

Приложение секундомер

Выполнили: Бурова Екатерина

Кахнович Мария

Activity_main

```
1 <?xml version="1.0" encoding="utf-8"?>
2 <LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
3     xmlns:app="http://schemas.android.com/apk/res-auto"
4     xmlns:tools="http://schemas.android.com/tools"
5     android:id="@+id/main1"
6     android:orientation="vertical"
7     android:layout_width="match_parent"
8     android:layout_height="match_parent"
9     tools:context=".MainActivity">
10
11
12     <Chronometer
13         android:id="@+id/textTime"
14         android:layout_width="wrap_content"
15         android:layout_height="wrap_content"
16         android:textSize="80sp"
17         android:layout_gravity="center"
18     />
19
20     <Button
21         android:id="@+id/btnStart"
22         android:layout_width="wrap_content"
23         android:layout_height="wrap_content"
24         android:text="Start"
25         android:layout_gravity="center"
26     />
```

```
27
28     <Button
29         android:id="@+id/btnPause"
30         android:layout_width="wrap_content"
31         android:layout_height="wrap_content"
32         android:text="Pause"
33         android:layout_gravity="center"
34     />
35
36     <Button
37         android:id="@+id/btnReset"
38         android:layout_width="wrap_content"
39         android:layout_height="wrap_content"
40         android:text="Reset"
41         android:layout_gravity="center"
42     />
43
44
45
46 </LinearLayout>
```

MainActivity

```
</> activity_main.xml </> strings.xml MainActivity.kt x
1 package com.example.lb6
2
3 import android.os.Bundle
4 import android.os.SystemClock
5 import android.widget.Button
6 import android.widget.Chronometer
7 import androidx.activity.enableEdgeToEdge
8 import androidx.appcompat.app.AppCompatActivity
9 import androidx.core.view.ViewCompat
10 import androidx.core.view.WindowInsetsCompat
11
12 <?> class MainActivity : AppCompatActivity() {
13
14     private lateinit var chronometr: Chronometer
15     private var running: Boolean = false
16     private var offset: Long = 0
17     private val OFFSET_KEY = "offset"
18     private val RUNNING_KEY = "running"
19     private val BASE_KEY = "base_key"
20
21     override fun onCreate(savedInstanceState: Bundle?) {
22         super.onCreate(savedInstanceState)
23
24         enableEdgeToEdge()
25         setContentView(R.layout.activity_main)
26
27         // Инициализация элементов интерфейса
28         chronometr = findViewById(R.id.textTime)
29         val btStart = findViewById<Button>(R.id.btnStart)
30         val btnPause = findViewById<Button>(R.id.btnPause)
31         val btnReset = findViewById<Button>(R.id.btnReset)
32
```

```
34
35         if (savedInstanceState != null) {
36             offset = savedInstanceState.getLong(OFFSET_KEY)
37             running = savedInstanceState.getBoolean(RUNNING_KEY)
38             chronometr.base = savedInstanceState.getLong(BASE_KEY)
39
40             if (running) {
41                 chronometr.start()
42             }
43
44             // Установка обработчиков событий на кнопки
45             btStart.setOnClickListener {
46                 if (!running) {
47                     setBaseTime()
48                     chronometr.start()
49                     running = true
50                 }
51             }
52
53             btnPause.setOnClickListener {
54                 if (running) {
55                     chronometr.stop()
56                     running = false
57                     // Сохраняем значение offset
58                     saveOffset()
59                 }
60             }
61
```

```
63         chronometr.stop()
64         running = false
65         offset = 0
66         // Сброс времени секундомера
67         chronometr.base = SystemClock.elapsedRealtime() // Устанавливаем на текущее время
68         chronometr.text = "00:00" // Сбрасываем текст на 0
69     }
70
71     // Применение отступов для операторов
72     ViewCompat.setOnApplyWindowInsetsListener(findViewById(R.id.main)) { v, insets ->
73         val systemBars = insets.getInsets(WindowInsetsCompat.Type.systemBars())
74         v.setPadding(systemBars.left, systemBars.top, systemBars.right, systemBars.bottom)
75         insets
76     }
77
78
79     override fun onSaveInstanceState(savedInstanceState: Bundle) {
80         savedInstanceState.putLong(OFFSET_KEY, offset)
81         savedInstanceState.putBoolean(RUNNING_KEY, running)
82         savedInstanceState.putLong(BASE_KEY, chronometr.base)
83         super.onSaveInstanceState(savedInstanceState)
84     }
85
86     private fun saveOffset() {
87         offset = SystemClock.elapsedRealtime() - chronometr.base
88     }
89
90     private fun setBaseTime() {
91         chronometr.base = SystemClock.elapsedRealtime() - offset
92
```

Запускаем секундомер

