

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное  
образовательное учреждение высшего образования  
«Комсомольский-на-Амуре государственный университет»

Факультет компьютерных технологий

Кафедра ПУРИС

Лабораторная работа №11  
по дисциплине «Программирование мобильных устройств»

Студент группы 0ВТб-1

Н.Д. Малышев

Преподаватель

В.А. Тихомиров

## 1 Разработка приложения

По заданию необходимо сделать две таблицы для базы данных: Авторы и Книги. Приложение должно выводить на экран список книг заданного автора и фото автора.

Создаются два макета страницы `activity_main.xml` (Листинг 1.4) и `user_main.xml` (Листинг 1.5). На первом макете выводится список авторов, на втором информация о выбранном авторе, список книг и фото этого автора.

В активности `MainActivity` (листинг 1.1) написан код, реализующий подключение базы данных и вывод таблицы `authors`.

В листинге 1.2 приведен код `DatabaseHelper`, который реализует подключение уже существующей базы данных `myLibrary.db`, которая была создана заранее и в нее уже занесены данные, в том числе и фото авторов с помощью типа данных `BLOB`. Также определены названия таблиц и их колонки.

В активности `UserActivity` (листинг 1.3) написан код, реализующий подключение базы данных и вывод в поля, таких данных как, фото автора, имя, дата рождения, описания и список книг который он написал.

### Листинг 1.1 – MainActivity

```
package ru.lab_111;

import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.view.View;
import android.widget.AdapterView;
import android.widget.SimpleCursorAdapter;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.os.Bundle;
import android.widget.ListView;
public class MainActivity extends AppCompatActivity {
    ListView userList; // Список для отображения книг
    DatabaseHelper databaseHelper; // Объект класса для работы с
    базой данных
    SQLiteDatabase db; // Объект для работы с базой данных
    Cursor userCursor; // Курсор для получения результатов
    запроса к базе данных
```

```

SimpleCursorAdapter userAdapter; // Адаптер для связи данных
из курсора с пользовательским интерфейсом

@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    userList = findViewById(R.id.list); // Инициализация
списка
    userList.setOnItemClickListener(new
AdapterView.OnItemClickListener() {
        @Override
        public void onItemClick(AdapterView<?> parent, View
view, int position, long id) {
            // Обработчик клика по элементу списка
            Intent intent = new
Intent(getApplicationContext(), UserActivity.class);
            intent.putExtra("id", id);
            startActivity(intent);
        }
    });

    databaseHelper = new
DatabaseHelper(getApplicationContext()); // Создание экземпляра
класса для работы с базой данных
    databaseHelper.create_db(); // Создание базы данных,
если она не существует
}

@Override
public void onResume() {
    super.onResume();
    db = databaseHelper.open(); // Открытие базы данных
    userCursor = db.rawQuery("select * from " +
DatabaseHelper.TABLE1, null); // Выполнение запроса к базе
данных
    String[] headers = new
String[]{DatabaseHelper.COLUMN_NAME,
DatabaseHelper.COLUMN_BIRTHDAY}; // Заголовки столбцов
    userAdapter = new SimpleCursorAdapter(this,
android.R.layout.two_line_list_item,
        userCursor, headers, new
int[]{android.R.id.text1, android.R.id.text2}, 0); // Создание
адаптера
    userList.setAdapter(userAdapter); // Установка адаптера
}

@Override
public void onDestroy() {
    super.onDestroy();
    db.close(); // Заккрытие базы данных
    userCursor.close(); // Заккрытие курсора
}

```

```

    }
}

```

## Листинг 1.2 – DatabaseHelper

```

package ru.lab_111;

import android.database.SQLException;
import android.database.sqlite.SQLiteOpenHelper;
import android.database.sqlite.SQLiteDatabase;
import android.content.Context;
import android.util.Log;

import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;

class DatabaseHelper extends SQLiteOpenHelper {
    private static String DB_PATH; // Полный путь к базе данных
    private static String DB_NAME = "myLibrary.db"; // Имя базы
данных
    private static final int SCHEMA = 1; // Версия базы данных
    static final String TABLE = "books"; // Название таблицы для
книг
    static final String TABLE1 = "authors"; // Название таблицы
для авторов
    // Названия столбцов
    static final String COLUMN_ID_BOOK = "_id";
    static final String COLUMN_TITLE = "title";
    static final String COLUMN_YEAR = "year";
    static final String COLUMN_AUTHOR = "id_author";
    static final String COLUMN_ID_AUTHOR = "_id";
    static final String COLUMN_NAME = "name";
    static final String COLUMN_BIRTHDAY = "birthday";
    static final String COLUMN_OTHER = "other";
    static final String COLUMN_PHOTO = "photo";
    private Context myContext;

    DatabaseHelper(Context context) {
        super(context, DB_NAME, null, SCHEMA);
        this.myContext=context;
        DB_PATH =context.getFilesDir().getPath() + DB_NAME; //
Получение полного пути к базе данных
    }

    @Override
    public void onCreate(SQLiteDatabase db) { }

    @Override
    public void onUpgrade(SQLiteDatabase db, int oldVersion,
int newVersion) { }

```

```

void create_db(){
    // Создание базы данных, если она не существует
    File file = new File(DB_PATH);
    if (!file.exists()) {
        try(InputStream myInput =
myContext.getAssets().open(DB_NAME);
        OutputStream myOutput = new
FileOutputStream(DB_PATH)) {
            byte[] buffer = new byte[1024];
            int length;
            while ((length = myInput.read(buffer)) > 0) {
                myOutput.write(buffer, 0, length);
            }
            myOutput.flush();
        }
        catch(IOException ex){
            Log.d("DatabaseHelper", ex.getMessage());
        }
    }
}

public SQLiteDatabase open()throws SQLException {
    // Открытие базы данных для чтения и записи
    return SQLiteDatabase.openDatabase(DB_PATH, null,
SQLiteDatabase.OPEN_READWRITE);
}
}

```

### Листинг 1.3 – UserActivity

```

package ru.lab_111;
import androidx.appcompat.app.AppCompatActivity;

import android.content.Intent;
import android.database.Cursor;
import android.database.sqlite.SQLiteDatabase;
import android.graphics.Bitmap;
import android.graphics.BitmapFactory;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.ImageView;
import android.widget.ListView;
import android.widget.SimpleCursorAdapter;
import android.widget.TextView;

public class UserActivity extends AppCompatActivity {
    TextView nameBox; // Поле для отображения имени автора
    TextView yearBox; // Поле для отображения года рождения
автора
    TextView otherBox; // Поле для отображения дополнительной
информации об авторе
    ImageView photoAuthor; // Изображение автора
    ListView userList; // Список для отображения книг

```

```

        DatabaseHelper sqlHelper; // Объект класса для работы с
        базой данных
        SQLiteDatabase db; // Объект для работы с базой данных
        Cursor userCursor; // Курсор для получения результатов
        запроса к базе данных
        Cursor bookCursor; // Курсор для получения результатов
        запроса к базе данных
        SimpleCursorAdapter userAdapter; // Адаптер для связи данных
        из курсора с пользовательским интерфейсом
        long userId = 0; // Идентификатор автора

        @Override
        protected void onCreate(Bundle savedInstanceState) {
            super.onCreate(savedInstanceState);
            setContentView(R.layout.activity_user);

            nameBox = findViewById(R.id.name); // Инициализация поля
            имени автора
            yearBox = findViewById(R.id.year); // Инициализация поля
            года рождения автора
            otherBox = findViewById(R.id.otherText); //
            Инициализация поля дополнительной информации об авторе

            userList = findViewById(R.id.bookList); // Инициализация
            списка
            photoAuthor = findViewById(R.id.photoAuthor); //
            Инициализация изображения автора

            sqlHelper = new DatabaseHelper(this); // Создание
            экземпляра класса для работы с базой данных
            db = sqlHelper.open(); // Открытие базы данных

            Bundle extras = getIntent().getExtras();
            if (extras != null)
                userId = extras.getLong("id");

            userCursor = db.rawQuery("select * from " +
            DatabaseHelper.TABLE1 + " where " +
            DatabaseHelper.COLUMN_ID_AUTHOR + "=?", new
            String[]{String.valueOf(userId)}); // Выполнение запроса к базе
            данных
            userCursor.moveToFirst(); // Перемещение курсора на
            первую запись
            nameBox.setText(userCursor.getString(1)); // Установка
            имени автора в поле
            yearBox.setText(userCursor.getString(2)); // Установка
            года рождения автора в поле
            otherBox.setText(userCursor.getString(3)); // Установка
            дополнительной информации об авторе в поле

            byte[] byteArray = userCursor.getBlob(4); // Получение
            байтового массива с изображением автора
            Bitmap bitmap = BitmapFactory.decodeByteArray(byteArray,

```

```

0, byteArray.length); // Преобразование байтового массива в
Bitmap
    photoAuthor.setImageBitmap(bitmap); // Установка
изображения автора

    userCursor.close(); // Закрытие курсора

    bookCursor = db.rawQuery("select * from " +
DatabaseHelper.TABLE + " where " +
        DatabaseHelper.COLUMN_AUTHOR + "=?", new
String[]{String.valueOf(userId)}); // Выполнение запроса к базе
данных
    String[] headers = new
String[]{DatabaseHelper.COLUMN_TITLE,
DatabaseHelper.COLUMN_YEAR}; // Заголовки столбцов
    userAdapter = new SimpleCursorAdapter(this,
android.R.layout.two_line_list_item,
        bookCursor, headers, new
int[]{android.R.id.text1, android.R.id.text2}, 0); // Создание
адаптера
    userList.setAdapter(userAdapter); // Установка адаптера

}
}

```

#### Листинг 1.4 – activity\_main

```

<?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:orientation="vertical"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <androidx.appcompat.widget.Toolbar
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:minHeight="?actionBarSize"
        android:theme="?actionBarTheme"
        android:background="@color/darkGray"
        tools:ignore="MissingConstraints">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="AUTHORS"
            android:textSize="35sp"
            android:textColor="@color/white"
            android:textStyle="bold"
            android:layout_gravity="center"/>
    </Toolbar>
</LinearLayout>

```

```

</androidx.appcompat.widget.Toolbar>

<ListView
    android:id="@+id/list"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginTop="3dp"
    android:dividerHeight="3dp"
    android:divider="@color/darkGray"/>

</LinearLayout>

```

### Листинг 1.5 – activity\_user

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:orientation="vertical"
    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">

    <androidx.appcompat.widget.Toolbar
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:background="@color/darkGray"
        android:minHeight="?actionBarSize"
        android:theme="?actionBarTheme">

        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:layout_gravity="center"
            android:text="ABOUT THE AUTHOR"
            android:textColor="@color/white"
            android:textSize="35sp"
            android:textStyle="bold" />
    </androidx.appcompat.widget.Toolbar>

    <ImageView
        android:id="@+id/photoAuthor"
        android:layout_width="180dp"
        android:layout_height="260dp"
        android:layout_gravity="center"
        android:adjustViewBounds="true"
        android:scaleType="centerCrop" />

    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:id="@+id/name"
        app:layout_constraintBottom_toTopOf="@+id/year"

```



```

        android:textColor="@color/darkGray"
        android:textStyle="bold"
        android:layout_gravity="center"/>

<TextView
    android:id="@+id/year"
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:layout_gravity="center"
    android:layout_marginTop="20dp"
    android:textColor="@color/darkGray"
    android:textStyle="bold" />

<TextView
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/otherText"
    android:textColor="@color/darkGray"
    android:textStyle="bold"
    android:layout_gravity="center" />

<ListView
    android:id="@+id/bookList"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:layout_marginTop="3dp"
    android:dividerHeight="3dp"
    android:divider="@color/darkGray"/>

</LinearLayout>

```

Работа приложения представлена на рисунке 1.1 – 1.2.

AUTHORS	
Николай Гоголь	Дата рождения 1 апреля 1809 г.
Михаил Лермонтов	Дата рождения 15 октября 1814 г.
Лев Толстой	Дата рождения 9 сентября 1828 г.
Александр Пушкин	Дата рождения 6 июня 1799 г.

Рисунок 1.1 – Пример работы приложения

ABOUT THE AUTHOR	
	
Михаил Лермонтов	
Дата рождения 15 октября 1814 г.	
Русский поэт, прозаик, драматург, художник. Поручик лейб-гвардии Гусарского полка.	
Бородино	Дата написания 1837
Парус	Дата написания 1841
Смерть поэта	Дата написания 1837
Мцыри	Дата написания 1840

Рисунок 1.2 – Пример работы приложения

### **Список использованных источников**

- 1 РД ФГБОУ ВО «КНАГТУ» 013-2016. Текстовые студенческие работы. Правила оформления. – Введ. 2016-03-10. – Комсомольск-на-Амуре: ФГБОУ ВО «КНАГТУ», 2016. – 55 с.