|  |
| --- |
| МИНОБРНАУКИ РОССИИ |
| Федеральное государственное бюджетное образовательное учреждение высшего образования **«МИРЭА  Российский технологический университет»**  **РТУ МИРЭА** |

**Институт информационных технологий (ИИТ)**

**Кафедра математического обеспечения и стандартизации ИТ**

**ОТЧЁТ ПО ПРАКТИЧЕСКОМУ ЗАНЯТИЮ №8**

**по дисциплине**

**«Разработка мобильных приложений»**

|  |  |  |  |
| --- | --- | --- | --- |
| Отчет представлен к  рассмотрению:  Студенты группы ИНБО-04-20 | « » февраля 2022 г. | (подпись) | Ло В.Х. |
|  |  |  |  |
| Преподаватель | « » 2022 г. | (подпись) | Фандеев И.И. |

Москва, 2022г.

**СОДЕРЖАНИЕ**

[ОТЧЕТ ВЫПОЛНЕНИЯ ПРАКТИЧЕСКОЙ РАБОТЫ. 3](#_Toc99280861)

[1. Отправка простых данных другим приложениям 3](#_Toc99280862)

[2 Получение простых данных от других приложений 4](#_Toc99280869)

[3. Простое добавление кнопки шаринга 6](#_Toc99280870)

[4. Организация обмена файлами 7](#_Toc99280871)

[5. Передача файла. Получение запроса на файл. 8](#_Toc99280872)

[ВЫВОД 12](#_Toc99280873)

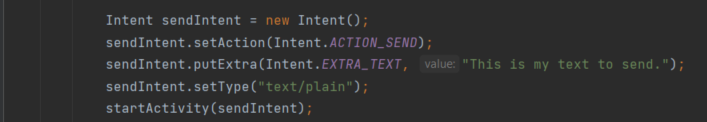
[СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ 12](#_Toc99280874)

# ОТЧЕТ ВЫПОЛНЕНИЯ ПРАКТИЧЕСКОЙ РАБОТЫ.

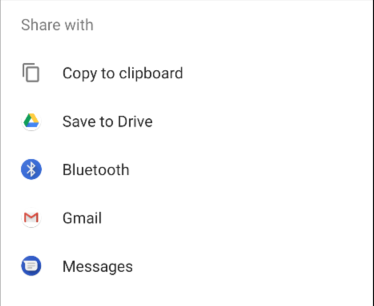
**1.** Отправка простых данных другим приложениям

**1.1** Отправка текста

Наиболее часто ACTION\_SEND используется для передачи текста из одного явления в другое. Например, браузер может передать адрес текущей страницы в виде текста любому другому приложению. Удобно использовать такой механизм, чтобы поделиться статьей или сайтом с друзьями, отправив им письмо или сообщение в социальной сети.

****

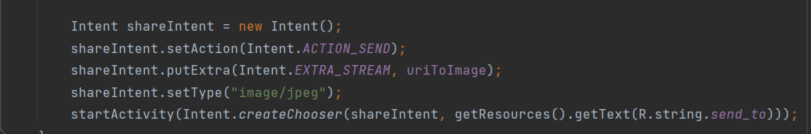
**Рисунок 1 –** Обмен данными

****

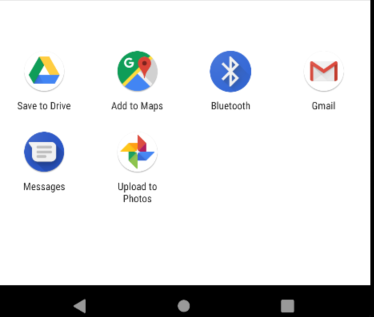
**Рисунок 2 – Результат**

**1.2** Передача бинарных данных

Бинарные данные также отправляются с использованием метода ACTION\_SEND, в сочетании с соответствующим MIME типом и передачей URI в расширение EXTRA\_STREAM. Обычно используется для передачи изображений, но может использоваться и для любых других бинарных данных:



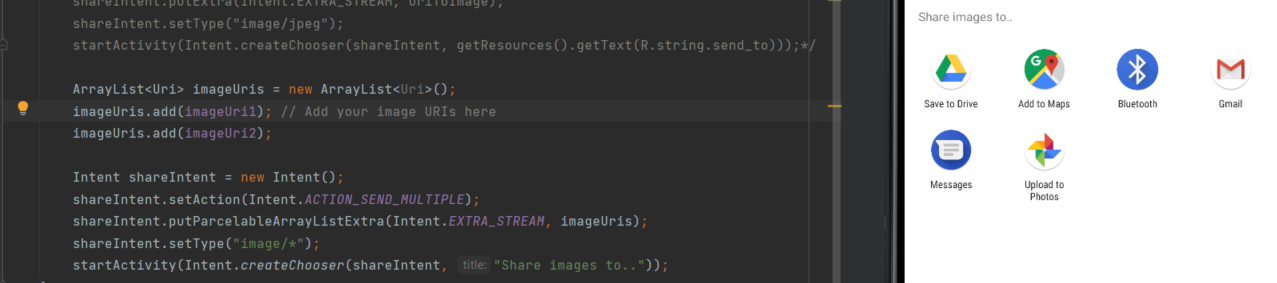
**Рисунок 3 –** Передача бинарных данных

****

**Рисунок 4 – Результат**

**1.3** Отправка нескольких частей данных

Чтобы отправить несколько частей данных, используйте действие ACTION\_SEND\_MULTIPLE совместно со списком URI-указателей. MIME тип может меняться в зависимости от данных. Например, если вы передаете 3 JPEG картинки, тип будет "images/jpeg". Для смеси различных типов картинок, тип должен быть "image/\*". Если вы передаете данные с большимколичеством разных типов, необходимо использовать "\*/\*". Как отмечалось ранее, это нужно для принимающего приложения, чтобы оно корректно обработало полученные данные.



**Рисунок 5 –** Отправка нескольких частей данных

**2** Получение простых данных от других приложений

2.1 Изменение файла манифеста

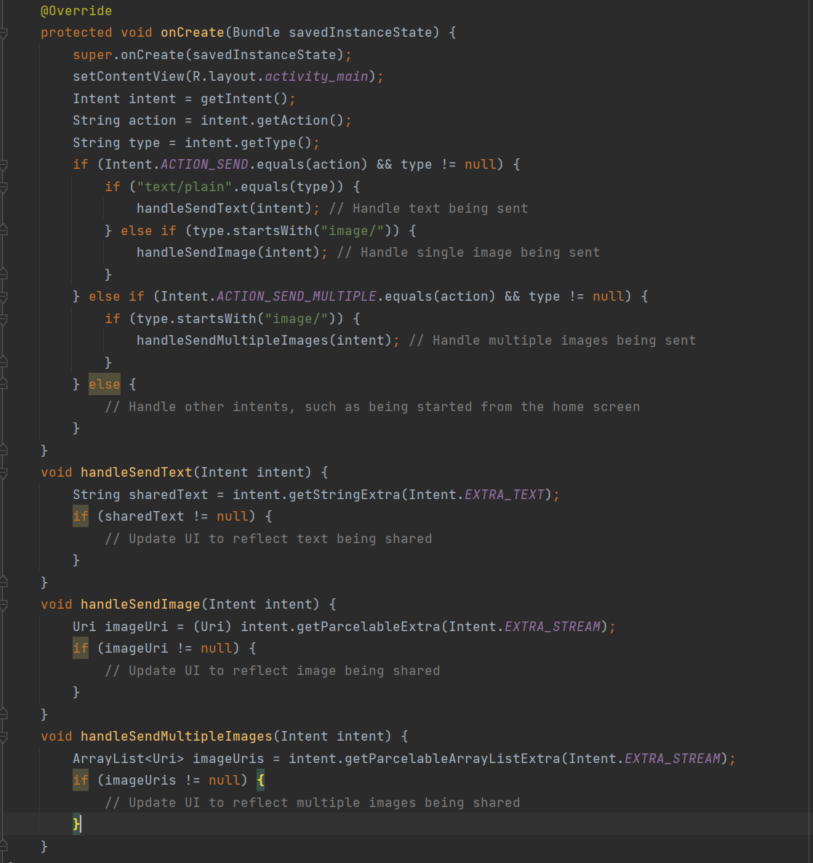
Фильтры намерений информируют систему о том, какие типы намерений может принять приложение. Также, как вы создавали намерения с действием ACTION\_SEND в предыдущем материале, вы можете создавать фильтры, чтобы получать намерения от других явлений. Для объявления фильтров в файле манифеста используется элемент <intent-filter>.



**Рисунок 6 –** Изменение файла манифеста

2.2 Обработка входящих данных

Для обработки данных, переданных в намерении, необходимо сначала получить экземпляр объекта типа Intent с помощью метода getIntent(). После этого вы можете изучить данные, чтобы решить, что с ними делать. Не забывайте, что если явление может быть запущено из разных приложений, а также из списка программ.

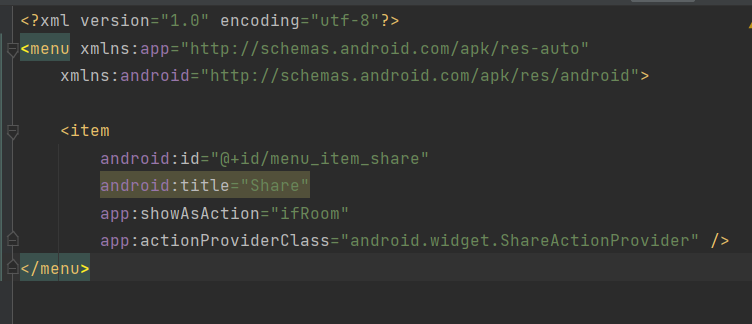


**Рисунок 7 –** Обработка входящих данных

3. Простое добавление кнопки шаринга

3.1 Изменяем меню

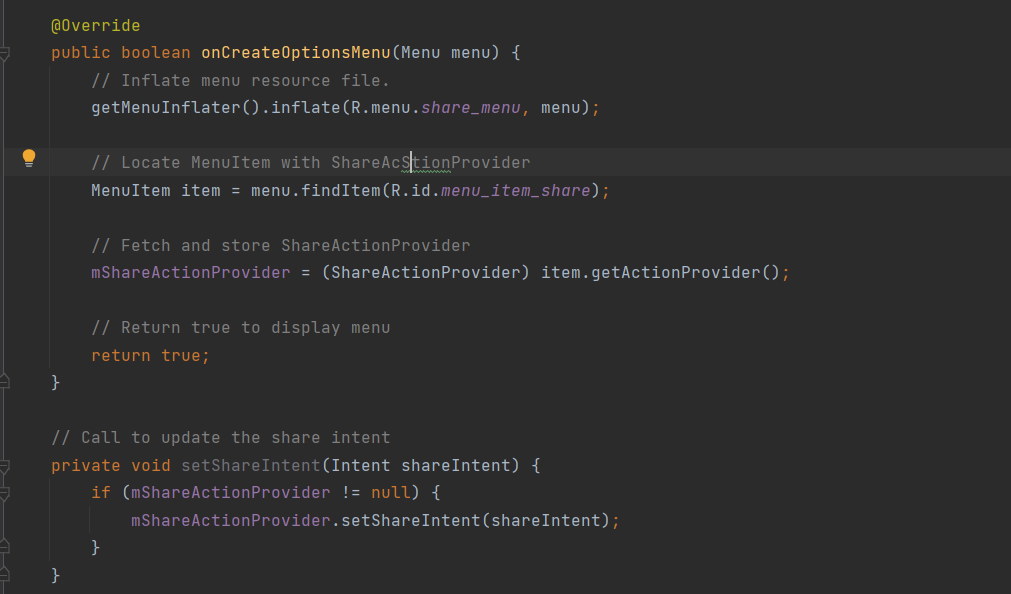
Чтобы использовать ShareActionProvider, добавьте атрибут android:actionProviderClass в соответствующий пункт <item> вашего файла меню.



**Рисунок 8 –** Изменяем меню

3.2 Создаем намерение расшаривания

Для использования ShareActionProvider, необходимо создать намерение расшаривания. Это намерение должно быть таким, как описано в уроке Отправка простых данных другим приложениям, должно использовать действие ACTION\_SEND и передавать дополнительные данные с помощью расширений EXTRA\_TEXT и EXTRA\_STREAM. Чтобы назначить намерение, необходимо сначала найти соответствующий пункт во время создания меню в явлении или фрагменте. Затем нужно вызвать метод MenuItem.getActionProvider(), чтобы получить экземпляр ShareActionProvider. Используйте метод setShareIntent(), чтобы связать намерение расшаривания с пунктом меню.

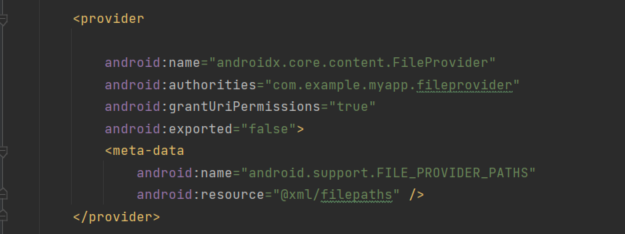


**Рисунок 9 –** Создаем намерение расшаривания

4. Организация обмена файлами

4.1 Указание объекта FileProvider

Для определения объекта FileProvider требуется добавить запись в файл манифеста. Эта запись дает право генерировать URI для данных, а так жесодержит имя XML файла, в котором указано какими файлами может поделиться ваше приложение. Следующий пример показывает, как добавить элемент <provider> в манифест, чтобы определить объект FileProvider, authority и имя XML файла:



**Рисунок 10 – Определить объект FileProvider, authority и имя XML файла**

4.2 Указание общих директорий

После добавления объекта FileProvider в манифест, необходимо указать директории, содержащие файлы, которыми вы готовы поделиться. Для указания директорий создайте файл filepath.xml в директории res/xml вашего проекта. В файле добавьте XML элементы для каждой директории. Далее приведен пример файла filepath.xml, который показывает как поделиться подкаталогом директории files/ из вашего внутреннего хранилища:

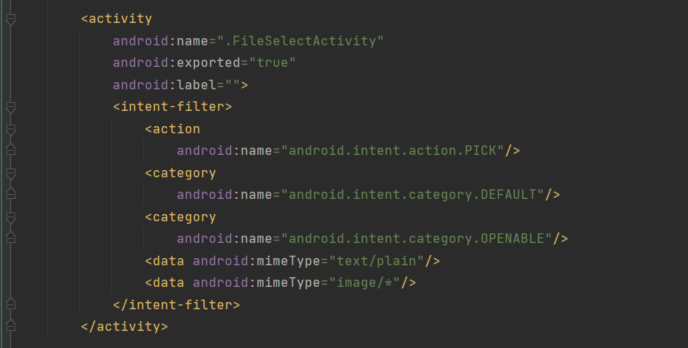


**Рисунок 11 –** Указание общих директорий

5. Передача файла. Получение запроса на файл.

5.1 Создание явления выбора файла

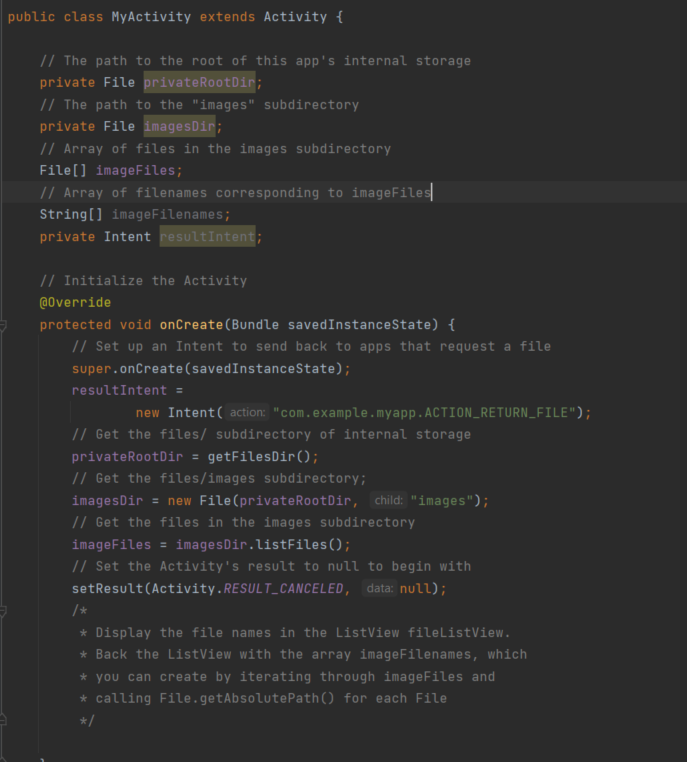
Чтобы создать явление выбора файл, сначала опишите явление в манифесте, включив в описание фильтр намерения с действием ACTION\_PICK и категориями CATEGORY\_DEFAULT и CATEGORY\_OPENABLE. Добавьте также MIME типы для файлов вашего приложения.



**Рисунок 12 –** Создание явления выбора файла

5.2 Программное создание явления выбора файла

Далее описан подкласс Activity, показывающий доступные файлы из директории files/images/ внутреннего хранилища и позволяющий пользователям выбрать необходимый файл:

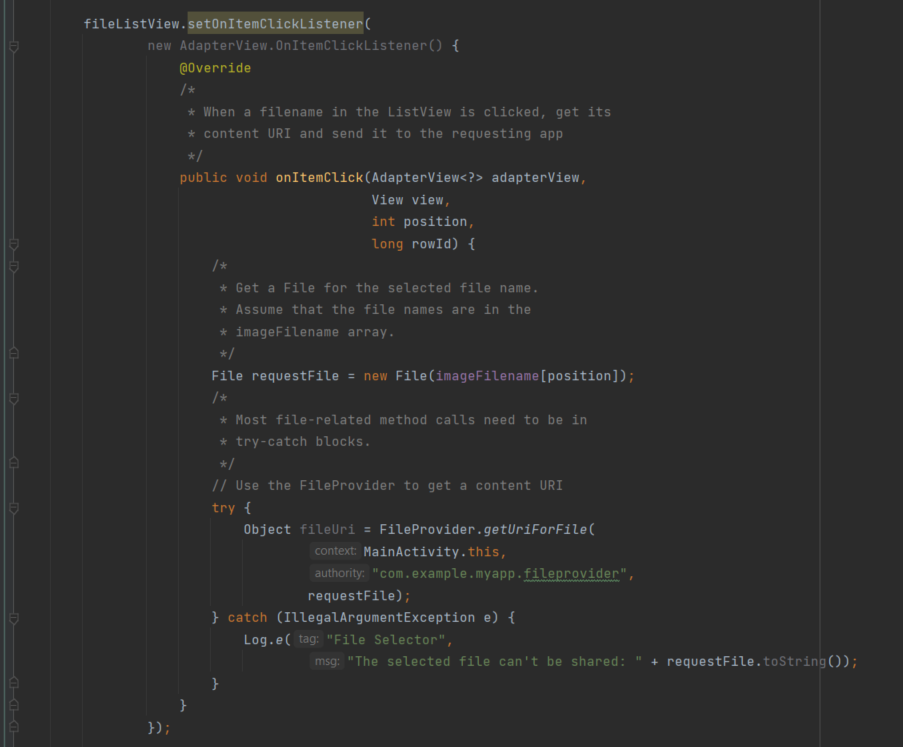


**Рисунок 13 – Программное создание явления выбора файла**

5.3 Обработчик выбора файла

При выборе файла, ваше приложение должно определить что за файл был выбран и сгенерировать для него URI. Явление показывает список доступных файлов с помощью компонента ListView. При клике по названию файласистема вызывает метод onItemClick(), в котором можно получить выбранный файл.

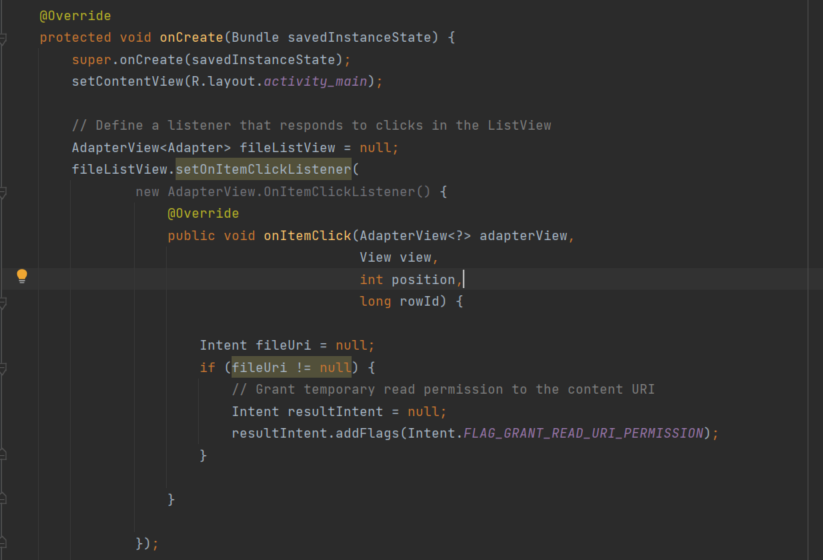
В методе onItemClick() получим объект типа File и передадим в качестве аргумента в метод getUriForFile() вместе с authority, который мы указали ранее в элементе <provider>. В результате мы получим URI, содержащий authority, путь до каталога и имя файла с расширением. Как класс FileProvider размечает директории и вычисляет для них пути на основе мета-данных из XML, мы рассмотрели в предыдущем материале.



**Рисунок 14 –** Обработчик выбора файла

5.4 Выдача разрешения на файл

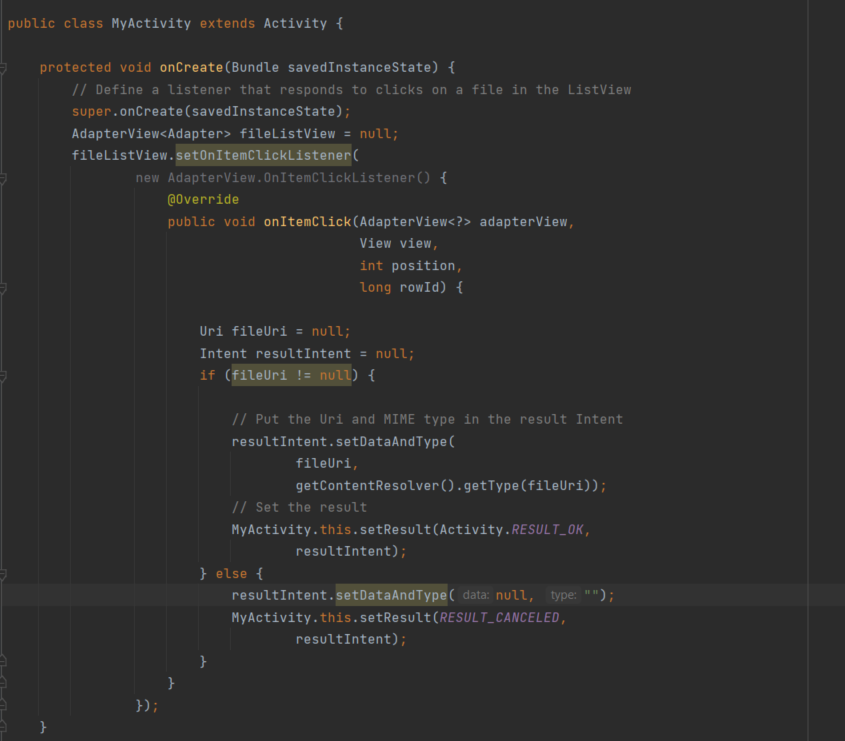
После получения URI необходимо выдать разрешение на доступ к этому файлу приложению, которое его запросило. Для этого нужно передать URI в намерение, а затем установить в намерении разрешающие флаги. Выданное разрешение будет временным и автоматически отменится после обработки файла другим приложением.



**Рисунок 15 –** Выдача разрешения на файл

**5.5** Делимся файлами с приложениями

Чтобы поделиться файлом с приложением, которое этот файл запросило, передайте намерение, включающее URI файла и разрешение в метод setResult(). После завершения вашего явления, оно передаст это намерение в клиентское приложение.

****

**Рисунок 16 –** Делимся файлами с приложениями\

# ВЫВОД

В ходе выполнения практической работы были изучены:

* Обмен простыми данными. Обмен файлами.

**СПИСОК ИСПОЛЬЗОВАННЫХ ИСТОЧНИКОВ**

1. Лекции по дисциплине «Разработка мобильных приложений» / И. В. Синицын, МИРЭА — Российский технологический университет, 2022.