**МИНИСТЕРСТВО ОБРАЗОВАНИЯ РЕСПУБЛИКИ БЕЛАРУСЬ БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ**

Факультет прикладной математики и информатики

Кафедра технологий программирования

Жуковский Павел Сергеевич

Программирование Мобильных и Встраиваемых Систем

Отчет по лабораторной работе №6

## «Разработка приложения с использованием Android Studio»

студента 3 курса 12 группы

**Преподаватель**

**Давидовская Мария Ивановна**

**Минск 2020**

**Вариант 4**

**Ход выполнения**

**Задание 1. Теоретические сведения**

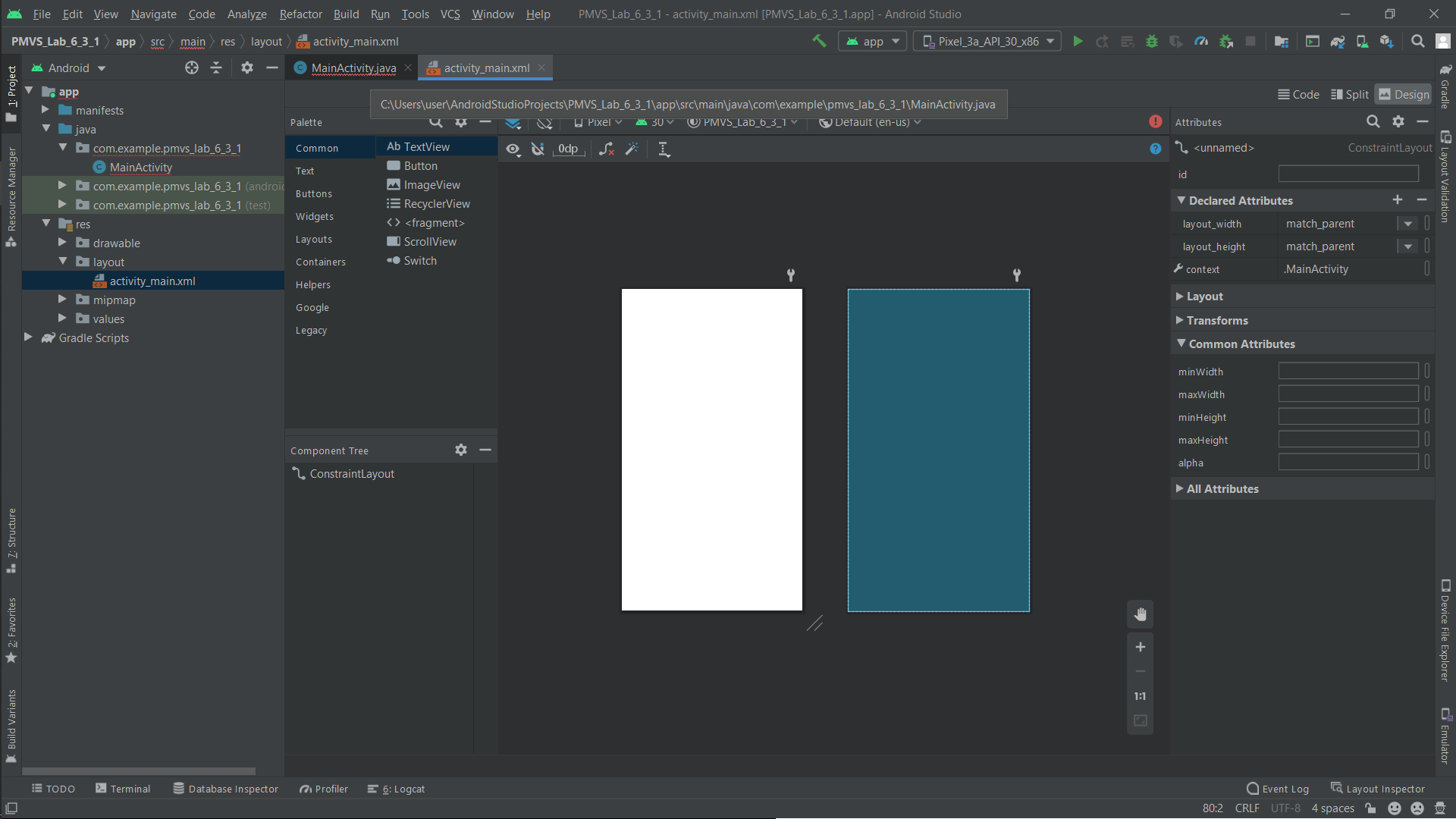
Ознакомился с теоретическими сведениями.

**Задание 2. Создать приложение «Угадай число»**

**2.1. Создать приложение и его структуру**

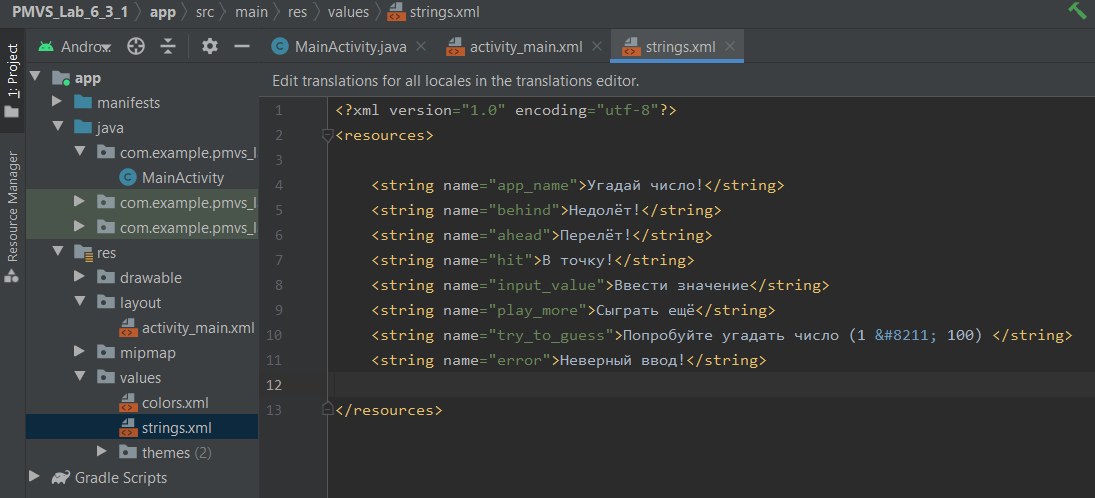
*Создайте новый проект в среде Android Studio. Процесс создания нового проекта и описание основных настроек подробно рассмотрен в лабораторной работе 5.*

Создал проект:

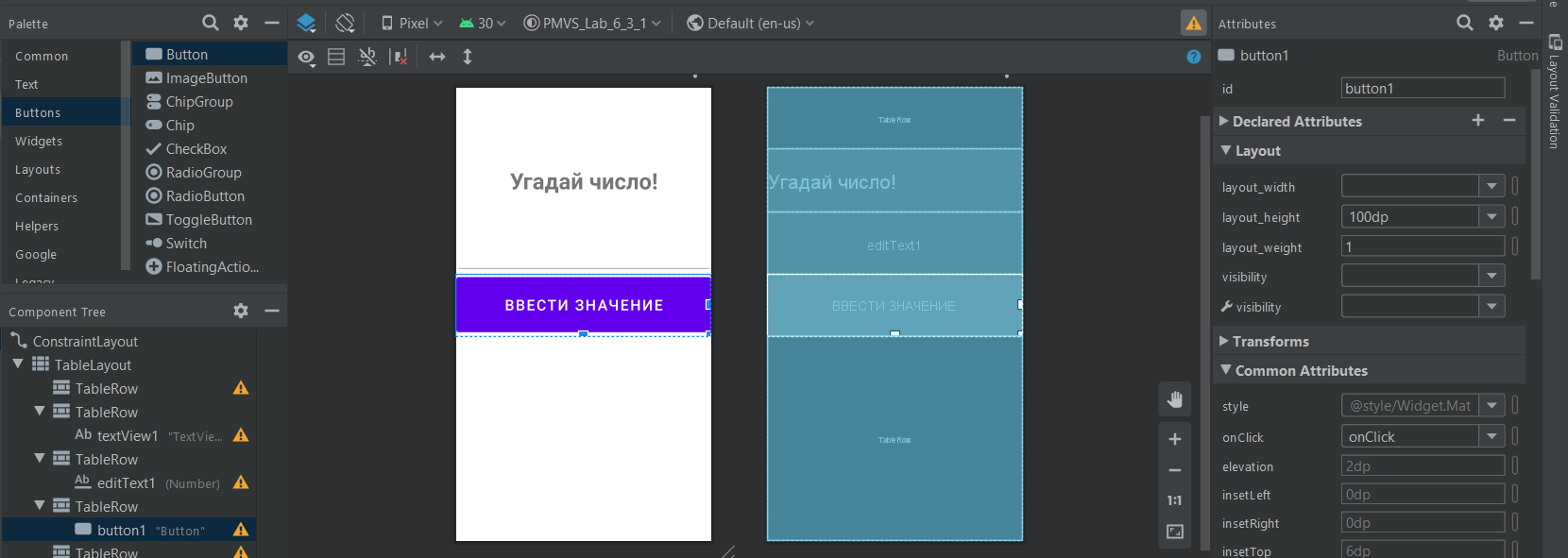


**2.2. Настроить интерфейс приложения**

Отредактировал файл strings.xml:

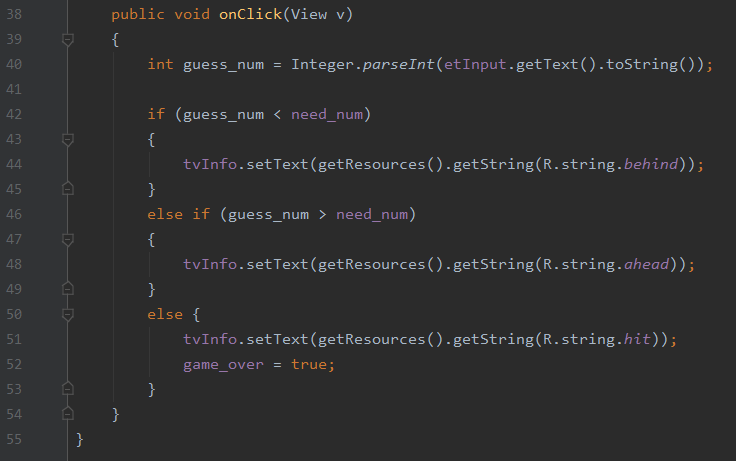


Расставил элементы интерфейса:

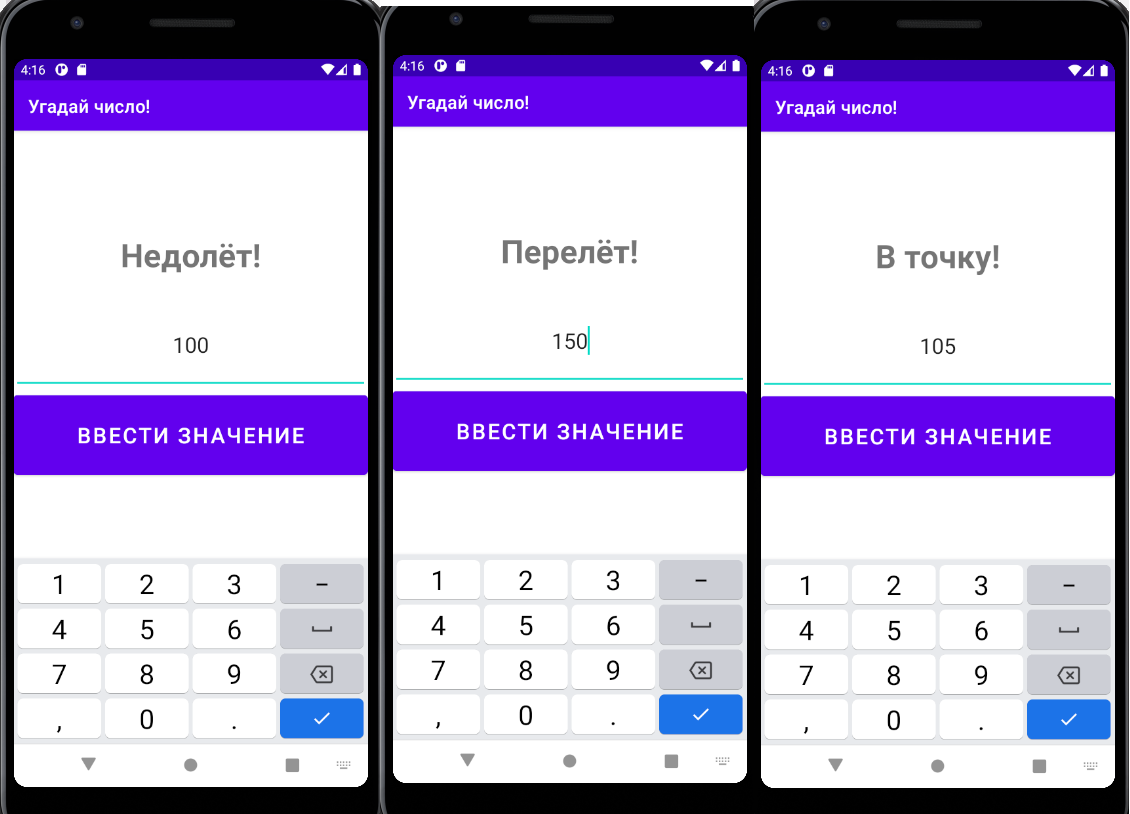


**2.3 Реализовать логику приложения**





Протестировал приложение:



**Задание 3. Задание для самостоятельной работы**

**Задание 3.1 Внести изменения в приложение «Угадай число»**

*Внести изменения в приложение, чтобы решались следующие проблемы:*

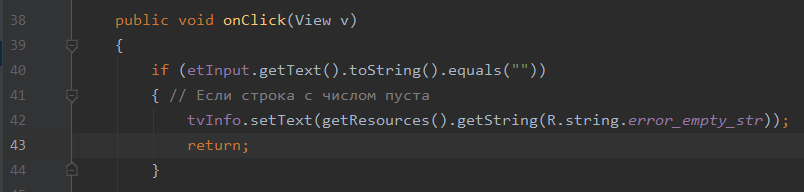
*- Что произойдёт, если кнопка будет нажата до ввода какого-либо числа?*

*Скорей всего приложение будет остановлено, необходимо как-то отслеживать этот вариант развития событий и адекватно реагировать.*

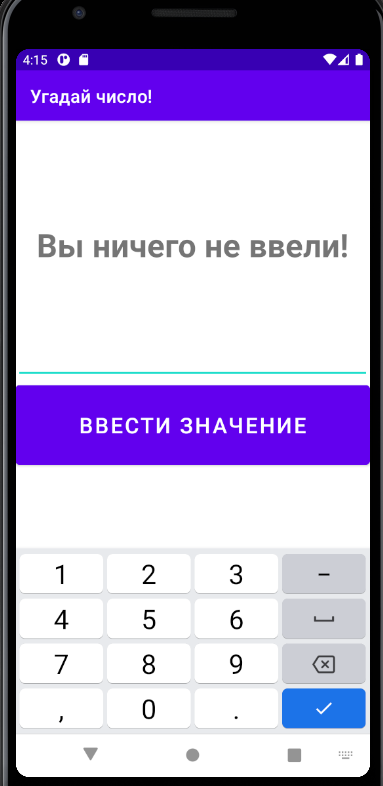
Добавил вот такую ошибку в strings.xml:

****

В методе onClick(View v) добавил следующую проверку:



И вот что получилось:



*- Что произойдёт, если пользователь введёт число меньшее нуля или большее*

*200? Скорей всего приложение обработает этот ввод, но было бы лучше, если*

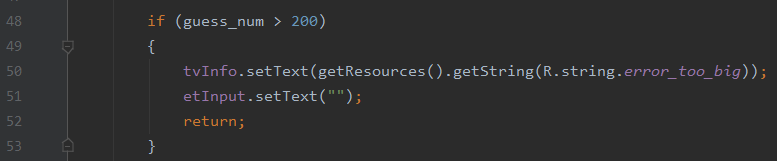
*бы появилось сообщение о том, что введённое число не удовлетворяет*

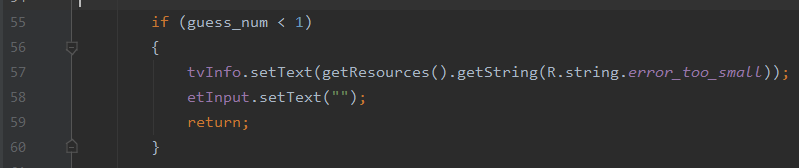
*условиям задачи.*

Добавил 2 ошибки в strings.xml:

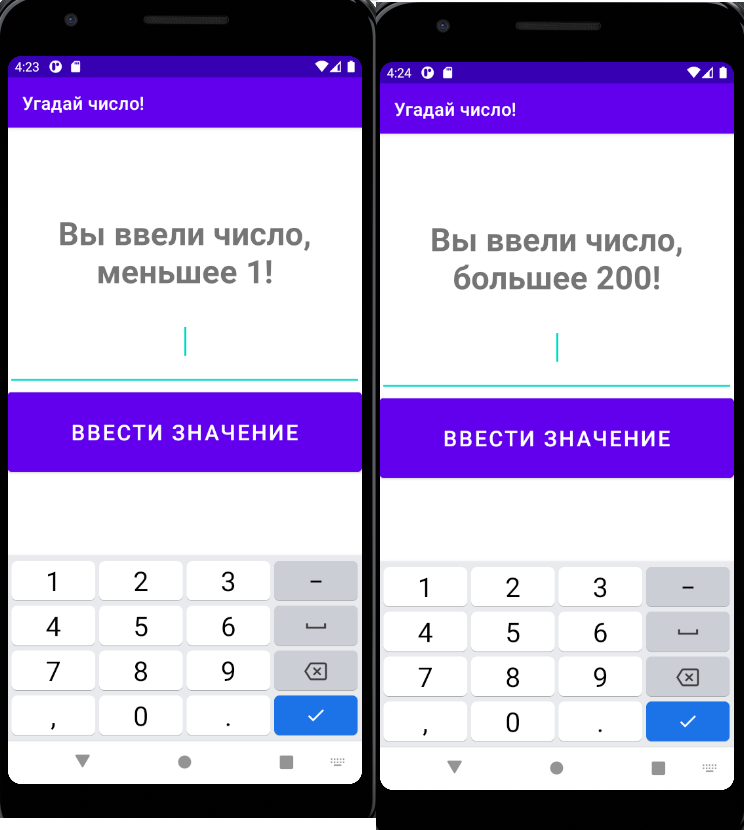


Потом добавил 2 проверки в метод:



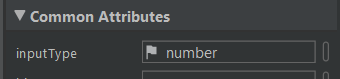


Я сделал небольшую модификацию, после ввода числа я сразу его стираю, чтобы пользователь смог быстрее ввести новое число. Вот, что получилось при вводе сначала нуля, а потом 201:



*- Что произойдет, если введено не число?*

Как я выяснил позже, ввести не число невозможно, так как в атрибутах у меня выбран тип «number»:

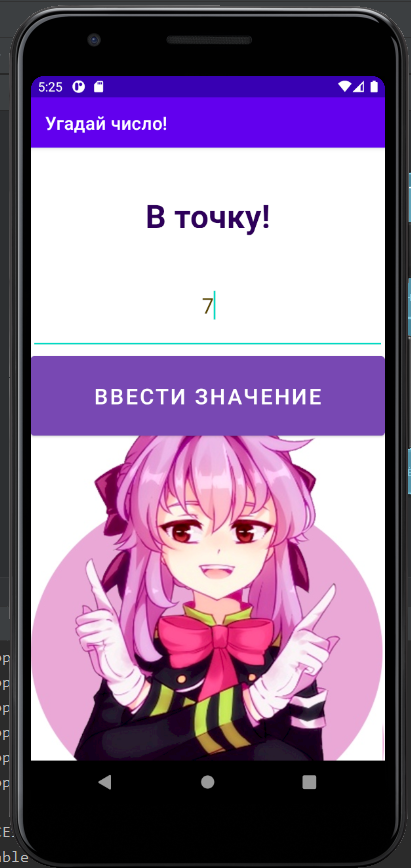


*- Как завершить приложение? И надо ли это делать?*

Как я выяснил позже, приложение закрывать не обязательно, ибо система Android делает это сама.

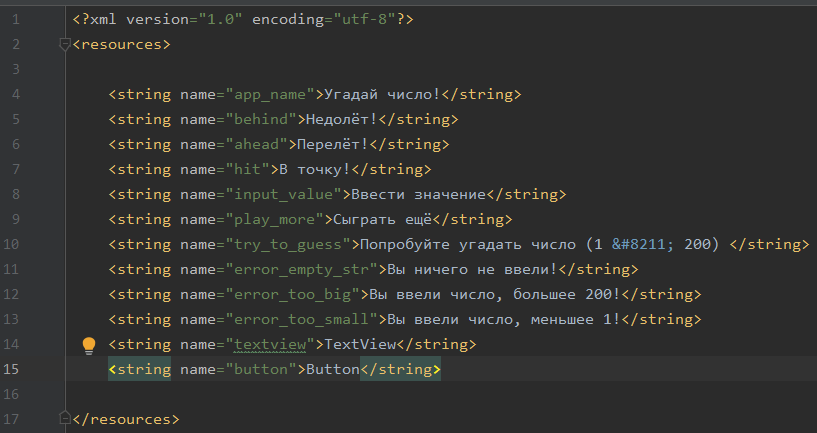
*- Изменить цвет фона, текста, кнопки.*

Поэкспериментировал:



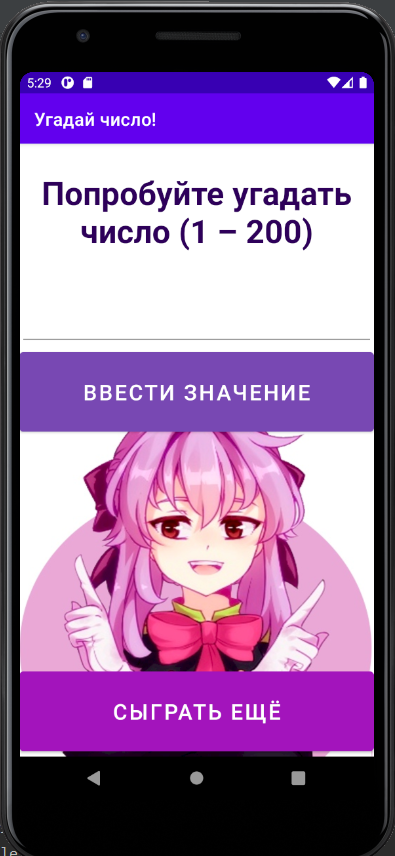
*- Добавить локализацию приложения для английского и русского языков.*

Это реализовано с помощью файла strings.xml:



*- Как еще можно улучшить приложение? Реализуйте эти улучшения.*

Я подумал, что будет неплохо добавить кнопку «сыграть ещё», которая генерирует новое число для новой игры:

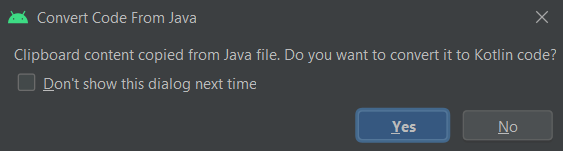


**Задание 3.2 Реализовать игру «Угадай число» на языке Kotlin**

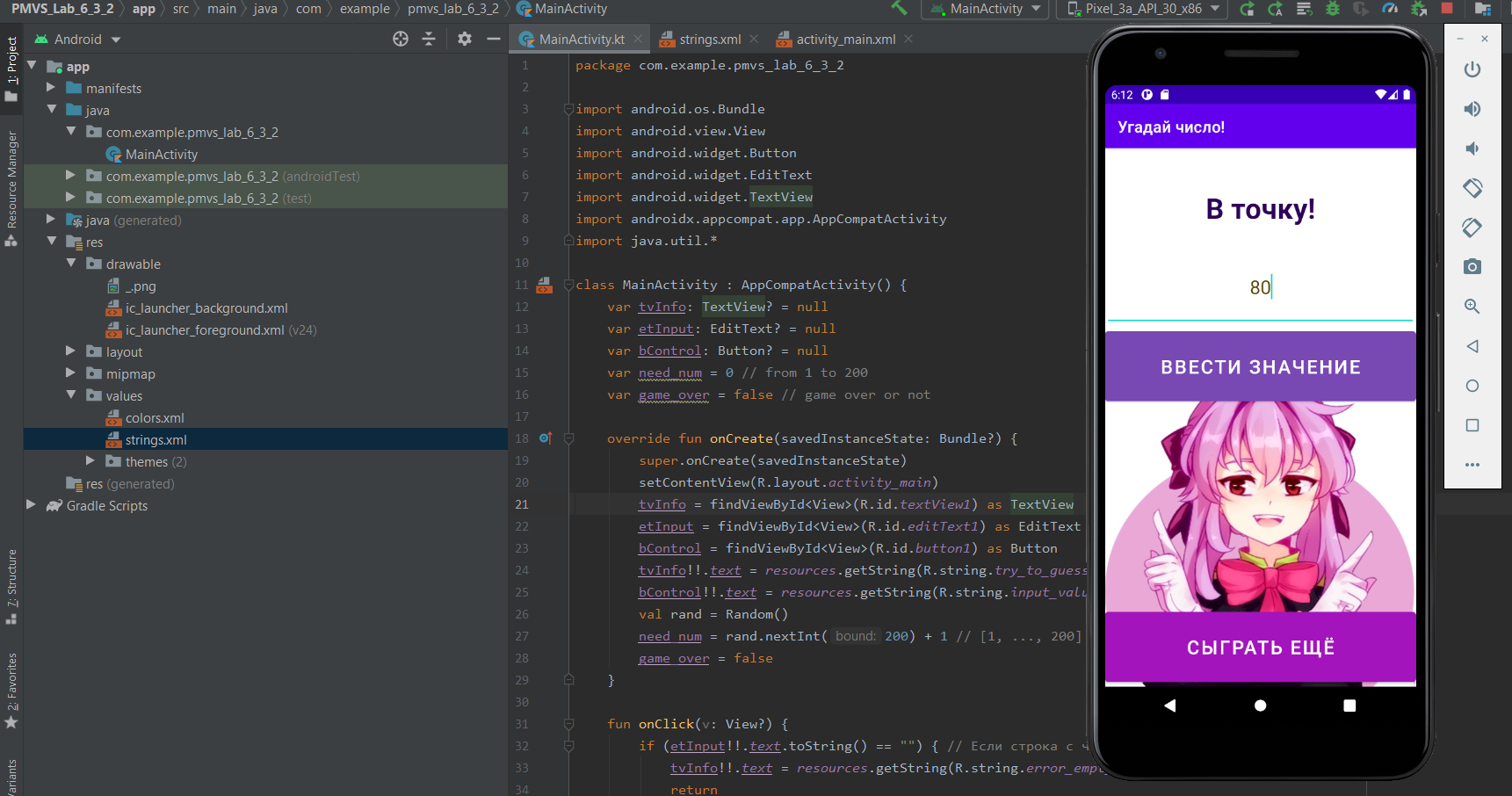
*Изучить синтаксис языка Kotlin в главах 1-4 книги Скин Джош, Гринхол Дэвид.*

*Kotlin. Программирование для профессионалов и реализовать приложение «Угадай число» на основе примера «A Simple Number Guessing Game» из книги Spath Peter. Learn Kotlin for Android Development (стр. 31-41) и с учетом требований, изложенных в задании 3.1.*

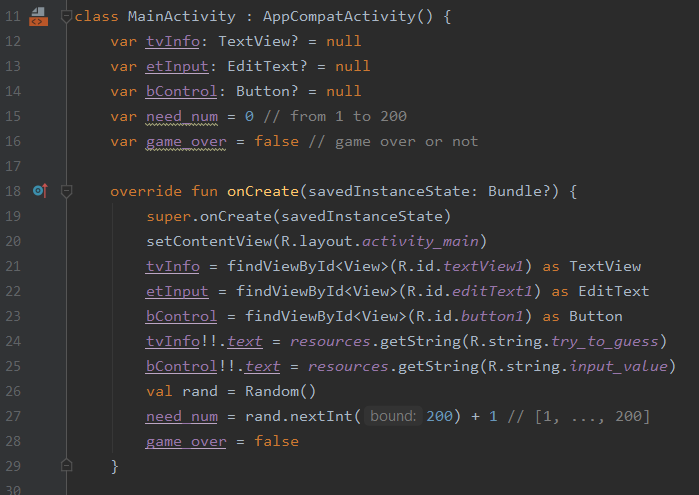
Итак, для начала я перекопировал все данные из strings.xml и activity\_main.xml в новый проект (на Kotlin), а затем попробовал перекопировать Java-исходники и появилось следующее сообщение:

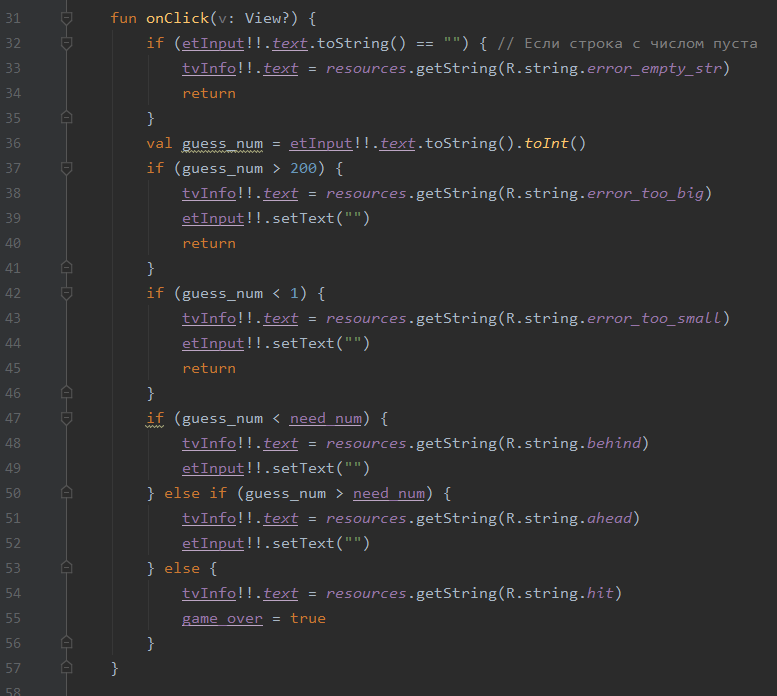


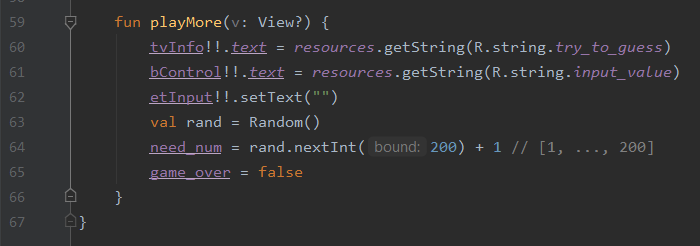
После этого мой проект запустился на языке Kotlin:



Сгенерированный код на Kotlin:







**Задание 3.3 Реализовать калькулятор по вариантам**

*Общие требования к вариантам:*

*- Предусмотреть возможность ввода дробных чисел через точку (например,*

*0.5).*

*- Предусмотреть возможность ввода отрицательных чисел.*

*- Создать графическое представление приложения Calculator для*

*горизонтальной ориентации экрана (пример приведен на рисунке 3.1, б).*

*- Программно реализовать обработку нажатий на кнопки с использованием*

*Activity в качестве единого обработчика.*

*- В случае деления на 0 выводить вместо результата сообщение о*

*недопустимости операции.*

*- Разработать перечень проверок и протестировать приложение Calculator.*

*- Опубликовать код приложения в репозиторий.*

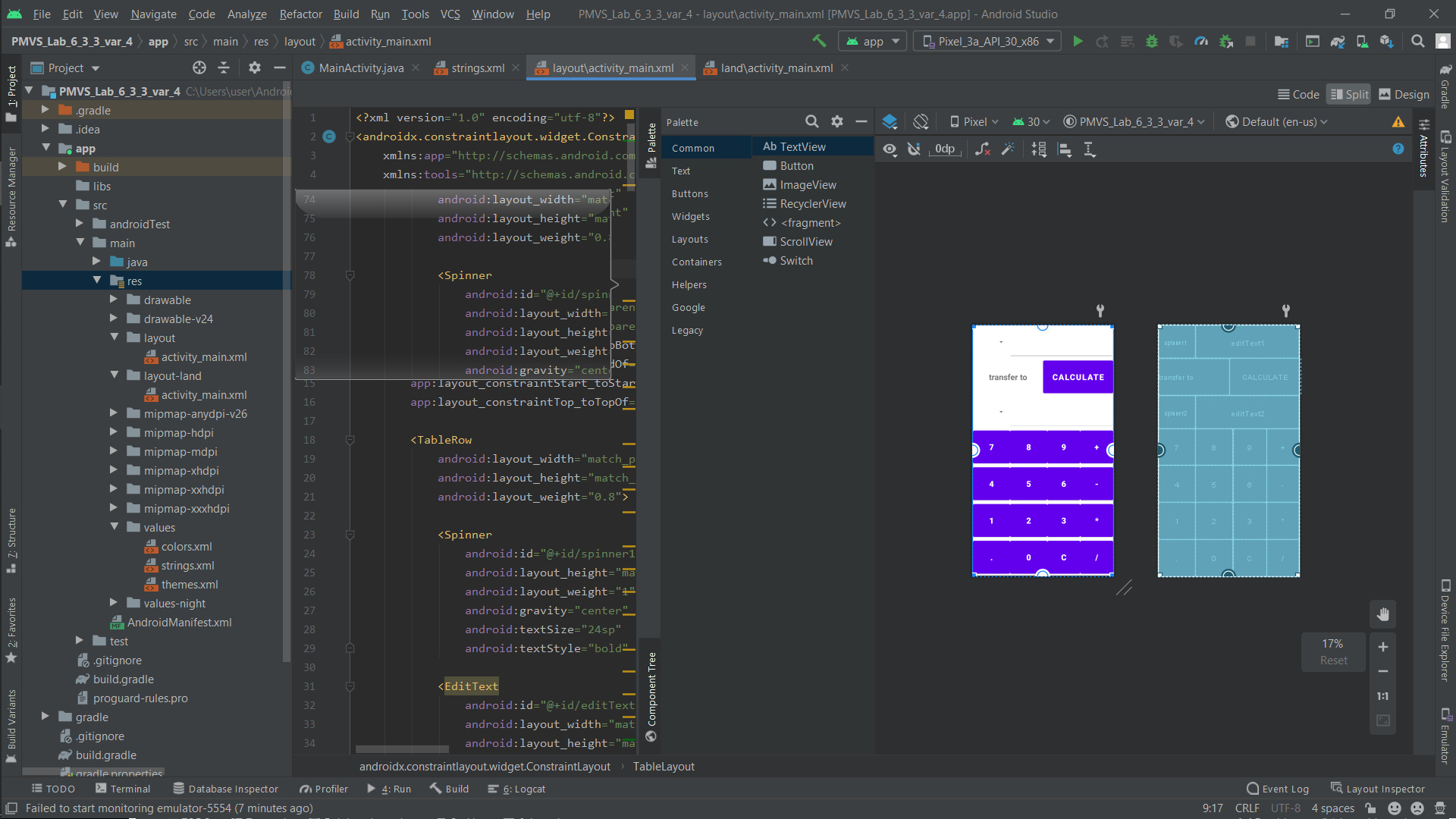
*- Продемонстрировать работу приложения Calculator на эмуляторе или*

*реальном устройстве.*

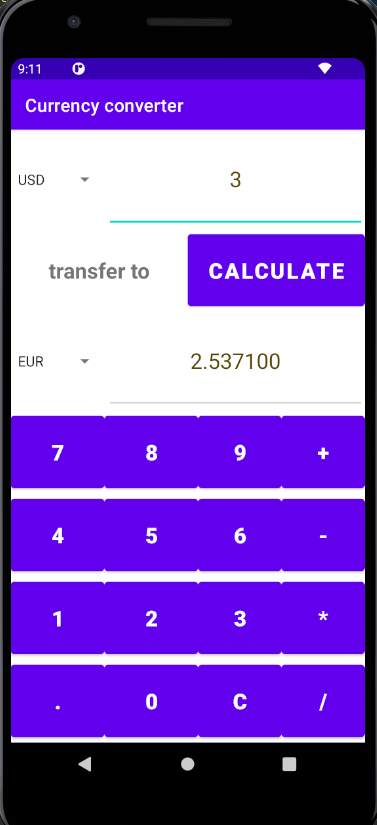
***Вариант 4.***

*Разработать приложение Calculator 4 c одним Activity, которая позволяет конвертировать валюты (не менее 10 валют). Реализовать с кнопками цифр, математических операций и блоком вывода результата. Локализовать приложение для английского и украинского языка.*

Создал проект и реализовал приложение:

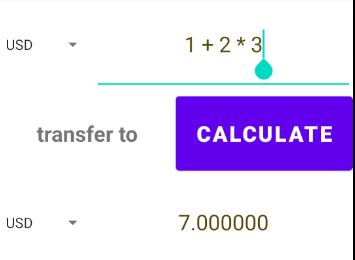


Демонстрация приложения:



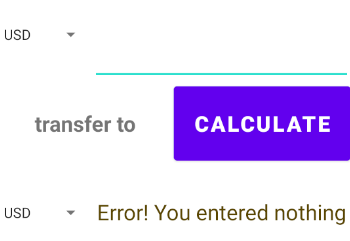
Для вычисления сложных математических выражений я воспользовался функцией eval из следующего источника: <https://fooobar.com/questions/28238/evaluating-a-math-expression-given-in-string-form>

Данная функция помогла мне из строки типа String распознавать математические выражения и выполнять их:

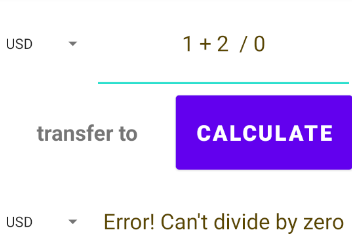


Кстати говоря, если не менять валюты, то данный конвертер вполне можно использовать как обычный калькулятор.

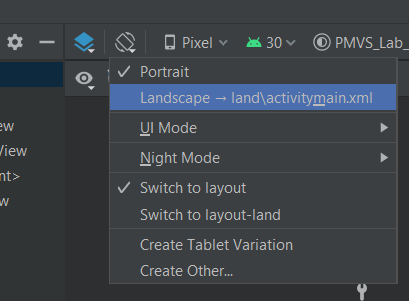
Также обработал пару ошибок, например, если пользователь ничего не введёт:

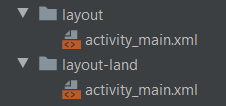


Проверка ошибки при делении на ноль:



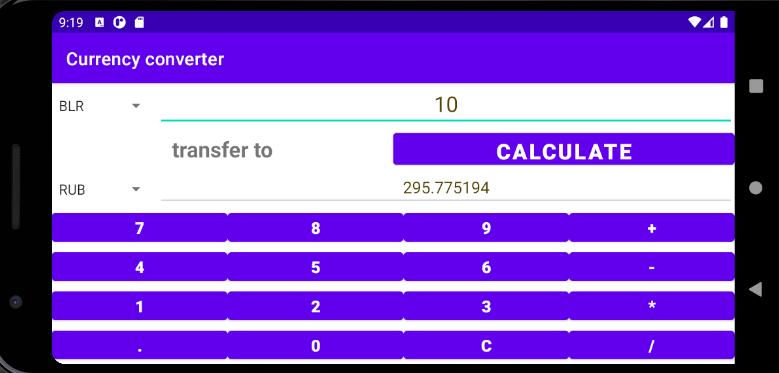
Для создания второго xml-файла для горизонтальной ориентации я воспользовался встроенными функциями Android-Studio и нажал кнопку «Create Landscape Variation», которая создала мне новую папку «layout-land» и идентичный xml-файл в ней:

э

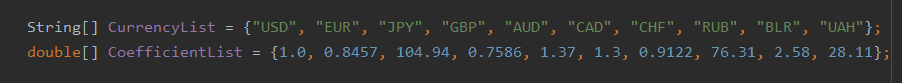


В новом xml-файле для корректного отображения мне пришлось редактировать лишь размер шрифта кнопок. В остальном с пропорциями проблем не возникло, так как я пользовался tableLayout и использовал для задания размеров тип dp.

Демонстрация приложения в горизонтальном режиме:



Здесь в MainActivity.java я храню информацию о всех 10-ти валютах и их коэффициентах относительно американского доллара:

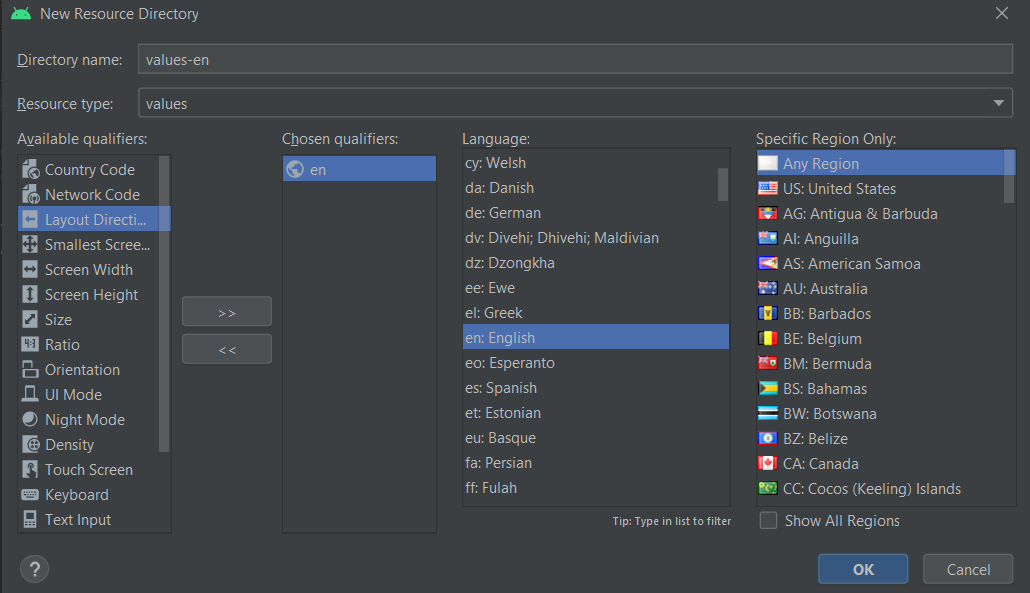


Я взял 7 самых популярных валют, а также российский и белорусский рубли, ну а также украинские UAH.

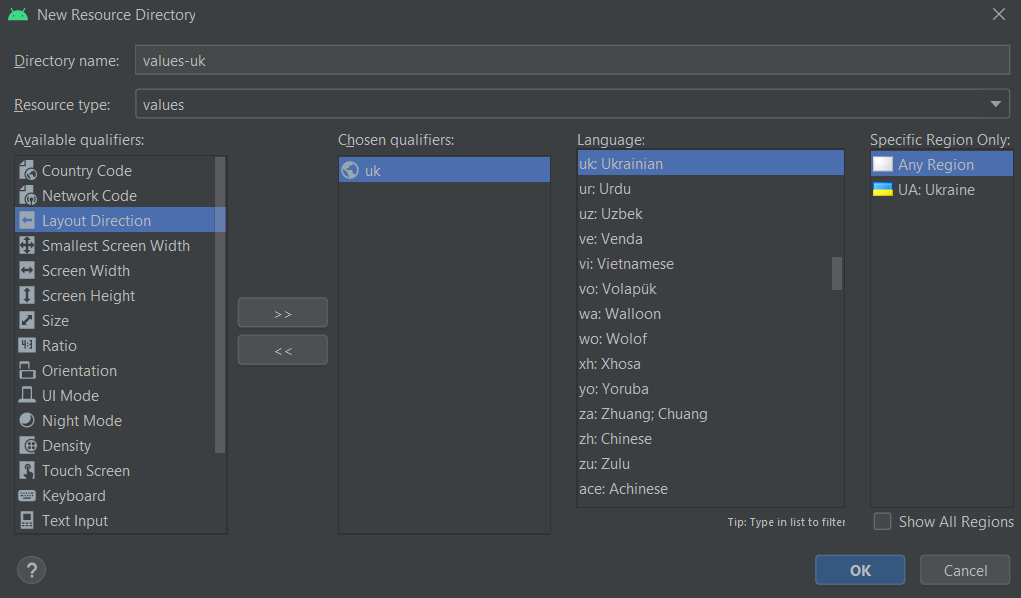
Касаемо локализации, мне нужно было добавить 2 локализации: английскую и украинскую.

Для этого я создал пару файлов в проекте…

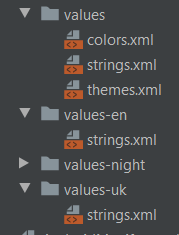
Сначала добавил английскую лоализацию:



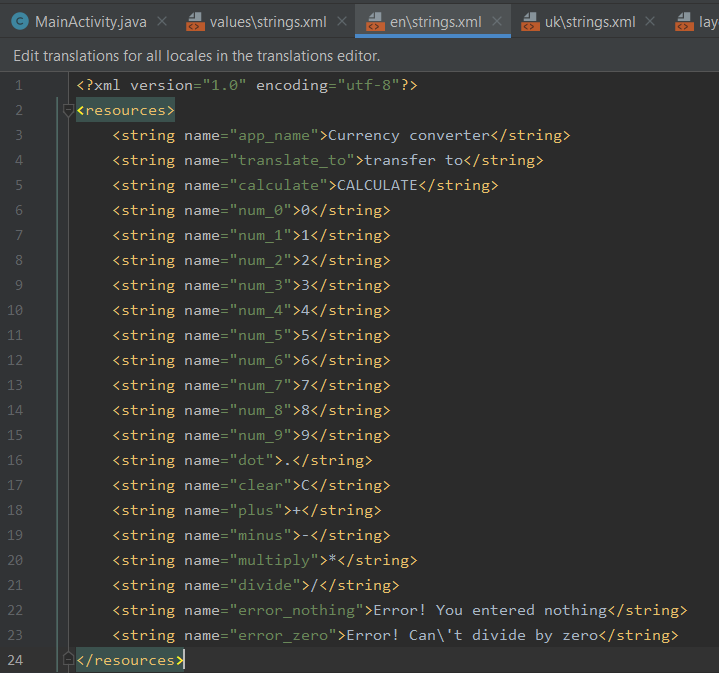
Затем украинскую:



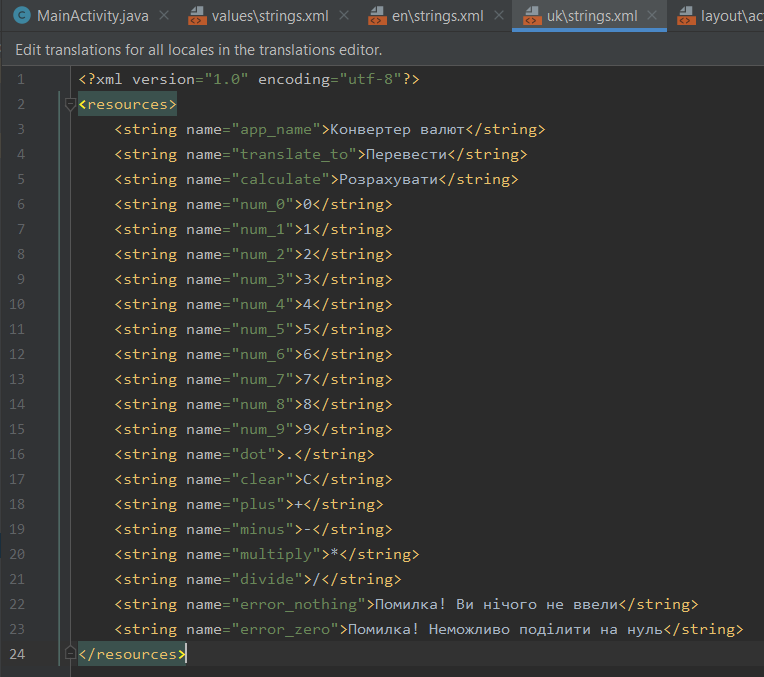
После этого в каждом из них я создал файл strings.xml с прописанными на соответствующих языках строками:



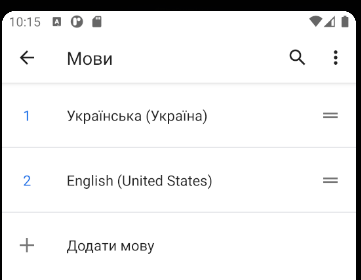
Файл strings.xml для английского совпадает с файлом strings.xml, который используется в проекте по умолчанию:



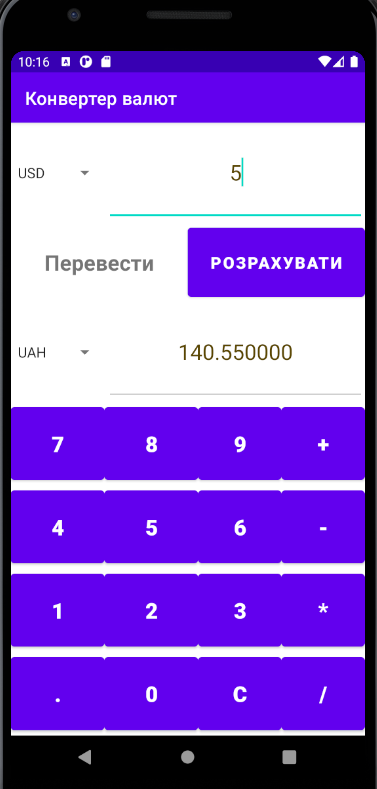
А вот так выглядит strings.xml для украинской локализации:

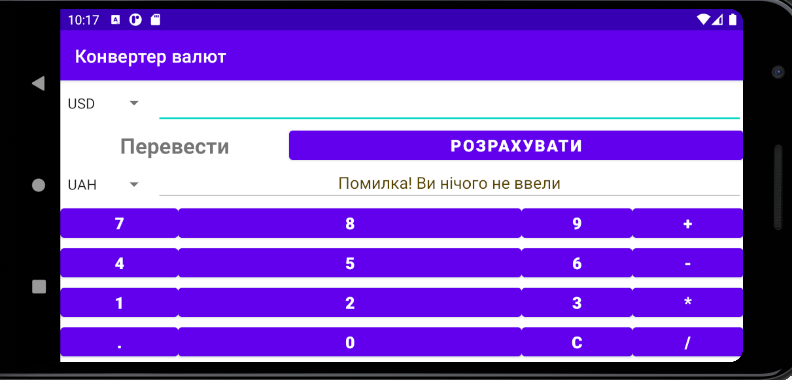


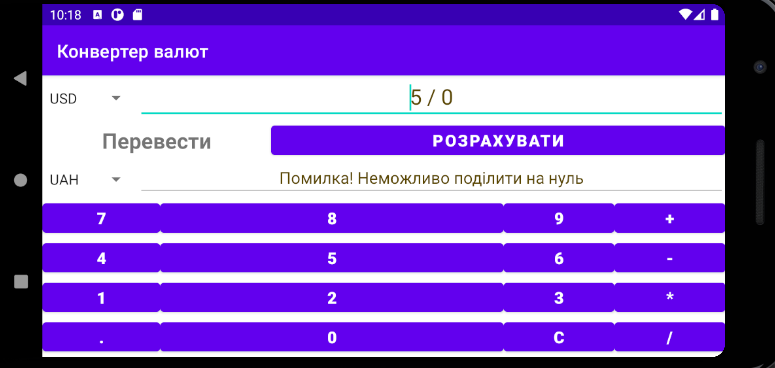
Чтобы протестировать украинскую локализацию я временно поменял настройки эмулятора:



Демонстрация для украинской локализации:







**Контрольные вопросы**

*1. Какие основные компоненты Android-приложения Вы знаете?*

Деятельность (Аctivity), служба (Service), приемник широковещательных намерений (Broadcast Receiver), контент-провайдер (Content Provider).

*2. Что такое графическое представление Аctivity?*

Графическое представление Activity формируется из различных компонентов (кнопка, поле ввода, чекбокс и т. д.), называемых View. Графическое представление каждого Activity в виде требуемого набора и взаимного расположения View-элементов хранится в xml-файле папки layout.

*3. Что такое Layout? Какие существуют виды Layout?*

Для размещения View-компонентов используются специальные контейнеры (ViewGroup), называемые Layout. Layout бывают различных типов (LinearLayout, RelativeLayout, FrameLayout, TableLayout, ConstraintLayout и т. д.) и отвечают за то, как будут расположены их дочерние View-компоненты на экране (таблицей, строкой, столбцом).

*4. Какие параметры (атрибуты) имеют View-элементы?*

View-элементы могу иметь следующие атрибуты:

android:layout\_width

android:layout\_height

android:text

android:id

android:background

android:padding

*5. Как создать Layout-файл для работы в горизонтальной ориентации экрана мобильного устройства? В каких случаях это необходимо?*

По умолчанию layout-файл настроен под вертикальную ориентацию экрана. Однако при повороте смартфона включится горизонтальная ориентация, что может привести к некорректному отображению View-элементов. Для устранения данной проблемы необходимо создать еще один layout-файл для горизонтальной ориентации экрана. Эту задачу можно решить с помощью опции Create Landscape.

*6. Для чего нужны методы setContentView, findViewById?*

Метод setContentView(int) – устанавливает содержимое Activity из layout-файла, в то время как findViewById() необходим для получения доступа к объекту по его id из кода программы.

*7. Какие существуют способы обработки событий в Аctivity?*

Обработку событий в Activity можно делать: 1) в коде приложения при помощи listener; 2) в xml файле при помощи специальных атрибутов.

*8. Какой файл содержит ссылки на все ресурсы, используемые в приложении?*

Файл с именем R.java

*9. Приведите правило именования ресурсов в проекте android*

Имя ресурса должно начинаться с его главной логической области. Если строка или идентификатор относятся строго к какому-то контексту (например, Activity), то после логического названия должно идти название этого контекста Названия должны начинаться с имени файла, в котором они находятся

*10.Приведите примеры записи полупрозрачного цвета в полной и краткой форме.*

В полной форме:

<color name="transparent\_red">#77FF0000</color>

В краткой форме:

<color name="transparent\_red">#77FF0</color>

*11. Какие функции выполняет класс TableLayout? Какие дочерние элементы класса? Привести примеры позиционирования и фрагменты кода*

Разметка TableLayout (Табличная разметка) позиционирует свои дочерние элементы в строки и столбцы, как это привыкли делать веб-мастера в теге table. Дочерние элементы TableRow

<TableLayout

android:id="@+id/tablelayout"

android:layout\_width="fill\_parent"

android:layout\_height="wrap\_content">

<TableRow>

<TextView

android:text="Щелкните на кнопке Усадка, чтобы увидеть, как длинный текст может уместиться в ячейке таблицы"

android:padding="3dip" />

</TableRow>

</TableLayout>

*12.Привести организации файловой структуры для локализации приложения в случае трех языков.*

Res

|\_values-ru // русский

|\_values-pl // польский

|\_values-en // английский

*13.Перечислить все возможности для индивидуальной настройки приложения в Android.*

• MCC (Mobile Country Code) и MNC (Mobile Network Code)

• Язык и регион

• Размер экрана (small, medium или large)

• «Широкоформатность» экрана (long или notlong)

• Ориентация экрана (port, land или square)

• Плотность пикселей на экране (ldpi, mdpi, hdpi или nodpi)

• Тип сенсорного экрана (notouch, stylus или finger)

• Доступность клавиатуры (keysexposed, keyshidden или keyssoft )

• Тип ввода (nokeys, qwerty или 12key )

• Способ навигации (nonav, dpad, trackball или wheel )