**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**

федеральное государственное автономное образовательное

учреждение высшего образования

**НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**ТОМСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ**

Инженерная школа информационных технологий и робототехники

Отделение информационных технологий

Отчет по лабораторной работе №6 по дисциплине

**«Разработка программного обеспечения для ОС Android»**

Фотография. Хранение данных

Выполнил:

Студент группы 8И7А \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А.В. Леонов

Проверил:

Ст. преп. ОИТ ИШИТР \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ В.А. Дорофеев

Томск 2020

# Задание

В предыдущей лабораторной работе внесенные пользователем данные пропадали при закрытии приложения, или даже при повороте устройства в альбомный режим. В реальном приложении это, конечно, недопустимо. Поэтому нужно разобраться как сохранять данные. Android предлагает несколько вариантов хранения данных: в виде файлов в памяти устройства, в базе данных SQLite, в хранилище настроек. Доработайте программу, добавив возможность удаления товара из списка.

# Текст программы

<https://github.com/8ait/android-course>

**MainActivity.kt**

package com.leonovalexandr.lab5

import android.app.Activity

import android.content.ContentValues

import android.content.Intent

import android.database.sqlite.SQLiteDatabase

import android.os.Build

import androidx.appcompat.app.AppCompatActivity

import android.os.Bundle

import android.view.View

import android.widget.AdapterView

import android.widget.ListView

import androidx.annotation.RequiresApi

class MainActivity : AppCompatActivity() {

private var items = ArrayList<Item>()

private lateinit var con: SQLiteDatabase;

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

// Настройка списка

val listView: ListView = findViewById(R.id.listItems)

listView.adapter = ItemAdapter(this, items)

listView.setOnItemClickListener { adapterView: AdapterView<\*>, view1: View, i: Int, l: Long ->

val intent = Intent(this, ItemActivity::class.java)

intent.putExtra("index", i)

intent.putExtra("item", items[i])

startActivityForResult(intent, 0)

}

val fab: View = findViewById(R.id.fab)

fab.setOnClickListener {

val intent = Intent(this, ItemActivity::class.java)

startActivityForResult(intent, 0)

}

val db = SQLiteHelper(this)

con = db.readableDatabase

getProducts()

}

@RequiresApi(Build.VERSION\_CODES.N)

override fun onActivityResult(requestCode: Int, resultCode: Int, data: Intent?) {

super.onActivityResult(requestCode, resultCode, data)

if (resultCode == Activity.RESULT\_OK)

{

val index: Int = data?.getIntExtra("index", -1) ?: -1

val item: Item = data?.getParcelableExtra("item") ?: Item()

val cv = ContentValues()

cv.put("title", item.title)

cv.put("kind", item.kind)

cv.put("price", item.price)

cv.put("weight", item.weight)

cv.put("photo", item.photo)

if (index == -1) {

items.add(item)

con.insert("products", null, cv)

} else if (index == -2) {

items.removeIf{i -> i.id == item.id}

con.delete("products", "id=?", arrayOf(item.id.toString()))

}

else {

items[index] = item

cv.put("id", item.id)

con.update("products", cv, "id=?", arrayOf(item.id.toString()))

}

val listView: ListView = findViewById(R.id.listItems)

(listView.adapter as ItemAdapter).notifyDataSetChanged()

}

}

private fun getProducts() {

val cursor = con.query("products",

arrayOf("id", "title", "kind", "price", "weight", "photo"),

null, null, null, null, null)

cursor.moveToFirst()

while (!cursor.isAfterLast) {

val s = Item()

s.id = cursor.getInt(0)

s.title = cursor.getString(1)

s.kind = cursor.getString(2)

s.price = cursor.getDouble(3)

s.weight = cursor.getDouble(4)

s.photo = cursor.getString(5)

items.add(s)

cursor.moveToNext()

}

cursor.close()

}

}

**ItemActivity.kt**

package com.leonovalexandr.lab5

import android.app.Activity

import android.content.Intent

import android.graphics.BitmapFactory

import androidx.appcompat.app.AppCompatActivity

import android.os.Bundle

import android.os.Environment

import android.provider.MediaStore

import android.view.Gravity

import android.view.Menu

import android.view.MenuItem

import android.widget.EditText

import android.widget.ImageView

import android.widget.Toast

import androidx.core.content.FileProvider

import java.io.File

class ItemActivity : AppCompatActivity() {

private var index = 0

private val INDEX\_TO\_DELETE = -2

private lateinit var item: Item

// Хранилище имени временного файла

private var currentPhotoPath: String = ""

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.item\_activity)

val intent = intent

index = intent?.getIntExtra("index", -1) ?: -1

item = intent?.getParcelableExtra("item") ?: Item()

val editTitle = findViewById<EditText>(R.id.title)

editTitle.setText(item.title)

val editKind = findViewById<EditText>(R.id.kind)

editKind.setText(item.kind)

val editPrice = findViewById<EditText>(R.id.price)

editPrice.setText(item.price.toString())

val editWeight = findViewById<EditText>(R.id.weight)

editWeight.setText(item.weight.toString())

if (item.photo != "") {

val bmp = BitmapFactory.decodeFile(item.photo)

findViewById<ImageView>(R.id.ivPhoto).setImageBitmap(bmp)

currentPhotoPath = item.photo

}

supportActionBar?.setDisplayHomeAsUpEnabled(true)

}

override fun onActivityResult(requestCode: Int, resultCode: Int, intent: Intent?) {

super.onActivityResult(requestCode, resultCode, intent)

if (resultCode == Activity.RESULT\_OK) {

val bmp = BitmapFactory.decodeFile(currentPhotoPath)

val ivPhoto = findViewById<ImageView>(R.id.ivPhoto)

ivPhoto.setImageBitmap(bmp)

}

else

currentPhotoPath = ""

}

override fun onCreateOptionsMenu(menu: Menu?): Boolean {

menuInflater.inflate(R.menu.add\_item, menu)

return super.onCreateOptionsMenu(menu)

}

override fun onOptionsItemSelected(item: MenuItem): Boolean {

if (item.itemId == android.R.id.home) {

finish()

return true

}

if (item.itemId == R.id.action\_save) {

this.item.title = findViewById<EditText>(R.id.title).

text.toString()

this.item.kind = findViewById<EditText>(R.id.kind).

text.toString()

this.item.price = findViewById<EditText>(R.id.price).

text.toString().toDouble()

this.item.weight = findViewById<EditText>(R.id.weight).text.toString().toDouble()

this.item.photo = currentPhotoPath

if (check(this.item)){

val intent = Intent()

intent.putExtra("index", index)

intent.putExtra("item", this.item)

setResult(Activity.RESULT\_OK, intent)

finish()

return true

} else {

var toast = Toast.makeText(applicationContext, "Неверный формат данных.", Toast.LENGTH\_LONG)

toast.setGravity(Gravity.TOP, 0,0)

toast.show()

return false

}

}

if (item.itemId == R.id.action\_photo) {

// Формирование имени временного файла

val photoFile = File.createTempFile(

"photo",

".jpg", getExternalFilesDir(Environment.DIRECTORY\_PICTURES))

// Сохранение имени для будущего использования

currentPhotoPath = photoFile.absolutePath

// Получение провайдера файла, чтобы камера могла // записать файл в нашу частную папку

val photoURI = FileProvider.getUriForFile(

this, BuildConfig.APPLICATION\_ID, photoFile)

// Формирование запроса на фото

val intent = Intent(MediaStore.ACTION\_IMAGE\_CAPTURE)

intent.putExtra(MediaStore.EXTRA\_OUTPUT, photoURI)

startActivityForResult(intent, 0)

return true

}

if (item.itemId == R.id.action\_delete) {

val intent = Intent()

intent.putExtra("index", INDEX\_TO\_DELETE)

intent.putExtra("item", this.item)

setResult(Activity.RESULT\_OK, intent)

finish()

return true

}

return super.onOptionsItemSelected(item)

}

// Поверка модели.

private fun check(item: Item): Boolean {

return item.title.isNotBlank() &&

item.kind.isNotBlank() &&

item.price >=0 &&

item.weight >= 0

}

}

**ItemAdapter.kt**

package com.leonovalexandr.lab5

import android.content.Context

import android.view.LayoutInflater

import android.view.View

import android.view.ViewGroup

import android.widget.BaseAdapter

import android.widget.TextView

class ItemAdapter(context: Context, items: ArrayList<Item>): BaseAdapter() {

var ctx: Context = context

var objects: ArrayList<Item> = items

var inflater: LayoutInflater = ctx. getSystemService(Context.LAYOUT\_INFLATER\_SERVICE) as LayoutInflater

// Формирование сетки содержащих строку данных.

override fun getView(position: Int, convertView: View?, parent: ViewGroup?): View {

// Если разметка ещё не существует, создаём её по шаблону

var view = convertView

if (view == null)

view = inflater.inflate(R.layout.listview\_layout\_item, parent, false)

// Получение объекта с информацией о студенте

val s = objects[position]

// Заполнение элементов данными из объекта

var v = view?.findViewById(R.id.item\_title) as TextView

v.text = s.title

v = view.findViewById(R.id.item\_kind) as TextView

v.text = s.kind

v = view.findViewById(R.id.item\_cost) as TextView

v.text = s.priceInfo

return view

}

// Получение элемента данных в указанной строке

override fun getItem(position: Int): Any {

return objects[position]

}

// Получение идентификатора элемента в указанной строке

// Часто вместо него возвращается позиция элемента

override fun getItemId(position: Int): Long {

return position.toLong()

}

// Получение количества элементов в списке

override fun getCount(): Int {

return objects.size

}

}

**Item.kt**

package com.leonovalexandr.lab5

import android.os.Parcel

import android.os.Parcelable

// Товар.

class Item() : Parcelable {

// Идентфиикатор.

var id: Int = 0

// Тип.

var kind: String = ""

// Название.

var title: String = ""

// Стоимость.

var price: Double = 0.0

// Вес.

var weight: Double = 0.0

// Путь для фото.

var photo: String = ""

val priceInfo: String

get() = "$price ₽"

constructor(parcel: Parcel) : this() {

id = parcel.readInt()

kind = parcel.readString() ?: ""

title = parcel.readString() ?: ""

photo = parcel.readString() ?: ""

price = parcel.readDouble()

weight = parcel.readDouble()

}

override fun writeToParcel(parcel: Parcel, flags: Int) {

parcel.writeInt(id)

parcel.writeString(kind)

parcel.writeString(title)

parcel.writeString(photo)

parcel.writeDouble(price)

parcel.writeDouble(weight)

}

override fun describeContents(): Int {

return 0

}

companion object CREATOR : Parcelable.Creator<Item> {

override fun createFromParcel(parcel: Parcel): Item {

return Item(parcel)

}

override fun newArray(size: Int): Array<Item?> {

return arrayOfNulls(size)

}

}

}

**SQLiteHelper.kt**

package com.leonovalexandr.lab5

import android.content.Context

import android.database.sqlite.SQLiteDatabase

import android.database.sqlite.SQLiteOpenHelper

class SQLiteHelper(context: Context?) :

SQLiteOpenHelper(context,"products.db", null, 1) {

override fun onCreate(db: SQLiteDatabase?) {

db?.execSQL("CREATE TABLE IF NOT EXISTS products (" +

"id INTEGER PRIMARY KEY AUTOINCREMENT, " +

"title TEXT, " +

"kind TEXT, " +

"price REAL, " +

"weight REAL, " +

"photo TEXT)");

}

override fun onUpgrade(p0: SQLiteDatabase?, p1: Int, p2: Int) {

TODO("Not yet implemented")

}

}

**activity\_main.xml**

<?xml version="1.0" encoding="utf-8"?>

<FrameLayout 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">

<ListView

android:id="@+id/listItems"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"/>

<com.google.android.material.floatingactionbutton.FloatingActionButton

android:id="@+id/fab"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:layout\_gravity="end|bottom"

android:src="@drawable/ic\_add"

android:tint="@color/design\_default\_color\_on\_primary"

android:layout\_margin="16dp"/>

</FrameLayout>

**item\_activity.xml**

<?xml version="1.0" encoding="utf-8"?>

<LinearLayout

xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

android:orientation="vertical"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:layout\_margin="20dp">

<androidx.cardview.widget.CardView

android:layout\_width="200dp"

android:layout\_height="200dp"

app:cardCornerRadius="100dp"

android:layout\_margin="10dp"

android:layout\_gravity="center">

<ImageView

android:id="@+id/ivPhoto"

android:layout\_height="200dp"

android:layout\_width="200dp"

android:layout\_gravity="center"

android:scaleType="center"

android:background="@drawable/ic\_empty"/>

</androidx.cardview.widget.CardView>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textSize="20dp"

android:text="Название:"/>

<EditText

android:id="@+id/title"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:inputType="textCapSentences"/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textSize="20dp"

android:text="Тип продукции:"/>

<EditText

android:id="@+id/kind"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:inputType="textCapSentences"/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textSize="20dp"

android:text="Стоимость:"/>

<EditText

android:id="@+id/price"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:inputType="numberDecimal"/>

<TextView

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textSize="20dp"

android:text="Вес:"/>

<EditText

android:id="@+id/weight"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:inputType="numberDecimal"/>

</LinearLayout>

**listview\_layout\_activity.xml**

<?xml version="1.0" encoding="utf-8"?>

<FrameLayout xmlns:android="http://schemas.android.com/apk/res/android"

android:orientation="vertical"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:padding="12dp">

<LinearLayout

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

android:orientation="vertical">

<TextView

android:id="@+id/item\_title"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"

android:textSize="16sp"

android:textColor="#000000"/>

<TextView

android:id="@+id/item\_kind"

android:layout\_width="match\_parent"

android:layout\_height="wrap\_content"/>

</LinearLayout>

<TextView

android:id="@+id/item\_cost"

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

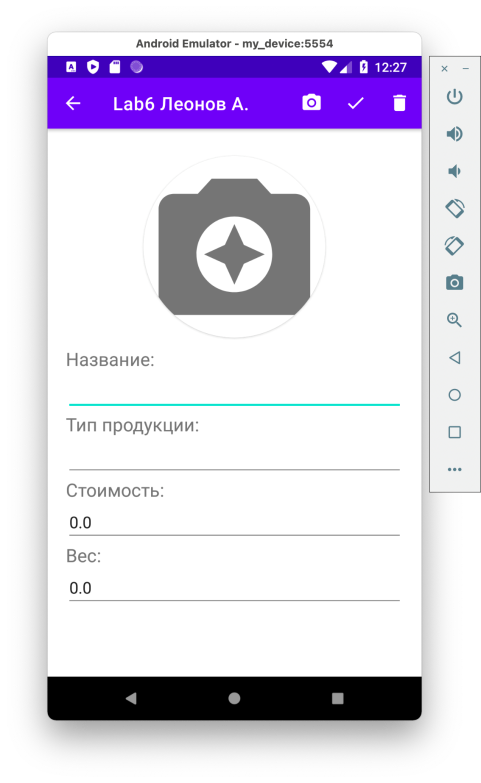
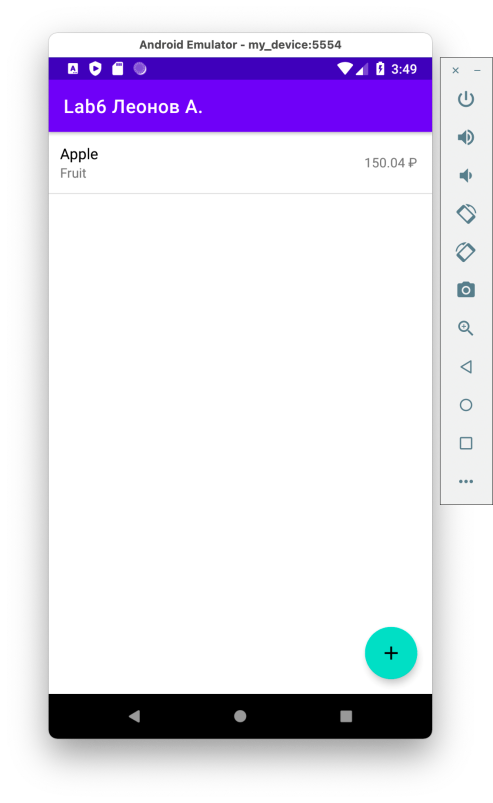
android:textSize="14sp"

android:layout\_marginRight="10px"

android:layout\_gravity="right|center"/>

</FrameLayout>

# Результаты работы



# Выводы

В результате выполнения лабораторной работы я познакомился с возможностями хранения фотографий и данных пользователя в приложении Android.