## Установка Android Studio. “Hello, World”

### Цель работы

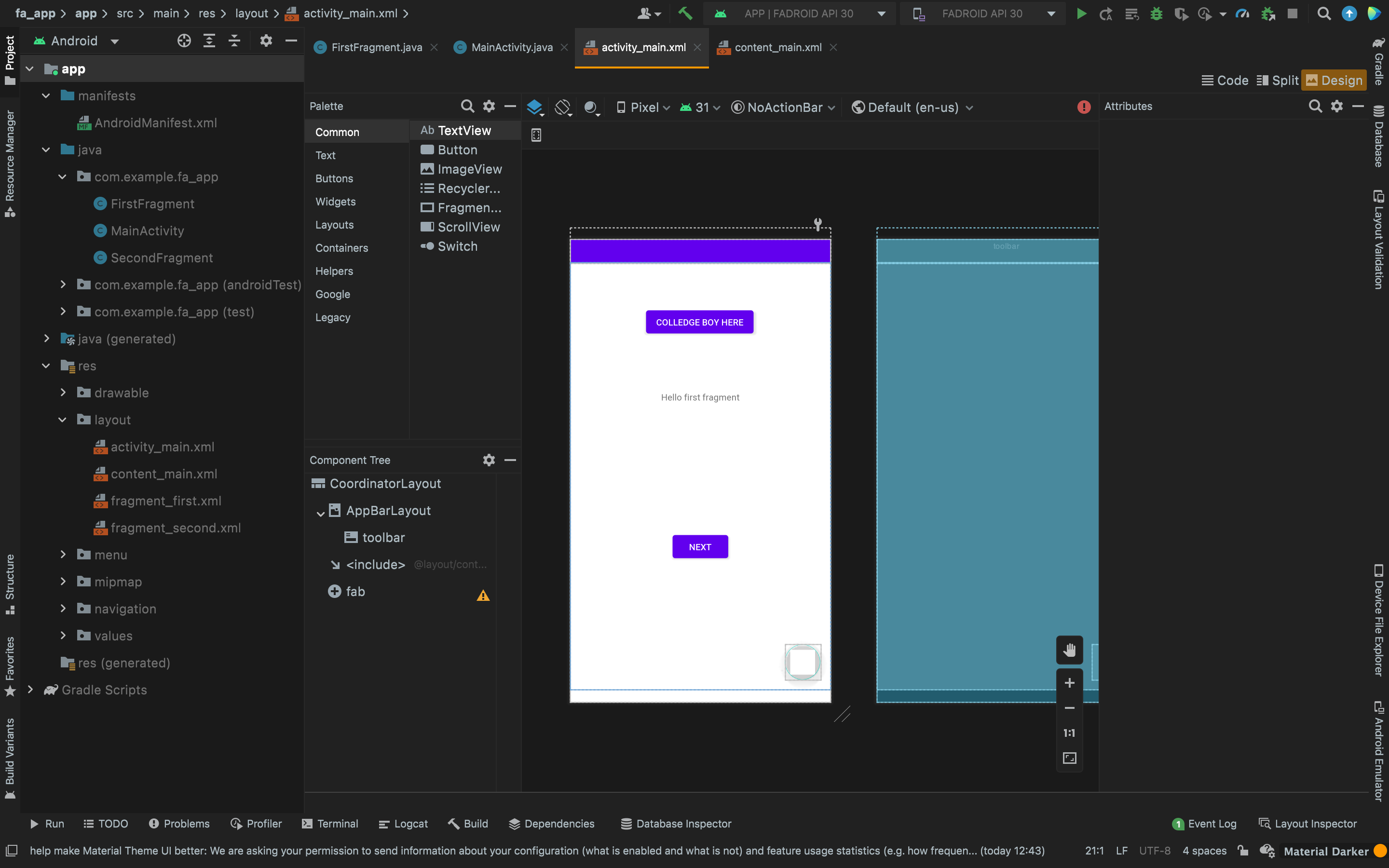
Получить навыки установки и первоначальной настройки среды разработки, получить базовое представление о структуре приложения и порядке его запуска.

### Задания для выполнения

1. Установите на компьютер среды разработки JDK и Android Studio.



1. Запустите Android Studio.



1. Создайте новый пустой проект на языке программирования Java.

Создан, API Level 30 (Android 11)

1. Познакомьтесь со структурой проекта и с содержанием главных файлов:
   1. MainActivity.java
   2. activity\_main.xml
   3. AndroidManifest.xml
   4. strings.xml

Изображение выглядит как текст, экран, снимок экрана

Автоматически созданное описание

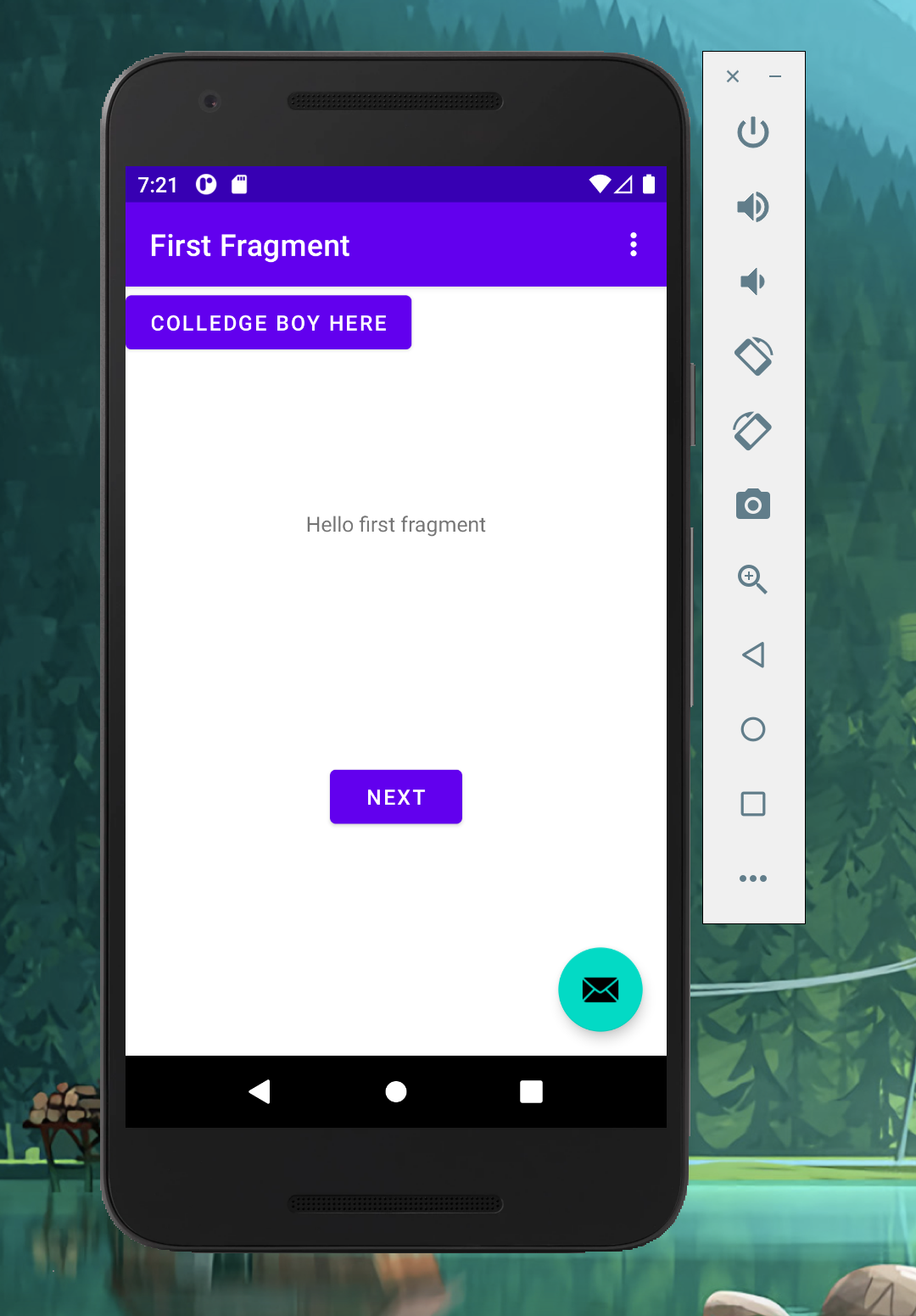
1. Найдите в справочном материале и кратко опишите своими словами назначение и содержание каждого из этих файлов.

* MainActivity.java - выполнение приложения Android по умолчанию начинается с класса MainActivity, который по умолчанию открыт в Android Studio. Класс MainActivity по сути представляет обычный класс java, в начале которого идет определение пакета данного класса.
* activity\_main.xml - это обычный текстовый файл с разметкой xml. Для переключения к коду нажмем на кнопку Code над графическим представлением.
* ConstraintLayout позволяет расположить вложенные элементы в определенных местах экрана. Вначале элемента ConstraintLayout идет определение пространств имен XML
* AndroidManifest.xml- каждое приложение содержит файл манифеста AndroidManifest.xml. Данный файл определяет важную информацию о приложении - название, версию, иконки, какие разрешения приложение использует, регистрирует все используемые классы activity, сервисы и т.д. Данный файл можно найти в проекте в папке manifests

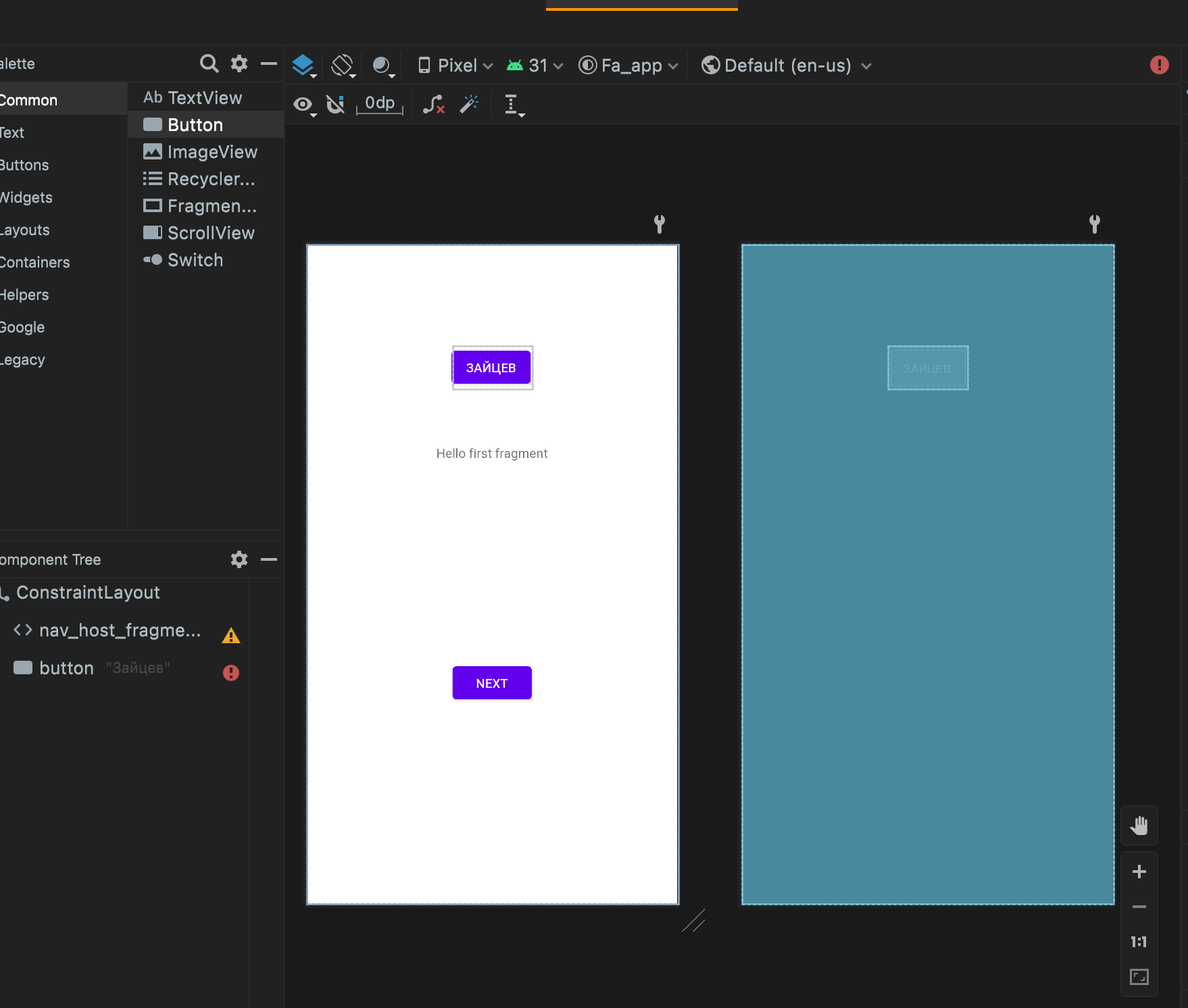
1. Познакомьтесь с присутствующими в проекте ресурсами.

Ресурс в приложении Android представляет собой файл, например файл разметки интерфейса или некоторое значение, например, простую строку. То есть ресурсы представляют собой и файлы разметки, и отдельные строки, и звуковые файлы, файлы изображений и т.д.

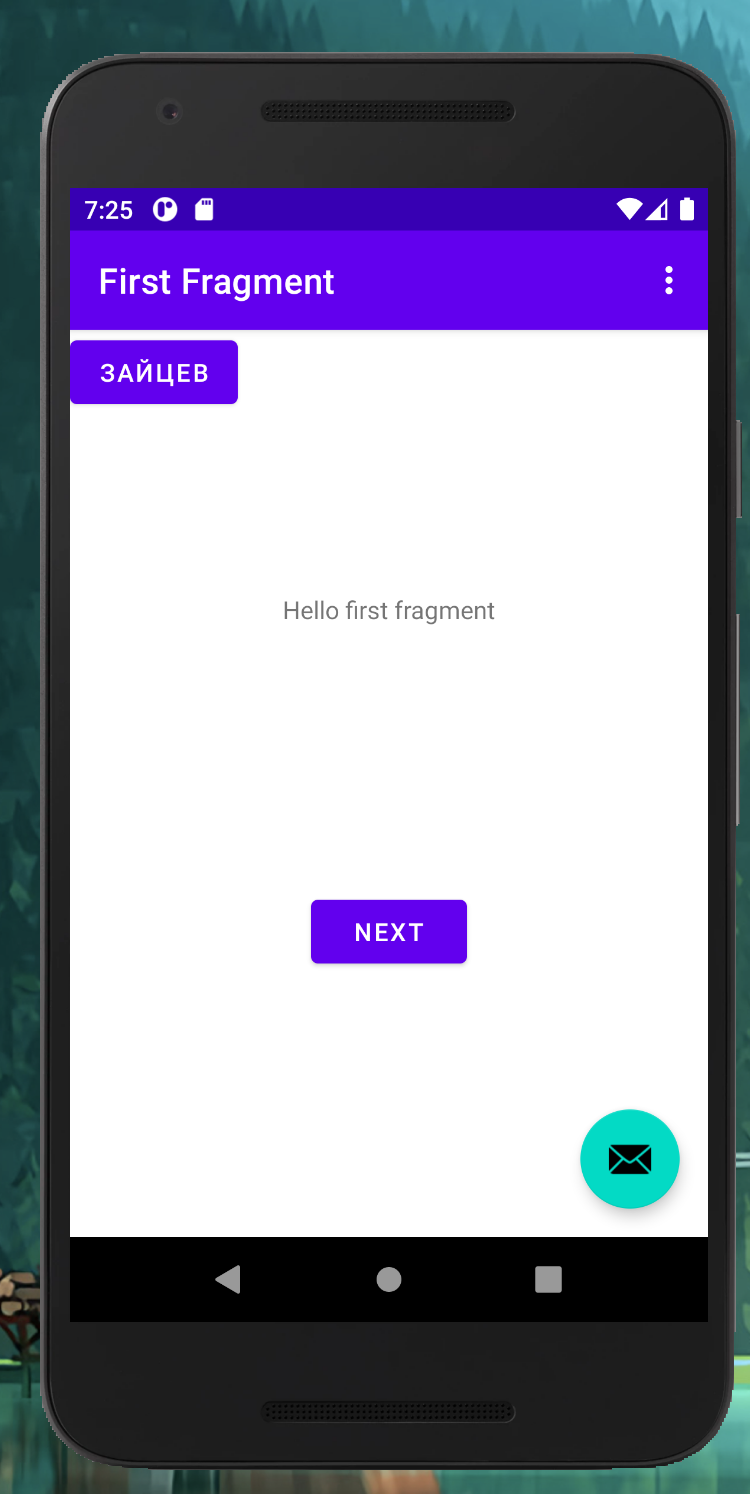
1. Запустите этот проект на виртуальном устройстве Android. Если виртуальных устройств не существует, создайте.



1. Модифицируйте проект таким образом, чтобы в названии приложения выводилась Ваша фамилия, а в строке на главном экране - любая другая строка по Вашему желанию.



1. Запустите данное приложение на нескольких виртуальных устройствах с разными версиями API и разными размерами, и разрешениями экрана.



### Контрольные вопросы

1. Что такое Activity и зачем оно нужно?

Каждый отдельный экран или страница в приложении описывается таким понятием как activity. Если мы запустим приложение на устройстве, то на экране мы по сути увидим определенную activity, которая предсталяет данный интерфейс

1. Что такое View и зачем оно нужно?

View – компонент, который пользователь может видеть и взаимодействовать с ним. В то время как ViewGroup является невидимым контейнером, который определяет структуру макета

1. Что такое Layout и зачем оно нужно?

Позволяет привязывать компонент к границам экрана или к другим компонентам. позволяет размещать View-компоненты в виде одной строки (horizontal) или одного столбца (vertical). настраивает положение каждого компонента относительно других

1. Почему все строки в Layout указываются косвенно из файла с ресурсами?

Потому что это ссылки на хранимые ресурсы. По этим ссылкам можно обращаться как в коде на странице activity\_main.xml, так и в графическом режиме (там они будут автоматически применяться)

1. Что такое версия API?

Уровень API — это целочисленное значение, однозначно идентифицирующее версию API фреймворка, предлагаемую платформой Android.

Обычно обновления API фреймворка платформы разрабатываются таким образом, чтобы новая версия API оставалась совместимой с более ранними версиями, поэтому большинство изменений в новом API являются аддитивными, а старые части API становятся устаревшими, но не удаляются

### Дополнительные задания

1. Замените RelativeLayout в главном окне приложения на LinearLayout с вертикальным расположением элементов.

<?xml version="1.0" encoding="utf-8"?>

<androidx.constraintlayout.widget.ConstraintLayout xmlns:android="http://schemas.android.com/apk/res/android"

xmlns:app="http://schemas.android.com/apk/res-auto"

xmlns:tools="http://schemas.android.com/tools"

android:layout\_width="match\_parent"

android:layout\_height="match\_parent"

tools:context=".MainActivity">

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Привет всем!"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintLeft\_toLeftOf="parent"

app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent" />

</androidx.constraintlayout.widget.ConstraintLayout>

1. Присвойте строке, отображаемой в TextView идентификатор следующим атрибутом:

<TextView

android:layout\_width="wrap\_content"

android:layout\_height="wrap\_content"

android:text="Привет всем!"

app:layout\_constraintBottom\_toBottomOf="parent"

app:layout\_constraintLeft\_toLeftOf="parent"

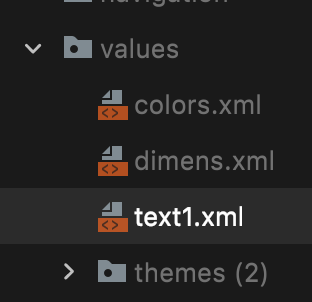
app:layout\_constraintRight\_toRightOf="parent"

app:layout\_constraintTop\_toTopOf="parent"

android:id=”@+id/main\_textview”

/>

1. Измените название ресурса строки на “text1”



1. Модифицируйте строку динамически из кода Java, добавив в обработчик события onCreate следующий код:

// 1. Access the TextView defined in layout XML

// and then set its text

mainTextView = (TextView) findViewById(R.id.main\_textview);

mainTextView.setText(“Set in Java!”);

