

Воробьев Илья Романович

бакалавр студент 3 курса

ПГУТИ

Россия, г. Самара

Морозов Дмитрий Денисович

бакалавр студент 3 курса

ПГУТИ

Россия, г. Самара

Секлетова Наталья Николаевна

к.п.н., доцент каф. «Информационные системы и технологии»

ПГУТИ

Россия, г. Самара

ПОГРУЖЕНИЕ В МОБИЛЬНУЮ РАЗРАБОТКУ

Аннотация: В статье подробно рассмотрен процесс создания приложения на языке программирования java. Предлагается рассматривать данную публикацию в качестве методического пособия для начинающих разработчиков мобильных приложений. Основной проблемой на ранних этапах освоения данной технологии является недостаточность тезисного пошагового материала, позволяющего приобрести практические навыки разработки. Решению данной проблемы и посвящена данная работа.

Ключевые слова: Разработка, элемент, список, файл, активити.

Annotation: The article describes in detail the process of creating an application in the java programming language. It is proposed to consider this publication as a methodological guide for beginner mobile application developers.

The main problem in the early stages of mastering this technology is the lack of thesis step-by-step material that allows you to acquire practical development skills. This work is devoted to the solution of this problem.

Keywords: Development, element, list, file, activity.

Введение

Разработка мобильных приложений – это процесс, при котором приложения разрабатываются для небольших портативных устройств, таких, как КПК, смартфоны или сотовые телефоны.

Теперь рассмотрим основную информацию, мобильные приложения пишутся на таких языках как: Java, Kotlin, Swift, C#. Мы будем писать приложение под андроид, поэтому язык Swift мы не рассматриваем, так как он нужен для разработки под IOS. В этой статье мы будем разрабатывать свое первое приложение на Java.

Первый шаг

-Необходимо скачать Android Studio (первая ссылка в интернете).

-После запуска Android Studio появляется окно с выбором активности (пока не заостряем внимание), выбираем пустое активности, как показано на рисунке 1.

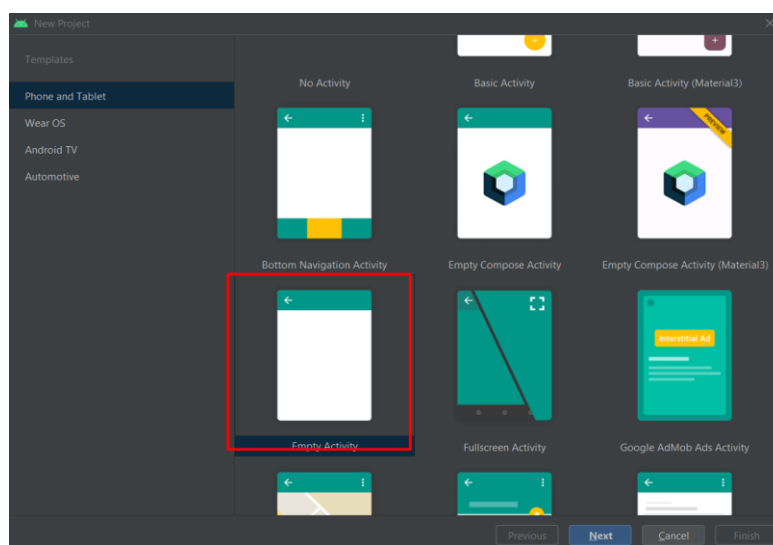


Рис. 1 (Пустое активности)

-Далее появится окно, где нужно назвать наш проект (1), указать, где будет находиться наш файл (2), выбрать язык, на котором будем писать (3), остальное оставить по дефолту, затем нажать на финиш.

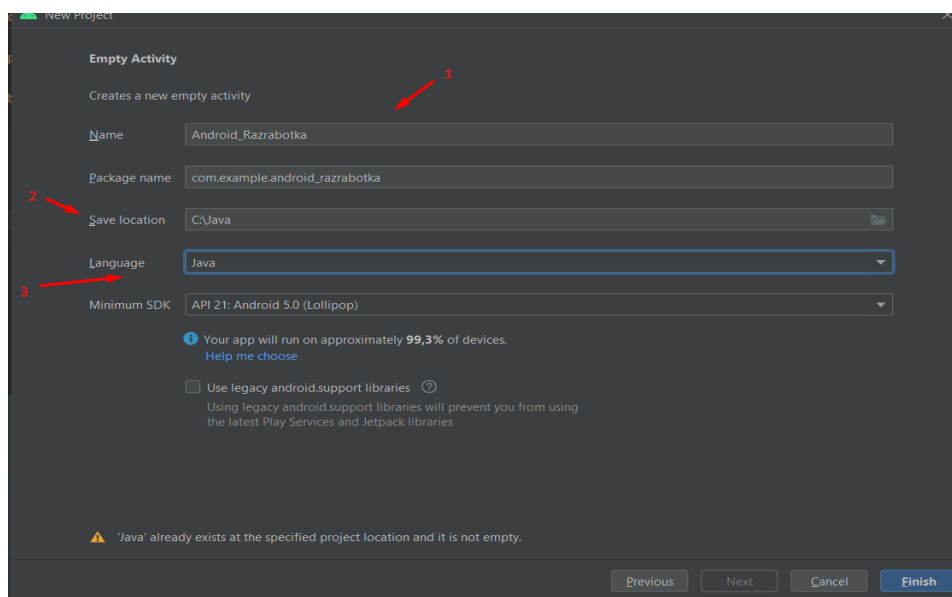


Рис. 2 (Создание проекта)

После полной загрузки проекта должно получиться так

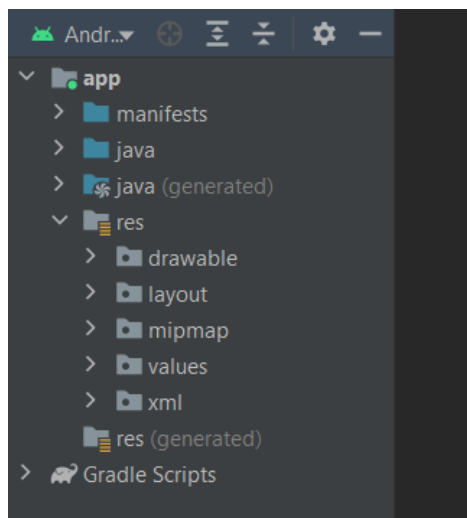


Рис. 3 (Начальное состояние)

Краткий рассказ о структуре. Полностью всё андроид приложение состоит из малых частей, которые собираются в один большой проект и являются самим приложением.

Выделим основные папки.

Manifests - внутри файлы конфигурации (малая инструкция), например там прописывается какое активити будет запускаться первым.

Java – основная папка, содержащая все активити (бек активити).

Res - в этой папке находятся все ресурсы.

Layout - папка в которой хранятся xml разметка активити (фронт активити).

Mipmap – в папке хранятся данные об иконке приложения

Values - папка значений, например: цвет

Drawable - содержит файлы png, jpg, jpeg.

Основы мобильной разработки

1) Активити

Что такое активити.

Активити – это фундаментальный элемент мобильного приложения, который представляет собой окно приложения.

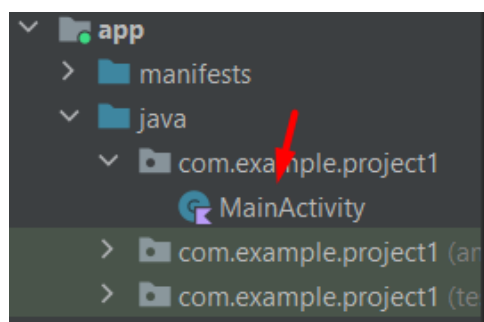


Рис. 4 (Активити)

Разберем на примере: мобильного приложения (ВКонтакте).

После того как приложение открылось, вы можете наблюдать первое активити, после перехода в сообщения вы видите уже другое активити, после перехода в список друзей вы видите ещё одно активити и так далее. Что представляет из себя активити надеемся вы поняли. В суть дела мы ещё вникнем на практике.

Как создать активити.

Для того чтобы создать активити нужно раскрыть папку с именем Java, далее раскрыть первую папку с именем com.example.project1, далее нажать на

неё правой кнопкой мыши и выполнить следующие действия показанные на рисунках.

Если все сделано правильно, то вуаля - вы создали активити.

По дефолту там уже создано одно активити, но для практики нам стоило попробовать.

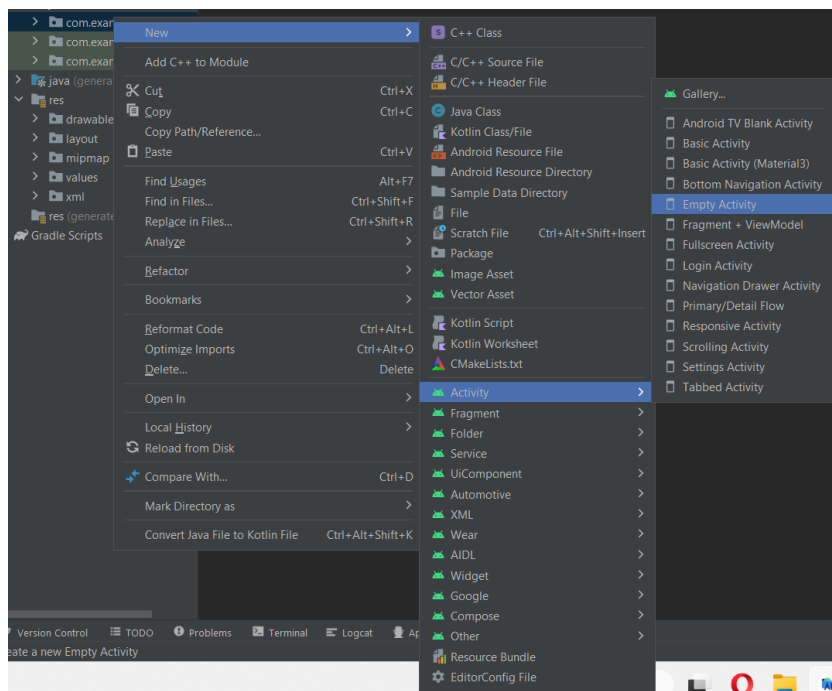


Рис. 5 (Создание активити)

2) Создание первого объекта xml разметки

Создание кнопки

Для того чтобы создать кнопку есть 2 варианта

Переходим в документ Res/layout/activity_main.xml, и видим следующее.

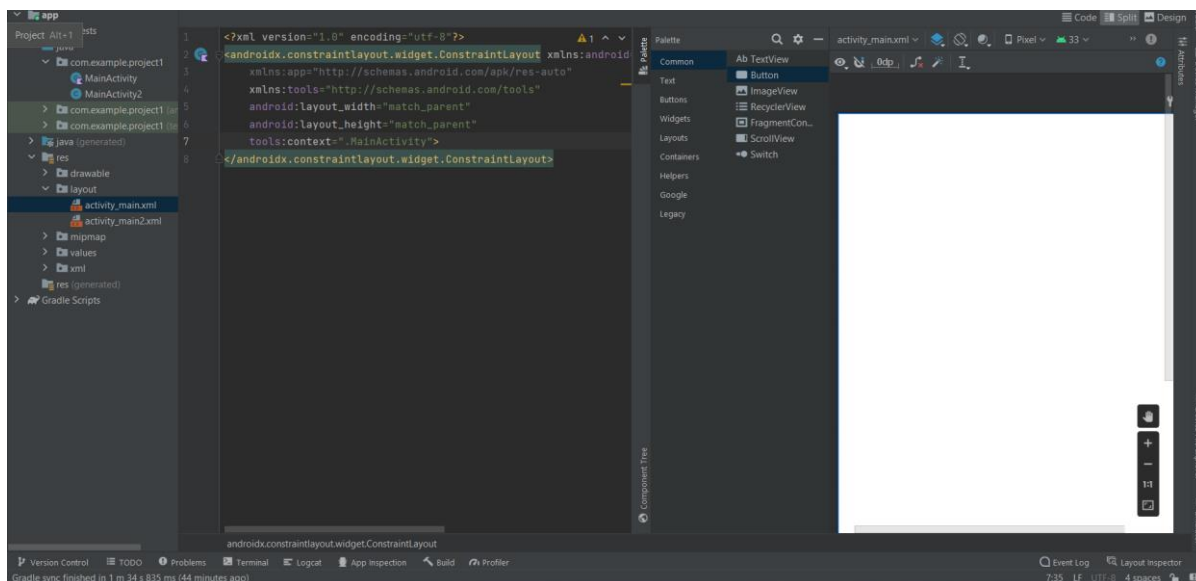


Рис. 6 (Создание кнопки)

Добавляем новый тег <Button>

(Код для копирования)

<Button

android:id="@+id/button"

android:layout_width="150dp"

android:layout_height="100dp

android:text="Button"

app:layout_constraintBottom_toBottomOf="parent"

app:layout_constraintEnd_toEndOf="parent"

app:layout_constraintHorizontal_bias="0.498"

app:layout_constraintStart_toStartOf="parent"

app:layout_constraintTop_toTopOf="parent"

app:layout_constraintVertical_bias="0.079" />

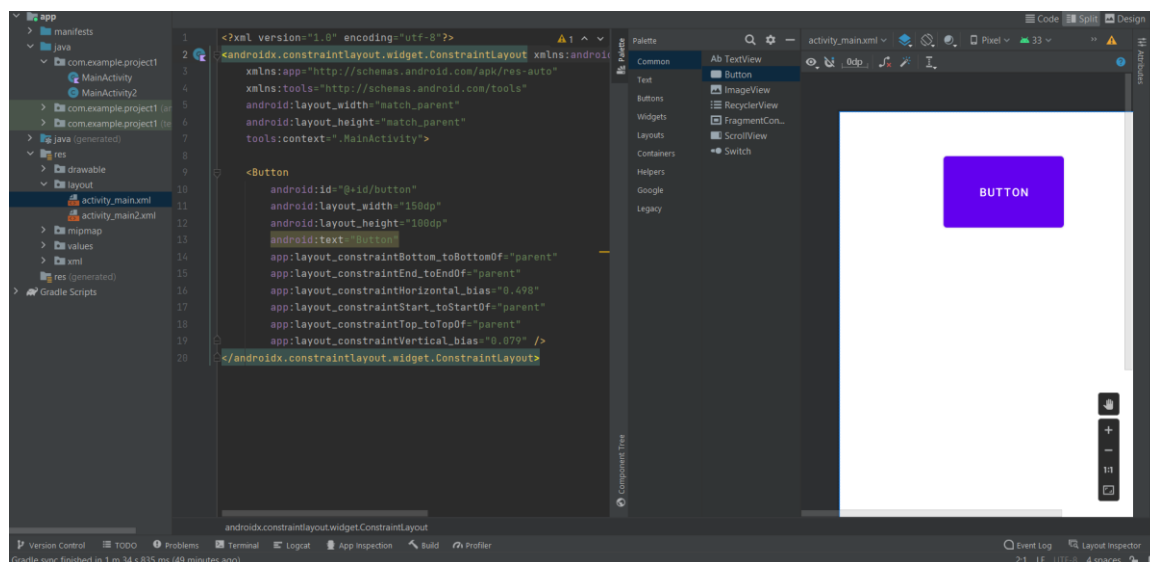


Рис. 7 (Получение кнопки)

Id – уникальный идентификатор

Layout_width – ширина

Layout_height – высота

Text – текст кнопки

Layout_constraint + Bottom,Top,Start,End – это привязка к границам экрана (она нужна чтобы объект был зафиксирован)

layout_constraint+Horizontal,Vertical_bias – атрибут отвечающий за расположение по горизонтали или вертикали.

- Создание текста

Для того чтобы создать текст нужно перейти в код файла activity_main.xml, так же как и в прошлом разделе.

(Код для вставки)

```
<TextView
```

```
android:id="@+id/textView"
```

```
android:layout_width="100dp"
```

```
android:layout_height="50dp"
```

```
android:text="Первые
```

шаги"

```
app:layout_constraintBottom_toBottomOf="parent"
```

```
app:layout_constraintEnd_toEndOf="parent"
```

```
app:layout_constraintStart_toStartOf="parent"
```

```
app:layout_constraintTop_toTopOf="parent"
```

```
app:layout_constraintVertical_bias="0.022" />
```

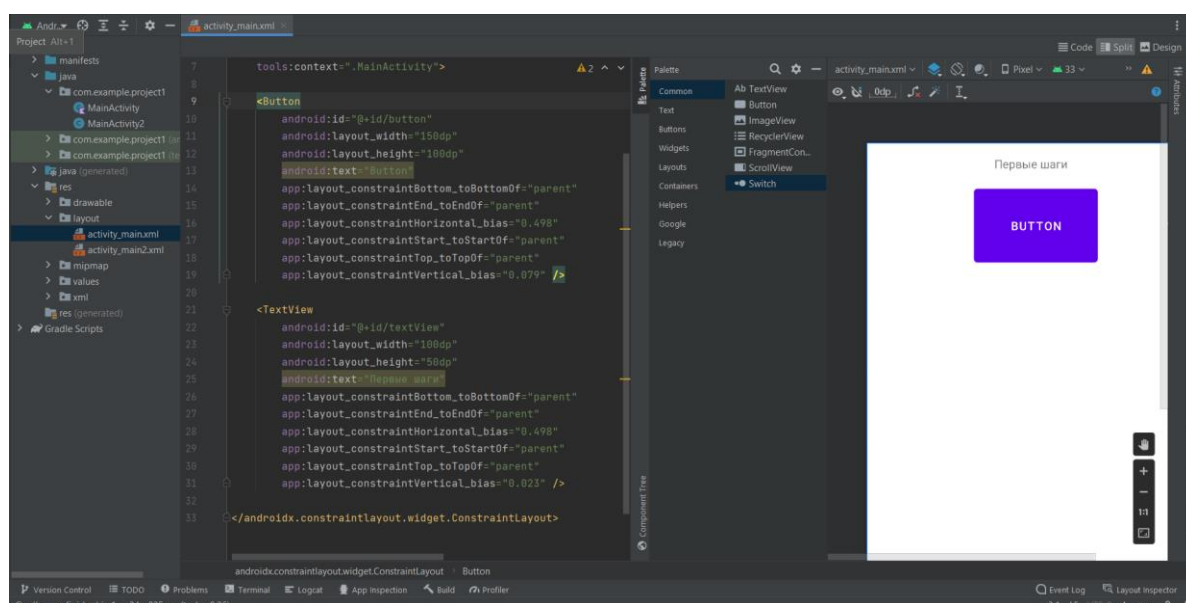


Рис. 8 (Создание текста)

3)Переход между активити

Далее мы сделаем переход на другое активити с помощью нажатия кнопки.

Для этого мы уже создали новое активити (смотреть выше).

Заходим в код Java/com.example.project1/MainActivity

Должно быть так

```
import android.content.Intent;
import android.os.Bundle;
import android.view.View;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }
}
```

Рис. 9 (Создание перехода на новое активити)

Вставляем код

(Код для копирования)

```
public void goToAnActivity2(View view) {
    Intent intent = new Intent(this, MainActivity2.class);
    startActivity(intent);
}
```

На экране мы видим следующее:


```

import android.content.Intent;
import android.os.Bundle;
import android.view.View;

public class MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);
    }

    public void goToAnActivity2(View view) {
        Intent intent = new Intent( packageContext: this, MainActivity2.class);
        startActivity(intent);
    }
}

```

Рис. 10 (Итоговый вариант)

а) Создаем класс onCreate в котором находится создание и отображение окна:

`super.onCreate(savedInstanceState);` - создает активность, настраивает статической активности интерфейса, основополагающая конструкция;

`savedInstanceState` – пакет Android содержащий данные для восстановления состояния активити;

`setContentView(R.layout.activity_main);` - это метод для отображения контента, в скобках указан путь на контент.

б) Создаем ещё один класс `goToActivity2` в котором создаем сам переход:

`Intent intent = new Intent(this, MainActivity2.class);` - это некое намерение (конструкцию нужно запомнить) которое необходимо для перехода на новую активность. В круглых скобках указываем первым контекстом из какого активити(в нашем случае `this`), вторым указываем куда будем переходить `MainActivity2.class`(новое активити);

`startActivity(intent);` - метод нужный для запуска активити (в круглых скобках объект на который ссылается метод).

4) Работа с изображениями

Для работы с изображениями существует папка drawable. В которой хранятся все наши картинки разных форматов.

- Чтобы добавить картинку в папку drawable можно просто перенести её в папку зажав её мышкой и перетащив в саму папку

Если нужен формат SVG, то делаем следующие действия.

После в открывшемся поле выбираем заранее заготовленную картинку SVG.

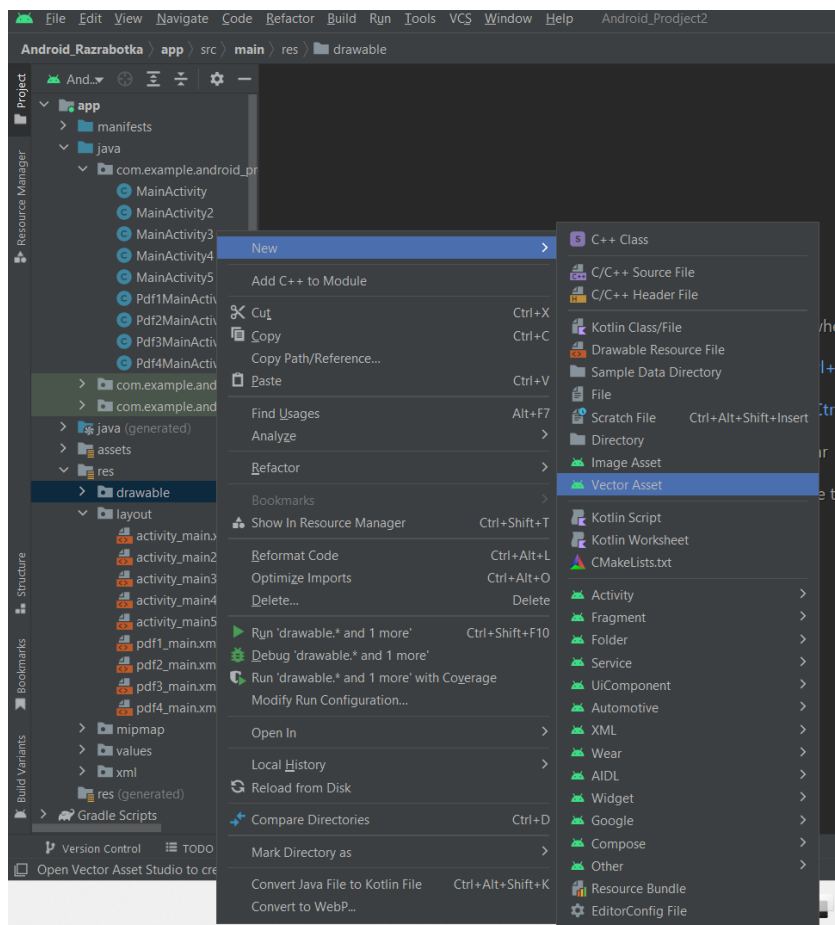


Рис. 11 (Выбор картинки)

-Открытие картинки по нажатию кнопки (Кнопку создавать мы уже научились выше)

Для реализации задуманного нам необходимо два активити (их создавать умеем). После создаем кнопку с добавлением к ней атрибута (onclick) и задаем имя метода, после переходим в активити где находилась наша кнопка и создаем метод с тем названием которое мы задали в атрибуте (onclick).Пишем следующий код для перехода на новое активити.

(Код для вставки)

```
public void goToAnActivity2(View view) {  
    Intent intent = new Intent(this, MainActivity2.class);  
    startActivity(intent);  
}
```

gotoAnActivity2 – название метода в атрибуте onclick.

Добавляем нашу картинку в папку drawable (название нашей картинки - android).

Далее переходим в xml файл второго активити, зажимаем опцию ImageView и перетаскиваем в наше окно. Нам выдаст меню где можно выбрать картинку, выбираем свою и нажимаем ок.

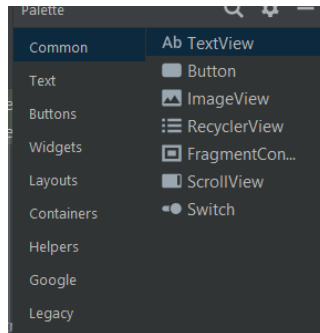


Рис. 12 (Результат)

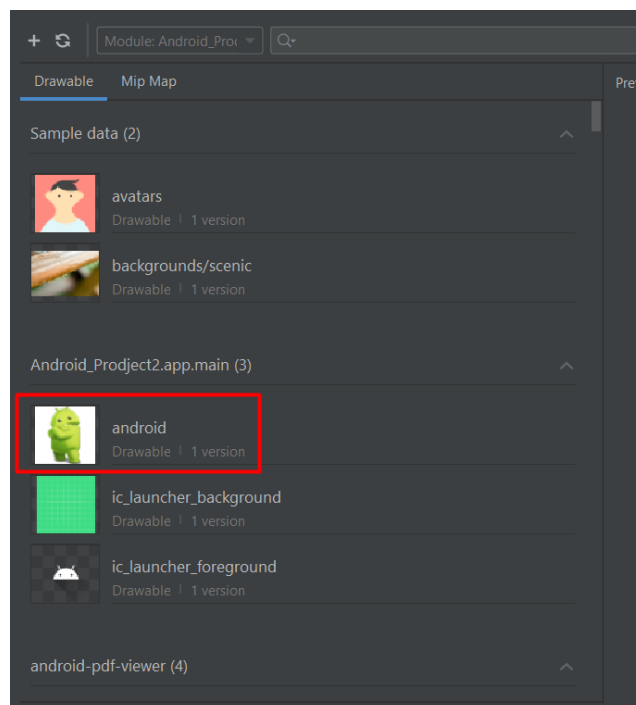


Рис. 13 (Картинка)

5) Работа с PDF

-Чтобы открыть файл PDF нужно подгрузить библиотеку с сайта, после перейти в папку build.gradle и вставить код

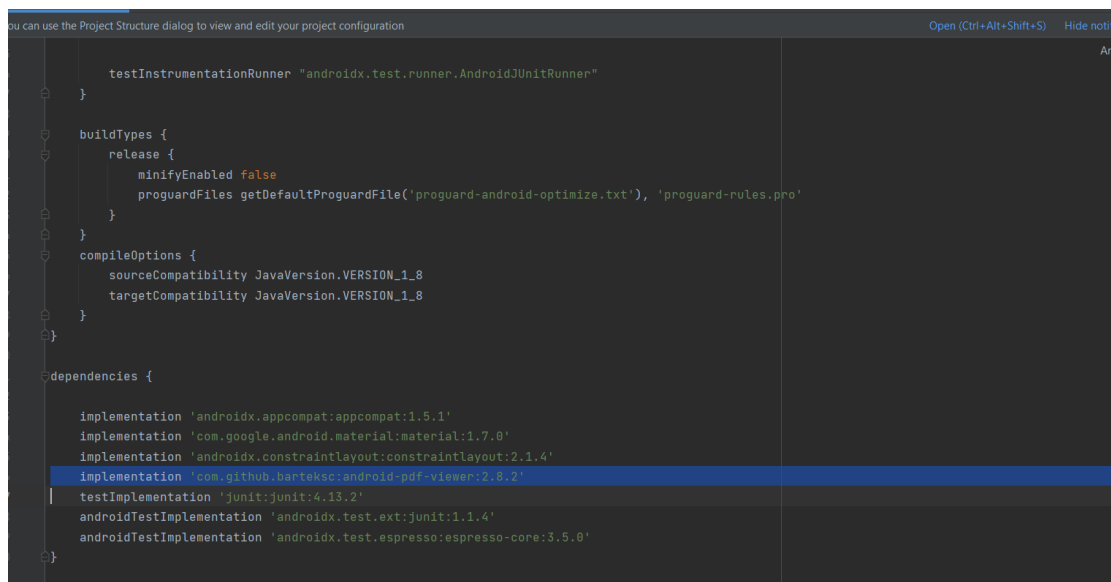


Рис. 14 (Код)

(Код для копирования)

implementation 'com.github.barteksc:android-pdf-viewer:2.8.2'

Далее создаем Обычное активити и доводим его до такого вида XML

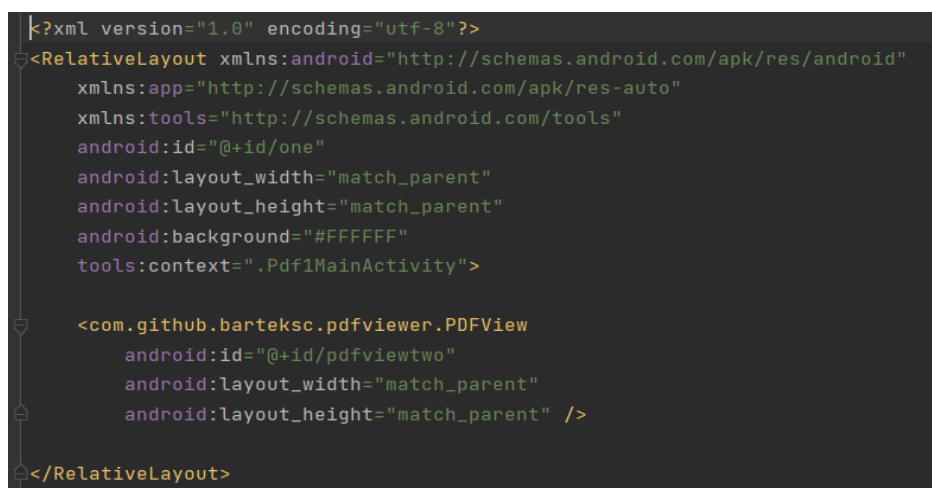


Рис. 15 (Конечный вид)

MainActivity (pdf1_main – название xml файла, pdfviewtwo – id объекта)

```

package com.example.android_project2;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import com.github.barteksc.pdfviewer.PDFView;

public class Pdf1MainActivity extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.pdf1_main);

        PDFView pdfView = findViewById(R.id.pdfviewtwo);
        pdfView.fromAsset("assetName: LR6_metodichka.pdf").load();
    }
}

```

Рис. 16 (Visual Studio Code)

После в дереве самого приложения (App) нужно создать новую папку assets, в которой хранятся файлы приложения. Далее загружаем туда свой файл в нашем случае это LR6-metodichka.pdf.

Заключение

Материал был подготовлен для начинающих создателей мобильных приложений, для того чтобы они могли опираться на материал статьи и создавать свои первые проекты. Статья в полном объеме раскрывает основные этапы разработки мобильного приложения.

Список источников

1. Федотенко М.А., Разработка мобильных приложений. Первые шаги. -2019.
2. Колисниченко Д.Н., Программирование для Android. -2020.
3. Черников В., Разработка мобильных приложений на С# для iOS и Android.-2020.
4. Хортон Д., Разработка Android-приложений с нуля. 3-е изд.-2023.