|  |  |
| --- | --- |
|  | |
| |  | | --- | | –МИНОБРНАУКИ РОССИИ | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |
| Институт кибербезопасности и цифровых технологий | |
| *(наименование института, филиала)* | |
| Кафедра КБ-2 «Информационно-аналитические системы кибербезопасности» | |
| *(наименование кафедры)* | |
| **ОТЧЁТ ПО ПРАКТИЧЕСКОЙ РАБОТЕ № 5**  по дисциплине «Разработка мобильных компонент анализа безопасности информационно-аналитических систем» | | |
|  | | |
|  | | |

|  |  |  |
| --- | --- | --- |
| Отчет представлен к рассмотрению: |  |  |
| Студент группы БИСО-03-20 | «     »                    20\_\_ г. | Смирнов М.В. |
|  |  | (подпись и расшифровка подписи) |
|  |  |  |
| Проверил преподаватель | «     »                    20\_\_ г. | Изергин Д.А. |
|  |  | (подпись и расшифровка подписи) |

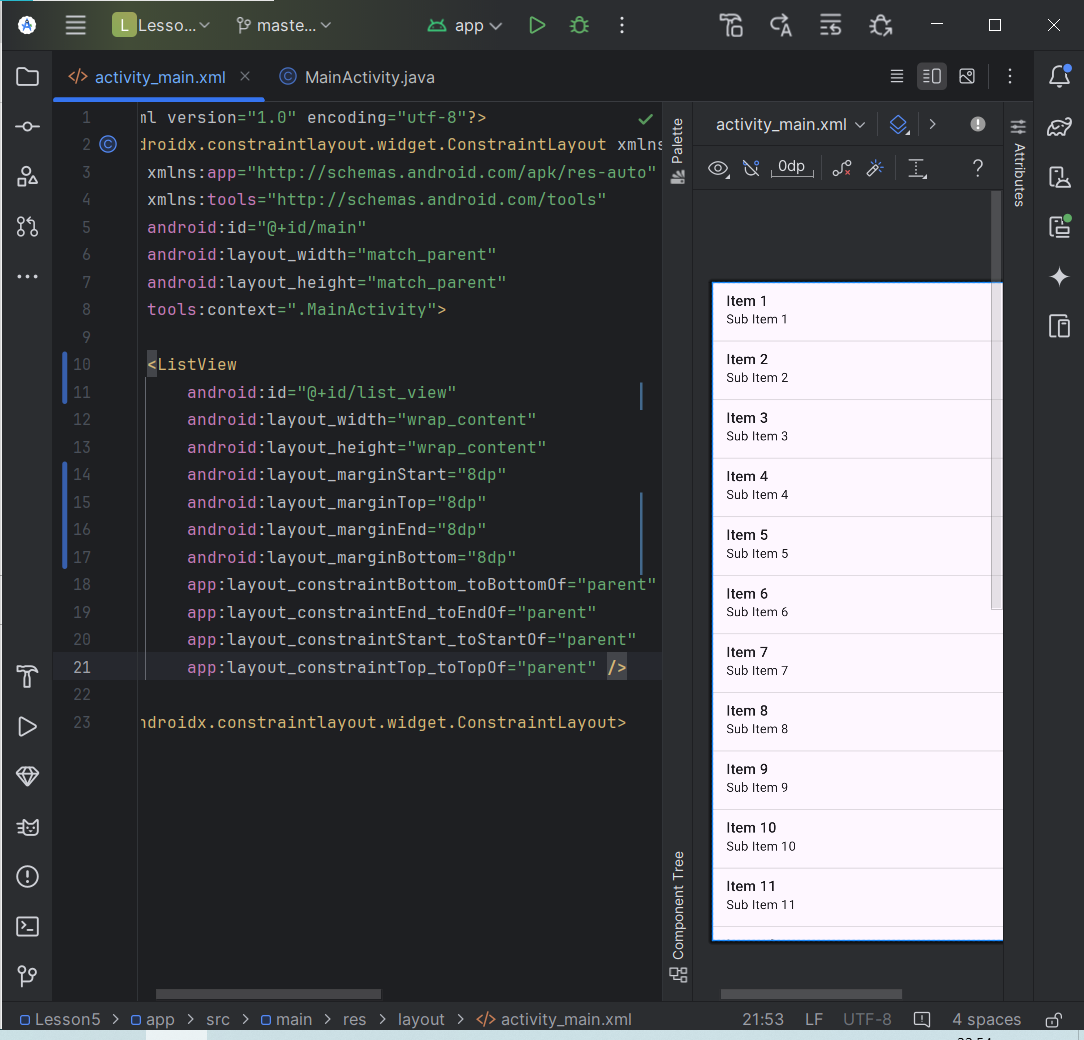
Москва 2025

1. ОСНОВЫ ИСПОЛЬЗОВАНИЯ АППАРАТНЫХ ВОЗМОЖНОСТЕЙ МОБИЛЬНЫХ УСТРОЙСТВ

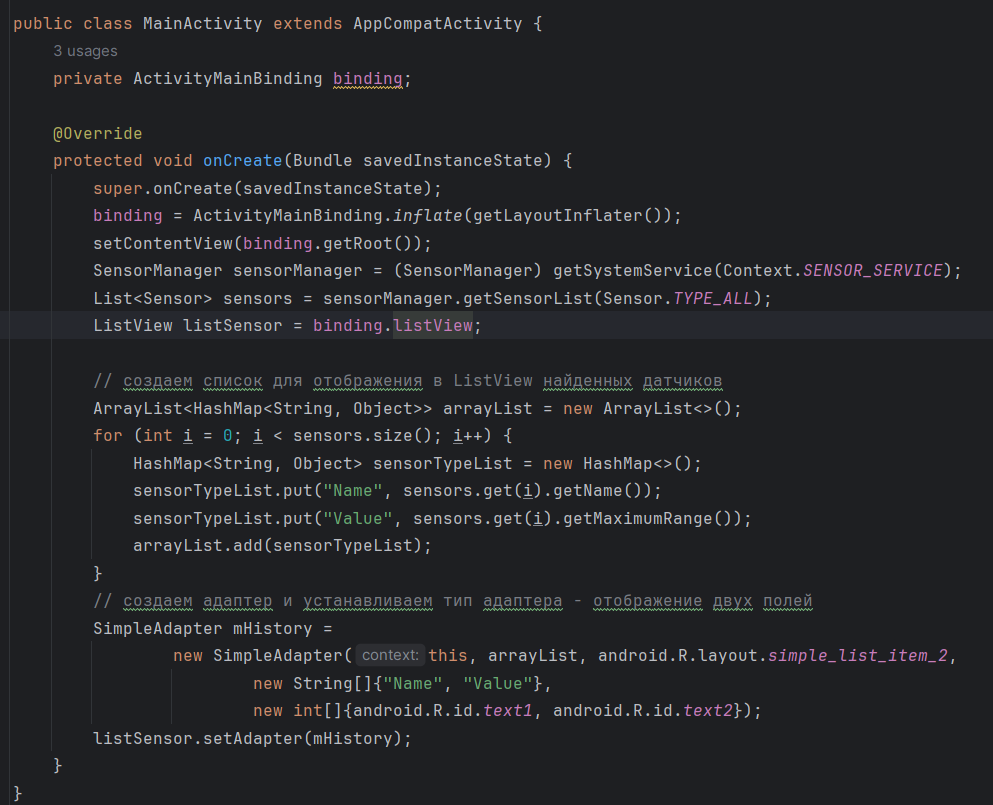


* 1. Задание. Список датчиков.

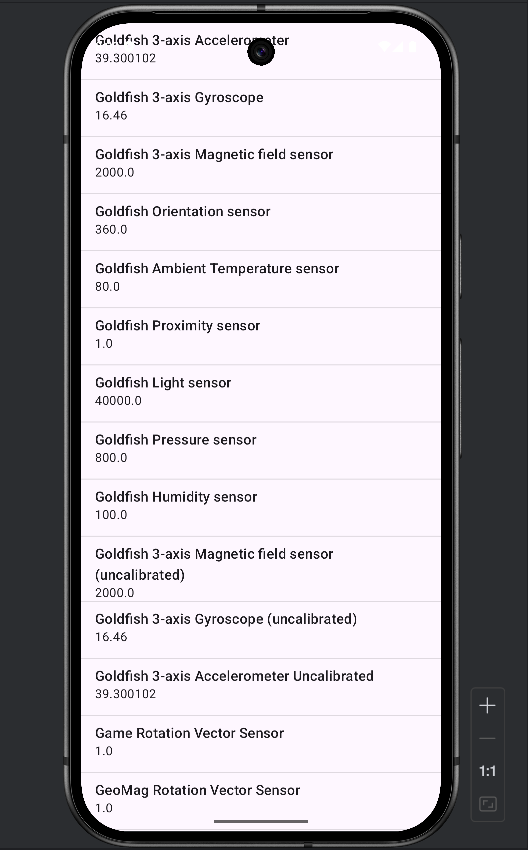
В «activity\_main» требуется добавить в файл разметки «ListView»:



Далее приведён код класса «MainActivity», предназначенный для получения списка датчиков и отображении их в компоненте список:

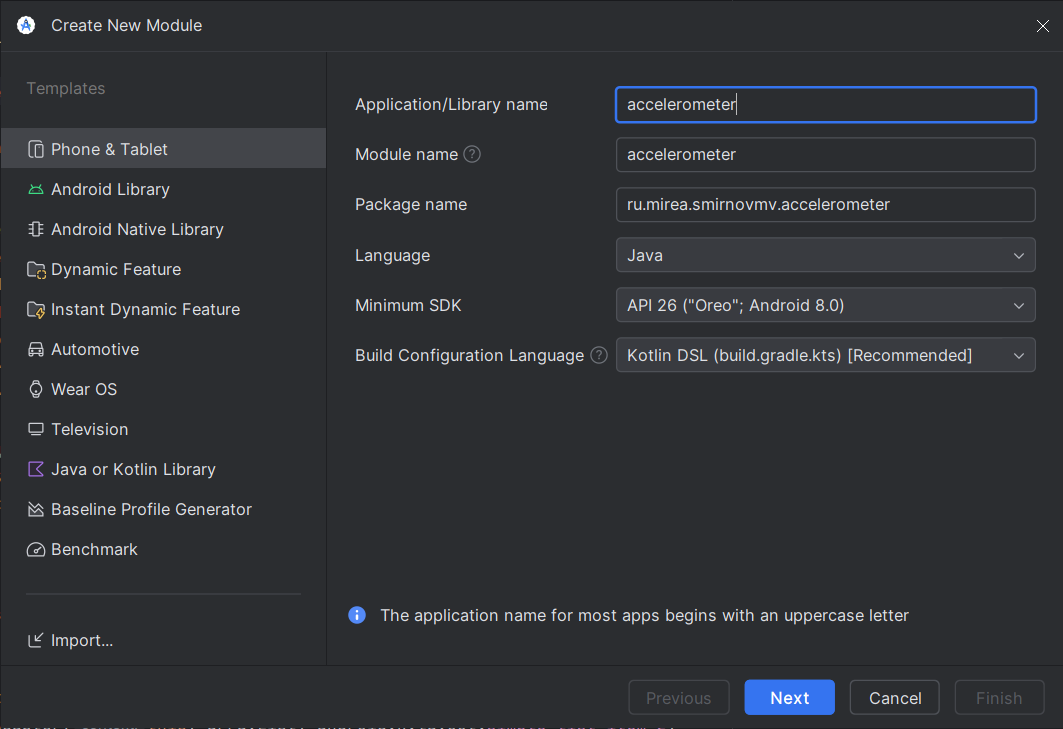


Результат:

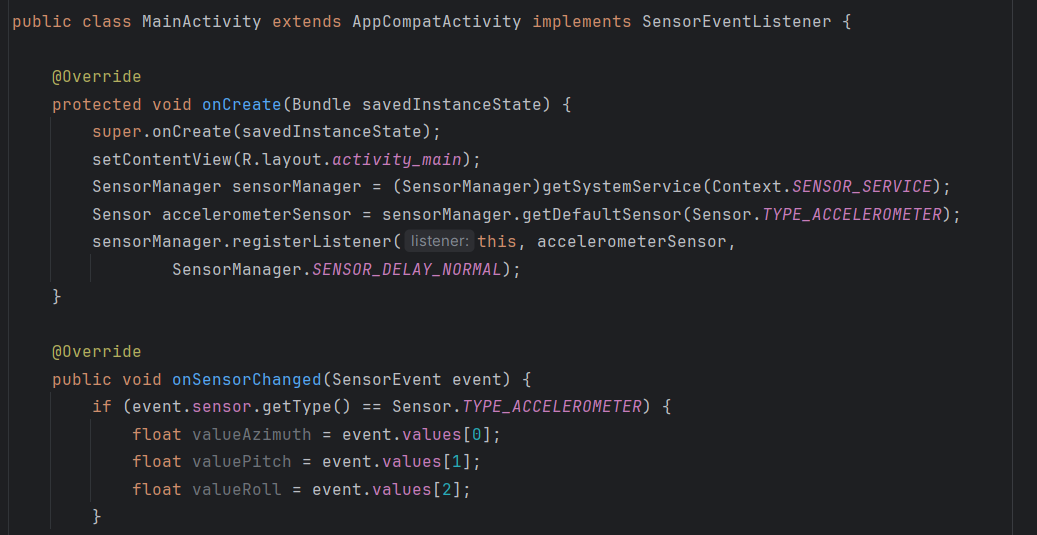


* 1. Показания акселерометра

Создаём новый модуль

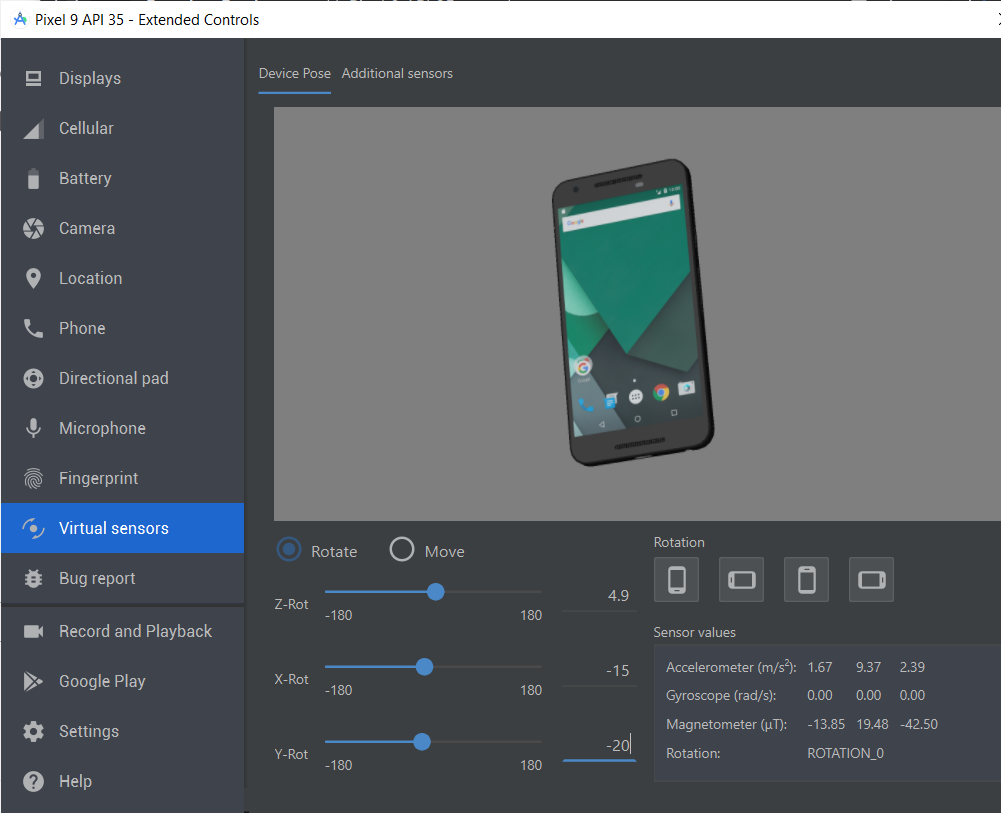


Пропишем MainActivity.java



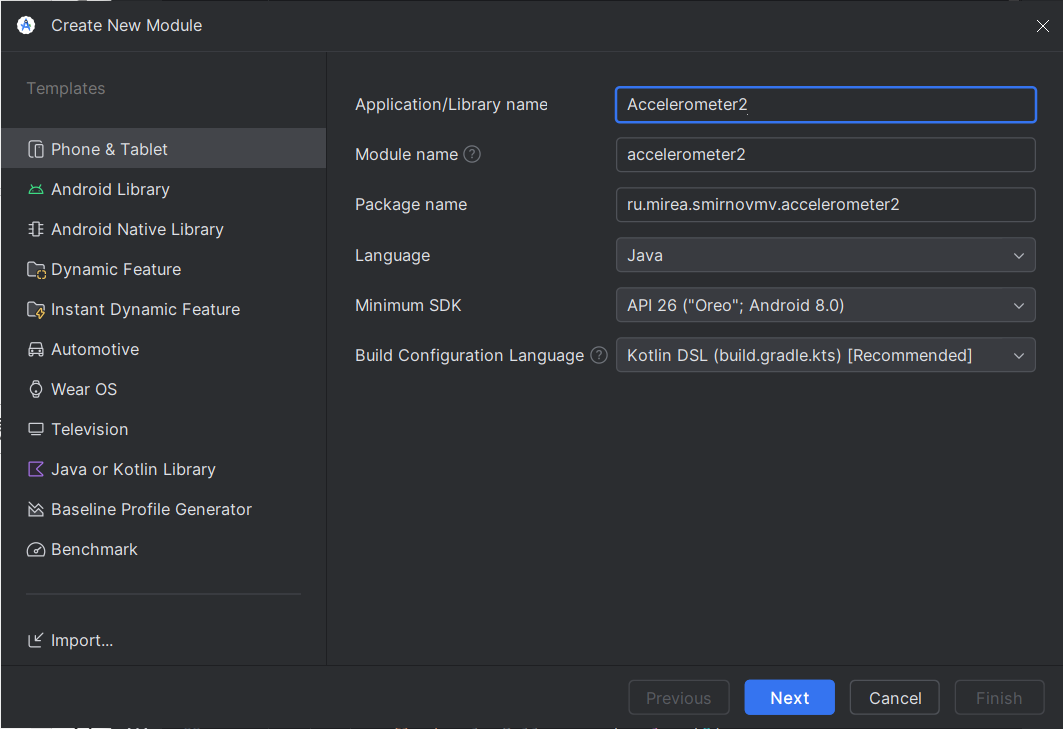
Далее при запуске нажмём на 3 точки в строке





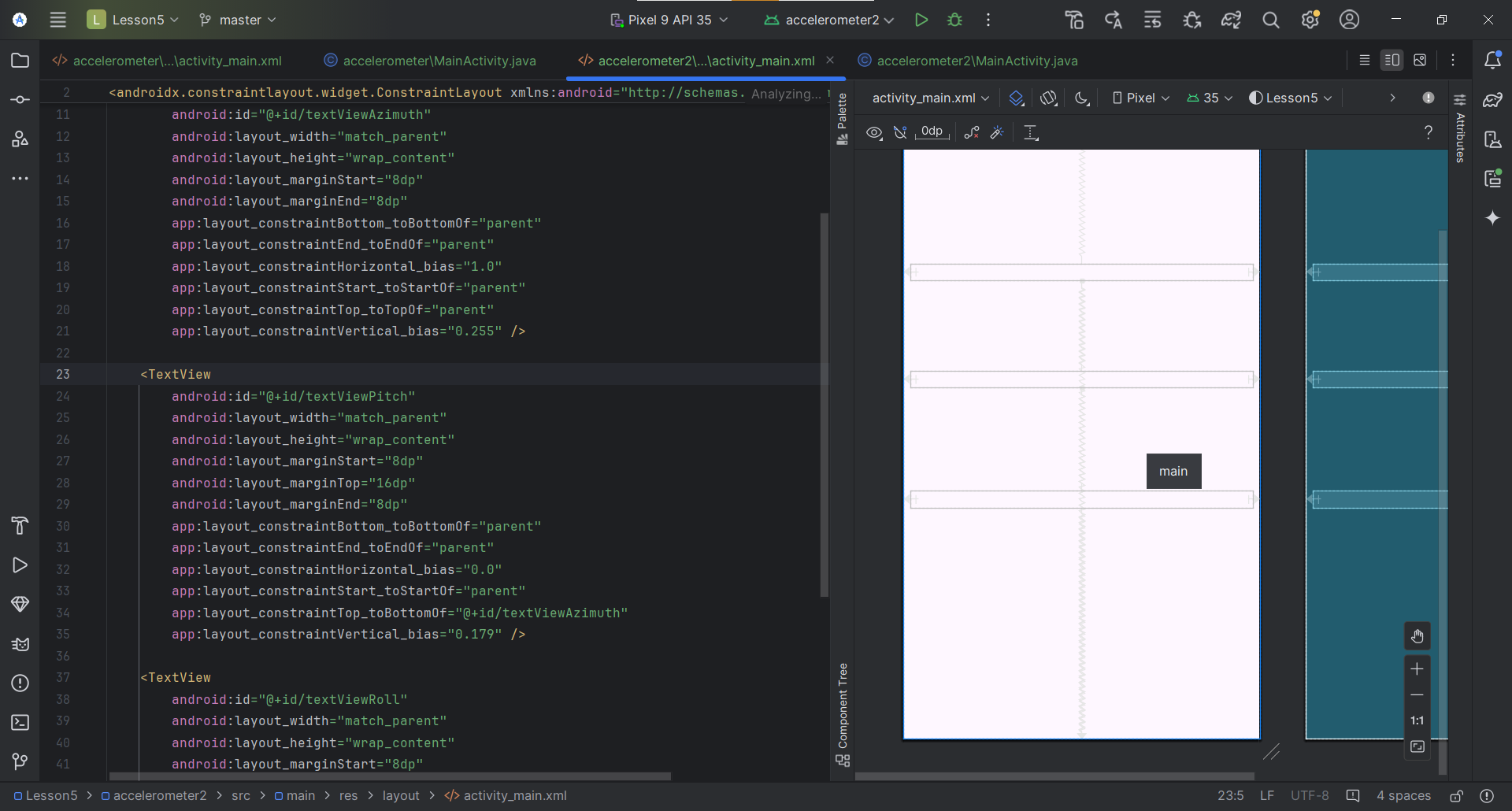
1. ЗАДАНИЕ.

Создадим новый модуль

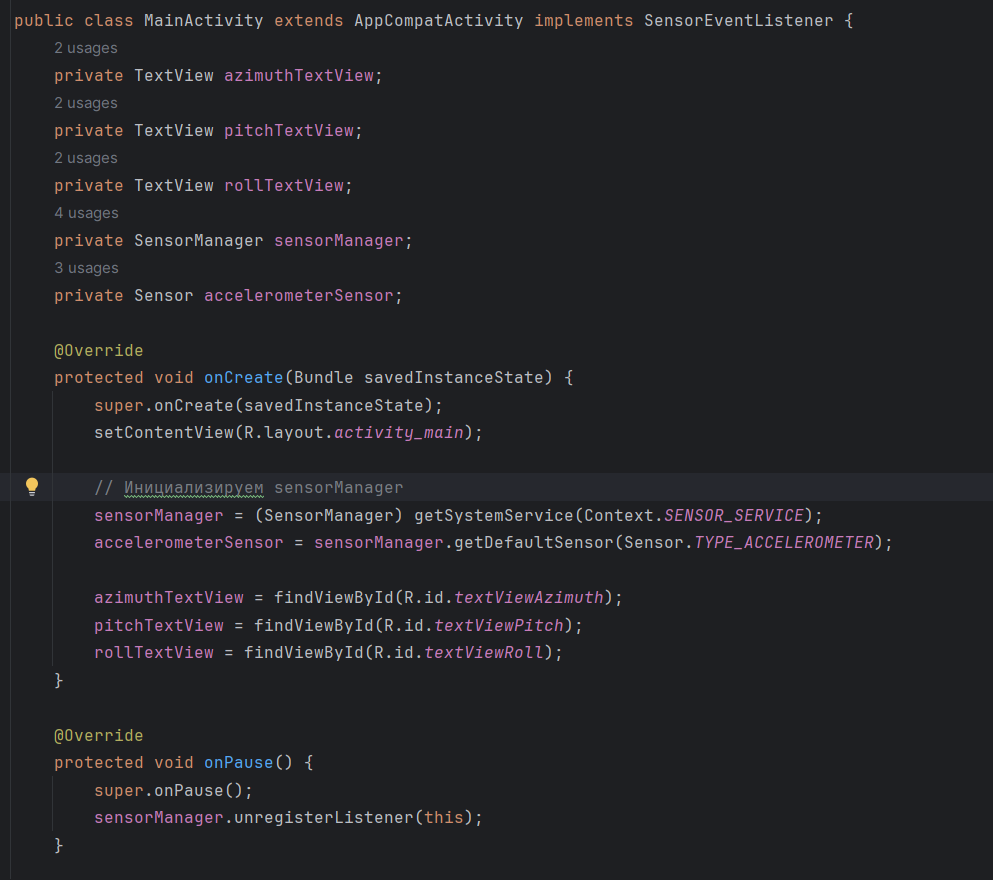


В первую очередь требуется реализовать экран с 3 текстовыми полями

в «activity\_main.xml».



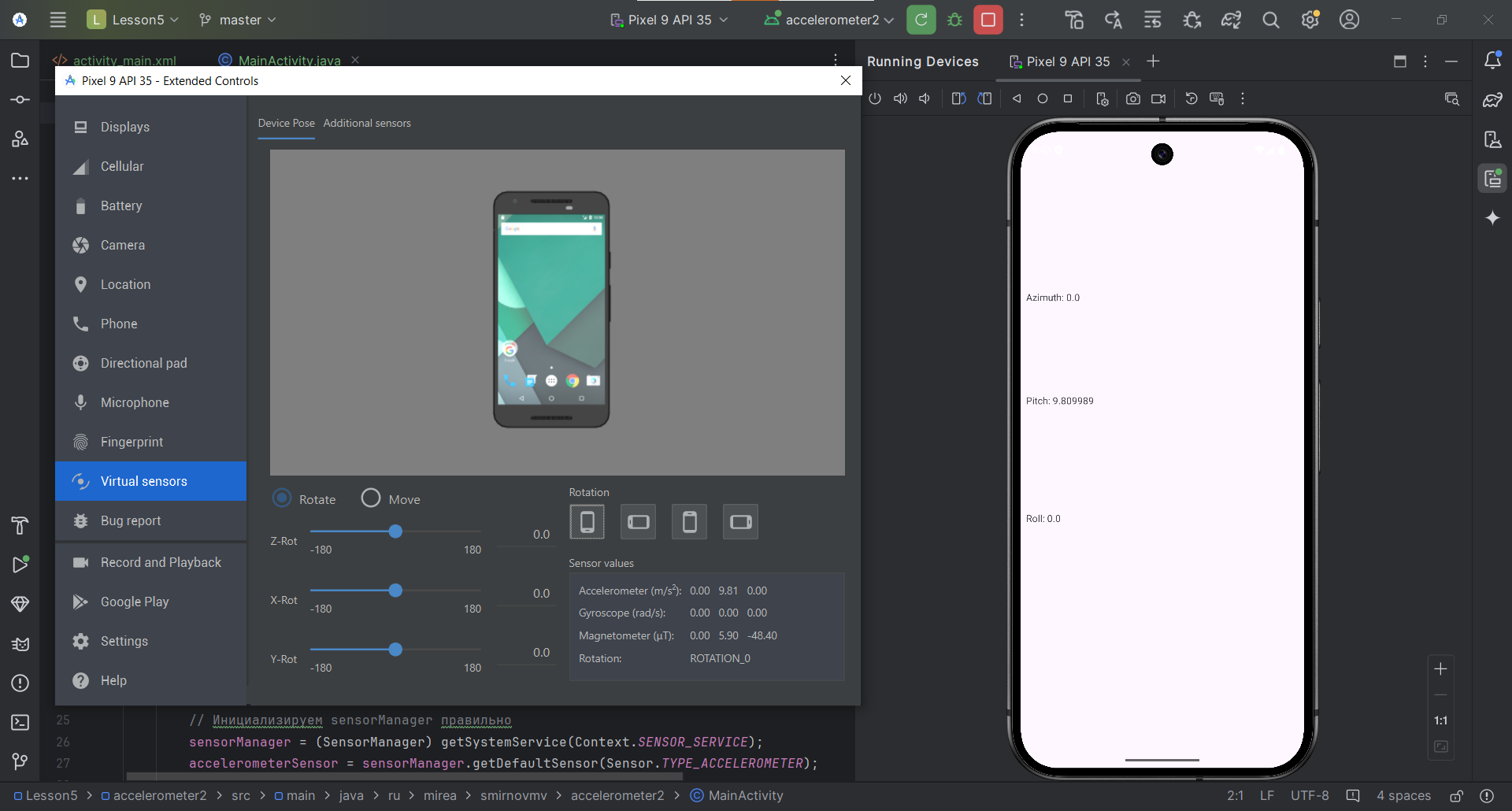
Далее немного исправляем объявление sensor manager в mainactivity.



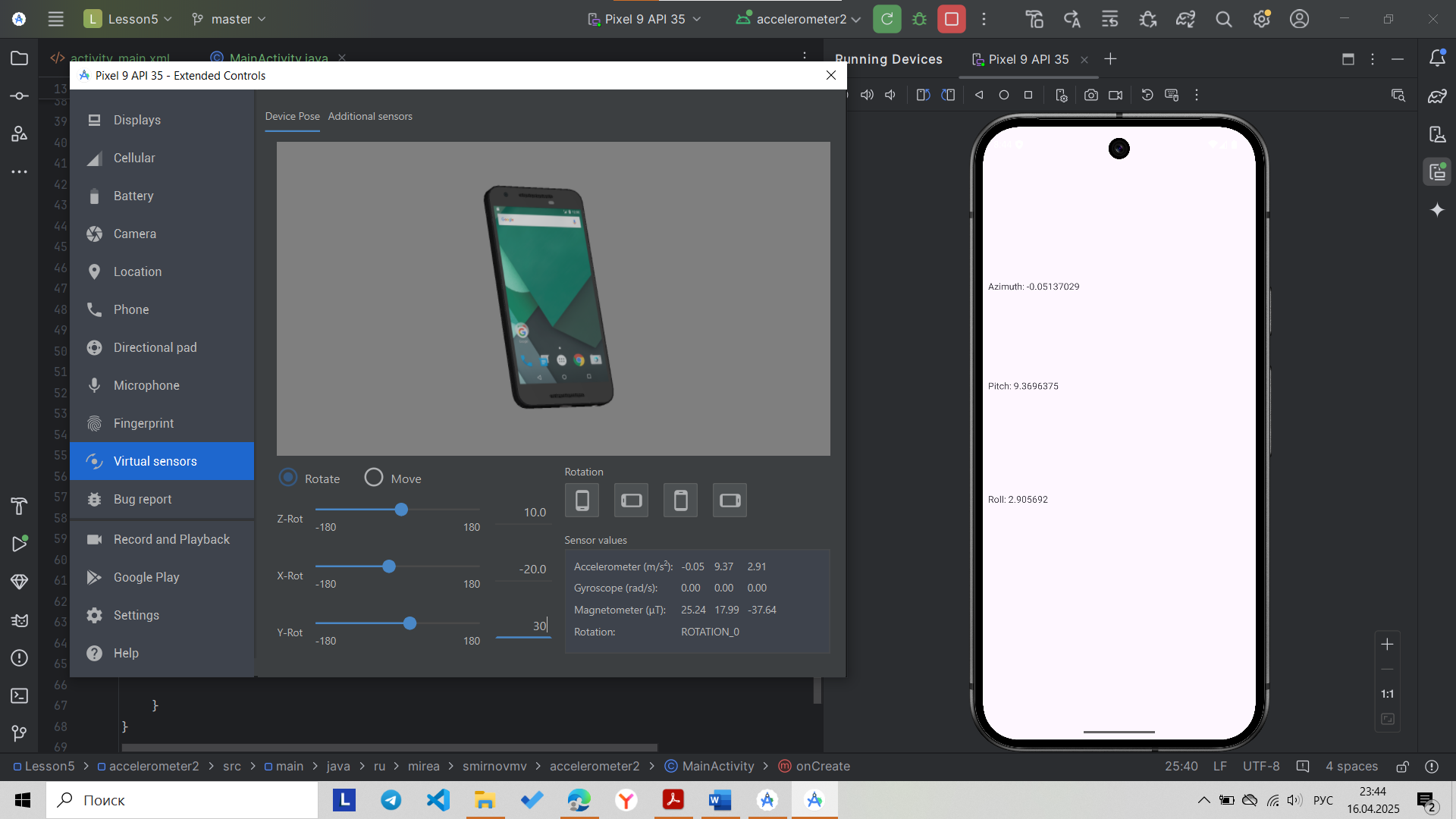


Запускаем программу

При 0 отклонении

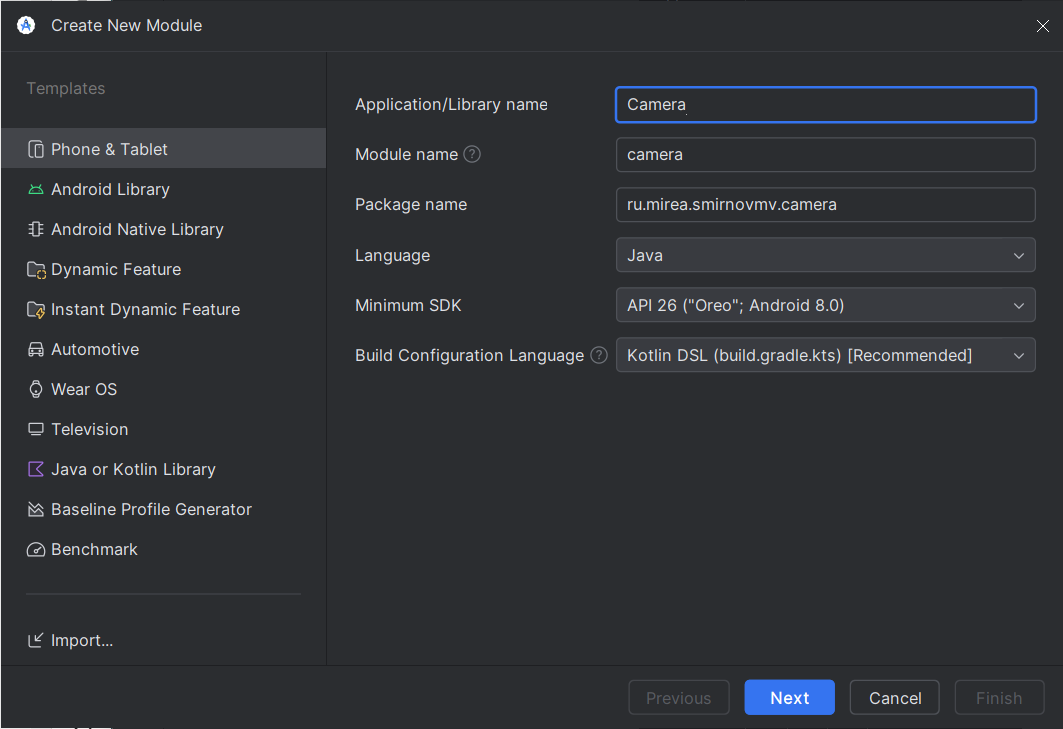


Видим, что при изменении значений меняются значения на экране.

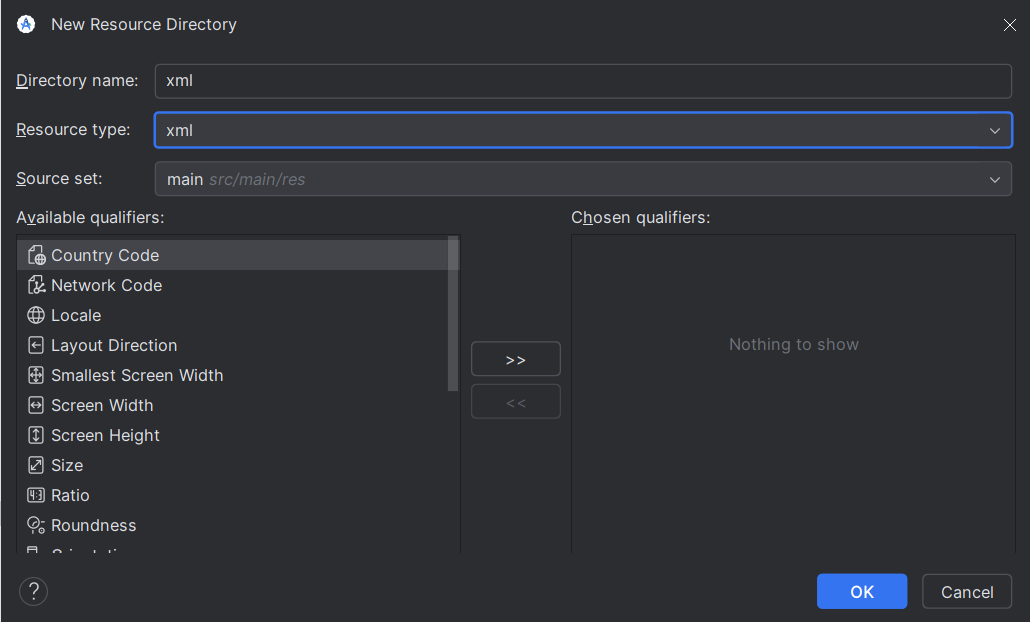


1. МЕХАНИЗМ РАЗРЕШЕНИЙ
2. ЗАДАНИЕ. КАМЕРА

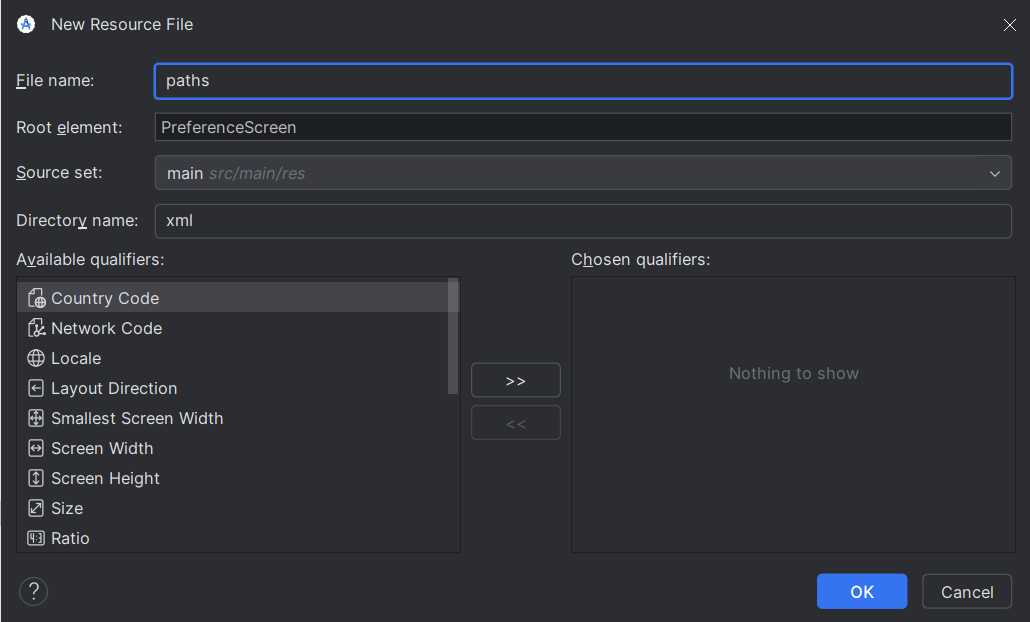
Создаём новый модуль.

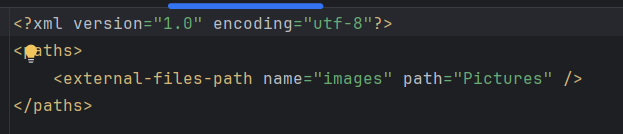


Создаём папку с ресурсами

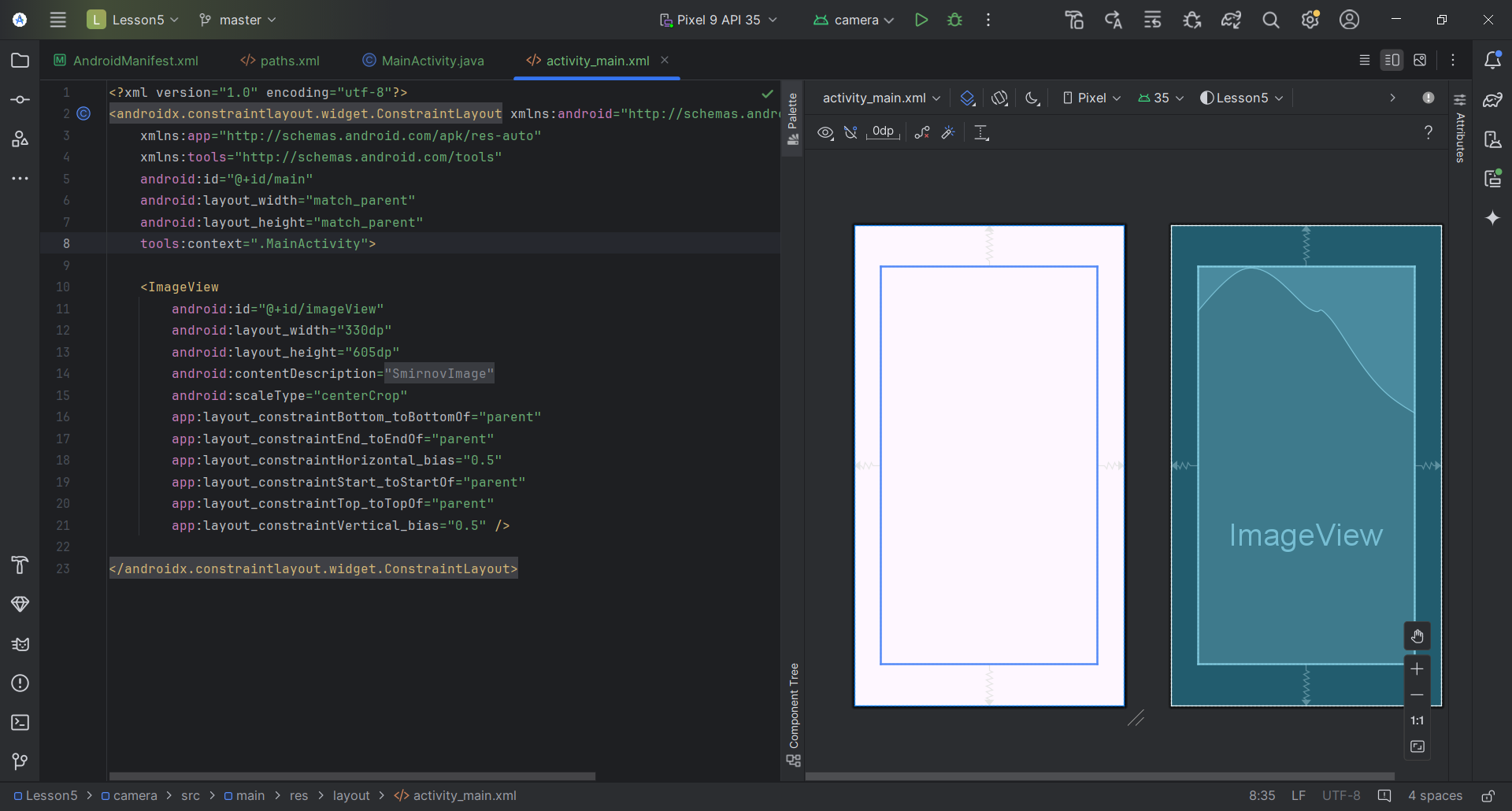


Создаём файл paths.xml

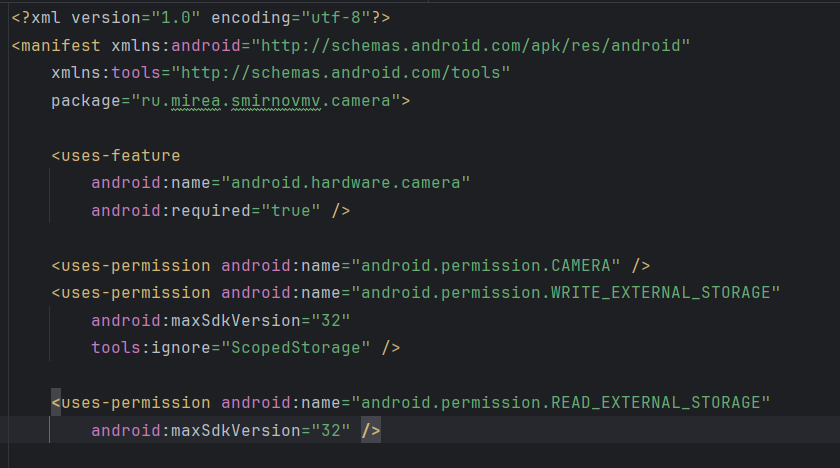


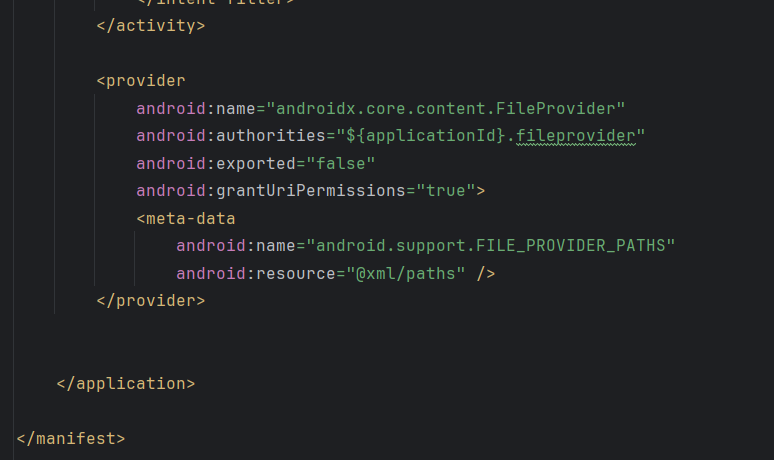


Добавляем imageView в activity\_main.xml

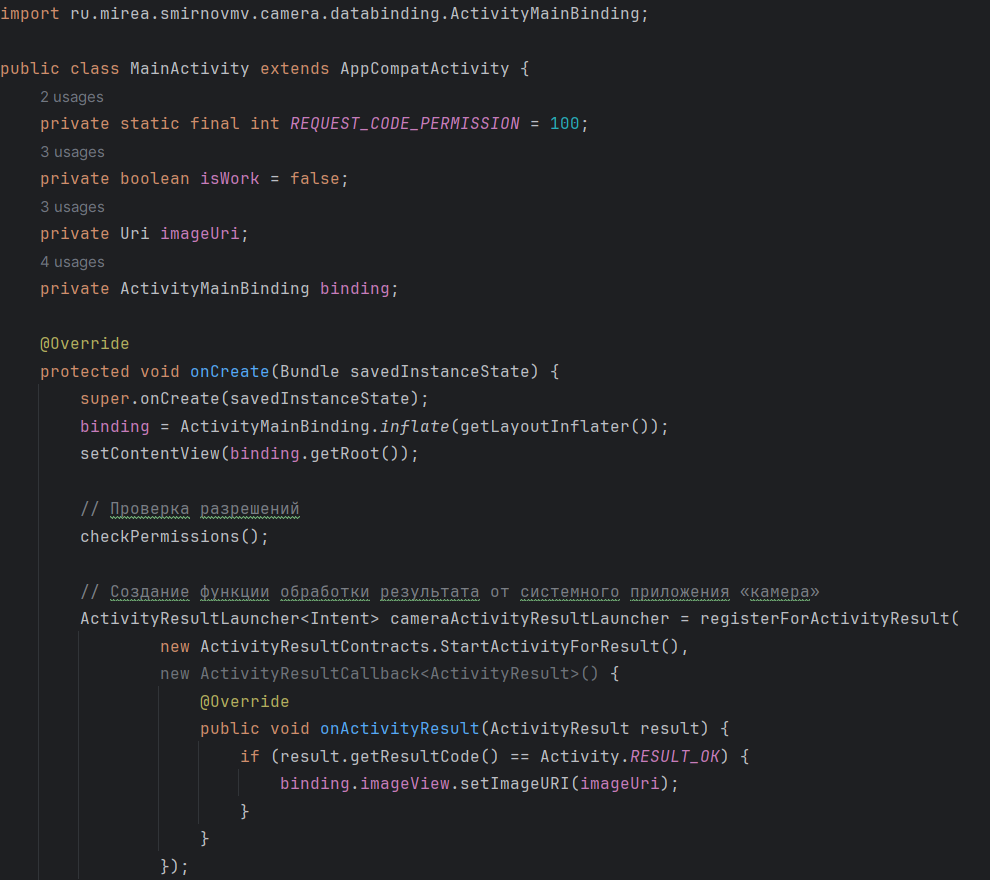


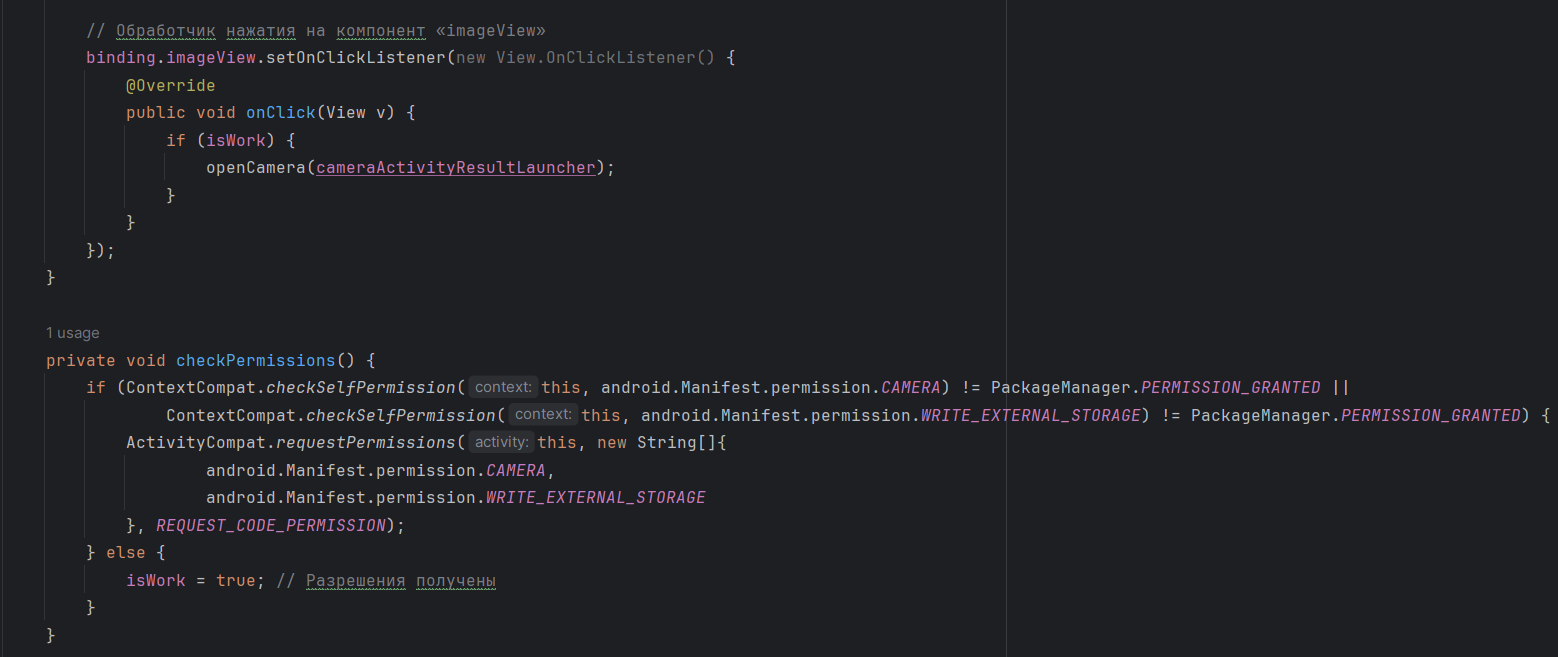
Проставим разрешения в Manifest

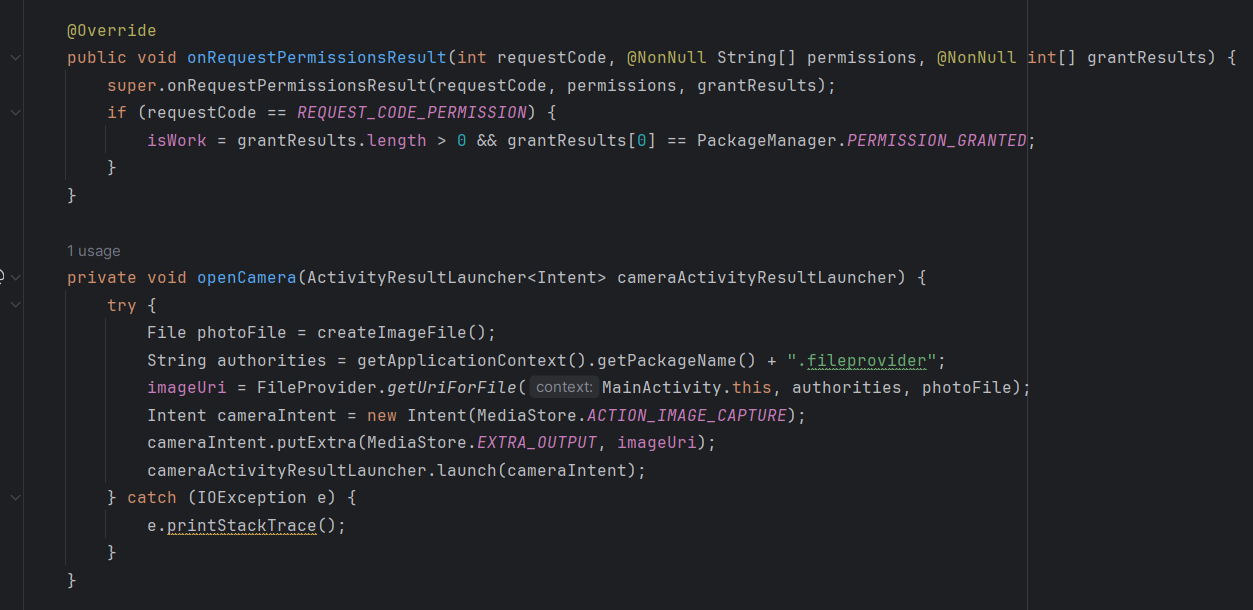




И конечно же пропишем MainActivity file



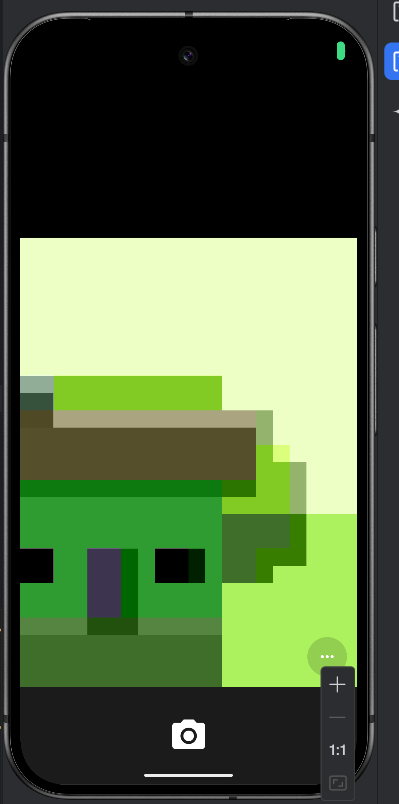




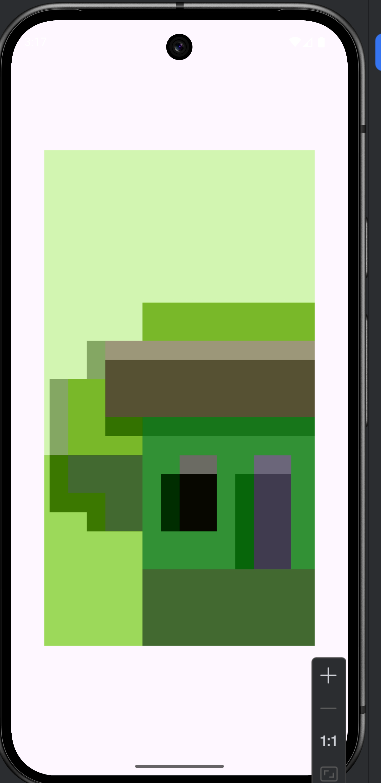


Наконец-то продемонстрирую результат

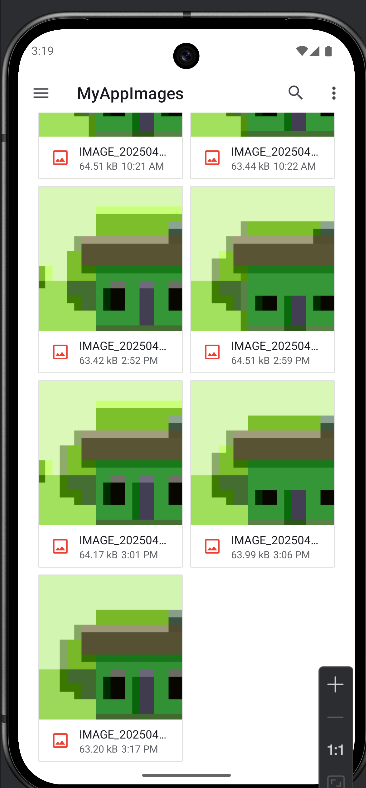
Нас встречает белый экран, но если на него нажать откроется камера, где будет видно моё лицо после всех практик



Нажимаем на знак фотоаппарата. Нам предалагает сохранить. После этого мы видим изображение на экране.

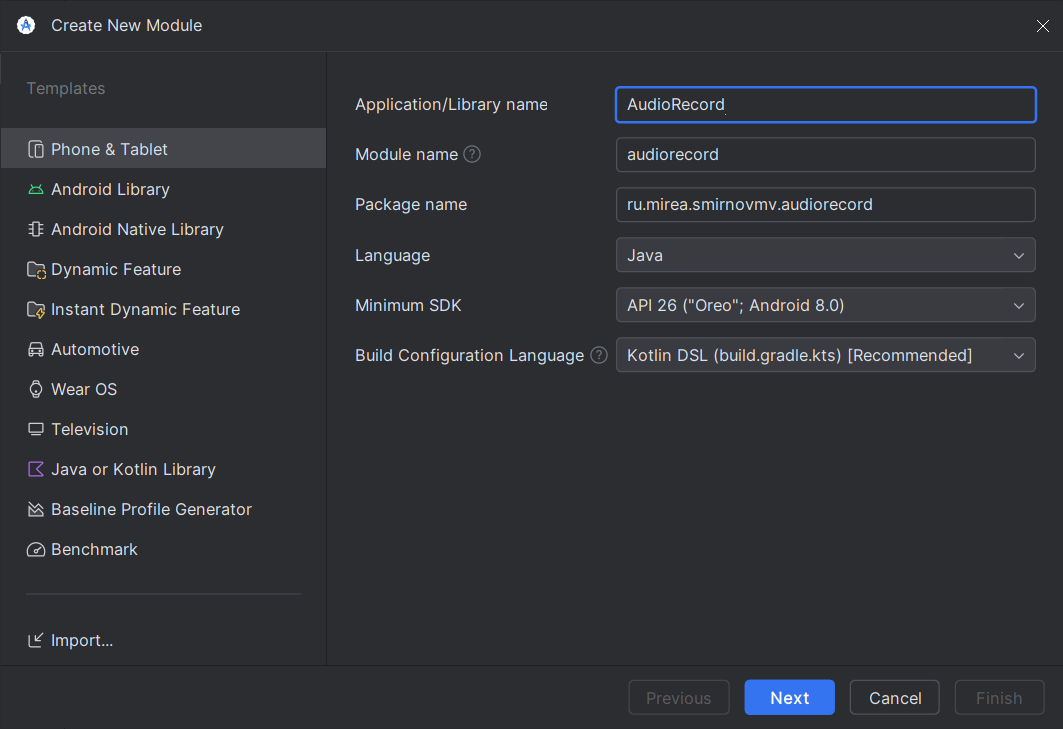


Видим все мои сохранённые снимки, потому что я долго не мог найти, куда он сохраняет. Искал в папке с проектом, но всё оказалось проще



1. МИКРОФОН. MEDIARECORDER

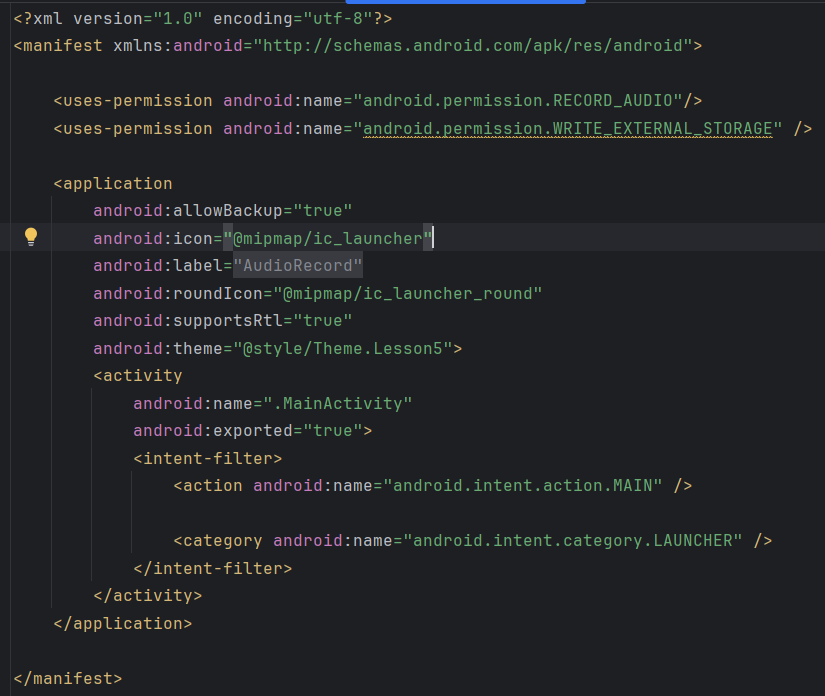
Создадим новый модуль



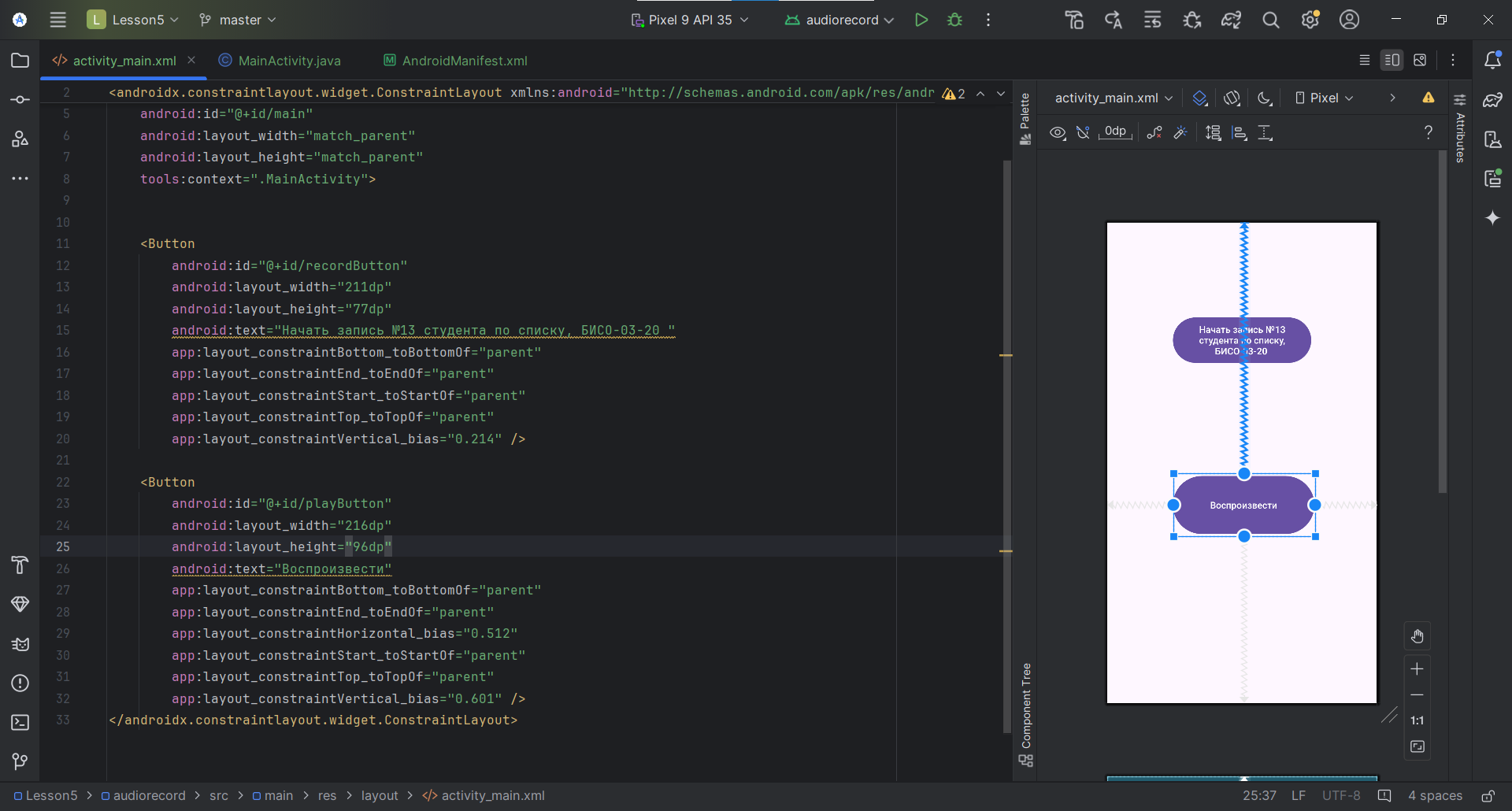
Что-то я подвис когда задание прочитал



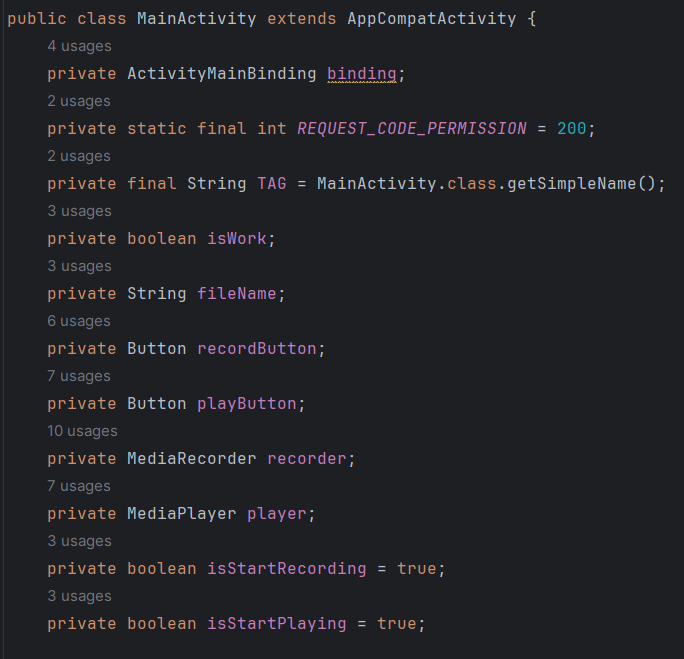
Сначала выдаём разрешения



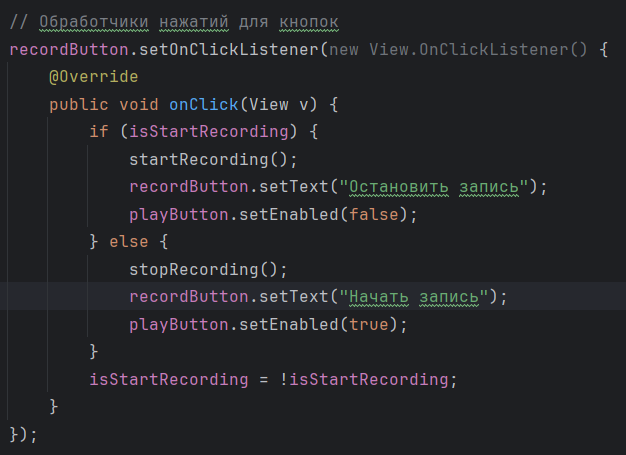
Затем создаём 2 кнопки



После этого выполняем все шаги из методички: проверяем разрешения, инициализируем кнопки, обрабатываем нажатия









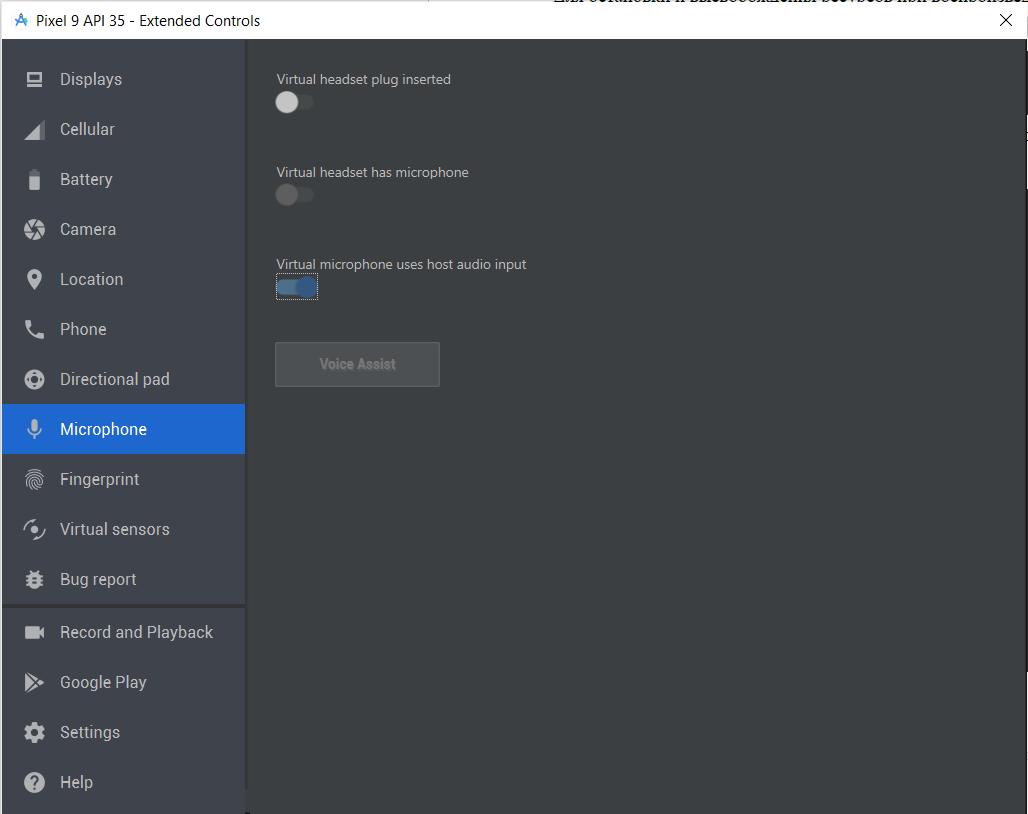




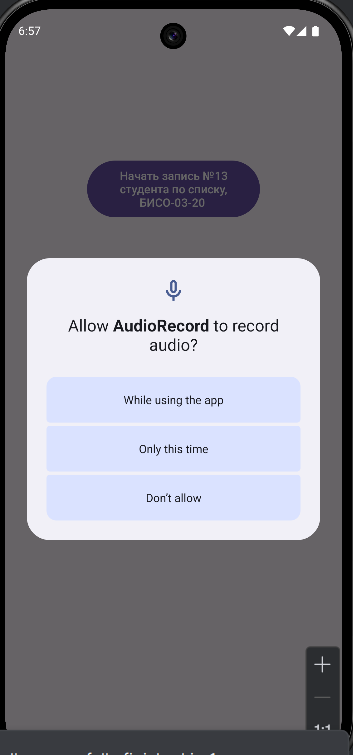
Чтобы записывать и прослушивать, используя системный микрофон, необходимо проставить разрешения

На 3 точки





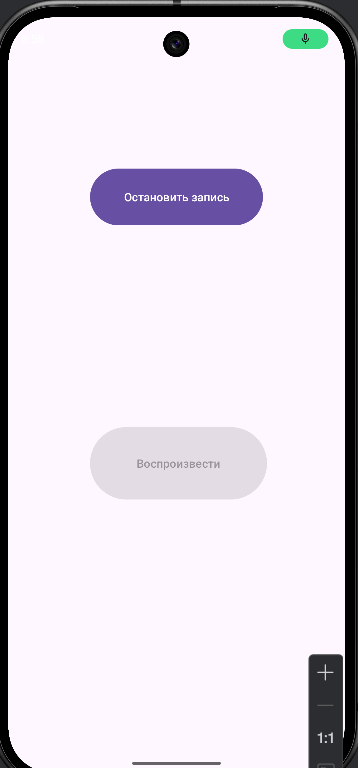
После этого при запуске эмулятора у нас попросят разрешение на доступ к микрофону



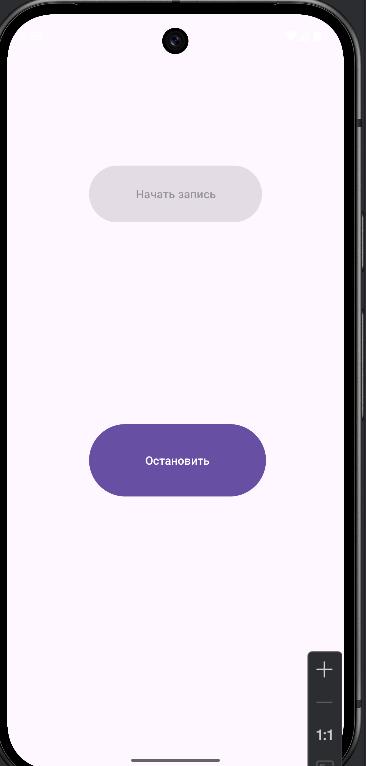
Мы видим наше окно с 2 кнопками



При нажатии на 1 у нас начнётся запись



Можно её остановить и нажать 2 кнопку

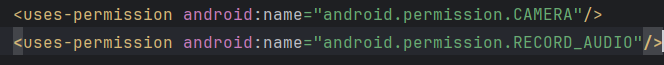


1. КОНТРОЛЬНОЕ ЗАДАНИЕ

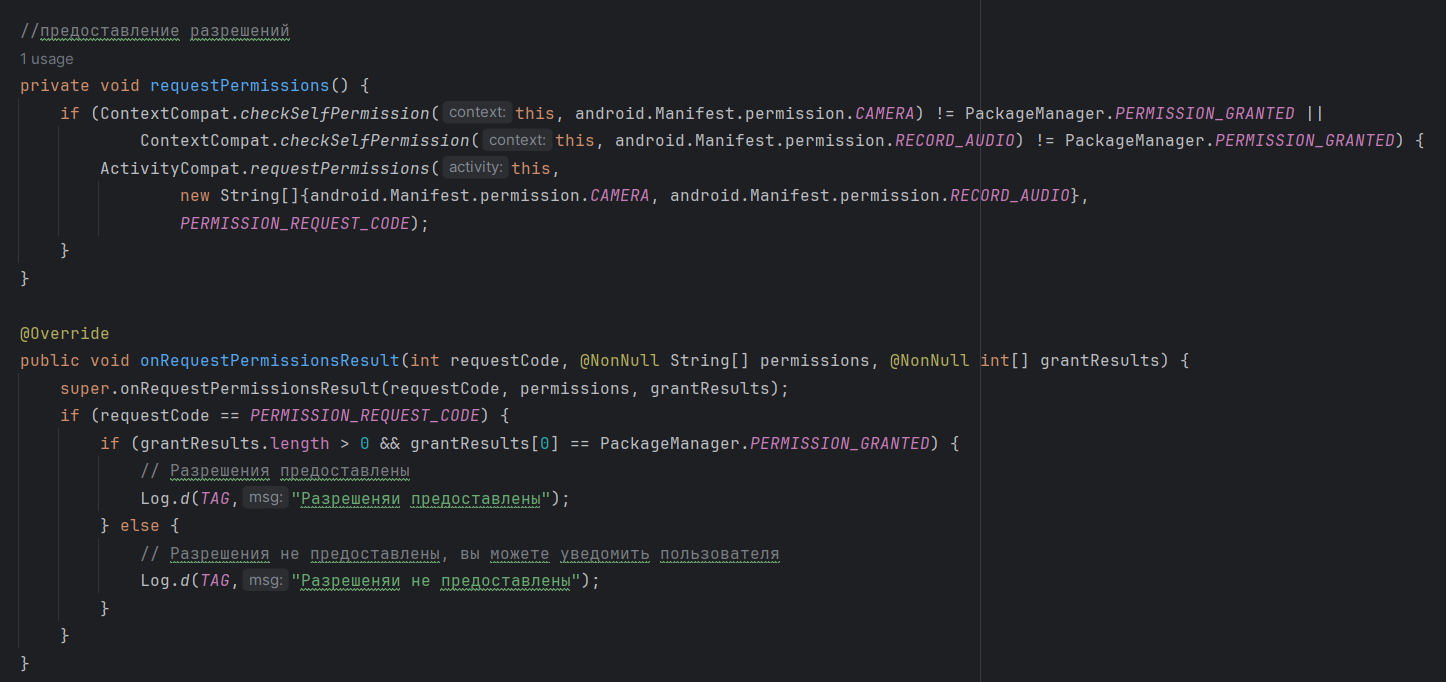
Вот это задание конечно

Для начала добавим в приложение механизм разрешений

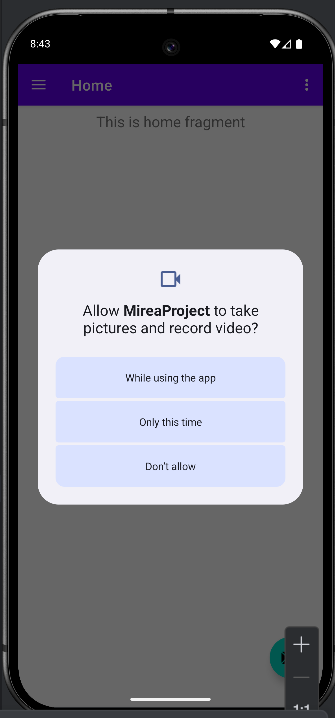
В manifest файле добавим разрешения



Добавим в MainActivity.java механизм обработки разрешений



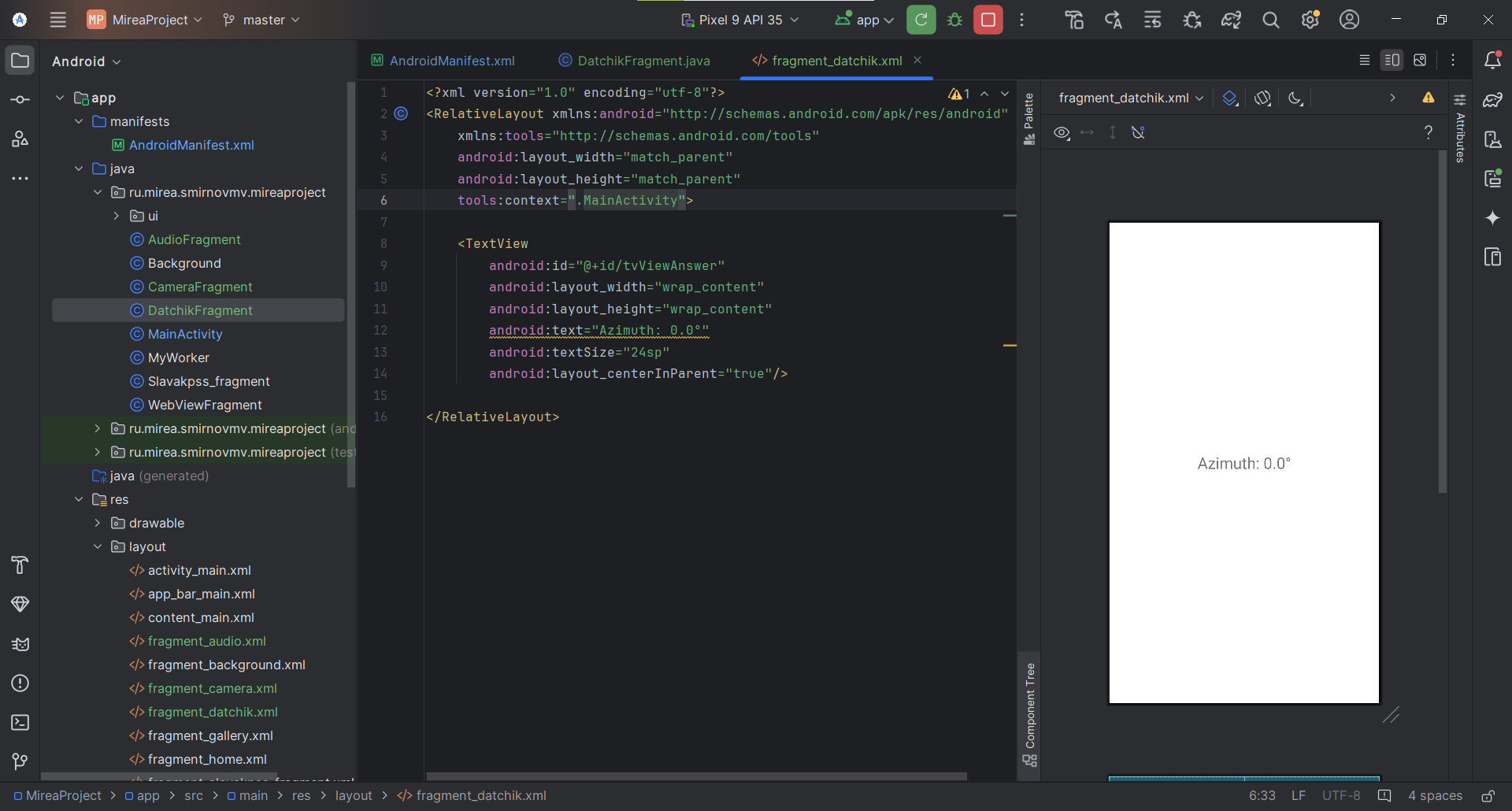
Результат:



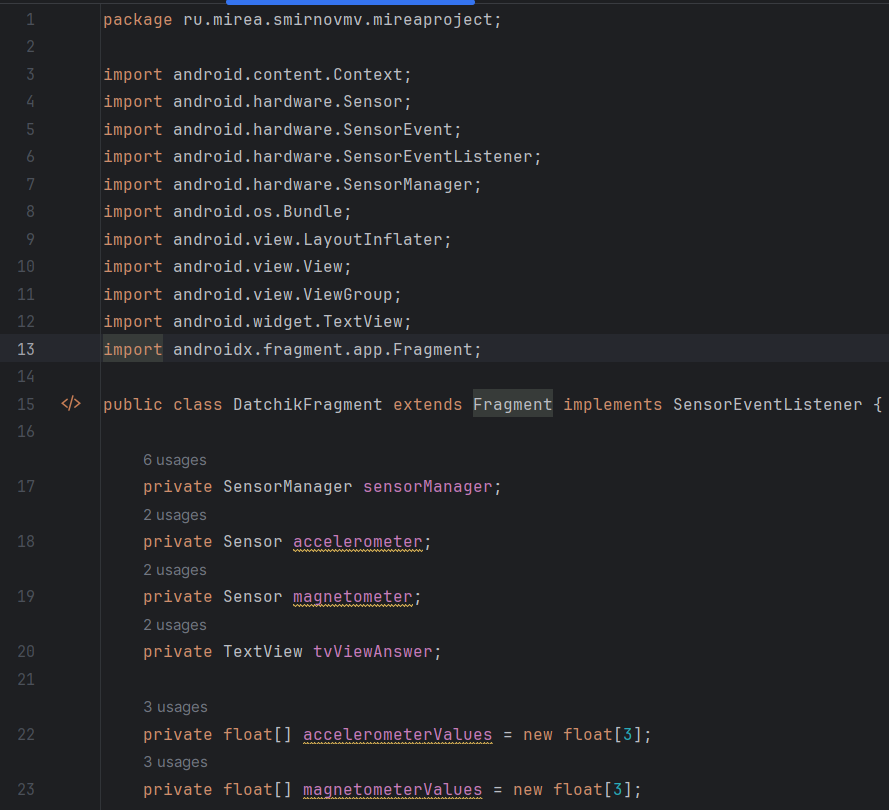
Далее ставим задачу: вычисление азимута\* угла с использованием датчиков акселерометра и магнитометра

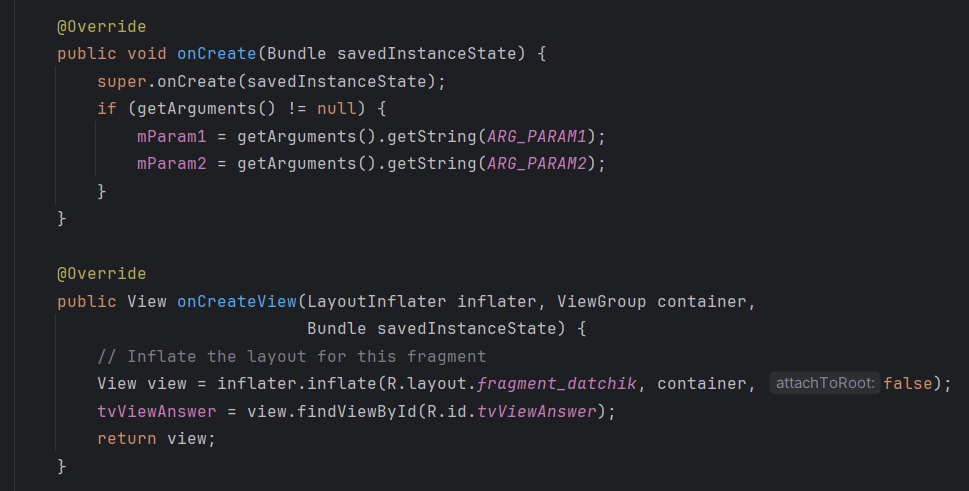
\*Азимут – это угол относительно магнитного севера

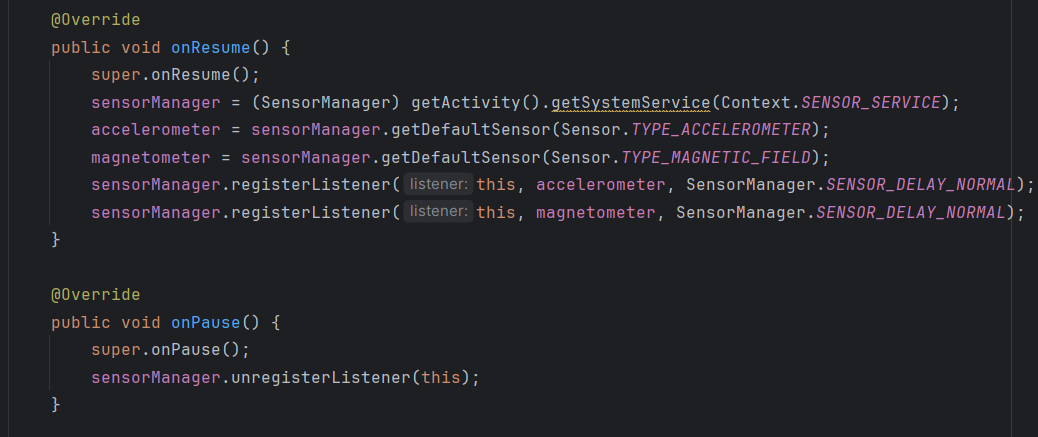
Для этого создаём экран с текстовым полем

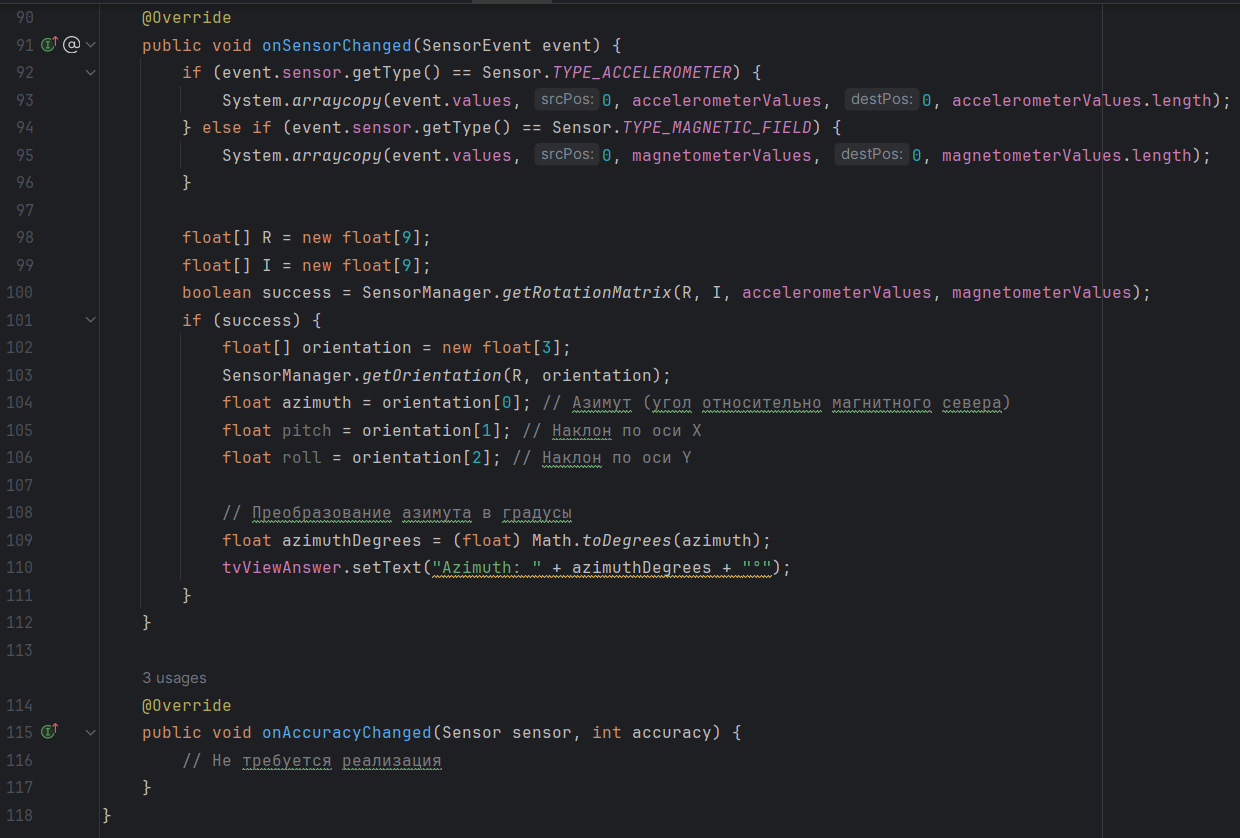


Пишем функцию, которая считывает показания датчиков и выводит на экран показатели азимута



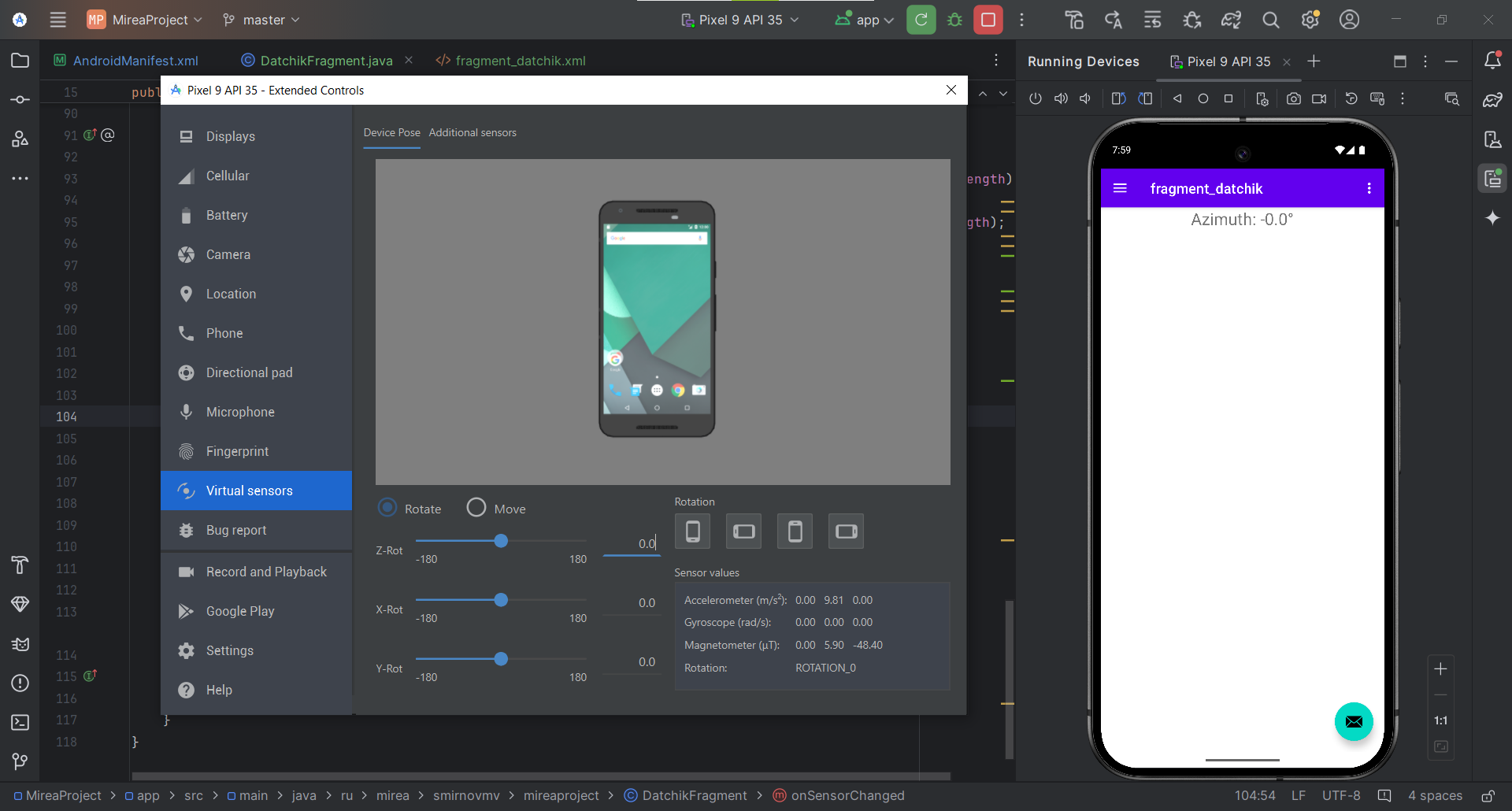




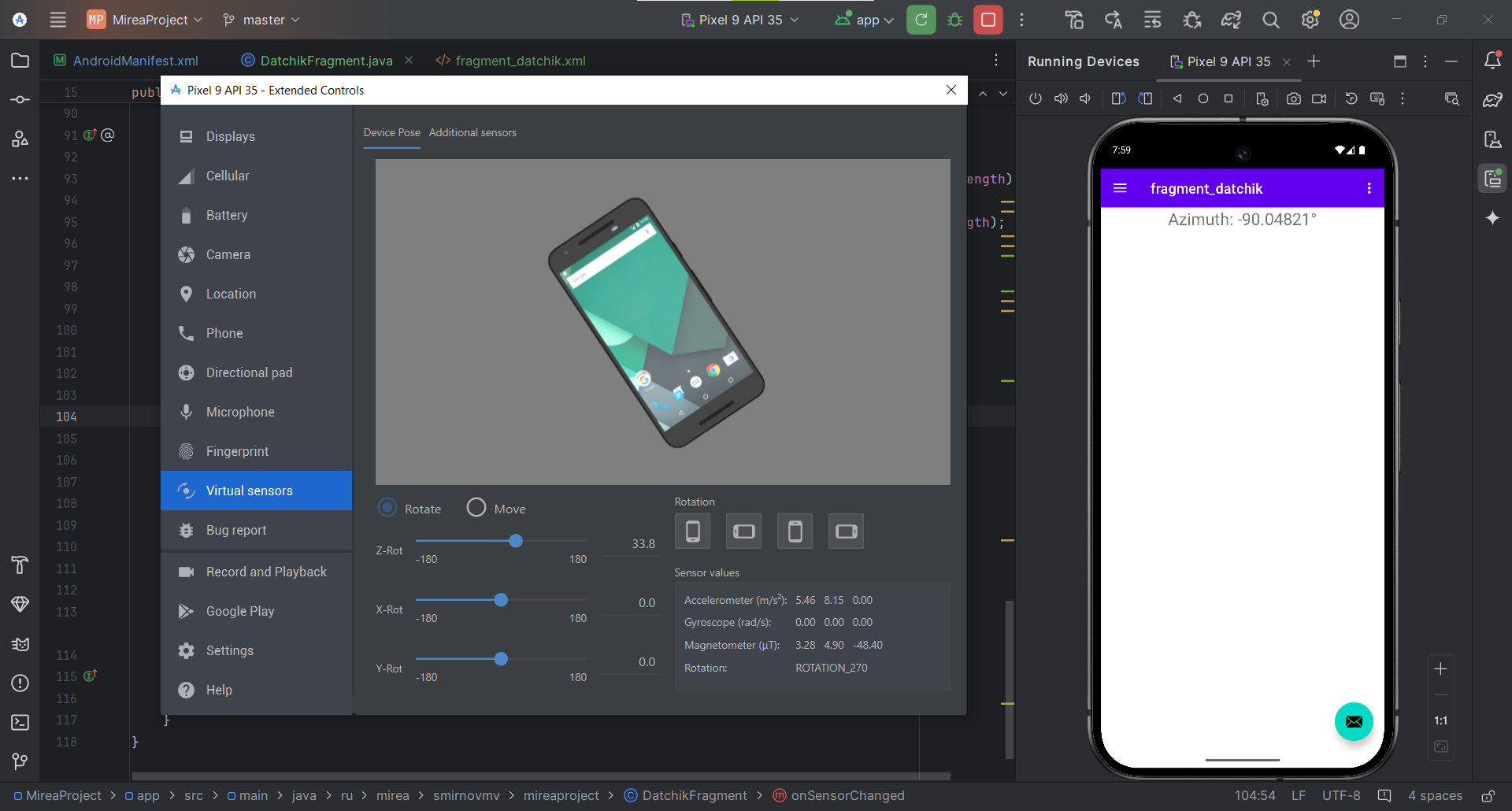


Результат:

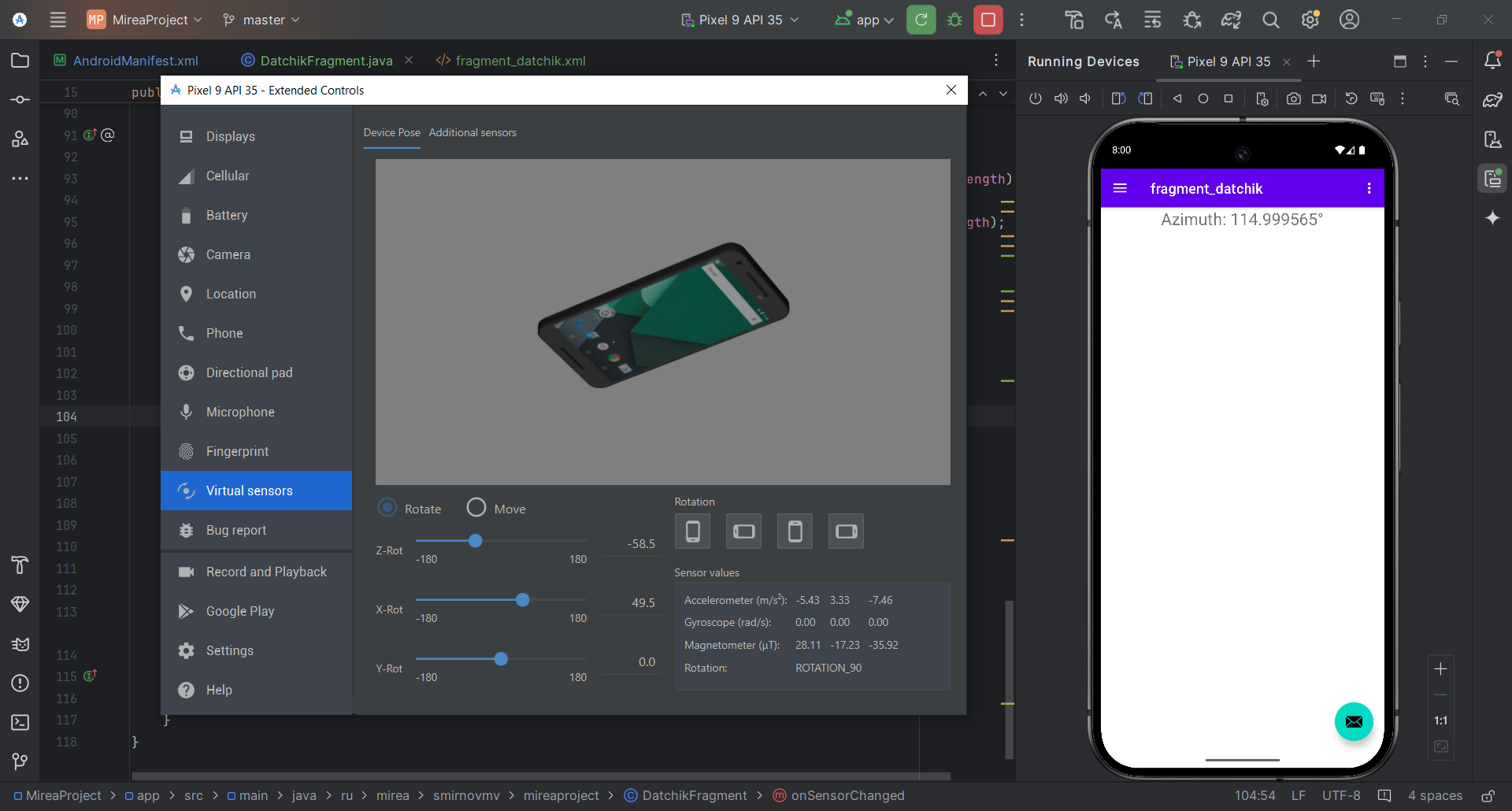
В спокойном положении



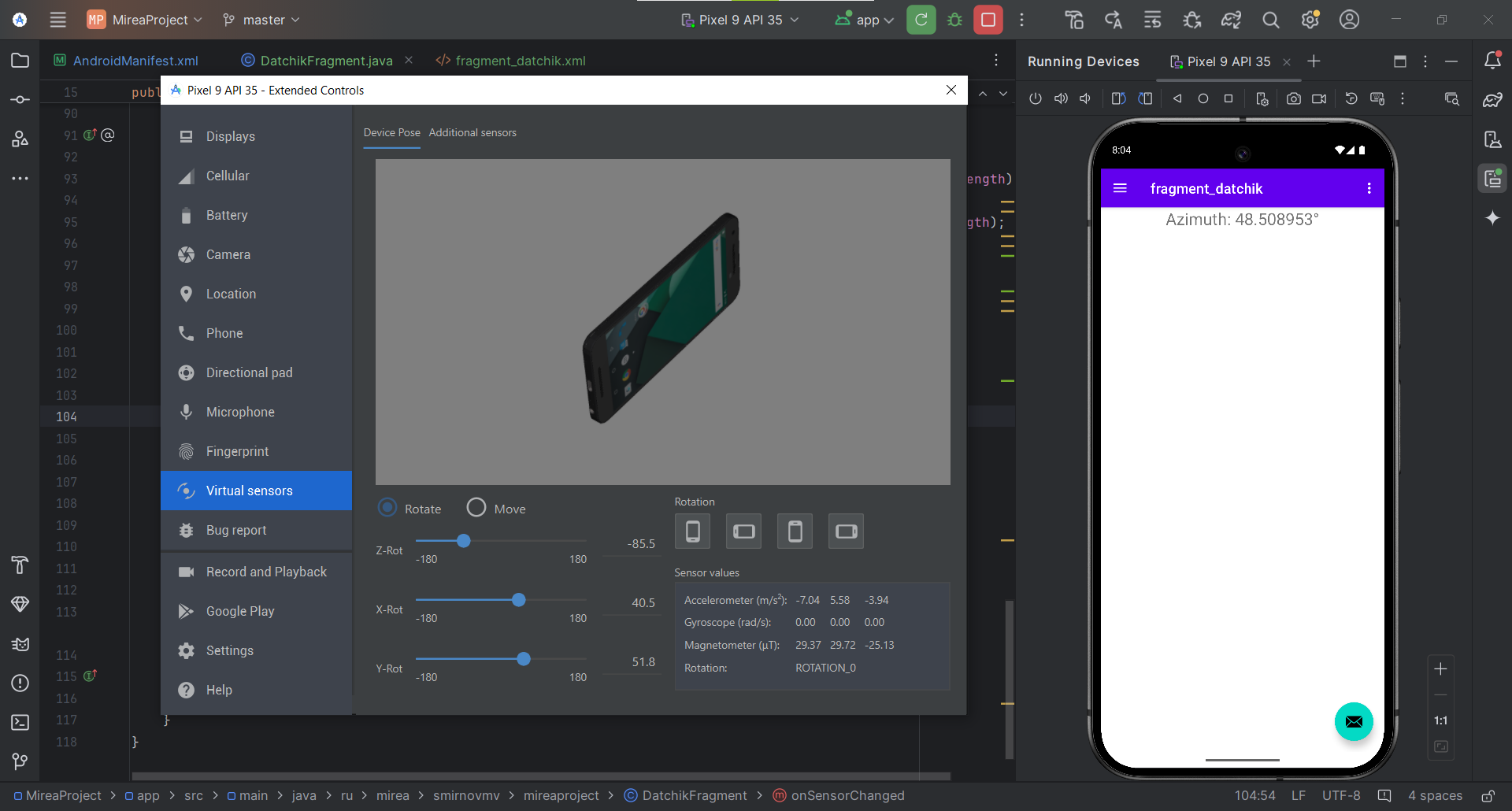
При полороте z оси



При повороте x оси

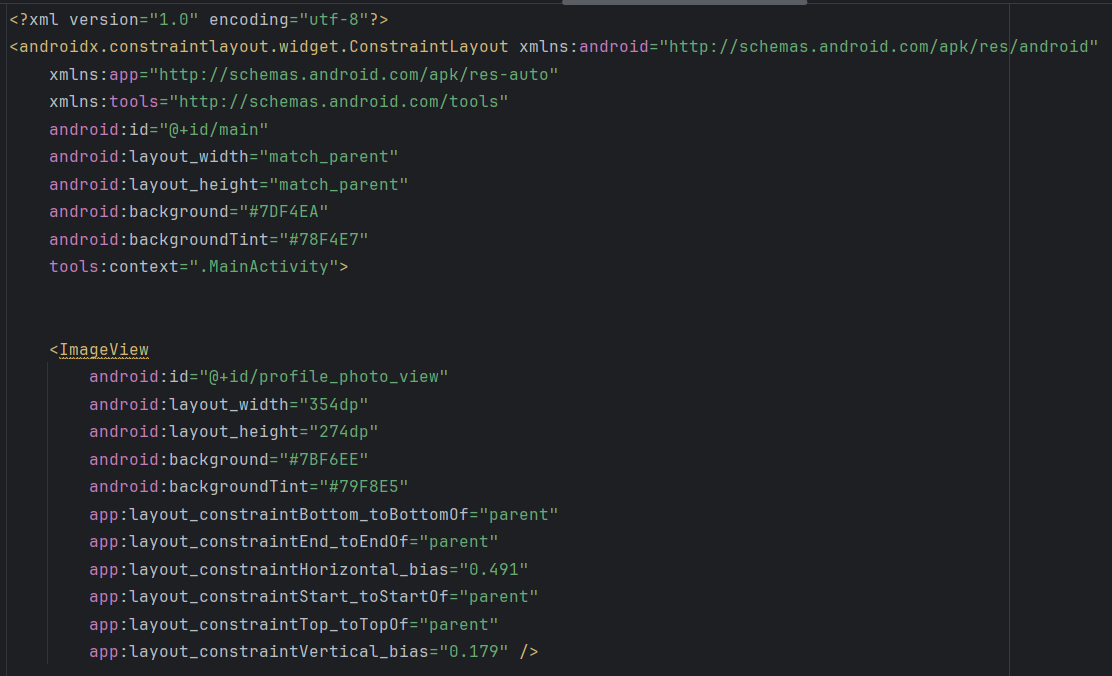


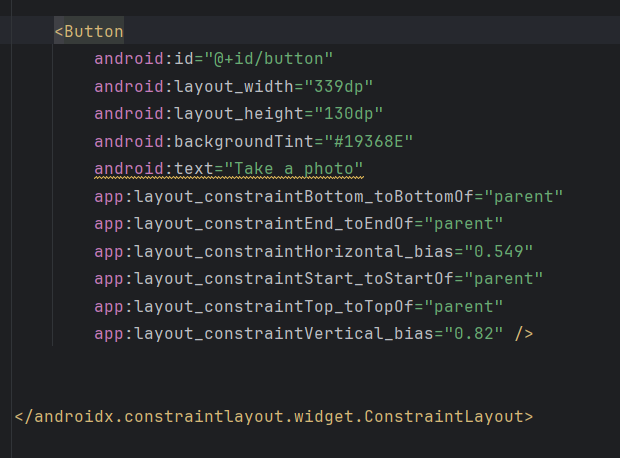
При повороте y оси



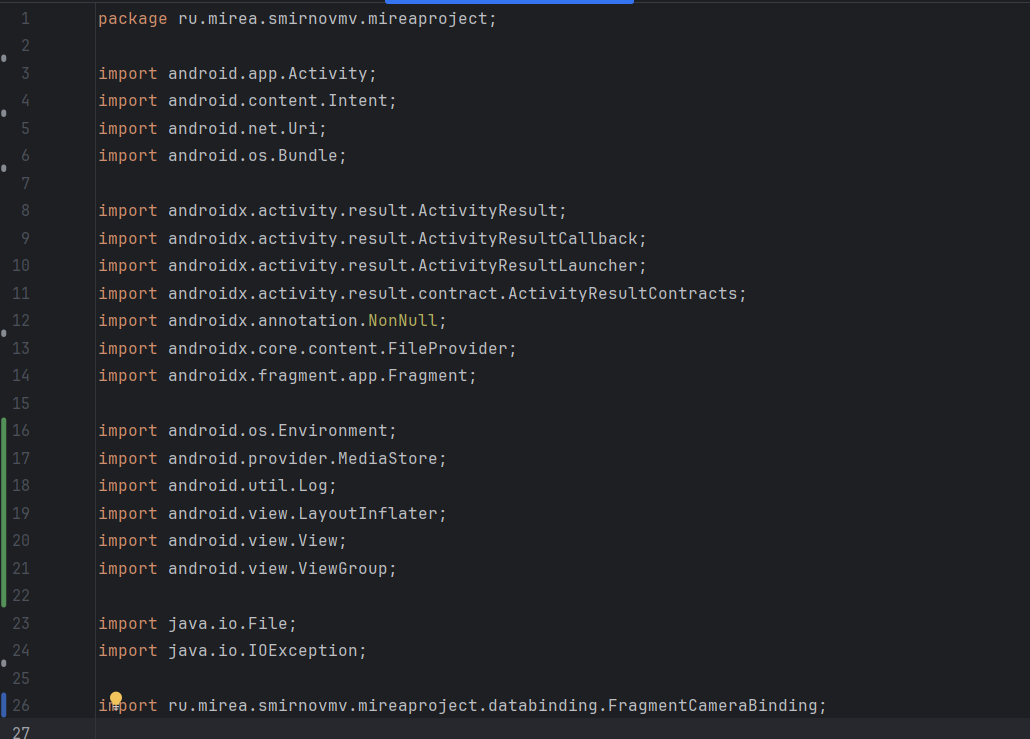
Далее интегрируем приложение камера в наш проект

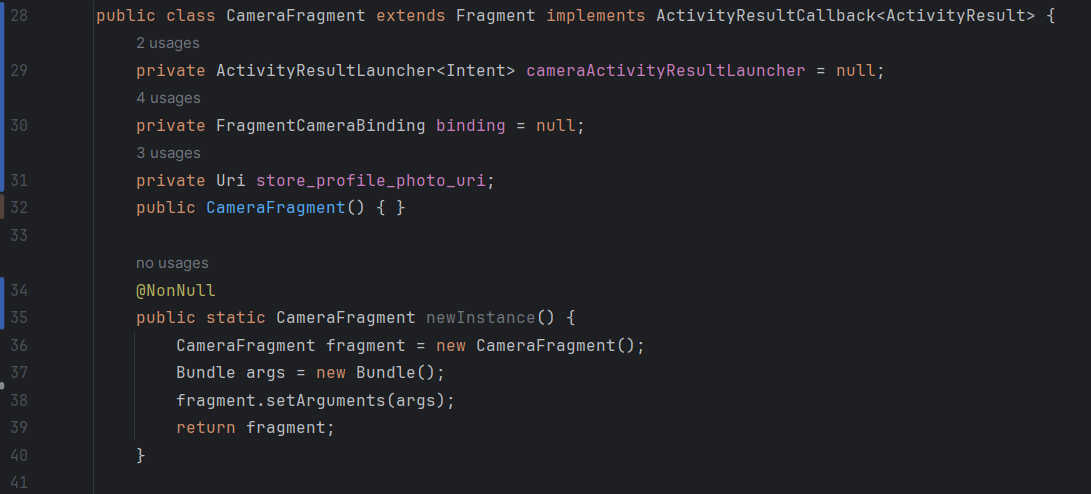
Создадим layout file

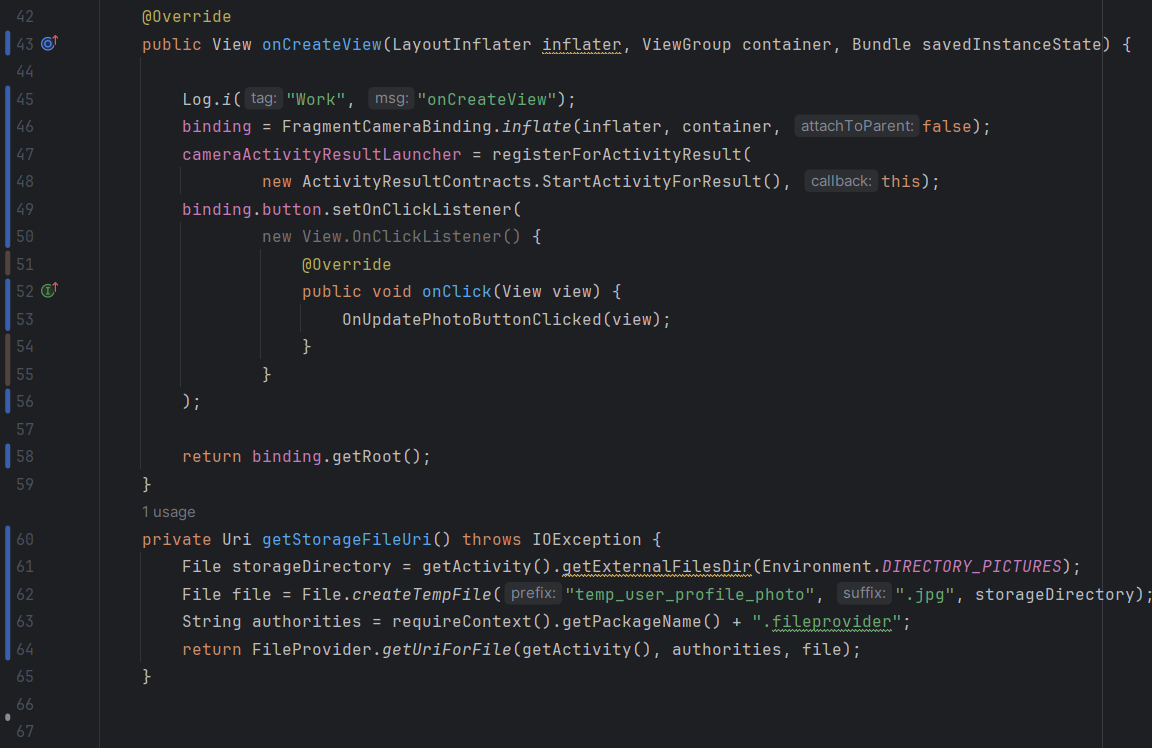




Также пропишем CameraActivity.java для того, чтобы прописать код того, как будет работать наша активность

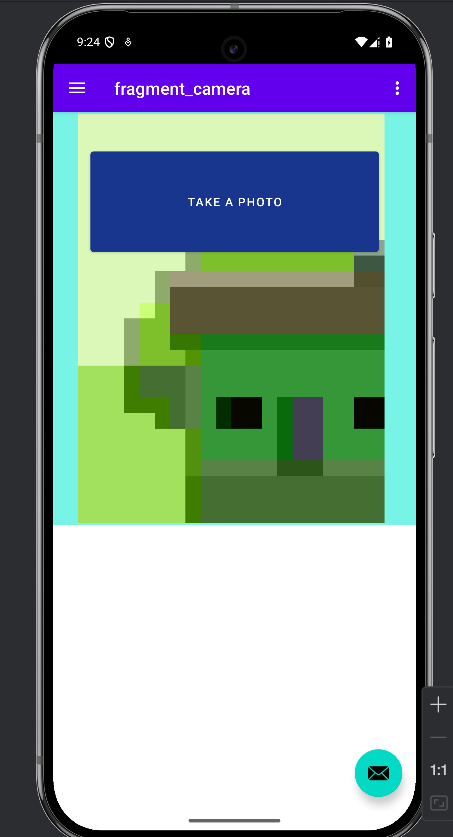






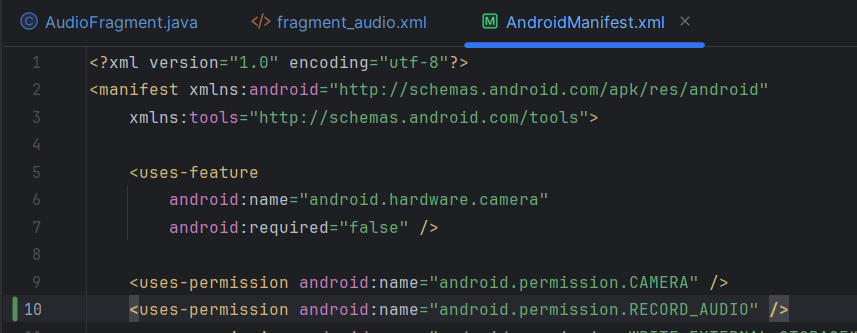


Результат: почему-то съезжает всё вверх экрана

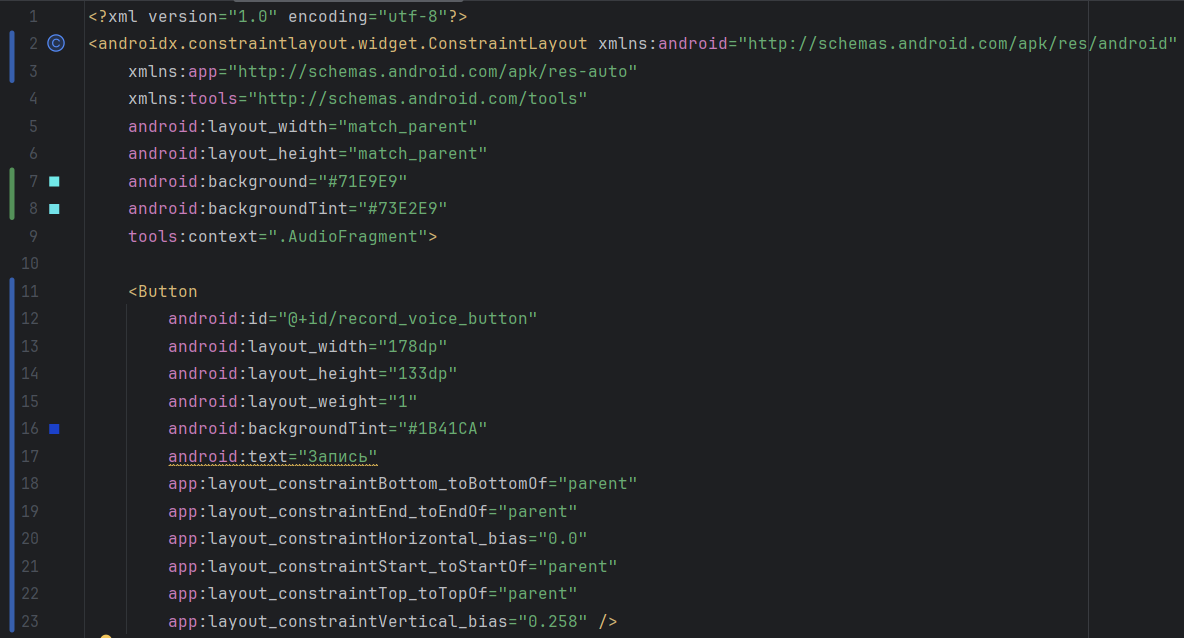


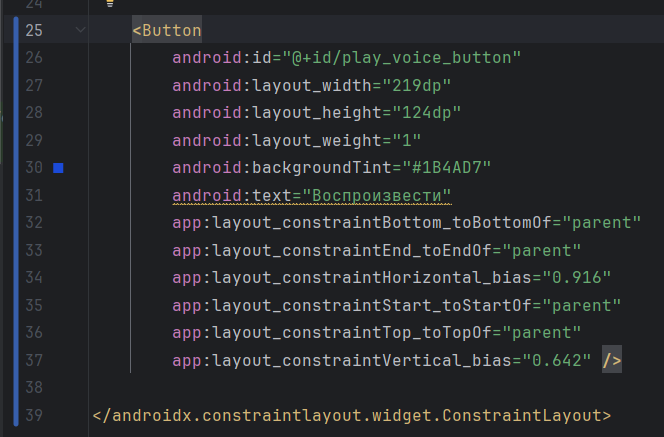
Далее создадим экран с функционалом микрофона

Пропишем разрешения

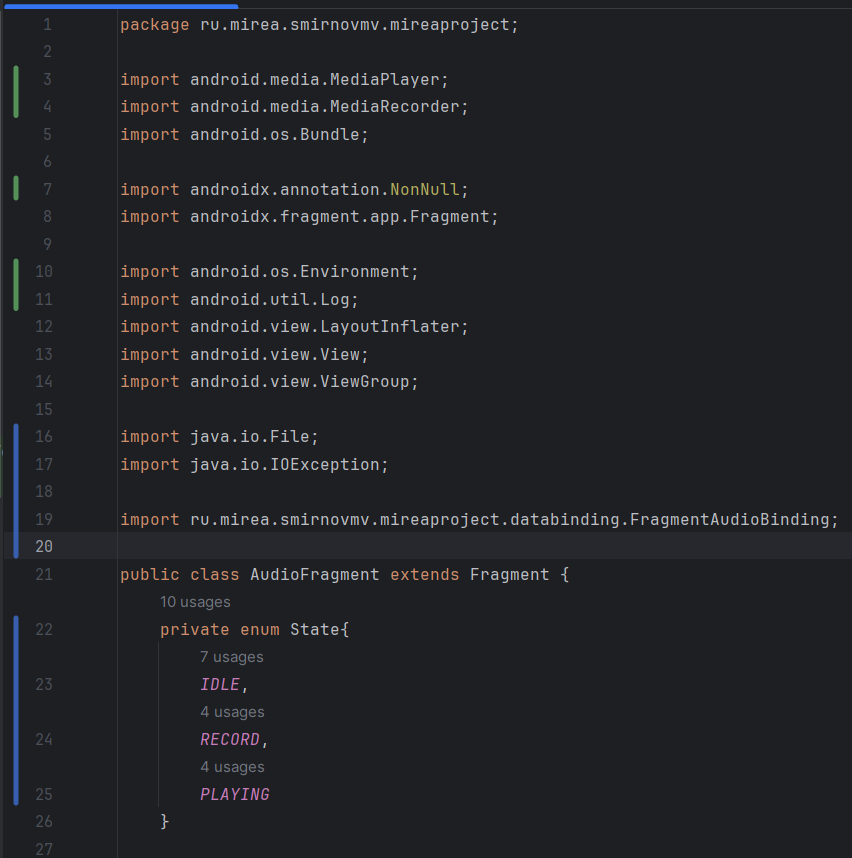


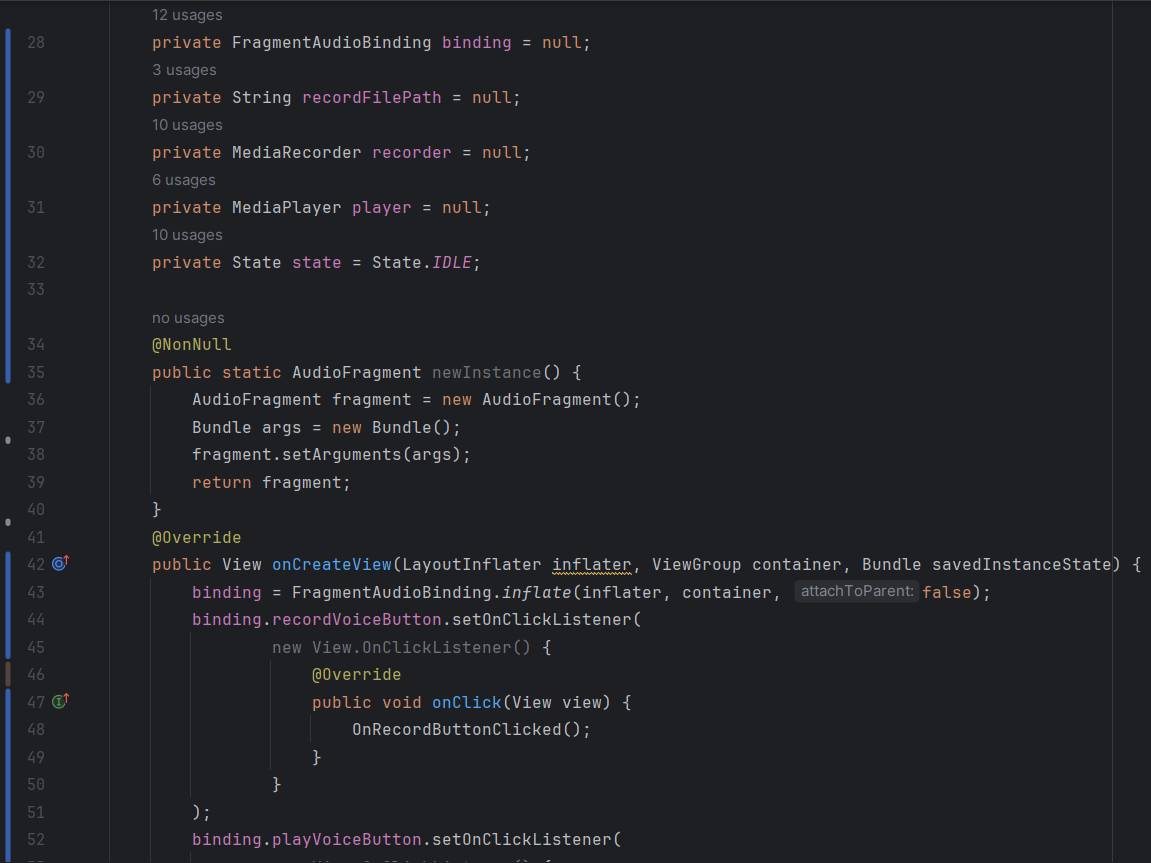
Создадим layout file





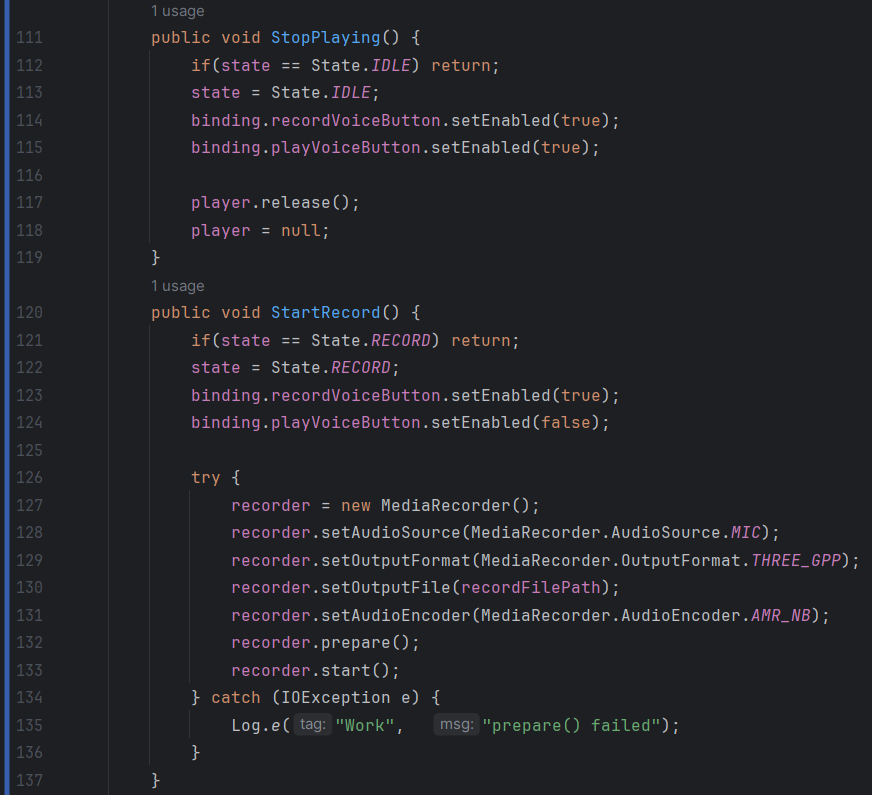
Создадим AudioFragment.java файл

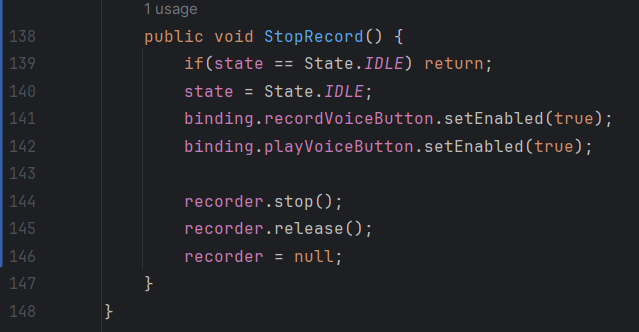












Результат:

