Министерство науки и высшего образования Российской Федерации

**Федеральное государственное бюджетное образовательное учреждение**

**высшего образования**

«Владимирский государственный университет имени Александра Григорьевича и Николая Григорьевича Столетовых»

(ВлГУ)

Кафедра информационных систем

и программной инженерии

**Лабораторная работа № 4-5**

**по дисциплине**

**«Технологии разработки мобильных приложений»**

Тема работы: «Работа с сенсорами на ОС Android», «Работа с БД на ОС Android»

Выполнил:

ст. гр. ПРИ-120

Ляпанов А.А.

Принял:

пр. каф. ИСПИ  
Петрова А.И.

Владимир, 2023

**Цель работы для лабораторной работы 4**

Изучить работу сенсоров на примере платформы Android и разработать мобильное приложение, которое будет считывать сигналы сенсоров устройства и представлять их в удобном для восприятия виде.

**Вариант 3**

**Задание**

1. Используя магнитный сенсор сделать компас, который будет примерно определять, в каком направлении расположен телефон.
2. Используя датчик освещенности менять яркость экрана в зависимости от уровня света в комнате.

**Ход работы**

1. Первым шагом был реализован пример из методических указаний.
2. Далее было необходимо выполнить преобразования, для выполнения работы по индивидуальному варианту.
3. Задумка была реализовать две Activity: 1 – начальная страница приложения, на которой отображаются данные, считываемые сенсорами; 2- компас, отображающий отклонение от севера в градусах и показывающий направление телефона.
4. В листинге 1 представлен код, демонстрирующий xml разметку первой Activity. (activity-main.xml).

**Листинг 1**

<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/textView4"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/accelerometr"  
 android:textColor="@color/red"  
 android:textSize="20dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.502"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.051" />  
  
 <TextView  
 android:id="@+id/mag3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hello World!"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.544" />  
  
 <TextView  
 android:id="@+id/mag2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hello World!"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.484" />  
  
 <TextView  
 android:id="@+id/acc2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hello World!"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.502"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.212" />  
  
 <TextView  
 android:id="@+id/light1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hello World!"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.502"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.932" />  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/light"  
 android:textSize="20dp"  
 android:textColor="@color/red"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.502"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.852" />  
  
 <TextView  
 android:id="@+id/prox1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hello World!"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.502"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.772" />  
  
 <TextView  
 android:id="@+id/textView3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/magnetic"  
 android:textColor="@color/red"  
 android:textSize="20dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.502"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.372" />  
  
 <TextView  
 android:id="@+id/mag1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hello World!"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.502"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.429" />  
  
 <TextView  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="@string/proximity"  
 android:textSize="20dp"  
 android:textColor="@color/red"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.502"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.692" />  
  
 <TextView  
 android:id="@+id/acc3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hello World!"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.502"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.292" />  
  
 <TextView  
 android:id="@+id/acc1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Hello World!"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.502"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.132" />  
  
 <TextView  
 android:id="@+id/textView2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="4 лаба Сани"  
 android:textSize="20dp"  
 android:textColor="@color/blue"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"/>  
  
 <Button  
 android:id="@+id/goToCompas"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Открыть компас"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.498"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="@+id/textView2"  
 app:layout\_constraintVertical\_bias="0.629" />  
  
  
</androidx.constraintlayout.widget.ConstraintLayout>

1. В листинге 2 представлен код, демонстрирующий логику первой Activity.

**Листинг 2**

package com.labsmobilka.lab4;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.annotation.SuppressLint;  
import android.content.Intent;  
import android.hardware.Sensor;  
import android.hardware.SensorEvent;  
import android.hardware.SensorEventListener;  
import android.hardware.SensorManager;  
import android.os.Bundle;  
import android.view.View;  
import android.view.WindowManager;  
import android.widget.Button;  
import android.widget.TextView;  
  
public class MainActivity extends AppCompatActivity implements SensorEventListener, View.OnClickListener {  
  
 TextView accelerometerX;  
 TextView accelerometerY;  
 TextView accelerometerZ;  
 TextView magneticX;  
 TextView magneticY;  
 TextView magneticZ;  
 TextView proximity;  
 TextView light;  
 Button goToCompas;  
 SensorManager sensorManager;  
 Sensor accelerometerSensor;  
 Sensor proximitySensor;  
 Sensor magneticSensor;  
 Sensor lightSensor;  
 Float value;  
  
 @SuppressLint("CutPasteId")  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_main);  
 goToCompas = (Button) findViewById(R.id.goToCompas);  
 accelerometerX = (TextView) findViewById(R.id.acc1);  
 accelerometerY = (TextView) findViewById(R.id.acc2);  
 accelerometerZ = (TextView) findViewById(R.id.acc3);  
 magneticX = (TextView) findViewById(R.id.mag1);  
 magneticY = (TextView) findViewById(R.id.mag2);  
 magneticZ = (TextView) findViewById(R.id.mag3);  
 light = (TextView) findViewById(R.id.light1);  
 proximity = (TextView) findViewById(R.id.prox1);  
 sensorManager = (SensorManager) getSystemService(SENSOR\_SERVICE);  
 accelerometerSensor = sensorManager.getDefaultSensor(Sensor.TYPE\_ACCELEROMETER);  
 proximitySensor = sensorManager.getDefaultSensor(Sensor.TYPE\_PROXIMITY);  
 magneticSensor = sensorManager.getDefaultSensor(Sensor.TYPE\_MAGNETIC\_FIELD);  
 lightSensor = sensorManager.getDefaultSensor(Sensor.TYPE\_LIGHT);  
 goToCompas.setOnClickListener(this);  
 }  
  
 @SuppressLint("SetTextI18n")  
 @Override  
 public void onSensorChanged(SensorEvent sensorEvent) {  
 if (sensorEvent.sensor.getType() == Sensor.TYPE\_ACCELEROMETER) {  
 accelerometerX.setText(Float.toString(sensorEvent.values[0]));  
 accelerometerY.setText(Float.toString(sensorEvent.values[1]));  
 accelerometerZ.setText(Float.toString(sensorEvent.values[2]));  
 }  
 if (sensorEvent.sensor.getType() == Sensor.TYPE\_MAGNETIC\_FIELD) {  
 magneticX.setText(Float.toString(sensorEvent.values[0]));  
 magneticY.setText(Float.toString(sensorEvent.values[1]));  
 magneticZ.setText(Float.toString(sensorEvent.values[2]));  
 }  
 if (sensorEvent.sensor.getType() == Sensor.TYPE\_PROXIMITY) {  
 proximity.setText(Float.toString(sensorEvent.values[0]));  
 }  
 if (sensorEvent.sensor.getType() == Sensor.TYPE\_LIGHT) {  
 light.setText(Float.toString(sensorEvent.values[0]));  
 value = sensorEvent.values[0];  
 WindowManager.LayoutParams layoutParams =getWindow().getAttributes();  
 layoutParams.screenBrightness = (value / 100);  
 getWindow().setAttributes(layoutParams);  
 }  
 }  
  
  
 @Override  
 protected void onStart() {  
 super.onStart();  
 sensorManager.registerListener(this, accelerometerSensor, SensorManager.SENSOR\_DELAY\_NORMAL);  
 sensorManager.registerListener(this, magneticSensor, SensorManager.SENSOR\_DELAY\_NORMAL);  
 sensorManager.registerListener(this, proximitySensor, SensorManager.SENSOR\_DELAY\_NORMAL);  
 sensorManager.registerListener(this, lightSensor, SensorManager.SENSOR\_DELAY\_UI);  
 }  
  
 @Override  
 protected void onStop() {  
 super.onStop();  
 sensorManager.unregisterListener(this, accelerometerSensor);  
 sensorManager.unregisterListener(this, magneticSensor);  
 sensorManager.unregisterListener(this, proximitySensor);  
 sensorManager.unregisterListener(this, lightSensor);  
 }  
  
 @Override  
 public void onAccuracyChanged(Sensor sensor, int i) {  
  
 }  
  
 @Override  
 public void onClick(View view) {  
 if (view.getId() == R.id.goToCompas) {  
 Intent intent = new Intent(this, Compas.class);  
 startActivity(intent);  
 }  
 }  
}

1. В листинге 3 представлен код, демонстрирующий логику второй Activity.

**Листинг 3**

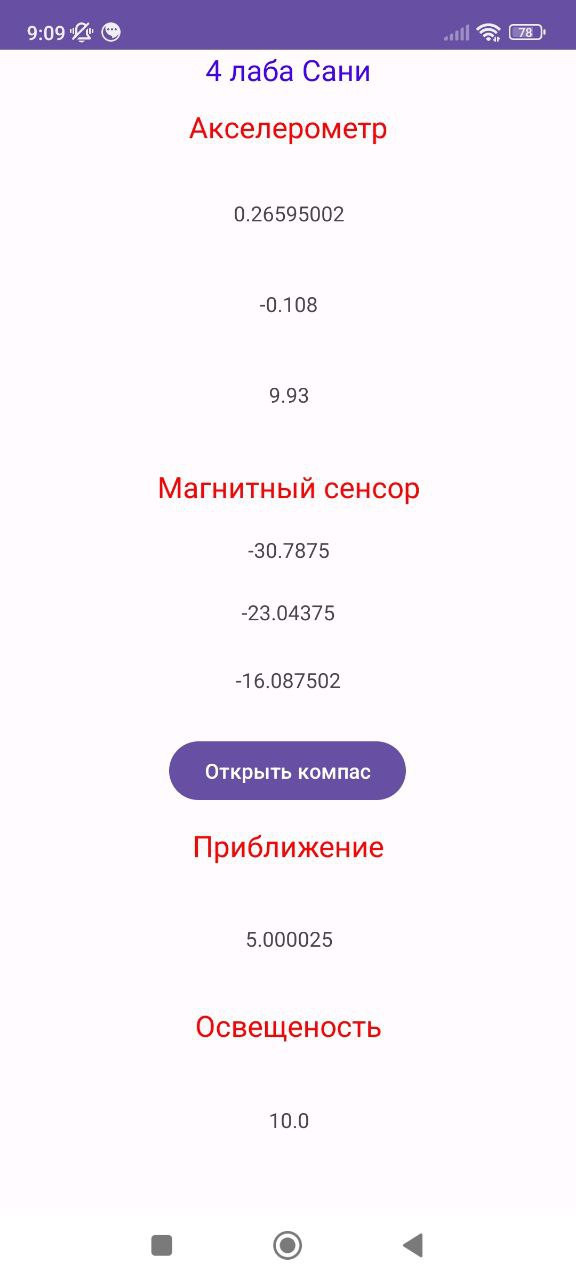
<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 android:background="@color/black"  
 tools:context=".Compas">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Компас"  
 android:textColor="@color/white"  
 android:textSize="20dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.022"  
 />  
  
 <TextView  
 android:id="@+id/degree"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:textColor="@color/white"  
 android:textSize="20dp"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.271" />  
  
 <ImageView  
 android:id="@+id/dynamicImage"  
 android:layout\_width="385dp"  
 android:layout\_height="385dp"  
 app:srcCompat="@drawable/img\_compass\_dinamic"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.8" />  
  
 <ImageView  
 android:id="@+id/staticImage"  
 android:layout\_width="385dp"  
 android:layout\_height="385dp"  
 app:srcCompat="@drawable/img\_compass\_static"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.8" />  
</androidx.constraintlayout.widget.ConstraintLayout>

1. В листинге 4 представлен код, демонстрирующий логику второй Activityю

**Листинг 4**

package com.labsmobilka.lab4;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.annotation.SuppressLint;  
import android.hardware.Sensor;  
import android.hardware.SensorEvent;  
import android.hardware.SensorEventListener;  
import android.hardware.SensorManager;  
import android.os.Bundle;  
import android.view.animation.Animation;  
import android.view.animation.RotateAnimation;  
import android.widget.ImageView;  
import android.widget.TextView;  
  
public class Compas extends AppCompatActivity implements SensorEventListener {  
 ImageView dynamicImage;  
 TextView tvDegree;  
 float currentDegree;  
 SensorManager sensorManager;  
 Sensor orientationSensor;  
  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.activity\_compas);  
 init();  
 }  
  
 private void init() {  
 dynamicImage = findViewById(R.id.dynamicImage);  
 tvDegree = findViewById(R.id.degree);  
 sensorManager = (SensorManager) getSystemService(SENSOR\_SERVICE);  
 orientationSensor = sensorManager.getDefaultSensor(Sensor.TYPE\_ORIENTATION);  
 }  
  
 @Override  
 protected void onResume() {  
 super.onResume();  
 sensorManager.registerListener(this, orientationSensor, SensorManager.SENSOR\_DELAY\_GAME);  
 }  
  
 @Override  
 protected void onPause() {  
 super.onPause();  
 sensorManager.unregisterListener(this);  
 }  
  
 @SuppressLint("SetTextI18n")  
 @Override  
 public void onSensorChanged(SensorEvent sensorEvent) {  
 float degree = Math.round(sensorEvent.values[0]);  
  
 tvDegree.setText("Отклонение от севера: " + Float.toString(degree) + "°");  
  
 RotateAnimation ra = new RotateAnimation(currentDegree, -degree, Animation.RELATIVE\_TO\_SELF, 0.5f, Animation.RELATIVE\_TO\_SELF, 0.5f);  
  
 ra.setDuration(210);  
  
 ra.setFillAfter(true);  
  
 dynamicImage.startAnimation(ra);  
 currentDegree = -degree;  
 }  
  
 @Override  
 public void onAccuracyChanged(Sensor sensor, int i) {  
  
 }  
}

1. На рисунке 1 представлен скриншот, демонстрирующий работу первой Activity.

  
Рисунок 1 (Скриншот, демонстрирующий первую Activity)

1. На рисунке 2 представлен скриншот, демонстрирующий работу второй Activity.

  
Рисунок 2 (Скриншот, демонстрирующий вторую Activity)Hbc

**Вывод по 4 лабораторной работе**

В ходе выполнения лабораторной работы была проделана работа по изучению сенсоров на примере платформы Android и разработано мобильно приложение, которое считывает сигналы сенсоров устройства и представляет данные в удобном для восприятия вид.

**Цель работы для лабораторной работы 5**

Изучить работу с базой данных в ОС Android на примере SQLite и получить практические навыки разработки мобильного приложения, взаимодействующего с базой данных.

**Вариант 1**

**Задание**

1. Реализовать функционал изменения записи.
2. Реализовать отображение информации о записях в БД на экране устройства.

**Ход работы**

1. Первым шагом был реализован пример из лабораторной работы.
2. Далее были внесены необходимые правки, которые нужные для реализации индивидуального задания.
3. В листинге 1 представлен код, демонстрирующий xml для Activity.

**Листинг 1**

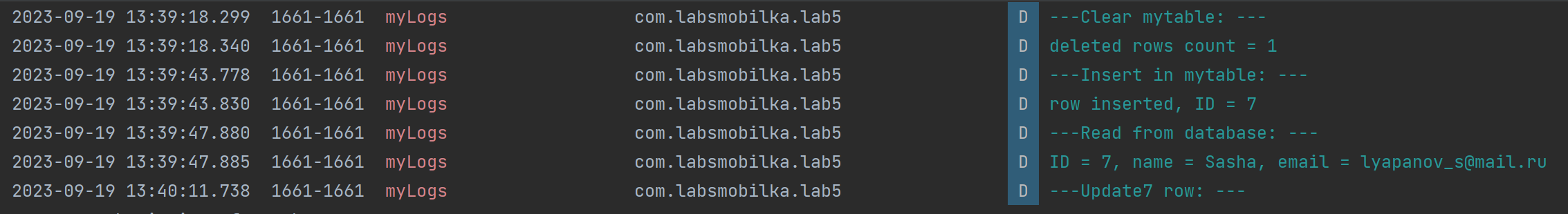
<?xml version="1.0" encoding="utf-8"?>  
<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"  
 xmlns:app="http://schemas.android.com/apk/res-auto"  
 xmlns:tools="http://schemas.android.com/tools"  
 android:layout\_width="match\_parent"  
 android:layout\_height="match\_parent"  
 tools:context=".MainActivity">  
  
 <TextView  
 android:id="@+id/textView1"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Name"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.08"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.125" />  
  
 <TextView  
 android:id="@+id/necID"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Necessary ID"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.03"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.367" />  
  
 <TextView  
 android:id="@+id/textView2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Email"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.08"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.247" />  
  
 <Button  
 android:id="@+id/button"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Add"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.168"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="@id/button2"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <Button  
 android:id="@+id/button4"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Update"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.592"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="@id/button2"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.609" />  
  
 <Button  
 android:id="@+id/button2"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Read"  
 tools:layout\_editor\_absoluteX="160dp"  
 tools:layout\_editor\_absoluteY="331dp"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintLeft\_toLeftOf="parent"  
 app:layout\_constraintRight\_toRightOf="parent" />  
  
 <Button  
 android:id="@+id/button3"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Clear"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.837"  
 app:layout\_constraintLeft\_toLeftOf="@id/button2"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.499" />  
  
 <Button  
 android:id="@+id/button5"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Find by id"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.422"  
 app:layout\_constraintLeft\_toLeftOf="@id/button2"  
 app:layout\_constraintRight\_toRightOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 app:layout\_constraintVertical\_bias="0.609" />  
  
 <EditText  
 android:id="@+id/editText1"  
 android:layout\_width="250dp"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="72dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.543"  
 app:layout\_constraintStart\_toEndOf="@+id/textView2"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <EditText  
 android:id="@+id/editText3"  
 android:layout\_width="250dp"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="244dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.541"  
 app:layout\_constraintStart\_toEndOf="@+id/textView2"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <EditText  
 android:id="@+id/editText2"  
 android:layout\_width="250dp"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="160dp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.543"  
 app:layout\_constraintStart\_toEndOf="@+id/textView2"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <TextView  
 android:id="@+id/dataFromDB"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:text="Data From DB"  
 android:textSize="20dp"  
 android:textColor="@color/red"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/button2" />  
  
</androidx.constraintlayout.widget.ConstraintLayout>

1. В листинге 2 представлен код, демонстрирующий код MainActivity.java, который содержит в себе всю логику работы с БД.

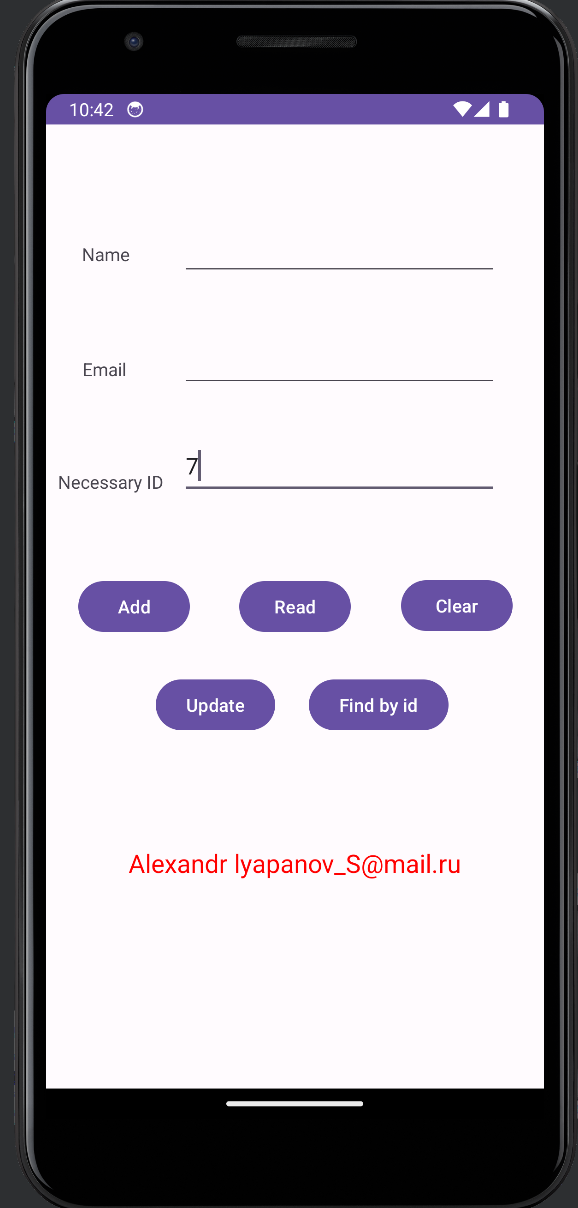
**Листинг 2**

package com.labsmobilka.lab5;  
  
import static android.app.DownloadManager.*COLUMN\_ID*;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.content.ContentValues;  
import android.content.Context;  
import android.database.Cursor;  
import android.database.sqlite.SQLiteDatabase;  
import android.database.sqlite.SQLiteOpenHelper;  
import android.os.Bundle;  
import android.util.Log;  
import android.view.View;  
import android.widget.Button;  
import android.widget.EditText;  
import android.widget.TextView;  
  
public class MainActivity extends AppCompatActivity {  
 final String LOG\_TAG = "myLogs";  
 Button buttonAdd, buttonRead, buttonClear, buttonUpdate, buttonFindByID;  
 EditText editName, editEmail, necId;  
 TextView dataFromDB;  
 DBHelper dbHelper;  
  
 @Override  
 protected void onCreate(Bundle savedInstanceState) {  
 super.onCreate(savedInstanceState);  
 setContentView(R.layout.*activity\_main*);  
 buttonAdd = (Button) findViewById(R.id.*button*);  
 buttonAdd.setOnClickListener(this:: onClick);  
 buttonRead = (Button) findViewById(R.id.*button2*);  
 buttonRead.setOnClickListener(this:: onClick);  
 buttonClear = (Button) findViewById(R.id.*button3*);  
 buttonClear.setOnClickListener(this:: onClick);  
 buttonUpdate = (Button) findViewById(R.id.*button4*);  
 buttonUpdate.setOnClickListener(this:: onClick);  
 buttonFindByID = (Button) findViewById(R.id.*button5*);  
 buttonFindByID.setOnClickListener(this:: onClick);  
 editName = (EditText) findViewById(R.id.*editText1*);  
 editEmail = (EditText) findViewById(R.id.*editText2*);  
 necId = (EditText) findViewById(R.id.*editText3*);  
 dataFromDB = (TextView) findViewById(R.id.*dataFromDB*);  
 dbHelper = new DBHelper(this);  
 }  
  
  
 protected void onClick(View v) {  
 //создаем объект для данных  
 ContentValues cv = new ContentValues();  
  
 //подключаемя к БД  
 SQLiteDatabase db = dbHelper.getWritableDatabase();  
 if (v.getId() == R.id.*button*) {  
 String name = editName.getText().toString();  
 String email = editEmail.getText().toString();  
 Log.*d*(LOG\_TAG, "---Insert in mytable: ---");  
 //Подготовим данные для вставки в виде пар: наименование стоблца - значение  
 cv.put("name", name);  
 cv.put("email", email);  
 //Вставляем запись и получаем ее id  
 long rowId = db.insert("mytable", null, cv);  
 Log.*d*(LOG\_TAG, "row inserted, ID = " + rowId);  
 editName.setText(null);  
 editEmail.setText(null);  
 necId.setText(null);  
 }  
 if (v.getId() == R.id.*button2*) {  
 Log.*d*(LOG\_TAG, "---Read from database: ---");  
 //Делаем запрос всех хранящихся данных в таблице и получаем объект Cursor  
 Cursor cursor = db.query("mytable", null, null, null, null, null, null);  
 //Ставим позицию курсор на первую строку выборки. Если же в полученной выборке нет строк, вернется false  
 if (cursor.moveToFirst()) {  
 //Определяем номер столбцов по имени в выборке  
 int idColIndex = cursor.getColumnIndex("id");  
 int nameColIndex = cursor.getColumnIndex("name");  
 int emailColIndex = cursor.getColumnIndex("email");  
 do {  
 //получаем значения по номерам столбцов и пишем все в лог  
 Log.*d*(LOG\_TAG,"ID = " + cursor.getInt(idColIndex) + ", name = "  
 + cursor.getString(nameColIndex) + ", email = "  
 + cursor.getString(emailColIndex));  
 } while (cursor.moveToNext());  
 } else  
 Log.*d*(LOG\_TAG, "0 rows");  
 cursor.close();  
 }  
 if (v.getId() == R.id.*button3*) {  
 Log.*d*(LOG\_TAG, "---Clear mytable: ---");  
 int clearCount = db.delete("mytable", null, null);  
 Log.*d*(LOG\_TAG, "deleted rows count = " + clearCount);  
 editName.setText(null);  
 editEmail.setText(null);  
 necId.setText(null);  
 }  
 if (v.getId() == R.id.*button4*) {  
 String name = editName.getText().toString();  
 String email = editEmail.getText().toString();  
 Log.*d*(LOG\_TAG, "---Update" + necId.getText().toString() + " row: ---");  
 ContentValues contentValues = new ContentValues();  
 contentValues.put("name", name);  
 contentValues.put("email", email);  
 db.update("mytable", contentValues, "id =" + necId.getText().toString(), null);  
 editName.setText(null);  
 editEmail.setText(null);  
 necId.setText(null);  
 }  
 if (v.getId() == R.id.*button5*) {  
 String id = necId.getText().toString();  
 if (id != null) {  
 Cursor cursor = db.rawQuery("select \* from mytable where ID=?", new String[]{String.*valueOf*(id)});  
 cursor.moveToFirst();  
 int nameColIndex = cursor.getColumnIndex("name");  
 int emailColIndex = cursor.getColumnIndex("email");  
 String name = cursor.getString(nameColIndex);  
 String email = cursor.getString(emailColIndex);  
 dataFromDB.setText(name + " " + email);  
 }  
 }  
 dbHelper.close();  
 }  
  
 class DBHelper extends SQLiteOpenHelper {  
 public static final String *COLUMN\_ID* = "id";  
  
 public DBHelper(Context context) {  
 super(context, "myDB", null, 1);  
 }  
 public void onCreate(SQLiteDatabase db) {  
 Log.*d*(LOG\_TAG, "---onCreate database");  
 //создаем таблицу и необходимые поля  
 db.execSQL("create table mytable ("  
 + "id integer primary key autoincrement,"  
 + "name text,"  
 + "email text" + ");");  
 }  
 public void onUpgrade(SQLiteDatabase db, int oldVersion, int newVersion) {  
  
 }  
 }  
}

1. На рисунке 1 представлен скриншот, демонстрирующий работу приложения через логи в консоли.

  
Рисунок 1 (Скриншот, демонстрирующий работу приложения через логи в консоли)

1. На рисунке 2- представлен скриншот, демонстрирующий работу приложения на экране устройства.

  
Рисунок 2 (Скриншот, демонстрирующий работу приложения на экране устройства)

**Вывод для лабораторной работы 5**

В ходе выполнения лабораторной работы были получены навыки работы с БД в ОС Android на примере SQLite и получены практические навыки разработки мобильного приложения, взаимодействующего с БД.

**Вывод**

В результате проделанных работ были получены навыки работы с Activity, сенсорами мобильных устройств, а также навыки по созданию, редактированию, чтению и удалению данных в БД для ОС Android. Весь код выложен в публичном репозитории <https://github.com/SashaLyapanov/Mobile_development>.