Министерство образования и молодежной политики Свердловской области



ГАПОУ СО «Екатеринбургский колледж транспортного строительства»

Отчёт по программе «**Практическое занятие 19\_2**

**Работа с JSON-данными.**»

Выполнил: Молотова В.А.

Группа: ПР-21

Преподаватель: Мирошниченко Г. В.

2023

**Задание:**

Разработать мобильное приложение, информация в приложении должна сохраняться в SharedPreferences, в формате JSON. Приложение должно содержать:

- приветственный экран с авторизацией пользователя, проверкой ввода на пустоту и сохранением информации о пользователе через механизм SharedPreferences;

- экран для ввода и просмотра сохраненной информации.

**Входные данные:**

* loginEd (String) – введенный пользователем логин;
* passwordEd (String) – введенный пользователем пароль;
* number (String) – введенный пользователем новый номер билета;
* type (String) – введенный пользователем новый тип билета.

**Выходные данные:**

* Список ListView, в котором отображаются все загруженные билеты;
* Сообщения пользователю об ошибках в Toast.makeText.

**Листинг программы:**

MainActivity.kt

package com.example.listoftraffictickets  
  
import android.content.Context  
import android.content.Intent  
import android.content.SharedPreferences  
import android.support.v7.app.AppCompatActivity  
import android.os.Bundle  
import android.view.Gravity  
import android.widget.Button  
import android.widget.EditText  
import android.widget.Toast  
  
class MainActivity : AppCompatActivity() {  
  
 private lateinit var registerButton: Button  
 private lateinit var enterButton: Button  
 private lateinit var sharedPreferences: SharedPreferences  
 private lateinit var loginEditText: EditText  
 private lateinit var passwordEditText: EditText  
  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_main*)  
  
 loginEditText = findViewById(R.id.*login*)  
 passwordEditText = findViewById(R.id.*pass*)  
 registerButton = findViewById(R.id.*save*)  
 enterButton = findViewById(R.id.*enter*)  
 sharedPreferences = getSharedPreferences("login", Context.*MODE\_PRIVATE*)  
  
 var login = sharedPreferences.getString("login", null)  
 var password = sharedPreferences.getString("password", null)  
  
 loginEditText.setText(login)  
 passwordEditText.setText(password)  
  
 registerButton.setOnClickListener **{** if (loginEditText.*text*.toString().*isEmpty*() ||  
 passwordEditText.*text*.toString().*isEmpty*()) {  
  
 val i = Toast.makeText(this, "Введите логин и пароль.",  
 Toast.*LENGTH\_SHORT*)  
 i.setGravity(Gravity.*TOP*, 0, 160)  
 i.show()  
 return@setOnClickListener  
 }  
  
 val loginEd = loginEditText.*text*.toString()  
 val passwordEd = passwordEditText.*text*.toString()  
  
 val ed = sharedPreferences.edit()  
 ed.putString("login", loginEd)  
 ed.putString("password", passwordEd)  
 ed.apply()  
  
 login = loginEd  
 password = passwordEd  
  
 val i = Toast.makeText(this, "Вы зарегестрировались.", Toast.*LENGTH\_SHORT*)  
 i.setGravity(Gravity.*TOP*, 0, 160)  
 i.show()  
 **}** enterButton.setOnClickListener **{** if (loginEditText.*text*.toString().*isEmpty*() ||  
 passwordEditText.*text*.toString().*isEmpty*()){  
  
 val i = Toast.makeText(this, "Введите логин и пароль.",  
 Toast.*LENGTH\_SHORT*)  
 i.setGravity(Gravity.*TOP*, 0, 160)  
 i.show()  
 return@setOnClickListener  
 } else if (loginEditText.*text*.toString() != login  
 || passwordEditText.*text*.toString() != password){  
  
 val i = Toast.makeText(this, "Пароль или логин неверны.",  
 Toast.*LENGTH\_SHORT*)  
 i.setGravity(Gravity.*TOP*, 0, 160)  
 i.show()  
 return@setOnClickListener  
 }  
 val intent = Intent(this, DisplayActivity::class.*java*)  
 startActivity(intent)  
 **}** }  
}

DisplayActivity.kt

package com.example.listoftraffictickets  
  
import android.annotation.SuppressLint  
import android.content.Context  
import android.content.Intent  
import android.support.v7.app.AppCompatActivity  
import android.os.Bundle  
import android.view.Gravity  
import android.widget.\*  
import android.view.inputmethod.InputMethodManager  
  
class DisplayActivity : AppCompatActivity() {  
  
 private lateinit var listView: ListView  
 private lateinit var deleteButton: Button  
 private lateinit var numberEditText: EditText  
 private lateinit var typeEditText: EditText  
 private lateinit var adapter: ArrayAdapter<String>  
  
 @SuppressLint("MissingInflatedId")  
 override fun onCreate(savedInstanceState: Bundle?) {  
 super.onCreate(savedInstanceState)  
 setContentView(R.layout.*activity\_display*)  
  
 listView = findViewById(R.id.*list\_view*)  
 deleteButton = findViewById(R.id.*delete*)  
 numberEditText = findViewById(R.id.*number\_edit\_text*)  
 typeEditText = findViewById(R.id.*type\_edit\_text*)  
  
 val saveButton: Button = findViewById(R.id.*add*)  
 saveButton.setOnClickListener **{** if (numberEditText.*text*.toString().*isEmpty*() || typeEditText.*text*.toString()  
 .*isEmpty*()  
 ) {  
 val i = Toast.makeText(this, "Введите номер и тип билета.",  
 Toast.*LENGTH\_SHORT*)  
 i.setGravity(Gravity.*TOP*, 0, 160)  
 i.show()  
 return@setOnClickListener  
 }  
  
 val number = numberEditText.*text*.toString()  
 val type = typeEditText.*text*.toString()  
  
 val tickets = SharedPreferencesHelper.loadTickets(this)  
 if (tickets.*any* **{ it**.number == number **}**) {  
 val i = Toast.makeText(  
 this,  
 "Билет с номером $number уже существует.",  
 Toast.*LENGTH\_SHORT* )  
 i.setGravity(Gravity.*TOP*, 0, 160)  
 i.show()  
 } else {  
 val ticket = Ticket(number, type)  
 val updatedTickets = tickets.*toMutableList*()  
 updatedTickets.add(ticket)  
 SharedPreferencesHelper.saveTickets(this, updatedTickets)  
  
 val i = Toast.makeText(this, "Билет № $number добавлен.",  
 Toast.*LENGTH\_SHORT*)  
 i.setGravity(Gravity.*TOP*, 0, 160)  
 i.show()  
  
 numberEditText.*text* = null  
 typeEditText.*text* = null  
  
 updateList()  
  
 val inputMethodManager = getSystemService(Context.*INPUT\_METHOD\_SERVICE*)  
 as InputMethodManager  
 inputMethodManager.hideSoftInputFromWindow(saveButton.*windowToken*, 0)  
 }  
 **}** adapter = createTicketAdapter()  
 listView.*adapter* = adapter  
  
 deleteButton.setOnClickListener **{** SharedPreferencesHelper.clearTickets(this)  
 adapter.clear()  
 adapter.notifyDataSetChanged()  
 updateList()  
 **}** val imageViewBackToMain: ImageView = findViewById(R.id.*imageViewBackToMain*)  
 imageViewBackToMain.setOnClickListener **{** val intent = Intent(this, MainActivity::class.*java*)  
 startActivity(intent)  
 **}** }  
  
 private fun createTicketAdapter(): ArrayAdapter<String> {  
 val tickets = SharedPreferencesHelper.loadTickets(this)  
 val ticketData = tickets.*map* **{** "${**it**.number} - ${**it**.type}" **}** val adapter = ArrayAdapter(this, android.R.layout.*simple\_list\_item\_1*, ticketData)  
 adapter.setDropDownViewResource(android.R.layout.*simple\_spinner\_dropdown\_item*)  
 return adapter  
 }  
  
 private fun updateList() {  
 val tickets = SharedPreferencesHelper.loadTickets(this)  
 val ticketData = tickets.*map* **{** "${**it**.number} - ${**it**.type}" **}** val adapter = ArrayAdapter(this, android.R.layout.*simple\_list\_item\_1*, ticketData)  
 adapter.setDropDownViewResource(android.R.layout.*simple\_spinner\_dropdown\_item*)  
 listView.*adapter* = adapter  
 }  
}

activity\_main.xml

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.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"  
 android:background="@color/background"  
 android:orientation="vertical">  
  
 <TextView  
 android:id="@+id/textView1"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginTop="16dp"  
 android:text="@string/pdd"  
 android:textAlignment="center"  
 android:textSize="33sp"  
 app:layout\_constraintTop\_toTopOf="parent"  
 tools:ignore="MissingConstraints"  
 tools:layout\_editor\_absoluteX="0dp" />  
  
 <EditText  
 android:id="@+id/login"  
 android:layout\_width="351dp"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="92dp"  
 android:fontFamily="sans-serif-medium"  
 android:hint="@string/login"  
 android:inputType="text"  
 android:textSize="20sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView1" />  
  
 <EditText  
 android:id="@+id/pass"  
 android:layout\_width="351dp"  
 android:layout\_height="50dp"  
 android:layout\_margin="10dp"  
 android:layout\_marginTop="12dp"  
 android:fontFamily="sans-serif-medium"  
 android:hint="@string/password"  
 android:inputType="textPassword"  
 android:textSize="20sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/login"  
 app:layout\_constraintVertical\_bias="0.102"  
 tools:ignore="MissingConstraints" />  
  
 <LinearLayout  
 android:id="@+id/linearLayout"  
 android:layout\_width="match\_parent"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginBottom="80dp"  
 android:orientation="vertical"  
 app:layout\_constraintBottom\_toBottomOf="parent"  
 tools:ignore="MissingConstraints"  
 tools:layout\_editor\_absoluteX="0dp">  
  
 <Button  
 android:id="@+id/enter"  
 android:layout\_width="242dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center\_horizontal"  
 android:layout\_margin="5dp"  
 android:text="@string/enter"  
 android:textSize="20sp"  
 android:textAllCaps="false"  
 android:background="@drawable/buttons\_1"/>  
  
 <Button  
 android:id="@+id/save"  
 android:layout\_width="242dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center\_horizontal"  
 android:text="@string/register"  
 android:textSize="20sp"  
 android:textAllCaps="false"  
 android:background="@drawable/buttons\_1"/>  
 </LinearLayout>  
  
</android.support.constraint.ConstraintLayou

activity\_display.xml

<?xml version="1.0" encoding="utf-8"?>  
<android.support.constraint.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/background"  
 tools:context=".DisplayActivity">  
  
 <TextView  
 android:id="@+id/textView"  
 android:layout\_width="265dp"  
 android:layout\_height="30dp"  
 android:layout\_marginTop="28dp"  
 android:layout\_marginEnd="75dp"  
 android:text="@string/listBilets"  
 android:textAlignment="center"  
 android:textSize="23sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="1.0"  
 app:layout\_constraintStart\_toEndOf="@+id/imageViewBackToMain"  
 app:layout\_constraintTop\_toTopOf="parent" />  
  
 <ImageView  
 android:id="@+id/imageViewBackToMain"  
 android:layout\_width="wrap\_content"  
 android:layout\_height="wrap\_content"  
 android:layout\_marginStart="24dp"  
 android:layout\_marginTop="28dp"  
 android:src="@drawable/buttunreturn"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toTopOf="parent"  
 tools:ignore="MissingConstraints" />  
  
 <EditText  
 android:id="@+id/number\_edit\_text"  
 android:layout\_width="320dp"  
 android:layout\_height="50dp"  
 android:layout\_marginTop="26dp"  
 android:fontFamily="sans-serif-medium"  
 android:hint="@string/numbertickets"  
 android:inputType="numberDecimal"  
 android:textSize="18sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.494"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/textView" />  
  
 <EditText  
 android:id="@+id/type\_edit\_text"  
 android:layout\_width="320dp"  
 android:layout\_height="50dp"  
 android:fontFamily="sans-serif-medium"  
 android:hint="@string/typetickets"  
 android:inputType="text"  
 android:textSize="18sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.494"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/number\_edit\_text" />  
  
 <ListView  
 android:id="@+id/list\_view"  
 android:layout\_width="391dp"  
 android:layout\_height="343dp"  
 android:layout\_marginTop="30dp"  
 android:padding="16dp"  
 android:textSize="24sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/delete"  
 tools:ignore="MissingConstraints" />  
  
 <Button  
 android:id="@+id/add"  
 android:layout\_width="242dp"  
 android:layout\_height="45dp"  
 android:layout\_gravity="center\_horizontal"  
 android:layout\_marginTop="20dp"  
 android:text="@string/addbilets"  
 android:textAllCaps="false"  
 android:background="@drawable/buttons\_1"  
 android:textSize="20sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/type\_edit\_text"  
 tools:ignore="MissingConstraints" />  
  
 <Button  
 android:id="@+id/delete"  
 android:layout\_width="242dp"  
 android:layout\_height="wrap\_content"  
 android:layout\_gravity="center\_horizontal"  
 android:layout\_marginTop="8dp"  
 android:background="@drawable/buttons\_1"  
 android:text="@string/delete"  
 android:textAllCaps="false"  
 android:textSize="20sp"  
 app:layout\_constraintEnd\_toEndOf="parent"  
 app:layout\_constraintHorizontal\_bias="0.497"  
 app:layout\_constraintStart\_toStartOf="parent"  
 app:layout\_constraintTop\_toBottomOf="@+id/add"  
 tools:ignore="MissingConstraints" />  
  
</android.support.constraint.ConstraintLayout>

SharedPreferencesHelper.kt

package com.example.listoftraffictickets  
  
import android.content.Context  
import com.google.gson.Gson  
  
object SharedPreferencesHelper {  
 private const val PREFS\_NAME = "MyPrefsFile"  
 private const val TICKETS\_KEY = "Tickets"  
  
 fun saveTickets(context: Context, tickets: List<Ticket>) {  
 val prefs = context.getSharedPreferences(PREFS\_NAME, Context.*MODE\_PRIVATE*)  
 val editor = prefs.edit()  
 val gson = Gson()  
 val json = gson.toJson(tickets)  
 editor.putString(TICKETS\_KEY, json)  
 editor.apply()  
 }  
  
 fun loadTickets(context: Context): List<Ticket> {  
 val prefs = context.getSharedPreferences(PREFS\_NAME, Context.*MODE\_PRIVATE*)  
 val gson = Gson()  
 val json = prefs.getString(TICKETS\_KEY, null)  
 return if (json != null) {  
 gson.fromJson(json, Array<Ticket>::class.*java*).*toList*()  
 } else {  
 *emptyList*()  
 }  
 }  
  
 fun clearTickets(context: Context) {  
 val preferences = context.getSharedPreferences(PREFS\_NAME, Context.*MODE\_PRIVATE*)  
 preferences.edit().remove(TICKETS\_KEY).apply()  
 }  
}

Ticket.kt

package com.example.listoftraffictickets  
  
data class Ticket(val number: String, val type: String)

strings.xml

<resources>  
 <string name="app\_name">List of traffic tickets</string>  
 <string name="login">Введите логин</string>  
 <string name="password">Введите пароль</string>  
 <string name="register">Зарегистрироваться</string>  
 <string name="enter">Войти</string>  
 <string name="pdd">Правила дорожного движения</string>  
 <string name="addbilets">Добавить</string>  
 <string name="numbertickets">Введите номер билета</string>  
 <string name="typetickets">Введите тип билета</string>  
 <string name="add">Добавить</string>  
 <string name="delete">Отчистить</string>  
 <string name="aggbilets">Добавить новые билеты</string>  
 <string name="listBilets">Список билетов</string>  
 <string name="obnov">Обновить</string>  
  
</resources>

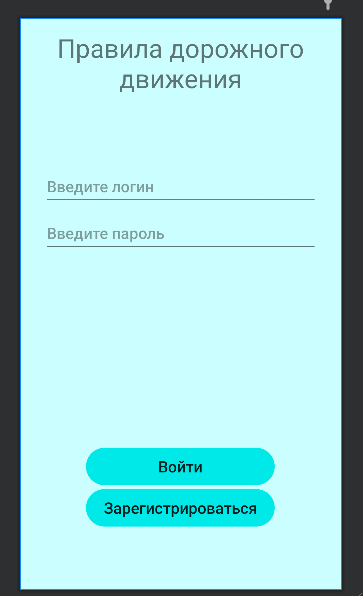
colors.xml

<?xml version="1.0" encoding="utf-8"?>  
<resources>  
 <color name="purple\_200">#FFBB86FC</color>  
 <color name="purple\_500">#FF6200EE</color>  
 <color name="purple\_700">#FF3700B3</color>  
 <color name="teal\_200">#FF03DAC5</color>  
 <color name="teal\_700">#FF018786</color>  
 <color name="black">#FF000000</color>  
 <color name="white">#FFFFFFFF</color>  
 <color name="LightCyan">#E0FFFF</color>  
 <color name="bluebut">#00E9E9</color>  
 <color name="background">#CBFFFF</color>  
  
</resources>

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">  
  
 <application  
 android:allowBackup="true"  
 android:dataExtractionRules="@xml/data\_extraction\_rules"  
 android:fullBackupContent="@xml/backup\_rules"  
 android:icon="@drawable/myicon"  
 android:label="@string/app\_name"  
 android:roundIcon="@drawable/myicon"  
 android:supportsRtl="true"  
 android:theme="@style/Theme.ListOfTrafficTickets"  
 tools:targetApi="31">  
 <activity  
 android:name=".DisplayActivity"  
 android:exported="false">  
 <meta-data  
 android:name="android.app.lib\_name"  
 android:value="" />  
 </activity>  
 <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>  
 <meta-data  
 android:name="android.app.lib\_name"  
 android:value="" />  
 </activity>  
 </application>  
</manifest>

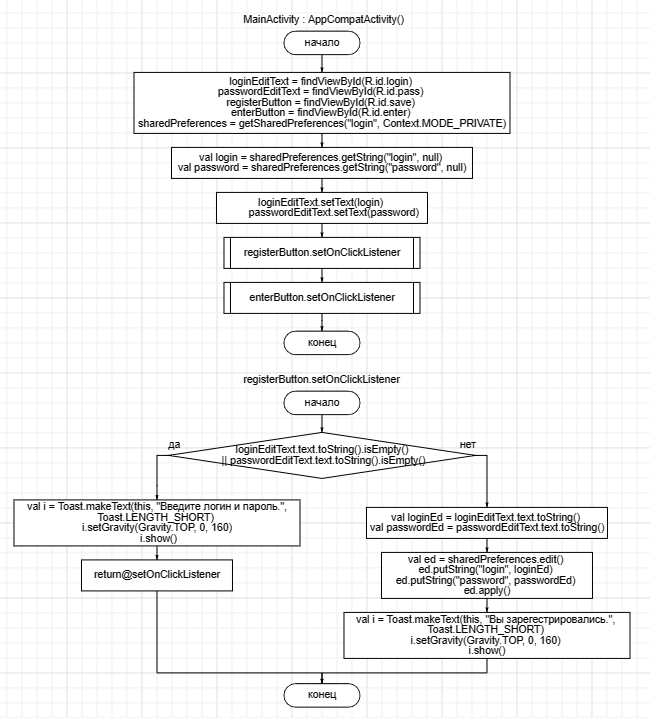
Экран 1.

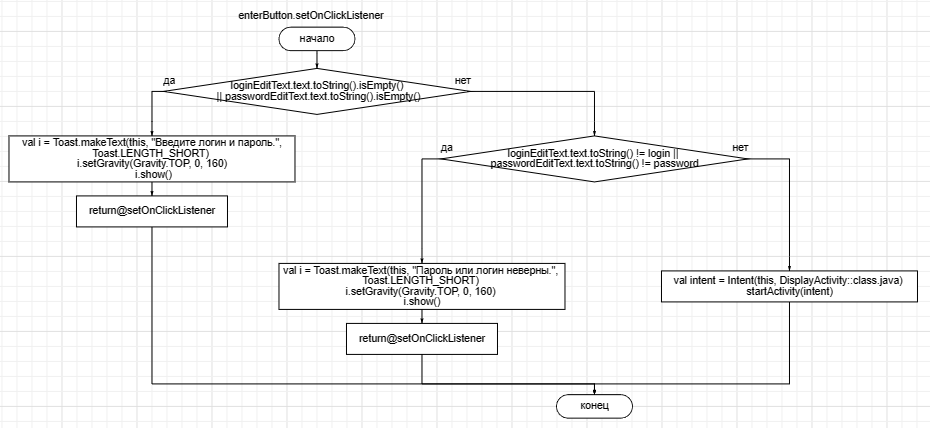


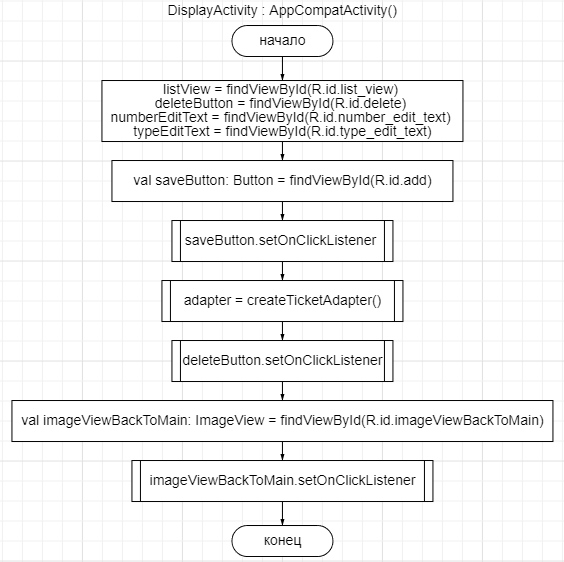
Экран 2.

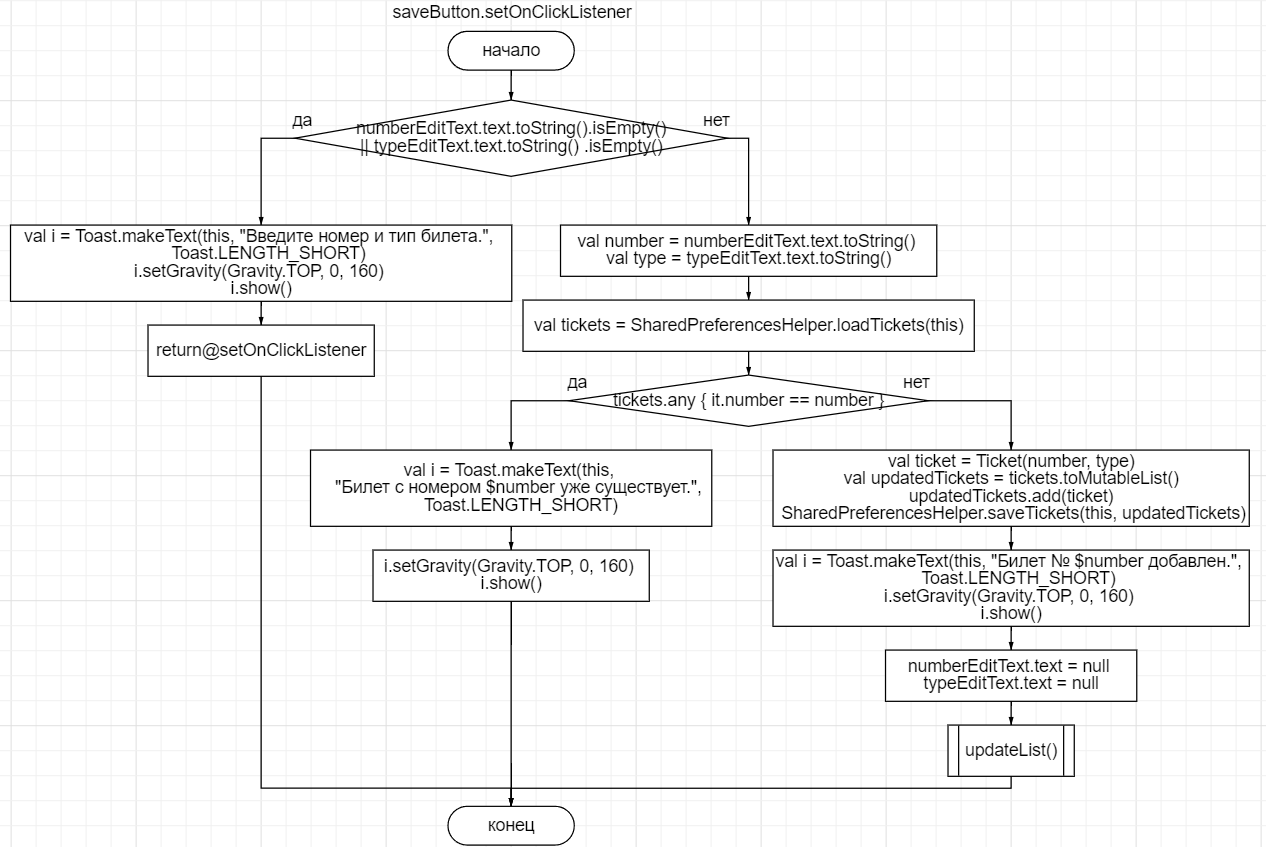


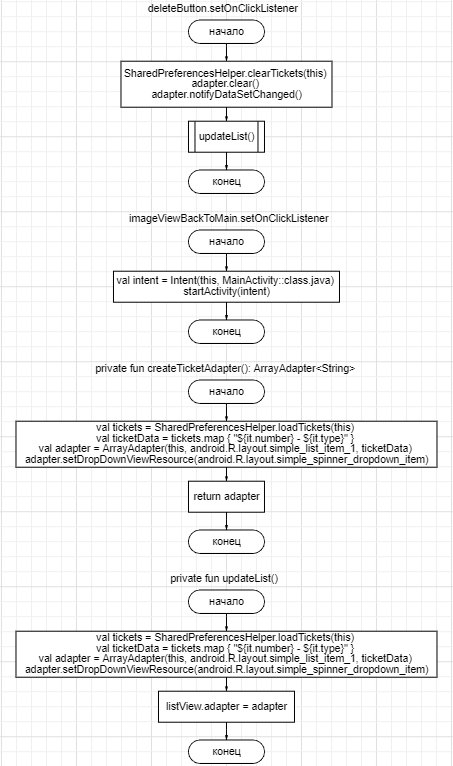
**Блок-схемы:**





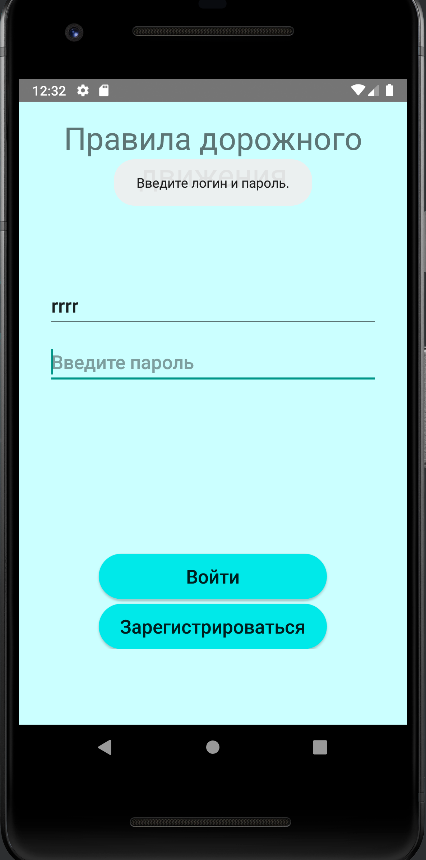






**Тестовые ситуации:**

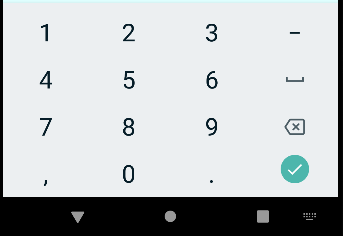
Имеется проверка на пустоту строк:



Имеется проверка на схожесть номеров билетов:

Нельзя вводить отрицательные числа (номер билета), благодаря элементу inputType

android:inputType="numberDecimal"



**Вывод:** благодаря практическому занятию 19\_2, я научилась работать с форматом данных JSON на языке Kotlin. С помощью механизма SharedPreference реализовал простые макеты экранов с сохранением информации в формате JSON.