

Сохранение данных в мобильном приложении с помощью SharedPreferences в Android Studio

Андрюенко Иван Сергеевич

*Приамурский государственный университет имени Шолом-Алейхема
Студент*

Аннотация

Целью данной статьи является создание приложения, использующего постоянное хранилище SharedPreferences. Для подключения SharedPreferences использовалась среда программирования Android Studio и языка программирования Java. В итоге было создано приложение, сохраняющее любой текст, набранный пользователем.

Ключевые слова: Java, SharedPreferences, база данных, Android studio

Saving data in a mobile app using SharedPreferences in Android Studio

Andrienko Ivan Sergeevich

*Sholom-Aleichem Priamursky State University
Student*

Abstract

The purpose of this article is to create an application using SharedPreferences persistent storage. To connect SharedPreferences, the Android Studio programming environment and the Java programming language were used. As a result, an application was created that saves any text typed by the user.

Keywords: Java, SharedPreferences, Database, Android studio

1 Введение

1.1 Актуальность

Часто при создании приложения разработчик сталкивается с проблемой, необходимо сохранить данные в виде пары: имя и значения. Именно в таких случаях и используется SharedPreference. С помощью данного хранилища можно легко хранить простые параметры конфигурации в виде карты. Преимущество данного класса в том, что с помощью него можно иметь, например таблицу с настройками приложения, при этом не подключая базу данных. Это позволяет оптимизировать приложение, ведь при задействовании обычной базы данных производительность вашего приложения может быть снижена.

1.2 Обзор исследований

В своей работе М.Д. Иванов предложил методы оптимизации архитектур мобильных приложений на базе операционной системы android.

[1]. А.Н. Казакова рассмотрела множество способов хранения данных, каждый из которых имеет свои преимущества и недостатки. [2]. В своей работе В.В. Осокин, В.А. Бендик провели реализацию CRM-системы для платформы Android. Система позволяет управлять клиентами, контактными лицами и контактами. В статье описан интерфейс данной системы и его реализация, а также реализация синхронизации данных с серверной частью. [3]. В своей работе Ю.В. Мурашко рассмотрел методы хранения данных в операционной системе Android. Также рассматриваются особенности хранения данных во внутреннем и внешнем хранилищах. Выделяются достоинства и недостатки каждого из хранилищ. [4]. В.С Дейч в своей работе описал библиотеки, которые облегчат разработку мобильного приложения под ОС Android, предоставляя набор общих функциональных возможностей. Использование библиотек позволяет сэкономить время и силы при создании приложений. [5].

1.3 Цель исследования

Цель исследования – создать приложение, сохраняющую и загружающую любую информацию, введённую пользователем, используя хранилище SharedPreferences.

2 Материалы и методы

Экземпляр класса SharedPreferences использовался для взаимодействия с данными хранилища, получаемый у произвольного объекта, который наследуется у класса android.content.Context.

3 Результаты и обсуждения

Существует множество разных способов хранения данных в приложении Android. У каждого есть свои преимущества и недостатки. SharedPreferences является самым простым в использовании, особенно если присутствует необходимость хранить дискретные примитивные типы данных. Во внутреннем и внешнем хранилище лучше всего хранить файлы, такие как музыка, видео и документы, в то время как SQLite выигрывает, если нужно выполнять быстрые поиски и запросы на свои данные. Выбранный метод хранения должен в конечном счете зависеть от типов данных, от времени, в течение которого нужны данные, и от того, насколько конфиденциальными они являются.

Для проверки хранилища, создадим обычное приложение с полем текста для ввода и двумя кнопками: сохранить и загрузить.

Для начала создадим макет нашего приложения. Пропишем код в файле с данными о дизайне, в файле «activity-main.xml» (рис.1).

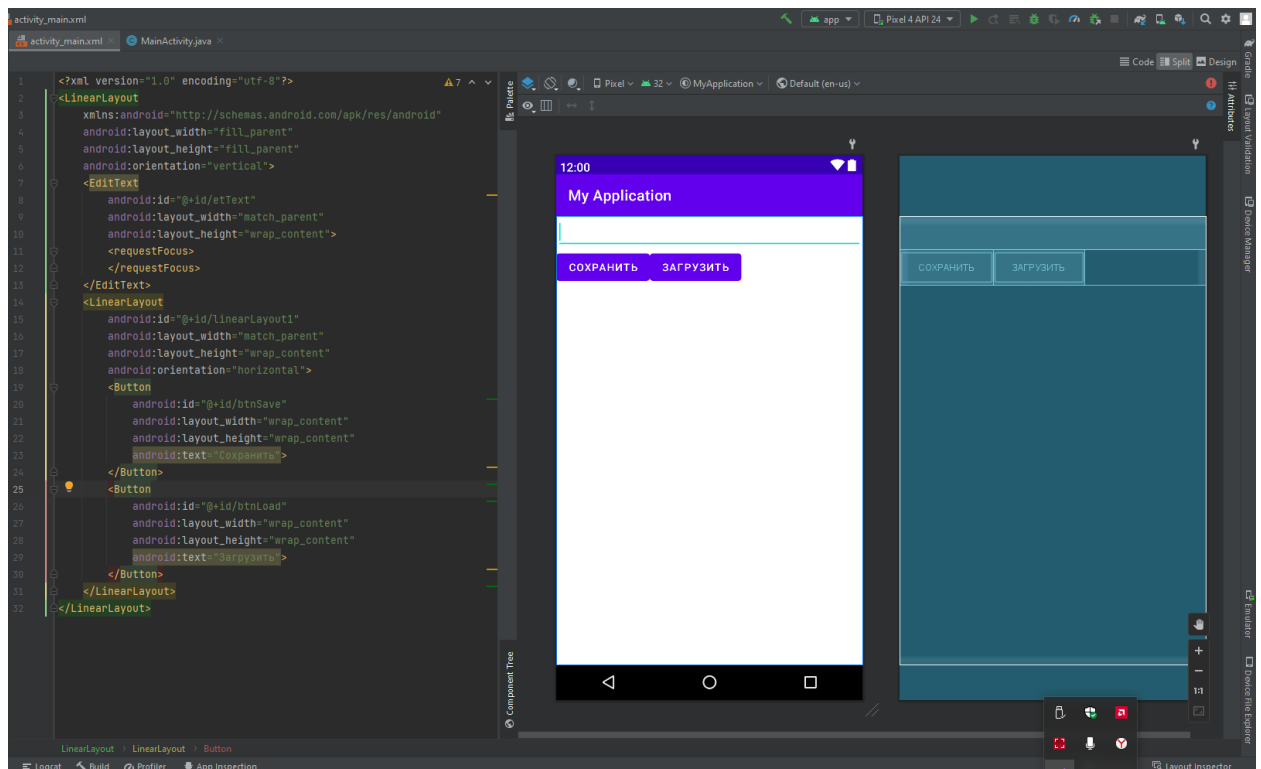


Рисунок 1 – Создание интерфейса

Теперь пишем код самой программы. Для начала импортируем необходимые библиотеки, присваиваем обработчики и реализуем метод `OnClickListener` (рис.2).

```
activity_main.xml x MainActivity.java x
1 package com.example.myapplication;
2
3 import android.content.SharedPreferences;
4 import android.os.Bundle;
5 import android.view.View;
6 import android.widget.Button;
7 import android.widget.EditText;
8 import android.widget.Toast;
9 import androidx.appcompat.app.AppCompatActivity;
10
11 public class MainActivity extends AppCompatActivity implements View.OnClickListener{
12
13     EditText etText;
14     Button btnSave, btnLoad;
15     SharedPreferences sPref;
16
17     final String SAVED_TEXT = "Текст загружен";
18
19
20     @Override
21     protected void onCreate(Bundle savedInstanceState) {
22         super.onCreate(savedInstanceState);
23         setContentView(R.layout.activity_main);
24
25         etText = (EditText)findViewById(R.id.etText);
26
27         btnSave = (Button)findViewById(R.id.btnSave);
28         btnSave.setOnClickListener(this);
29         btnLoad = (Button)findViewById(R.id.btnLoad);
30         btnLoad.setOnClickListener(this);
31
32         loadText();
33     }
```

Рисунок 2 – Импорт необходимых библиотек, присваивание обработчиков и реализация метода OnClickListener.

Теперь рассмотрим вызываемые методы OnClick. По нажатию на кнопку «Сохранить» будет вызван метод «saveText», а по нажатию на кнопку «Загрузить» будет вызван метод «loadText» (рис.3).

```
34
35     @Override
36     public void onClick(View v) {
37         switch (v.getId()){
38             case R.id.btnSave:
39                 saveText();
40                 break;
41             case R.id.btnLoad:
42                 loadText();
43                 break;
44             default:
45                 break;
46         }
47     }
48
49 }
50
```

Рисунок 3 – Методы OnClick

Следующим шагом необходимо реализовать эти методы. Благодаря методу `getPreferences` получаем объект `sPref` класса `sharedPreferences`, с помощью которого будет реализована работа с данными, считывание и запись их (рис.4).

```
53     private void saveText() {
54         sPref = getSharedPreferences( name: "MyDate", MODE_PRIVATE);
55         SharedPreferences.Editor ed = sPref.edit();
56         ed.putString(SAVED_TEXT, etText.getText().toString());
57         ed.commit();
58         Toast.makeText( context: MainActivity.this, text: "Текст сохранен", Toast.LENGTH_SHORT).show();
59     }
60
61 }
62
63
64     private void loadText() {
65         sPref = getSharedPreferences( name: "MyDate", MODE_PRIVATE);
66         String savedText = sPref.getString(SAVED_TEXT, "");
67         etText.setText(savedText);
68         Toast.makeText( context: MainActivity.this, text: "Текст загружен", Toast.LENGTH_SHORT).show();
69     }
70
71     @Override
72     protected void onDestroy() {
73         super.onDestroy();
74         saveText();
75     }
76 }
77
```

Рисунок 4 – Реализация методов

Теперь проверим созданное приложение. Введем текст в поле ввода и сохраним его (рис.5).

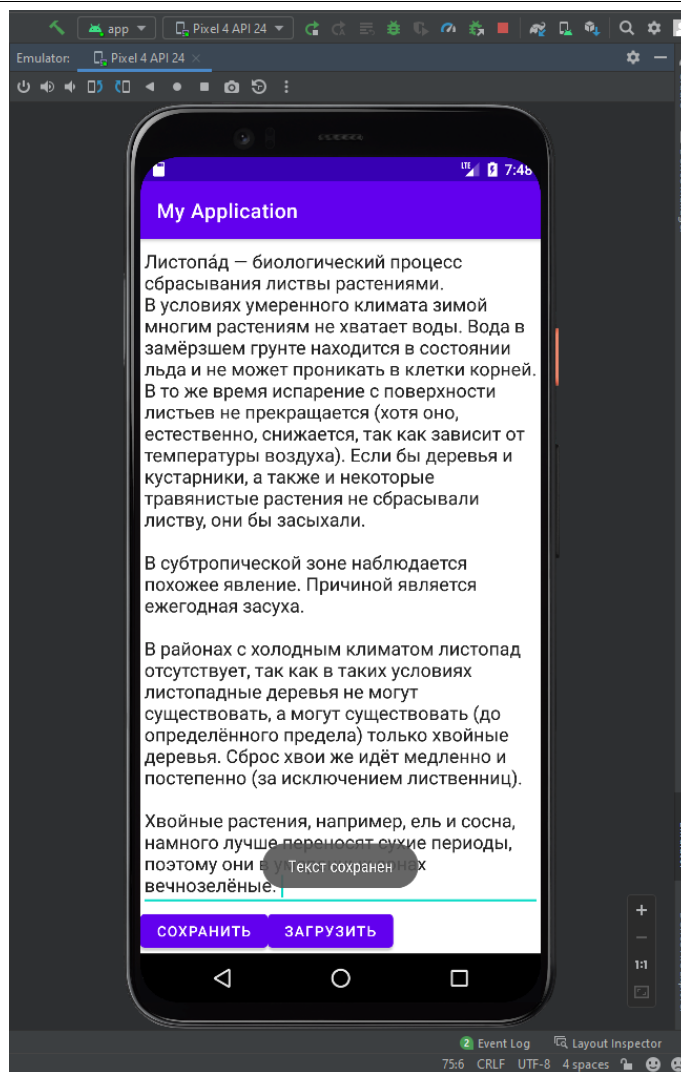


Рисунок 5 – Ввод и сохранение текста

Для того, чтобы проверить сохраняет ли приложение введенный текст, перезагрузим устройство и попробуем загрузить ранее сохраненный текст (рис.6).

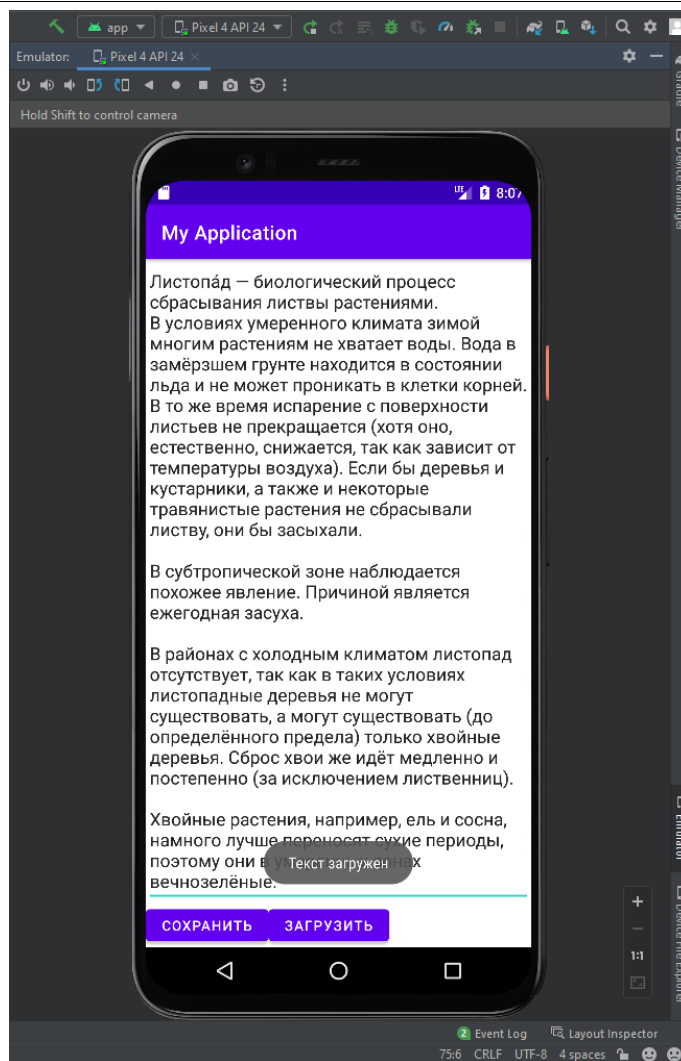


Рисунок 6 – Загрузка сохраненного текста

Созданное приложение успешно сохраняет и загружает введенный текст пользователем.

Выводы

В данной работе было создано приложение, сохраняющее и загружающее любую информацию, введенную пользователем, используя хранилище SharedPreferences.

Библиографический список

1. Иванов М.Д. Оптимизация архитектуры Android-приложений. // Образование России и актуальные вопросы современной науки. сборник статей Всероссийской научно-практической конференции. 2018. С. 121-124.
2. Казакова А.Н. Исследование способов хранения информации в android приложениях. // Материалы конференций ГНИИ "НАЦРАЗВИТИЕ". Декабрь 2017. 2018. С. 38-39.
3. Осокин В.В., Бендик В.А. Android как платформа для разработки

- корпоративных приложений на примере сrm. // Интеллектуальные системы. Теория и приложения. 2014. Т. 18. № 2. С. 179-188.
4. Мурашко Ю.В. Методы хранения данных в ос android. // Решение. 2019. Т. 1. С. 144-146.
 5. Дейч В.С. 10 библиотек, которые облегчат android-разработку в 2021 году. // Цифровизация и кибербезопасность: современная теория и практика. сборник научных трудов по материалам Международной научно-практической конференции. Омск, 2021. С. 16-20.