

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение высшего образования

**«МИРЭА – Российский технологический университет»**

РТУ МИРЭА

Институт Информационных технологий

Кафедра Математического обеспечения и стандартизации информационных технологий

Отчет по практической работе № 9

по дисциплине «Проектирование и разработка мобильных приложений»

# Выполнил:

Студент группы ИКБО-68-23 Клейменов М.Д.

Москва, 2025 г.

**Оглавление**

[Задание 3](#__RefHeading___Toc1509_1805702352)

[Ход работы 4](#__RefHeading___Toc1511_1805702352)

[1. Задание 1 4](#__RefHeading___Toc632_4235534367)

[2. Реализовать сохранение состояния приложения при повороте экрана. 10](#__RefHeading___Toc396_4056371770)

[3. Задание 3 12](#__RefHeading___Toc398_4056371770)

[Вывод 19](#__RefHeading___Toc496_390074292)

[Ссылка на репозиторий 1-2 задачи: репозиторий 19](#__RefHeading___Toc400_4056371770)

[Ссылка на репозиторий 3 задачи: репозиторий 19](#__RefHeading___Toc402_4056371770)

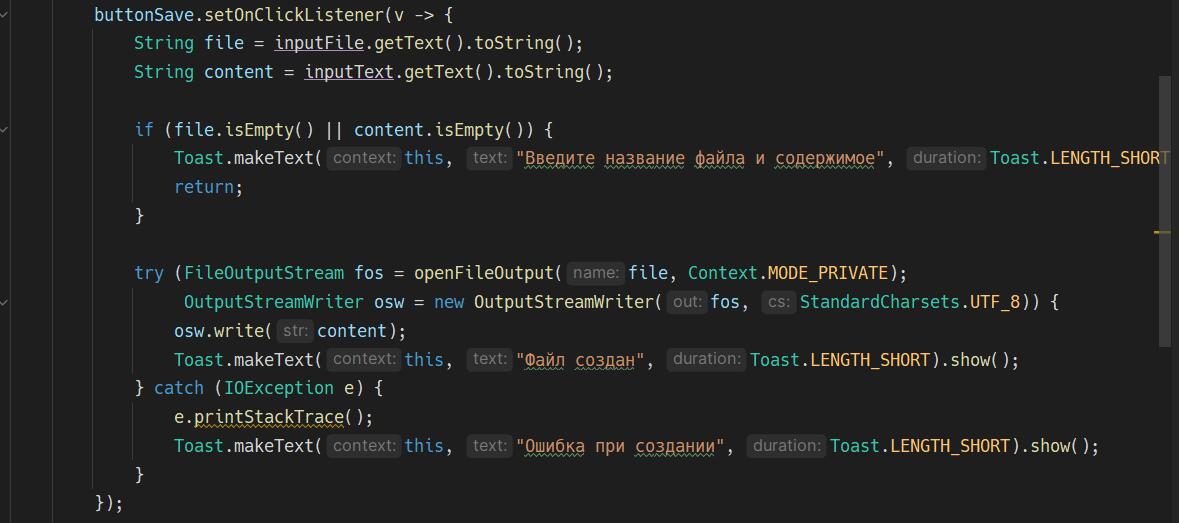
# Задание

1. На экране сделать несколько полей: поле, в котором будет прописано название создаваемого файла (EditText), поле для ввода содержимого файла (EditText). Также должны быть реализованы кнопки для создания файла, записи дополнительной информации в конец существующего файла, кнопка для чтения файла, а также для удаления файла.
2. Реализовать сохранение состояния приложения при повороте экрана.
3. Реализовать создание, запись и удаление во внешней памяти на экране одного приложения и чтение выбранного файла в другом приложении

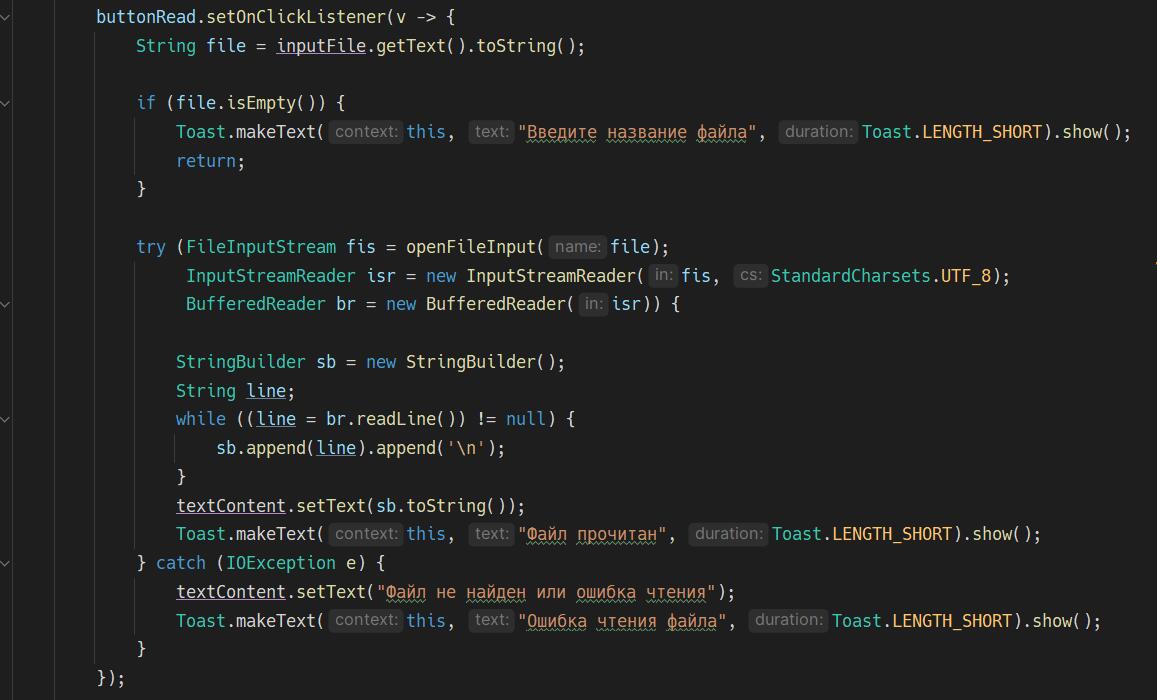
# Ход работы

# 1. На экране сделать несколько полей: поле, в котором будет прописано название создаваемого файла (EditText), поле для ввода содержимого файла (EditText). Также должны быть реализованы кнопки для создания файла, записи дополнительной информации в конец существующего файла, кнопка для чтения файла, а также для удаления файла.

Для того, чтобы создать и сохранить файл обрабатываем кнопку сохранения.

Рисунок 1 — открываем файл через FileOutputStream и OutputStreamWriter.

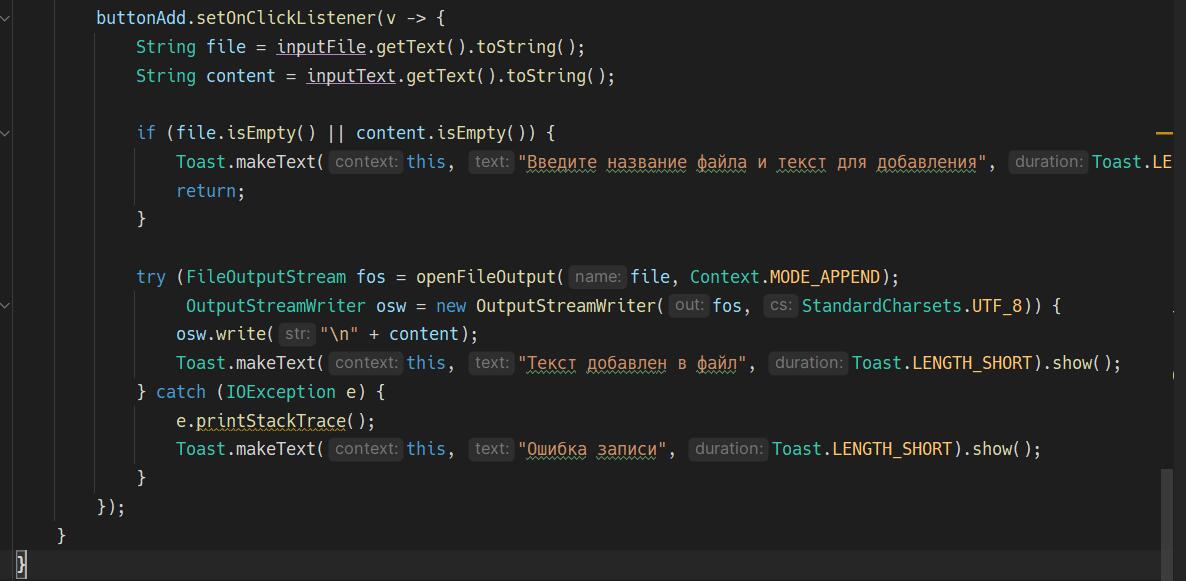
Чтобы прочитать файл обрабатываем кнопку buttonRead.

Рисунок 2 — открываем файл через FileInputStream и BufferReader.

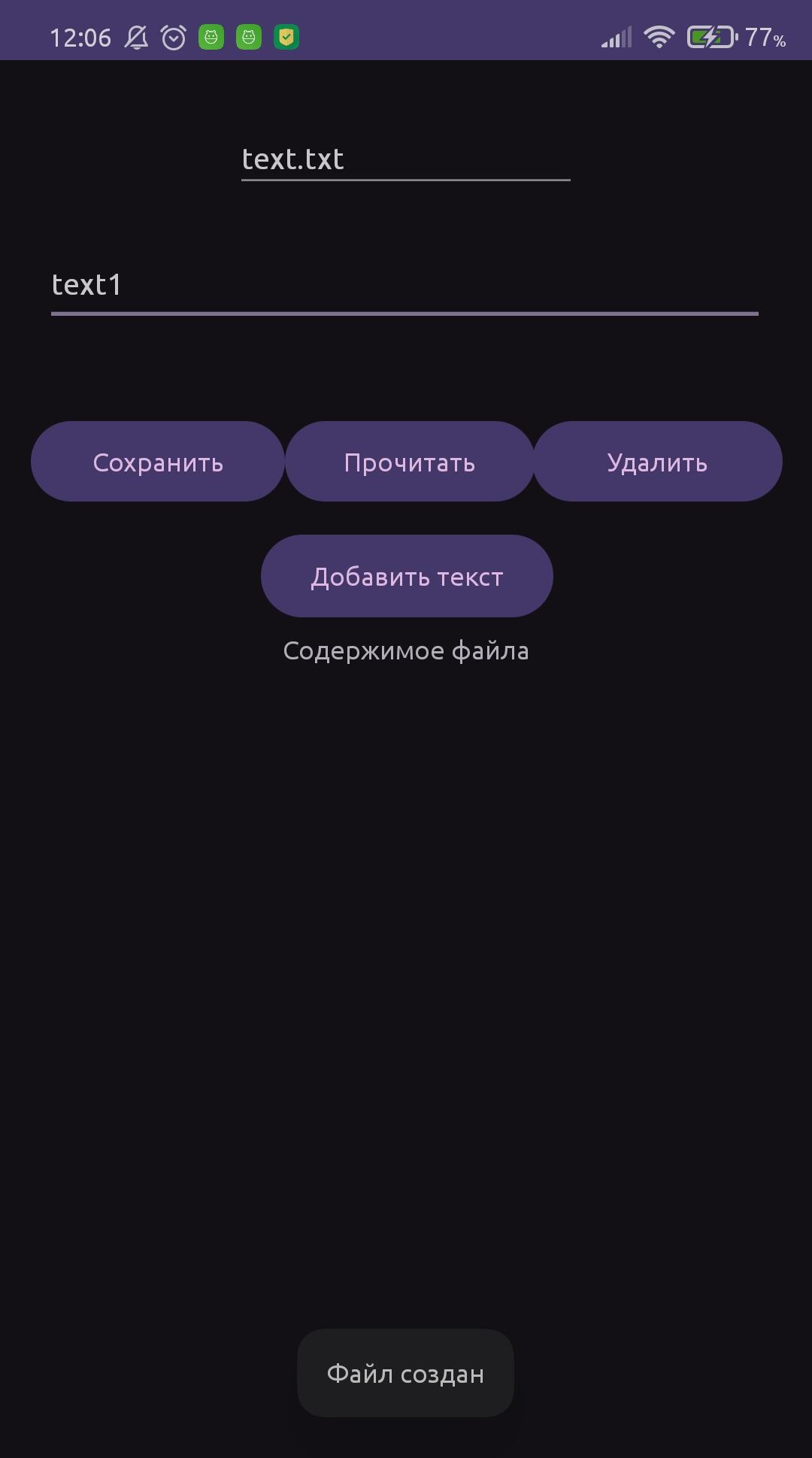
Чтобы удалить файл обрабатываем кнопку buttonDelete.

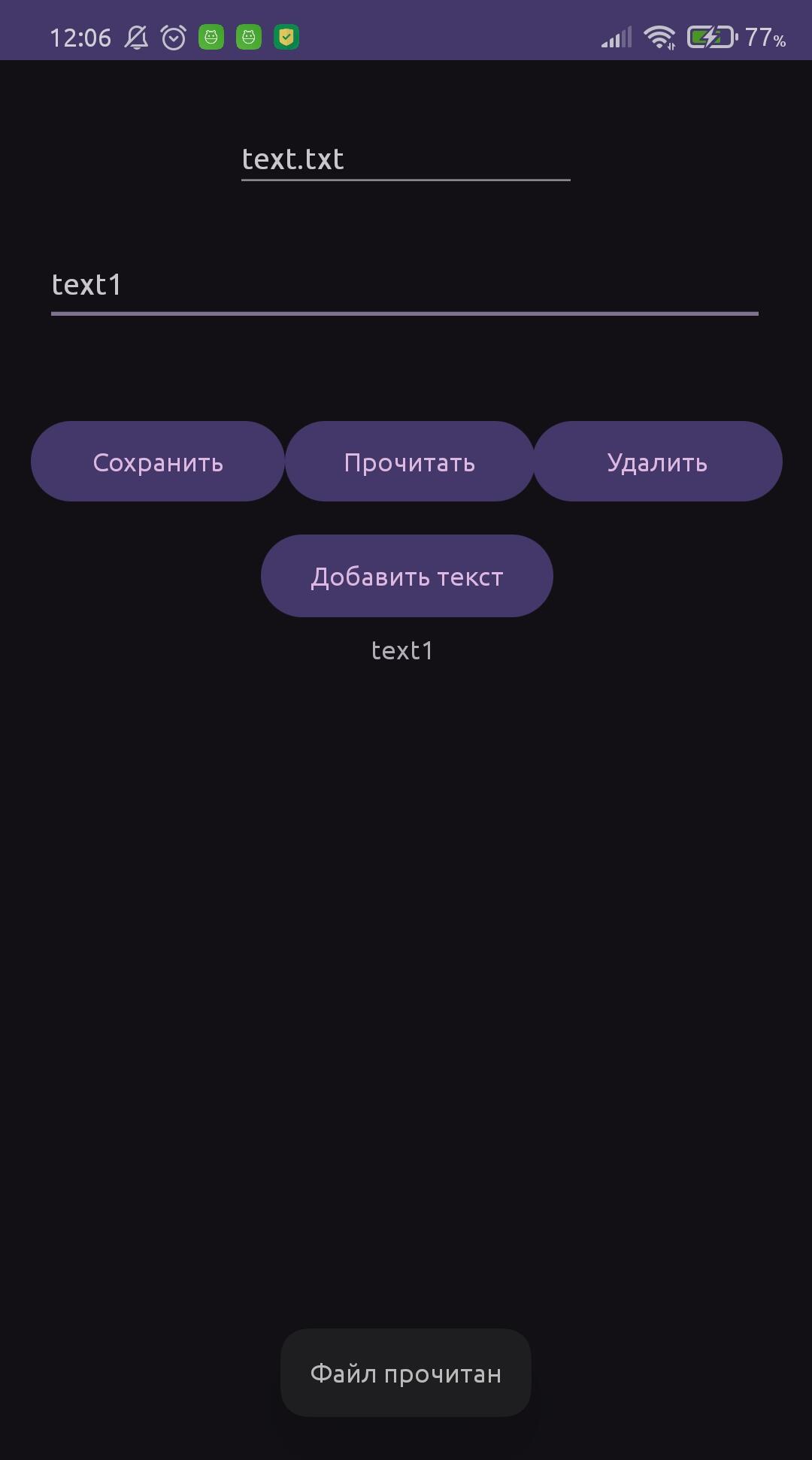
Рисунок 3 — для потверждения удаления создаем AlertDialog.

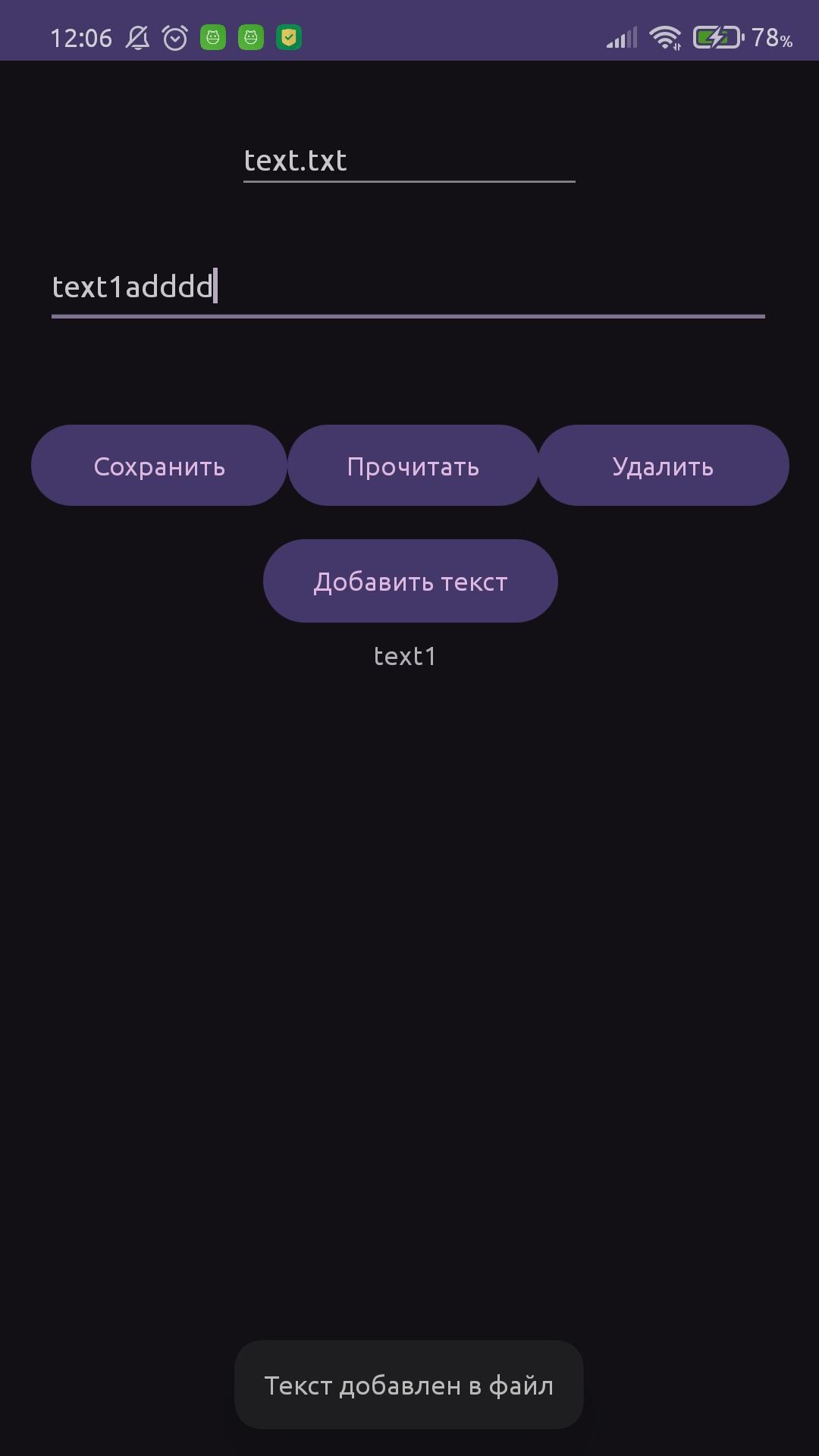
Чтобы добавить текст в уже созданный файл обрабатываем соотвественную кнопку.

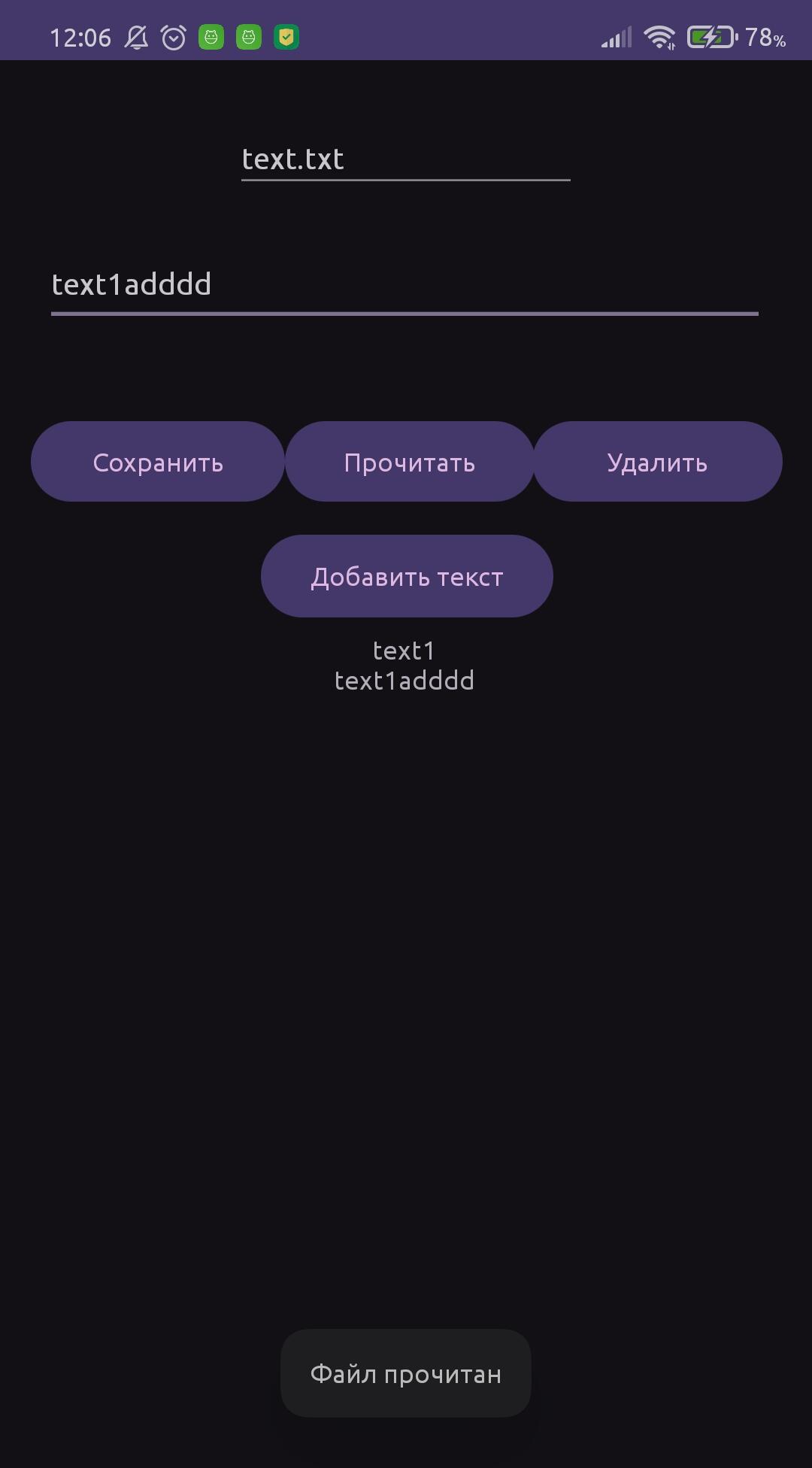
Рисунок 4 — открываем файл с MODE\_APPEND.

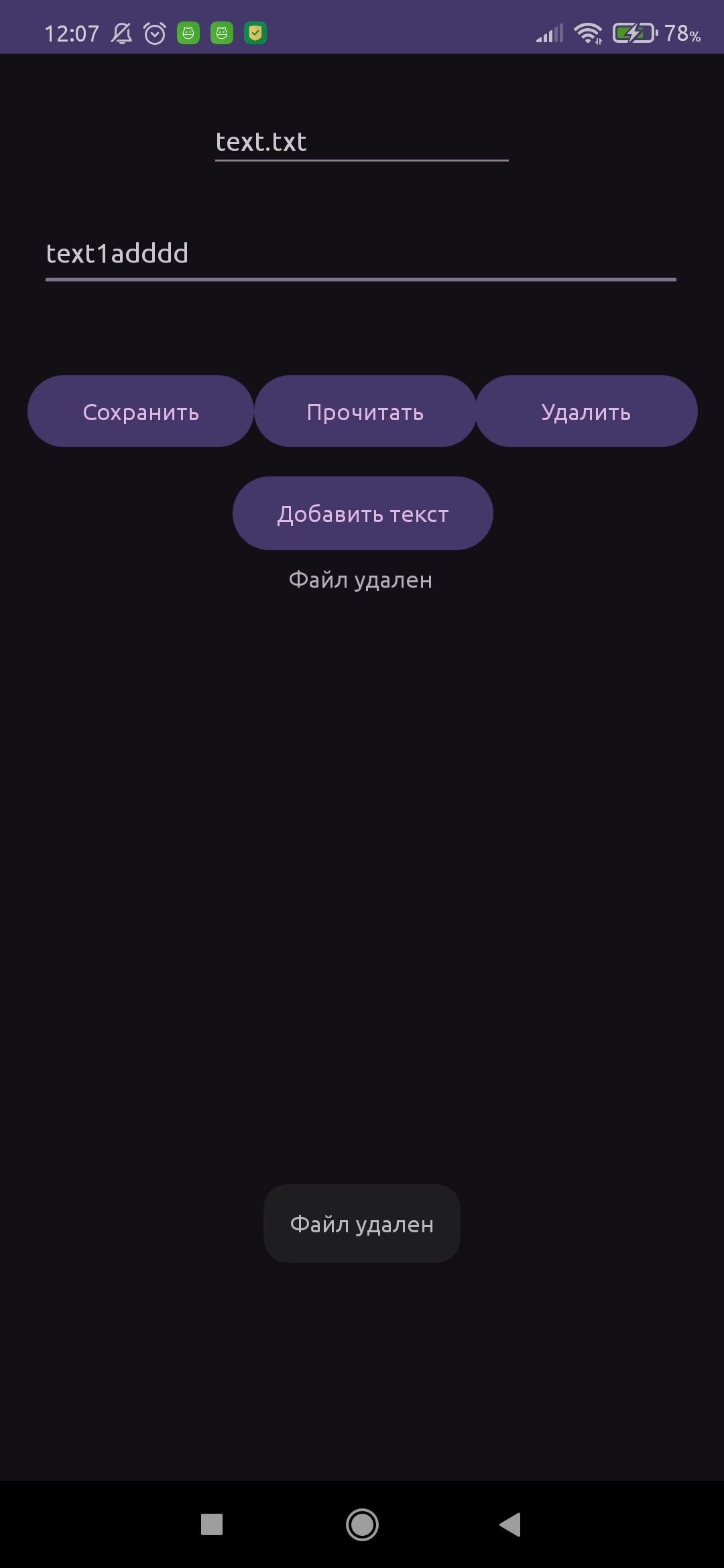
Проверяем работу.

Рисунок 5 — создание файла с текстом.

Рисунок 6 — чтение файла и вывод текста.

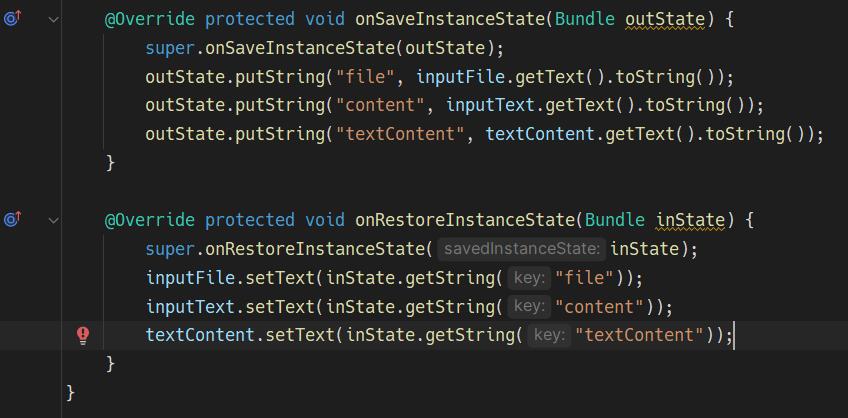
Рисунок 7 — добавление текста в файл.

Рисунок 8 — читаем добавленный текст.

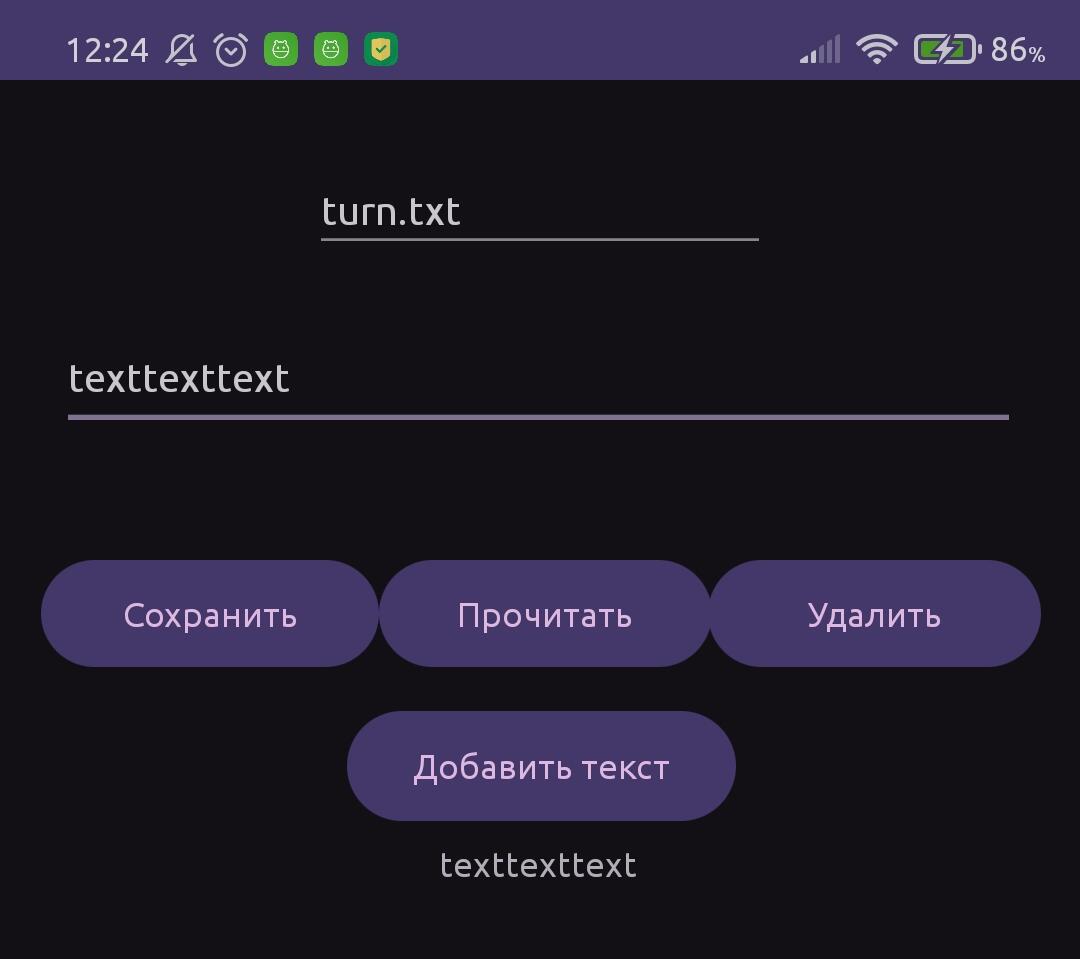
Рисунок 9 — удаление файла.

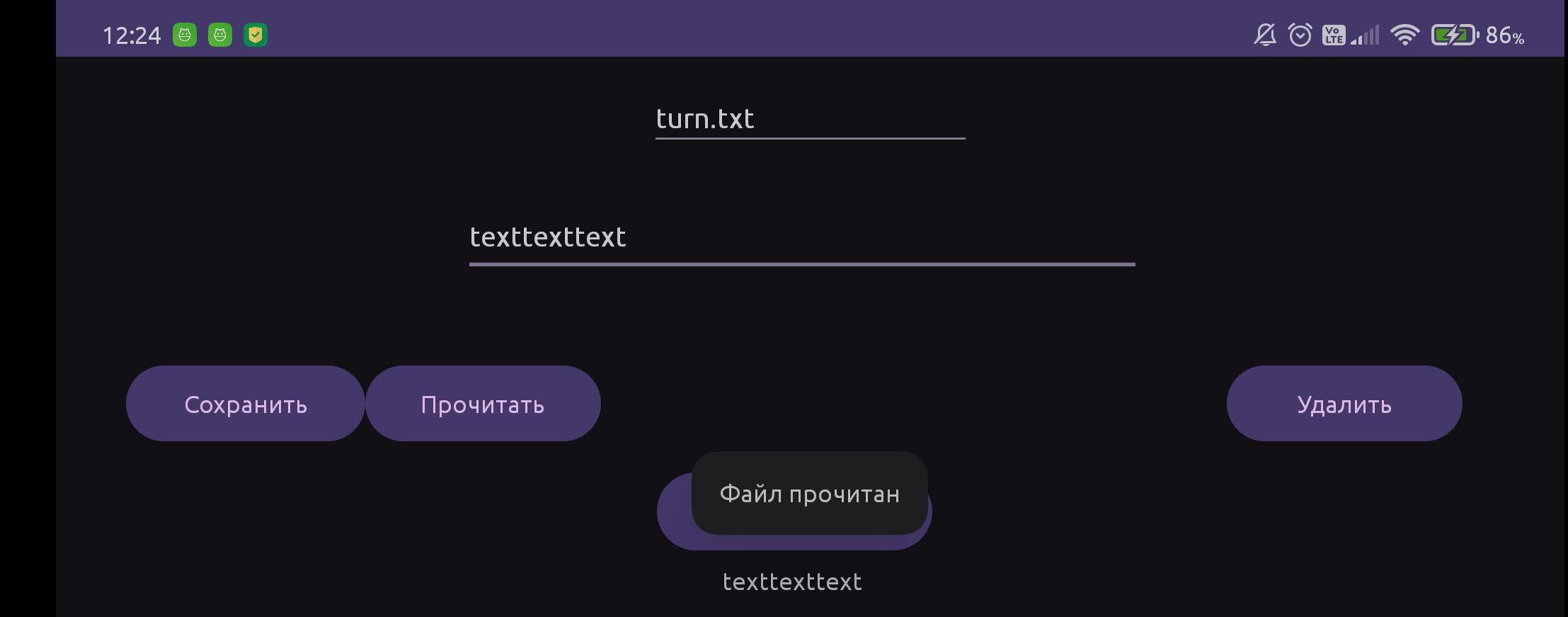
# 2. Реализовать сохранение состояния приложения при повороте экрана.

Для того, чтобы при повороте экрана данные не сбивались, создаем два метода: onSaveInstanceState и onRestoreInstanceState

Рисунок 10 — в метода save сохраняем в Bundle данные, в методе restore восстанавливаем их.

Проверяем работу.

Рисунок 11 — создаем новый файл.

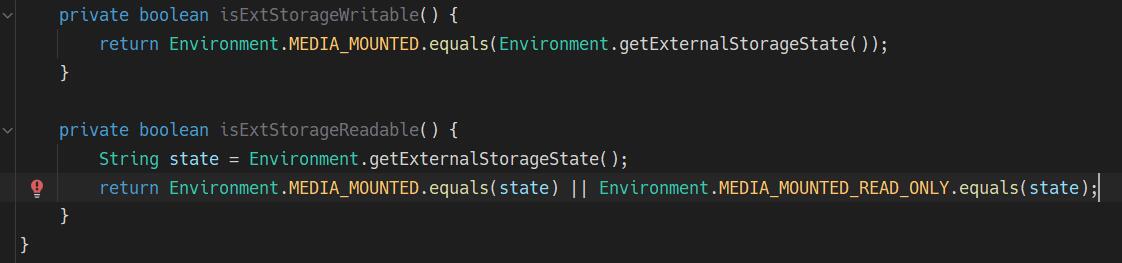
Рисунок 12 — при повороте экрана данные сохранились.

# 3. Реализовать создание, запись и удаление во внешней памяти на экране одного приложения и чтение выбранного файла в другом приложении

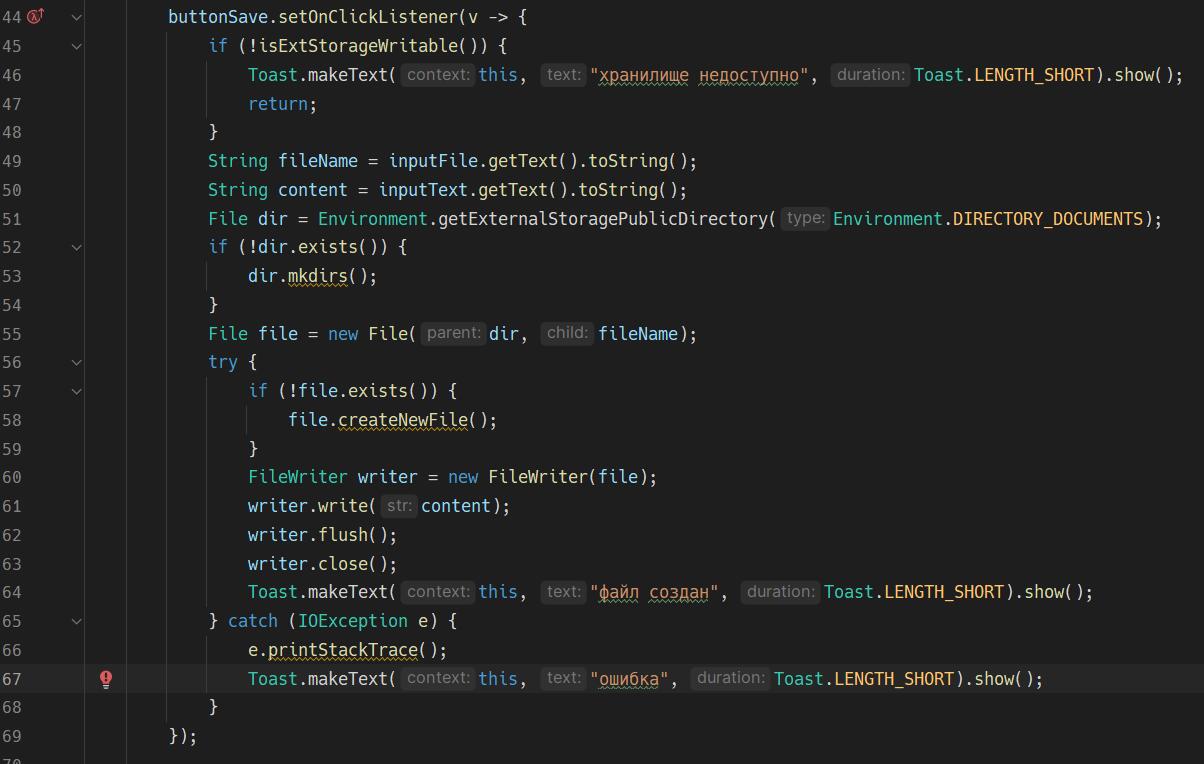
Для того, чтобы файл хранился во внешней памяти, изменяем первую программу. Для начала в AndroidManifest впишем разрешения.

Рисунок 13 — два разрешения на запись и чтения с внешней памяти, а так же requestLegacyExternalStorage.

Чтобы сократить код, создаем два приватных метода для проверки возможности чтения и записи.

Рисунок 14 — два метода.

Изменяем обработки кнопок на работу с внешнем хранилищем.

Рисунок 15 — вместо потоков как в прошлый раз, используем File и FileWriter.

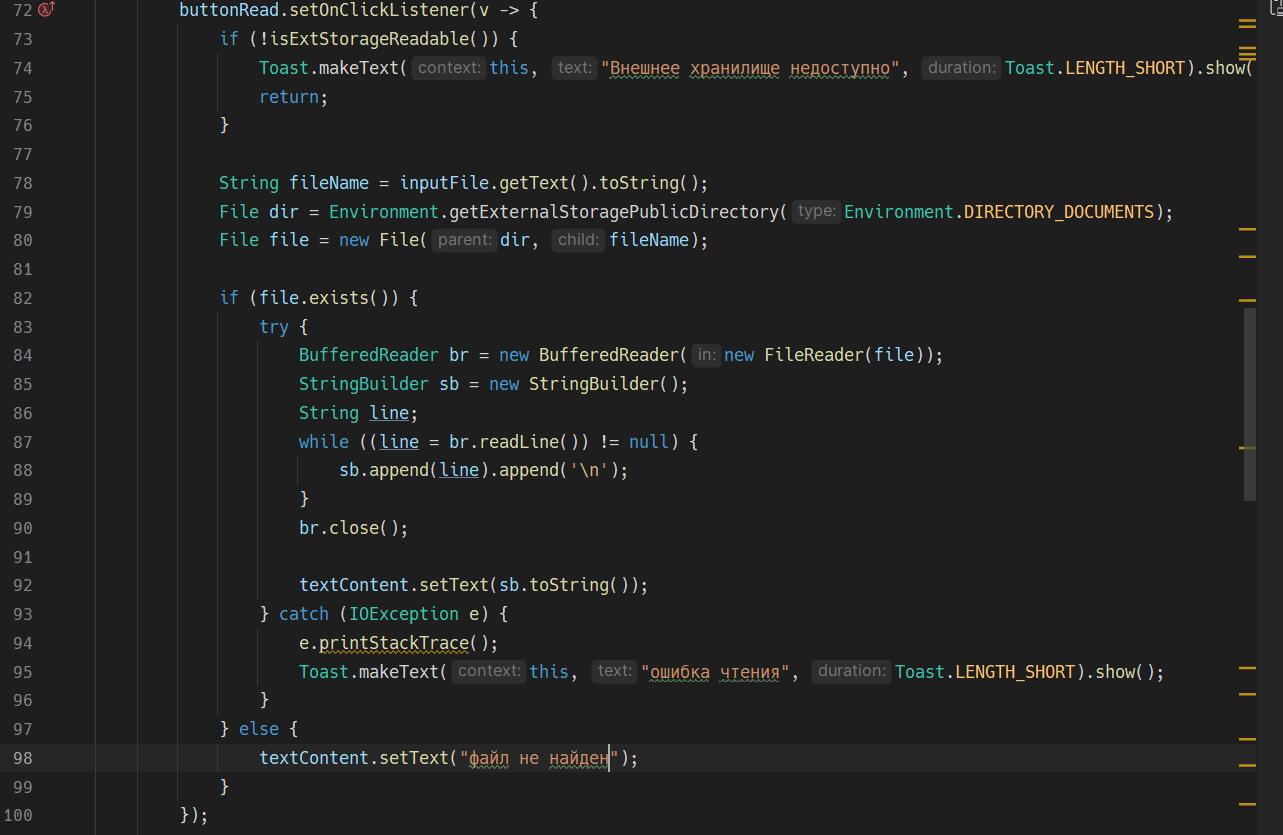
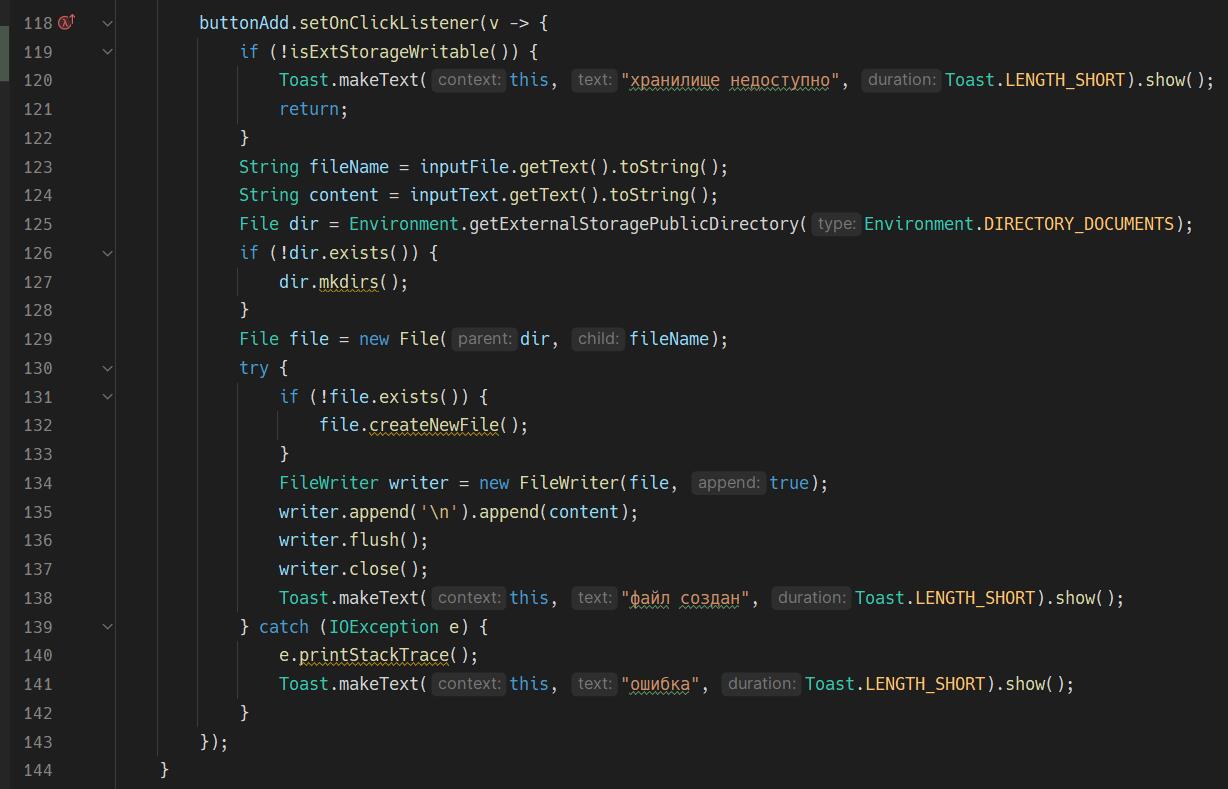
Рисунок 16 — измененная кнопка чтения.

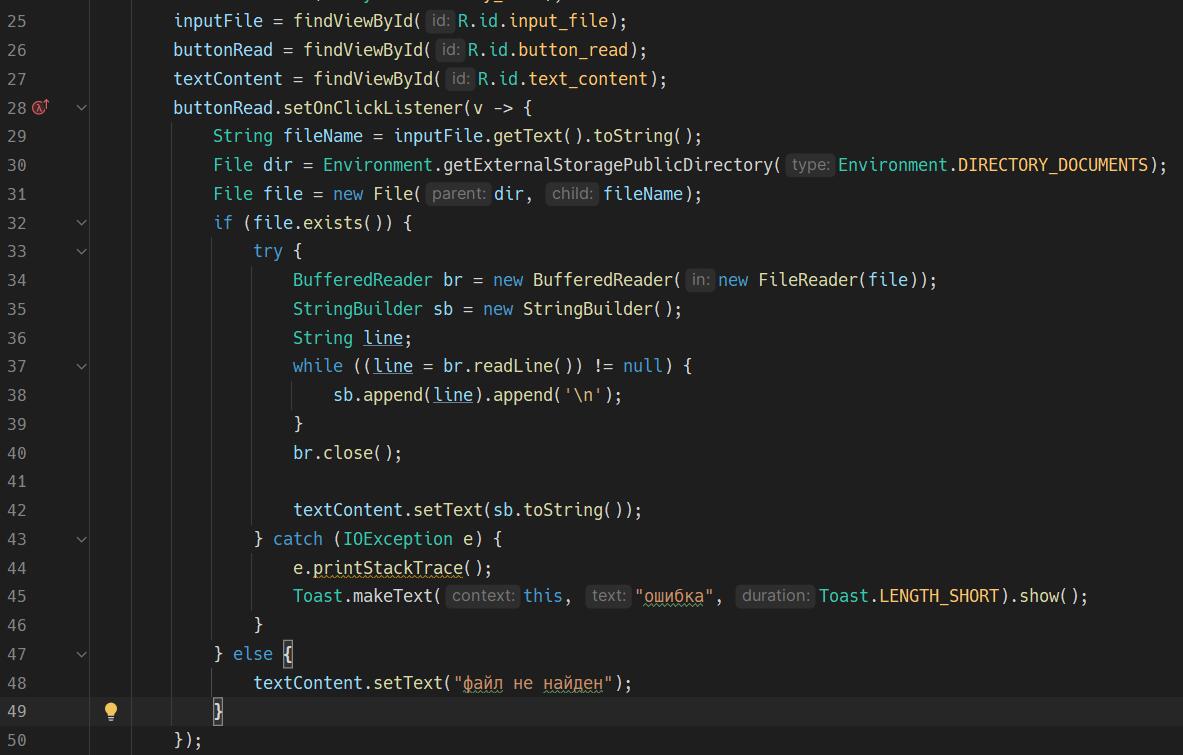
Рисунок 17 — измененная кнопка удаления.

Рисунок 18 — измененная кнопка добавления текста в файл.

Создаем вторую программу, которая будет выводить текст из файла. Добавляем разрешения.

Рисунок 19 — разрешения в AndroidManifest во второй программе.

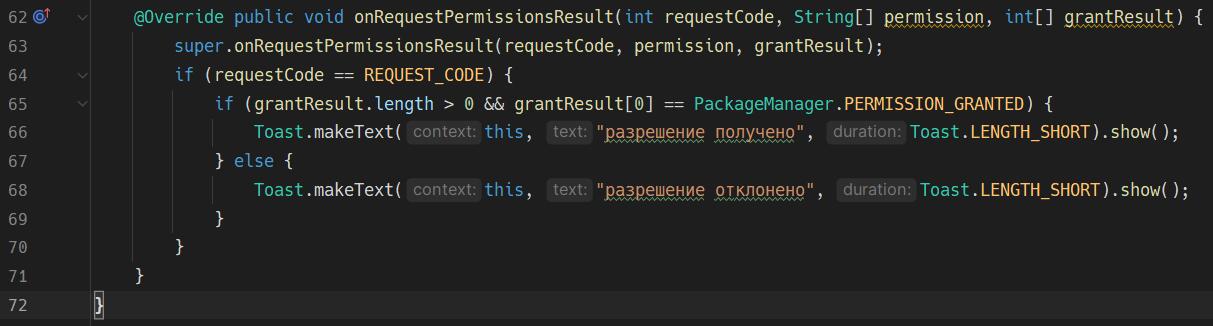
Обрабатываем кнопку чтения.

Риисунок 20 — кнопка чтения во второй программе.

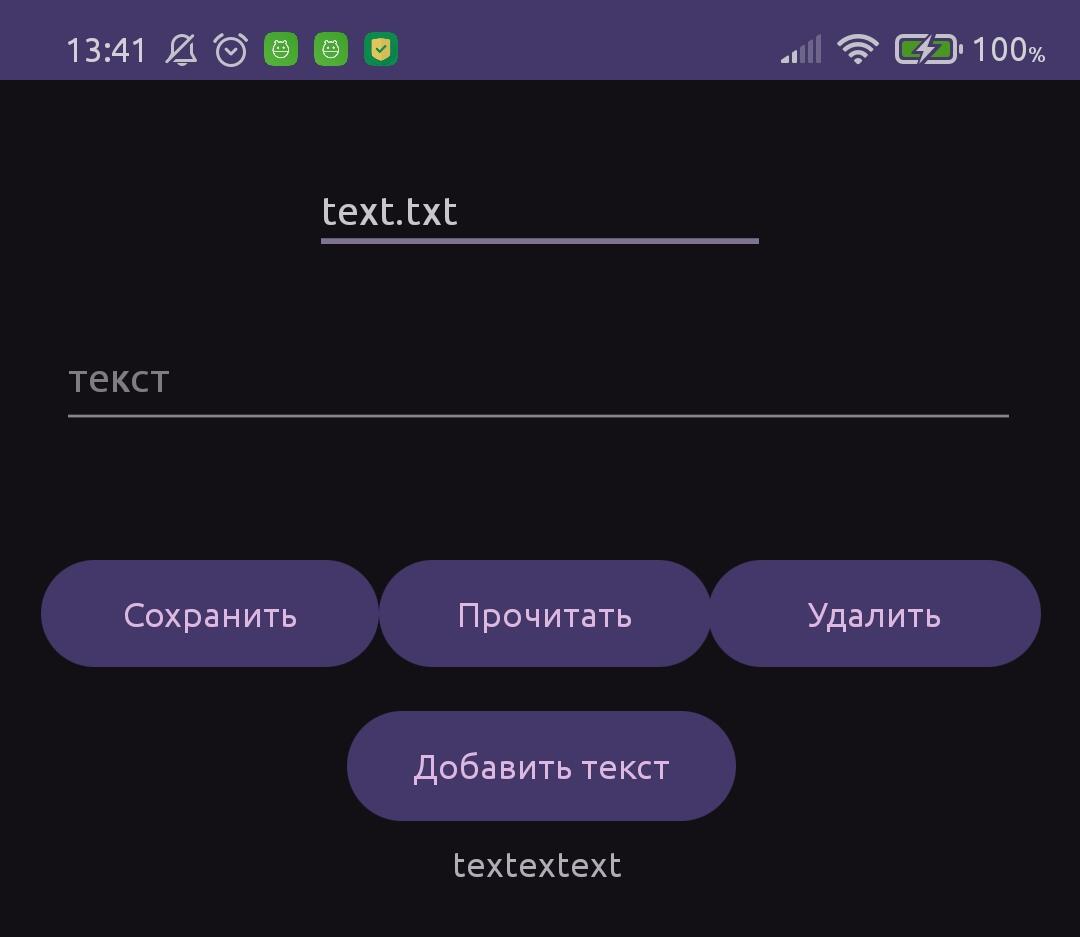
Если разрешение на чтения внешней памяти не предоставлено, то запрашиваем разрешение с request кодом.

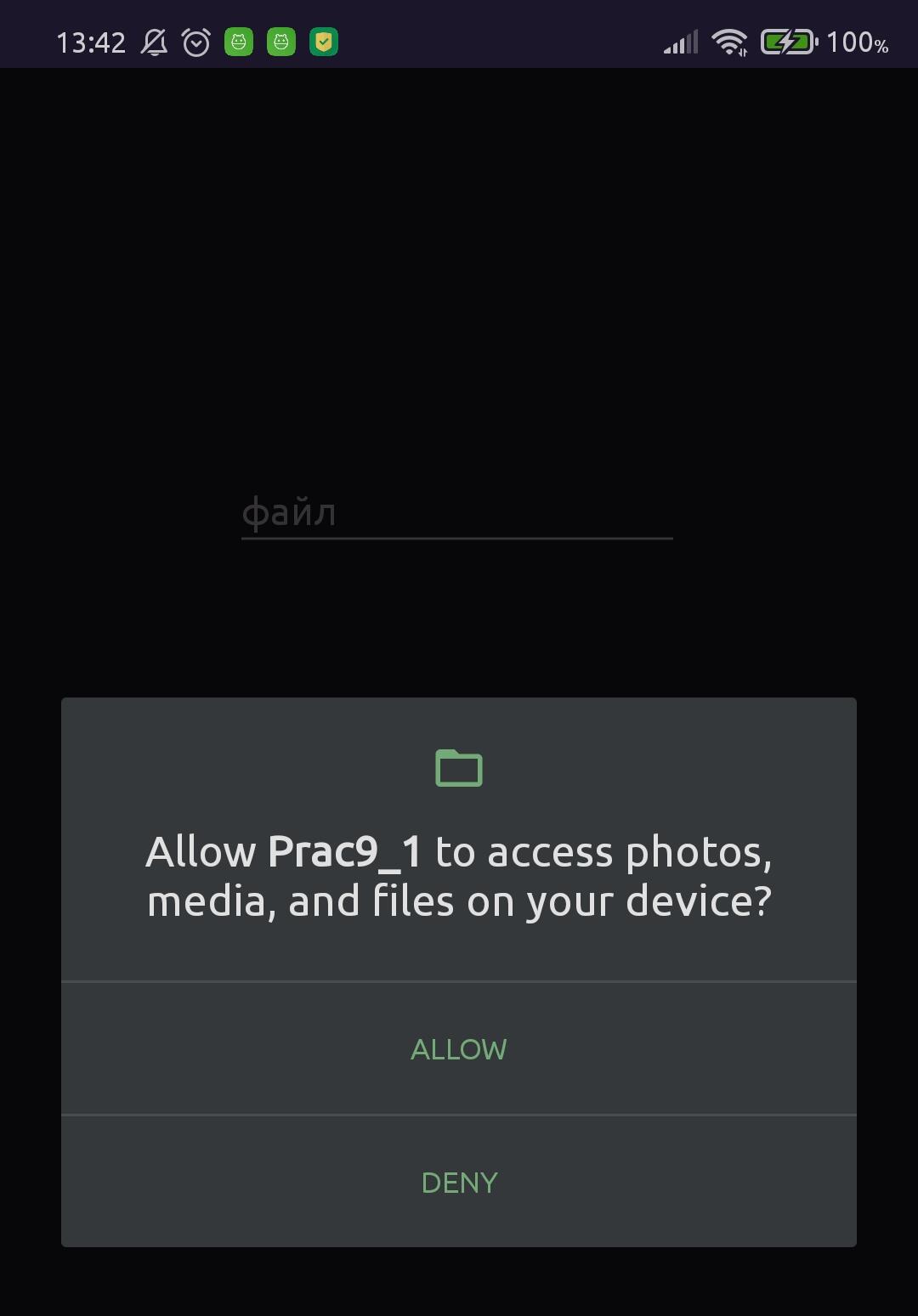
Рисунок 21 — requestPermissions – запрашивание разрешения.

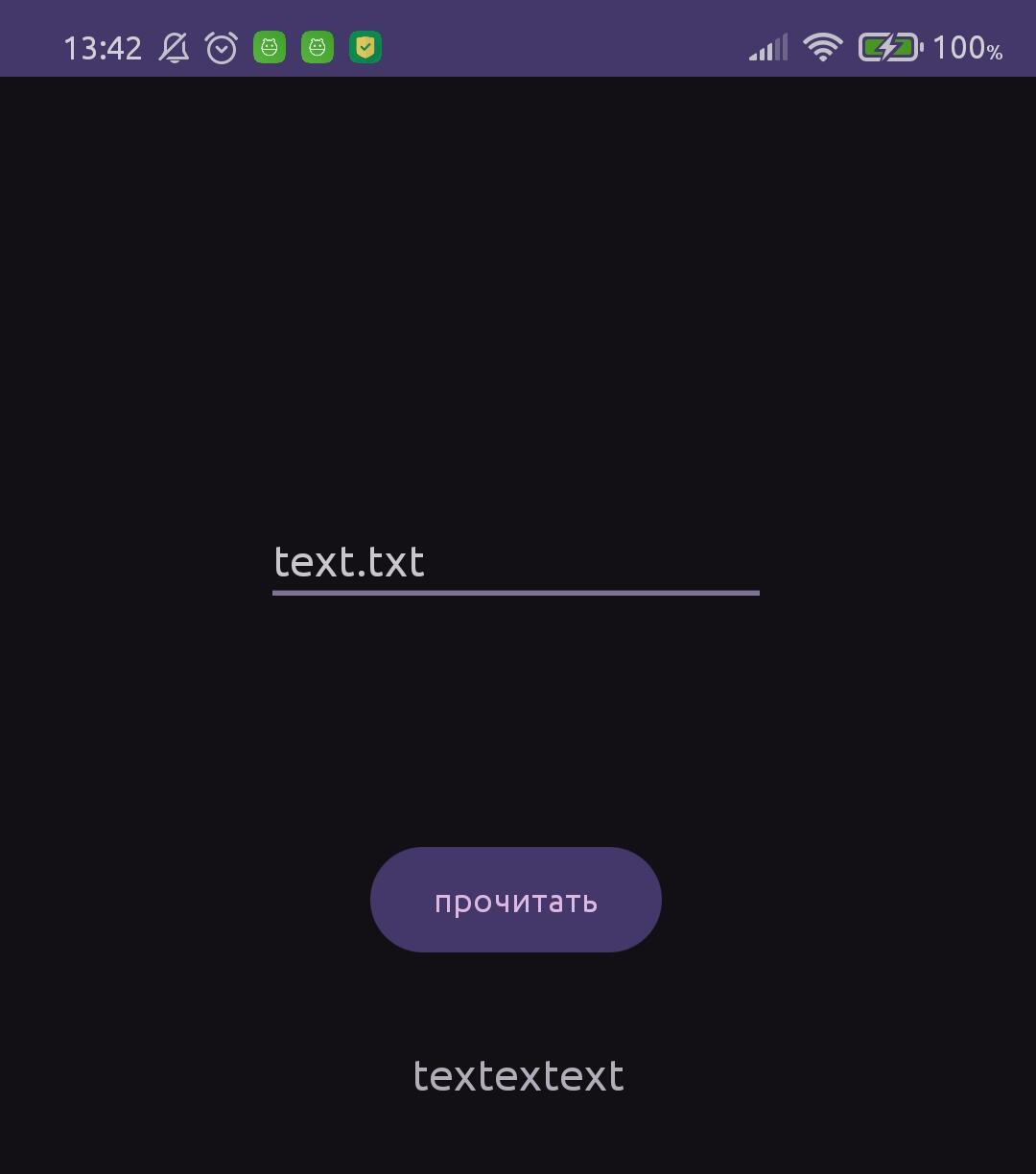
Создаем метод, чтобы Андроид просил у пользователя разрешение.

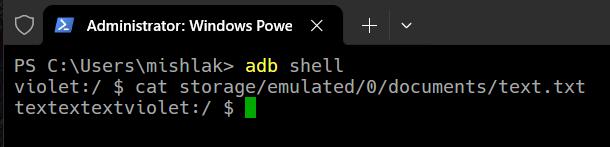
Рисунок 22 — метод onRequestPermissionsResult.

Проверяем работу.

Рисунок 23 — в первой программе создаем файл с текстом (на примере он не удалялся с прошлого задания)

Рисунок 24 — при заходе во вторую программу Андроид спрашивает разрешение.

Рисунок 25 — файл успешно прочитан.

Рисунок 26 — удостоверяемся, что файл лежит там, где надо.

# Вывод

В данной работе с работой с файлами, а также с разрешениями.

# Ссылка на репозиторий 1-2 задачи: [репозиторий](https://github.com/mihailkl/mobile-razrabotka_semestr4/tree/main/Prac9)

# Ссылка на репозиторий 3 задачи: [репозиторий](https://github.com/mihailkl/mobile-razrabotka_semestr4/tree/main/Prac9_1)

# 