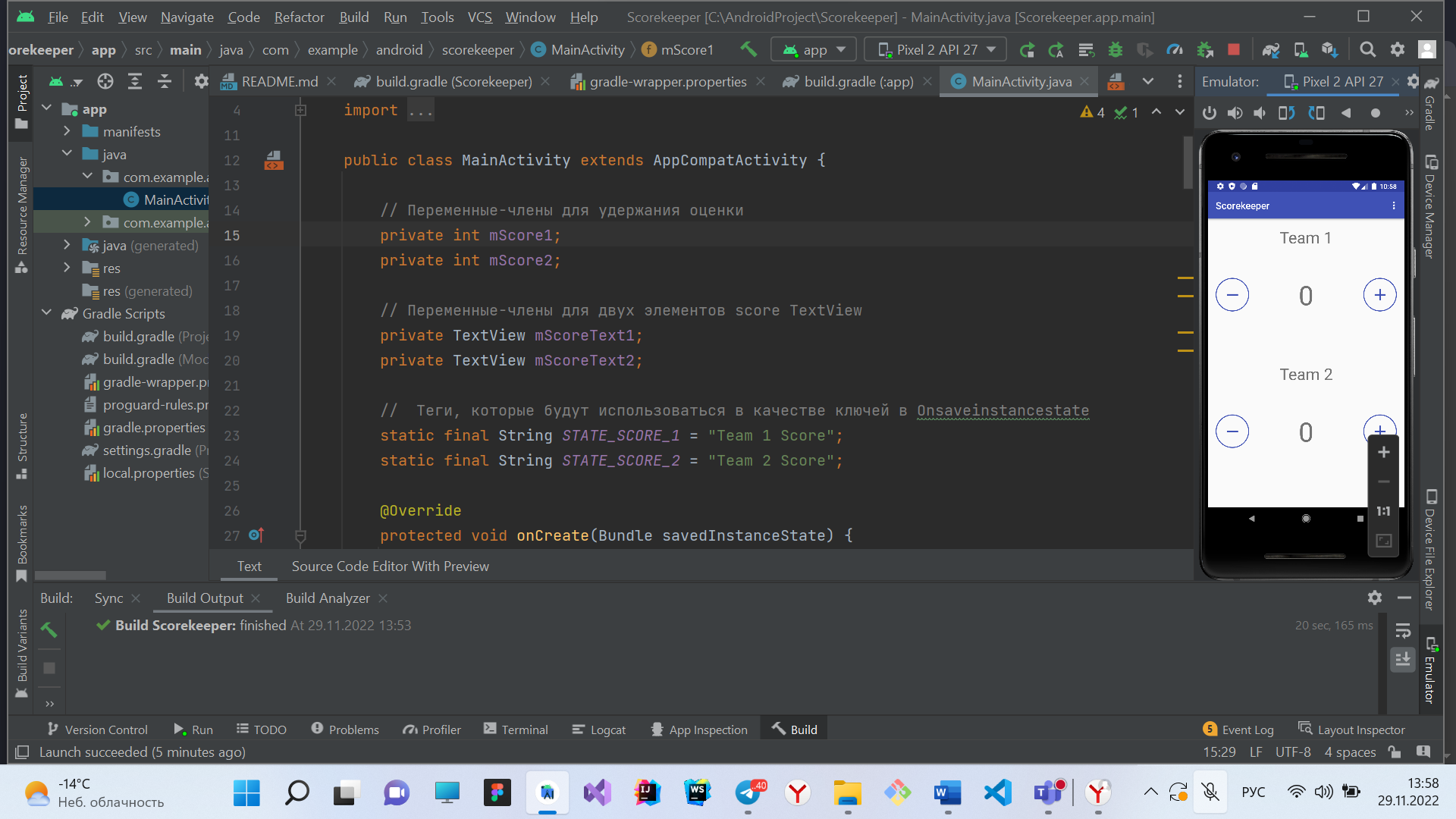
На рисунке 1 показано начальное состояние элекментов в программе.

Рисунок 1 – Начальное состояние программы

На рисунке 2 показан результат клика на кнопку плюс или минус, изменяется состояние элемента.

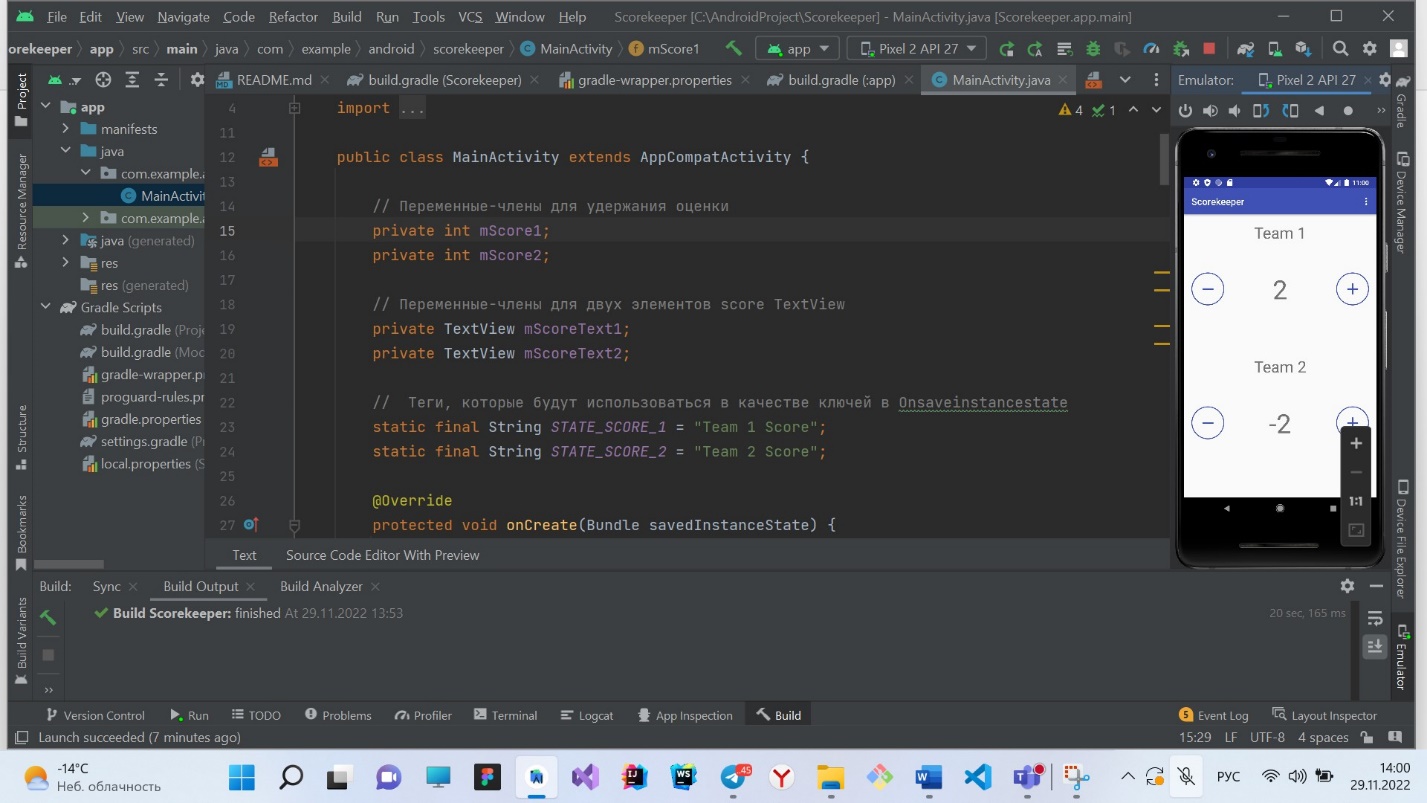


Рисунок 2 – Состояние программы после нажатия на кнопки плюс и минус

## **Анализ полученных результатов**

В результате выполнения данной лабораторной работы мной был разработан проект, в котором я научился определять ресурс стиля. Как применить стиль View. Как применить тему к Activity приложению или в XML и программно. Как использовать Drawable ресурсы.

.