Комитет по образованию Мингорисполкома

Учреждение образования

«Минский государственный колледж цифровых технологий»

Учебный предмет

«Разработка приложений для мобильных устройств»

Специальность: 2-39 03 02 «Программируемые мобильные системы»

**Отчет**

**по лабораторной работе №21**

**Тема «Обработка жизненного цикла активности.»**

Выполнил учащийся гр. 68МС Казак М.П.

Проверил: Бровка Д. С.

Минск

2024

**Цель работы:** научиться обрабатывать жизненный цикл активности.

**Ход работы**

**Задание 1**

Создать приложение «Секундомер» следуя материалу и инструкциям в нем. Выполненное приложение демонстрируется преподавателю и защищается кратким опросом по коду.

package com.laba.lr21;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
import android.os.Handler;  
import android.view.View;  
import android.widget.TextView;  
  
import java.util.Locale;  
  
public class StopwatchActivity extends AppCompatActivity {  
  
 private int seconds = 0; //В переменных seconds и running хранится количество прошедших секунд и флаг работы секундомера  
 private boolean running;  
 private boolean wasRunning;  
  
 @Override  
 public void onSaveInstanceState(Bundle savedInstanceState) {  
 super.onSaveInstanceState(savedInstanceState);  
 savedInstanceState.putInt("seconds", seconds);  
 savedInstanceState.putBoolean("running", running);  
 savedInstanceState.putBoolean("WasRunning",wasRunning);  
 }  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_stopwatch*);  
 if (savedInstanceState != null) {  
 seconds = savedInstanceState.getInt("seconds");  
 running = savedInstanceState.getBoolean("running");  
 wasRunning = savedInstanceState.getBoolean("WasRunning");  
 }  
 runTimer();  
 }  
  
 @Override protected void onPause() {  
 super.onPause();  
 wasRunning = running;  
 running = false;  
 }  
  
 @Override protected void onResume() {  
 super.onResume();  
 if(wasRunning)running = true;  
 }  
  
 public void onClickStart(View view) {//Вызывается при щелчке на кнопке Start  
 running = true; //Запустить секундомер  
 wasRunning = true;  
 }  
  
 public void onClickStop(View view) {//Вызывается при щелчке на кнопке Stop  
 running = false; //Остановить секундомер  
 wasRunning = false;  
 }  
  
 public void onClickReset(View view) {//Вызывается при щелчке на кнопке Reset  
 running = false;  
 wasRunning = false;  
 seconds = 0;//Остановить секундомер и присвоить счетчику секунд 0  
 }  
  
 private void runTimer() {  
 final TextView timeView = (TextView) findViewById(R.id.*textView*);  
 final Handler handler = new Handler();  
 handler.post(new Runnable() {  
 @Override  
 public void run() {  
 int hours = seconds / 3600;  
 int minutes = (seconds % 3600) / 60;  
 int secs = seconds % 60;  
 String time = String.*format*(Locale.*getDefault*(), "%d:%02d:%02d", hours, minutes, secs);  
 timeView.setText(time);  
 if (running) {  
 seconds++;  
 }  
 handler.postDelayed(this, 1000);  
 }  
 });  
 }  
  
}

**Контрольные вопросы**

1. Что можно делать с помощью объекта Handlers?

Handler — класс Android, который может использоваться для планирования выполнения кода в некоторый момент в будущем. Также класс может использоваться для передачи кода, который должен выполняться в другом программном потоке.

2. Опишите жизненный цикл активности и его элементы.

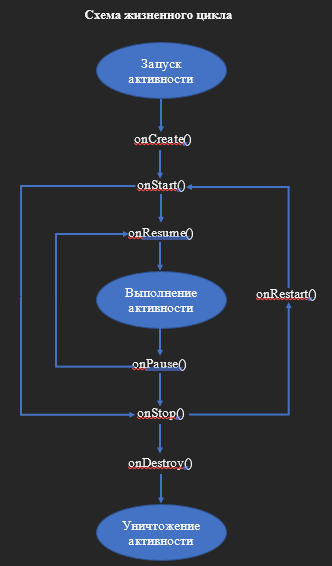


Рис. 1 – общая схема жизненного цикла

3. Как сохранить текущее состояния?

Небоходимо реализовать метод onSaveInstanceState() где состояние сохраняется перед приостановкой/уничтожением активности. А в методе onCreate() восстанавливать значения активности.

4. Приведите пример реализации методов onPause и onResume.

@Override protected void onPause() {  
 super.onPause();  
 wasRunning = running;  
 running = false;  
}  
  
@Override protected void onResume() {  
 super.onResume();  
 if(wasRunning)running = true;  
}