



Министерство науки и высшего образования Российской Федерации
Калужский филиал
федерального государственного бюджетного
образовательного учреждения высшего образования
**«Московский государственный технический университет имени Н.Э.
Баумана (национальный исследовательский университет)»**
(КФ МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ ИУК «Информатика и управление»

КАФЕДРА ИУК4 «Программное обеспечение ЭВМ, информационные технологии»

ЛАБОРАТОРНАЯ РАБОТА №5

**«Создание и использование собственных источников данных.
Работа со стандартными источниками данных»**

ДИСЦИПЛИНА: «Разработка мобильного ПО»

Выполнил: студент гр. ИУК4-52Б

_____ (_____ Губин Е.В. _____)
(Подпись) (Ф.И.О.)

Проверил:

_____ (_____ Прудяк П.Н. _____)
(Подпись) (Ф.И.О.)

Дата сдачи (защиты):

Результаты сдачи (защиты):

- Балльная оценка:

- Оценка:

Калуга, 2024 г.

Цель: формирование навыков создания собственных источников данных.

Задачи

1. Научиться создавать собственные источники данных (ContentProvider).
2. Научиться использовать собственные источники данных.
3. Научиться работать со стандартными источниками данных: аудио файлами, графическими файлами, списком контактов.
4. Разработать эффективные приложения с учетом аппаратных ограничений мобильных устройств.
5. Научиться реализовывать логику работы приложения с учетом специфики платформы Android

Вариант 9

Вывести список композиций (исполнитель, название) длительность которых находится в определенном диапазоне. Диапазон выбирается с помощью компонентов EditText, находящихся на одной Activity, а список выводится на втором. Диапазон указывается в минутах и секундах

Листинг программы:

MainActivity.kt:

```
package com.example.lw_5

import android.content.Intent
import androidx.appcompat.app.AppCompatActivity
import android.os.Bundle
import android.widget.Button
import android.widget.EditText
import android.widget.Toast
class MainActivity : AppCompatActivity() {
    override fun onCreate(savedInstanceState: Bundle?) {
        super.onCreate(savedInstanceState)
        setContentView(R.layout.activity_main)
        val fromMinutes = findViewById<EditText>(R.id.fromMinutes)
        val fromSeconds = findViewById<EditText>(R.id.fromSeconds)
        val toMinutes = findViewById<EditText>(R.id.toMinutes)
        val toSeconds = findViewById<EditText>(R.id.toSeconds)
        val searchButton = findViewById<Button>(R.id.searchButton)
        searchButton.setOnClickListener {
            try {
                val minSecondsValue =
                    fromSeconds.text.toString().toIntOrNull() ?: 0
                val maxSecondsValue =
```

```

        toSeconds.text.toString().toIntOrNull() ?: 0
    if (minSecondsValue >= 60 || maxSecondsValue >= 60) {
        Toast.makeText(this, "Seconds must be under 60",
            Toast.LENGTH_SHORT).show()
        return@setOnClickListener
    }
    val minDuration =
        (fromMinutes.text.toString().toIntOrNull() ?: 0) * 60 +
            minSecondsValue
    val maxDuration =
        (toMinutes.text.toString().toIntOrNull() ?: 0) * 60 +
            maxSecondsValue
    if (maxDuration < minDuration) {
        Toast.makeText(this, "Max duration must be greater than min duration",
            Toast.LENGTH_SHORT).show()
        return@setOnClickListener
    }
    val intent = Intent(this, TracksActivity::class.java).apply {
        putExtra("MIN_DURATION", minDuration * 1000L)
        putExtra("MAX_DURATION", maxDuration * 1000L)
    }
    startActivity(intent)
} catch (e: NumberFormatException) {
    Toast.makeText(this, "Please enter valid numbers",
        Toast.LENGTH_SHORT).show()
}
}
}
}

```

TracksActivity.kt:

```

package com.example.lw_5

import android.Manifest
import android.content.pm.PackageManager
import android.os.Bundle
import android.provider.MediaStore
import android.util.Log
import android.widget.ArrayAdapter
import android.widget.ListView
import android.widget.Toast
import androidx.appcompat.app.AppCompatActivity
import androidx.core.app.ActivityCompat
import androidx.core.content.ContextCompat
class TracksActivity : AppCompatActivity() {
    companion object {
        private const val PERMISSION_REQUEST_CODE = 1
    }
    private var minDuration: Long = 0

```

```

private var maxDuration: Long = 0
override fun onCreate(savedInstanceState: Bundle?) {
    super.onCreate(savedInstanceState)
    setContentView(R.layout.activity_tracks)
    minDuration = intent.getLongExtra("MIN_DURATION", 0)
    maxDuration = intent.getLongExtra("MAX_DURATION", Long.MAX_VALUE)
    if (checkPermission()) {
        loadTracks()
    } else {
        requestPermission()
    }
}

private fun getPermission(): String {
    return if (android.os.Build.VERSION.SDK_INT < 33) {
        Manifest.permission.READ_EXTERNAL_STORAGE
    } else {
        Manifest.permission.READ_MEDIA_AUDIO
    }
}

private fun checkPermission(): Boolean {
    return ContextCompat.checkSelfPermission(
        this,
        getPermission()
    ) == PackageManager.PERMISSION_GRANTED
}

private fun requestPermission() {
    ActivityCompat.requestPermissions(
        this,
        arrayOf(getPermission()),
        PERMISSION_REQUEST_CODE
    )
}

override fun onRequestPermissionsResult(
    requestCode: Int,
    permissions: Array<out String>,
    grantResults: IntArray
) {
    super.onRequestPermissionsResult(requestCode, permissions,
        grantResults)
    if (requestCode == PERMISSION_REQUEST_CODE &&
        grantResults.isNotEmpty() && grantResults[0] ==
        PackageManager.PERMISSION_GRANTED) {
        loadTracks()
    }
}

private fun loadTracks() {
    val tracks = mutableListOf<String>()
    val projection = arrayOf(
        MediaStore.Audio.Media.ARTIST,
        MediaStore.Audio.Media.TITLE,

```

```

        MediaStore.Audio.Media.DURATION
    )
    val selection = "${MediaStore.Audio.Media.DURATION} BETWEEN ? AND ?"
    val selectionArgs = arrayOf(minDuration.toString(),
        maxDuration.toString())
    contentResolver.query(
        MediaStore.Audio.Media.EXTERNAL_CONTENT_URI,
        projection,
        selection,
        selectionArgs,
        null
   )?.use { cursor ->
        val artistColumn =
            cursor.getColumnIndexOrThrow(MediaStore.Audio.Media.ARTIST)
        val titleColumn =
            cursor.getColumnIndexOrThrow(MediaStore.Audio.Media.TITLE)
        while (cursor.moveToNext()) {
            val artist = cursor.getString(artistColumn)
            val title = cursor.getString(titleColumn)
            tracks.add("$artist - $title")
        }
    }
    val listView = findViewById<ListView>(R.id.tracksListView)
    val adapter = ArrayAdapter(this, android.R.layout.simple_list_item_1,
        tracks)
    listView.adapter = adapter
    if (tracks.isEmpty()) {
        Toast.makeText(this, "No tracks found in the specified duration range",
            Toast.LENGTH_SHORT).show()
    }
}
}

```

activity_main.xml:

```

<?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:id="@+id/main"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".MainActivity">

    <TextView
        android:id="@+id/textView"
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:layout_marginTop="80dp"
    >

```

```
android:text="Длительность"
android:textSize="20sp"
android:textStyle="bold"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toTopOf="parent" />
```

<TextView

```
android:id="@+id/textView2"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginStart="70dp"
android:layout_marginTop="40dp"
android:text="От"
android:textSize="18sp"
android:textStyle="bold|italic"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/textView" />
```

<TextView

```
android:id="@+id/textView3"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="40dp"
android:layout_marginEnd="70dp"
android:text="До"
android:textSize="18sp"
android:textStyle="bold|italic"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintTop_toBottomOf="@+id/textView" />
```

<Button

```
android:id="@+id/searchButton"
android:layout_width="wrap_content"
android:layout_height="wrap_content"
android:layout_marginTop="50dp"
android:text="Display compositions"
app:layout_constraintEnd_toEndOf="parent"
app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/fromSeconds" />
```

<EditText

```
android:id="@+id/fromMinutes"
android:layout_width="100dp"
android:layout_height="48dp"
android:layout_marginStart="40dp"
android:layout_marginTop="40dp"
android:ems="10"
android:hint="Минуты"
android:inputType="number"
```

```

app:layout_constraintStart_toStartOf="parent"
app:layout_constraintTop_toBottomOf="@+id/textView2" />

<EditText
    android:id="@+id/toMinutes"
    android:layout_width="100dp"
    android:layout_height="48dp"
    android:layout_marginTop="40dp"
    android:layout_marginEnd="40dp"
    android:ems="10"
    android:hint="Минуты"
    android:inputType="number"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/textView3" />

<EditText
    android:id="@+id/fromSeconds"
    android:layout_width="100dp"
    android:layout_height="48dp"
    android:layout_marginStart="40dp"
    android:layout_marginTop="40dp"
    android:ems="10"
    android:hint="Секунды"
    android:inputType="number"
    app:layout_constraintStart_toStartOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/fromMinutes" />

<EditText
    android:id="@+id/toSeconds"
    android:layout_width="100dp"
    android:layout_height="48dp"
    android:layout_marginTop="40dp"
    android:layout_marginEnd="40dp"
    android:ems="10"
    android:hint="Секунды"
    android:inputType="number"
    app:layout_constraintEnd_toEndOf="parent"
    app:layout_constraintTop_toBottomOf="@+id/toMinutes" />
</androidx.constraintlayout.widget.ConstraintLayout>

```

activity_tracks.xml

```

<?xml version="1.0" encoding="utf-8"?>
<ListView xmlns:android="http://schemas.android.com/apk/res/android"
    android:id="@+id/tracksListView"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:padding="16dp" />

```

AndroidManifest.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:tools="http://schemas.android.com/tools">

    <uses-permission android:name="android.permission.READ_MEDIA_AUDIO" />
    <uses-permission android:name="android.permission.READ_EXTERNAL_STORAGE"
/>

    <application
        android:allowBackup="true"
        android:dataExtractionRules="@xml/data_extraction_rules"
        android:fullBackupContent="@xml/backup_rules"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportRtl="true"
        android:theme="@style/Theme.LW_5"
        tools:targetApi="31">
        <activity
            android:name=".TracksActivity"
            android:exported="false" />
        <activity
            android:name=".MainActivity"
            android:exported="true">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
        </activity>
    </application>
</manifest>
```


Результаты работы:

6:06

Длительность

От До

2 90

8 5

Display compositions

This screenshot shows a mobile application interface with a dark background. At the top, the status bar displays the time 6:06 and various icons. The main heading is 'Длительность' (Duration). Below it, there are two columns of input fields for filtering tracks. The first column is labeled 'От' (From) and contains the numbers 2 and 8. The second column is labeled 'До' (To) and contains the numbers 90 and 5. At the bottom, there is a light blue button labeled 'Display compositions'.

6:06

Ольга Бузова - Танцуй под Бузову

Nature Sounds - The Sound Of Birds
Non-Stop Mix

Звуки природы - 06 Звуки тропического
леса после дождя

This screenshot shows the same mobile application interface after the filter is applied. The status bar still shows 6:06. The list of tracks is displayed, including 'Ольга Бузова - Танцуй под Бузову', 'Nature Sounds - The Sound Of Birds Non-Stop Mix', and 'Звуки природы - 06 Звуки тропического леса после дождя'.

Вывод: в ходе лабораторной работы было реализовано получение информации о треках (автор, название и длительность) с использованием встроенного content provider и выборка треков.