

Разработка Android-приложений

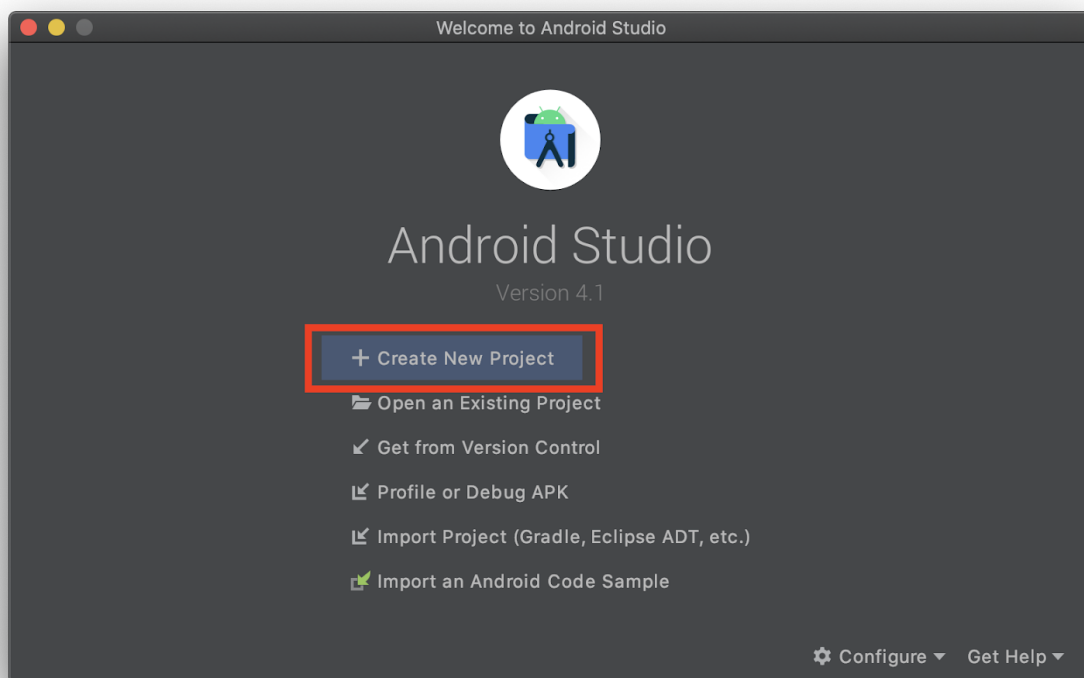
Домашнее задание

Задание 1. Создание Android-приложения

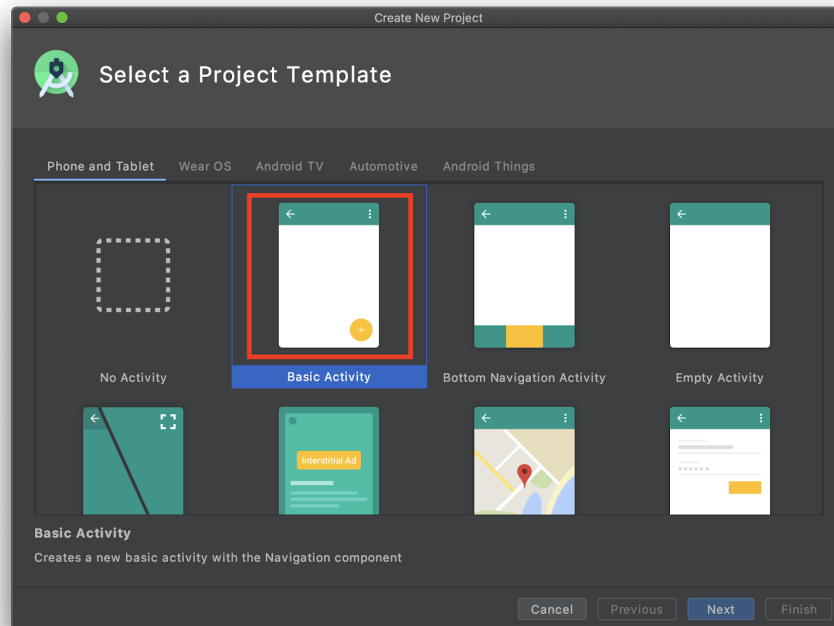
Что нужно сделать

Создайте Android-приложение так же, как это было показано в видеоуроке, а затем измените цвета экранов и надписей «День» и «Ночь» на любые другие на ваше усмотрение.

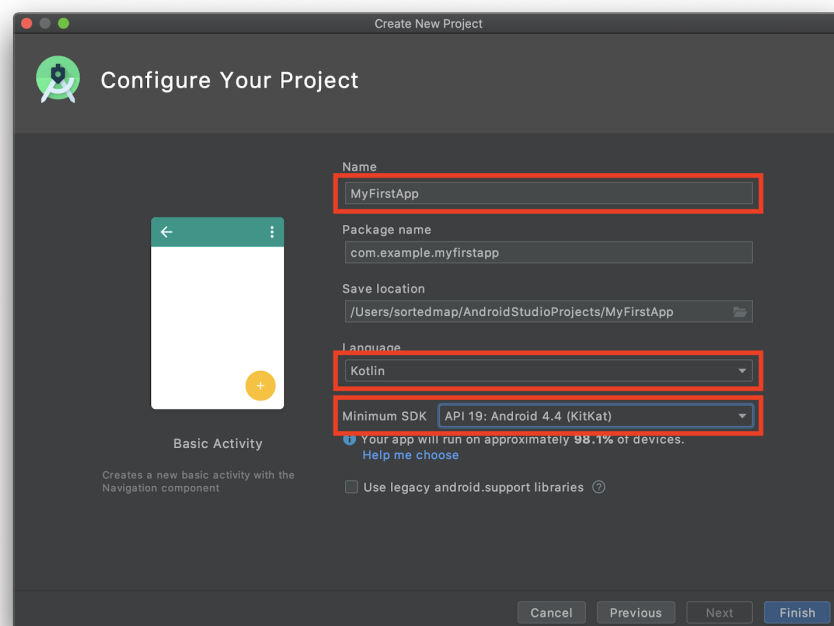
- 1.1. Скачайте и установите на свой компьютер среду разработки [Android Studio](#).
- 1.2. Запустите её и в появившемся окне нажмите **Create New Project**:



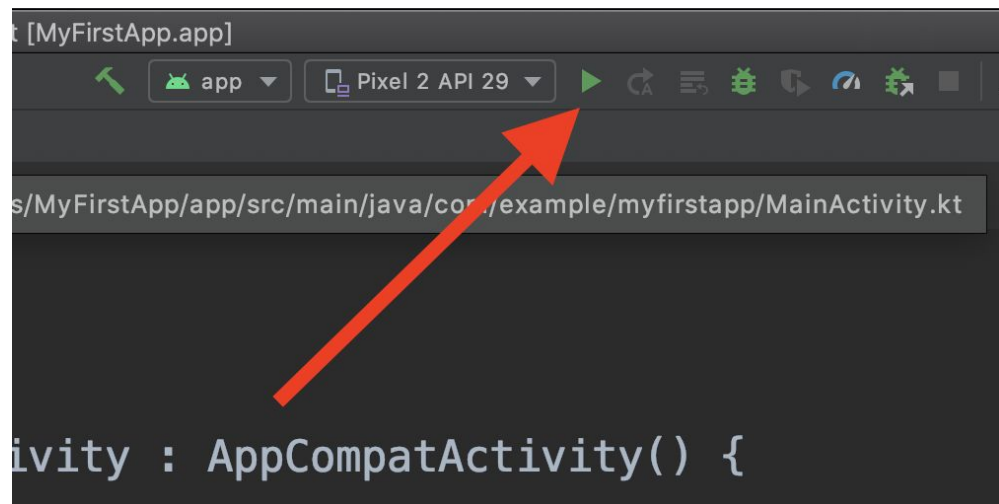
- 1.3. Выберите шаблон приложения **Basic Activity** и нажмите **Next**:



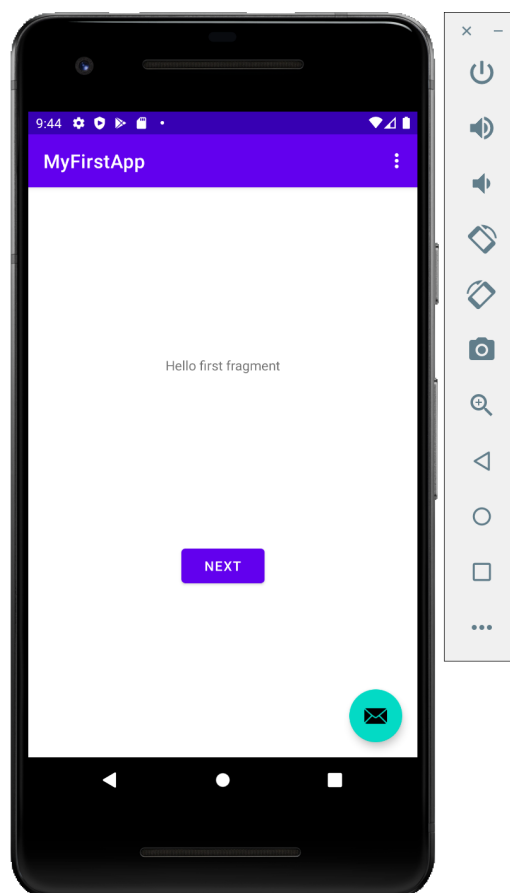
- 1.4. Назовите приложение по-английски, убедитесь, что в форме выбран язык программирования Kotlin, выберите версию Android SDK 4.4 KitKat и нажмите кнопку **Finish**:



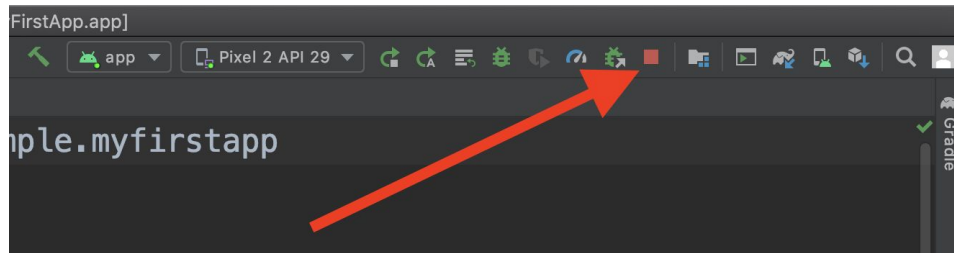
- 1.5. Подождите некоторое время: проект будет создан автоматически, после вы сможете его запустить. Для этого нажмите кнопку **Run**, расположенную справа вверху:



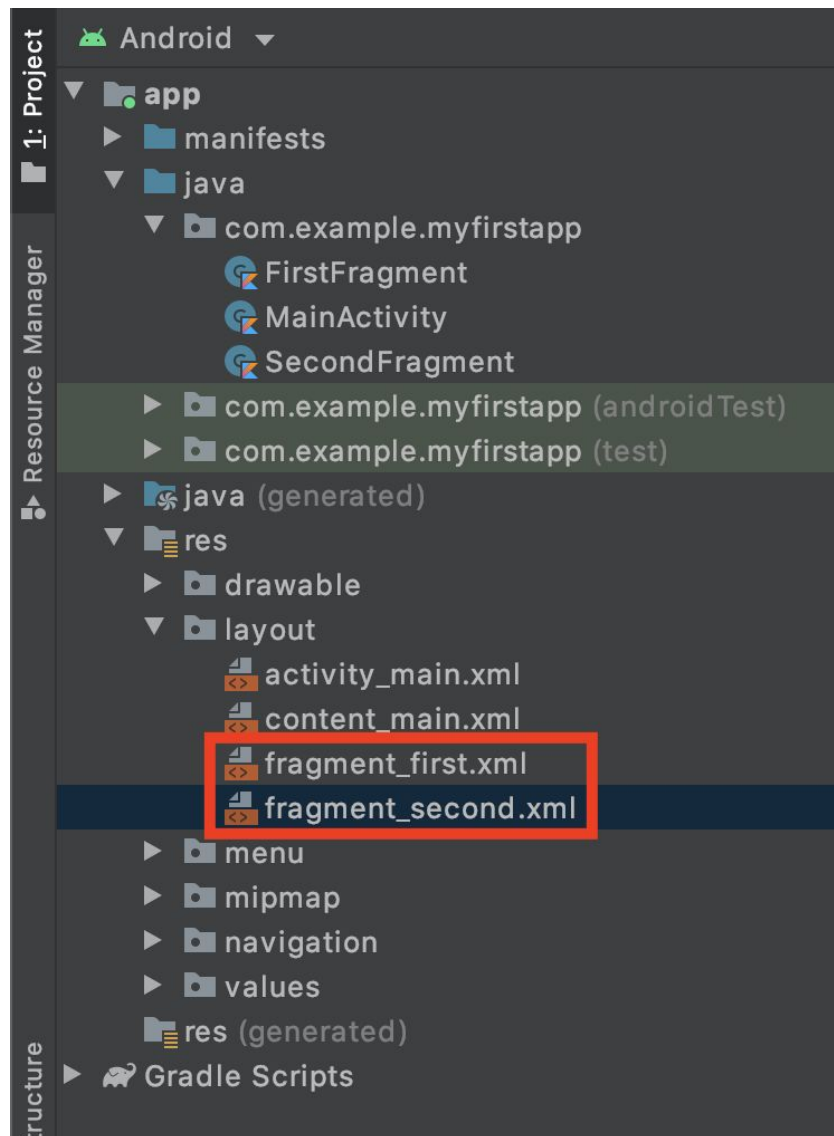
- 1.6. После запуска сначала запустится симулятор реального Android-смартфона, на котором запустится приложение. Этот процесс может занять длительное время. Запущенное приложение будет выглядеть так:



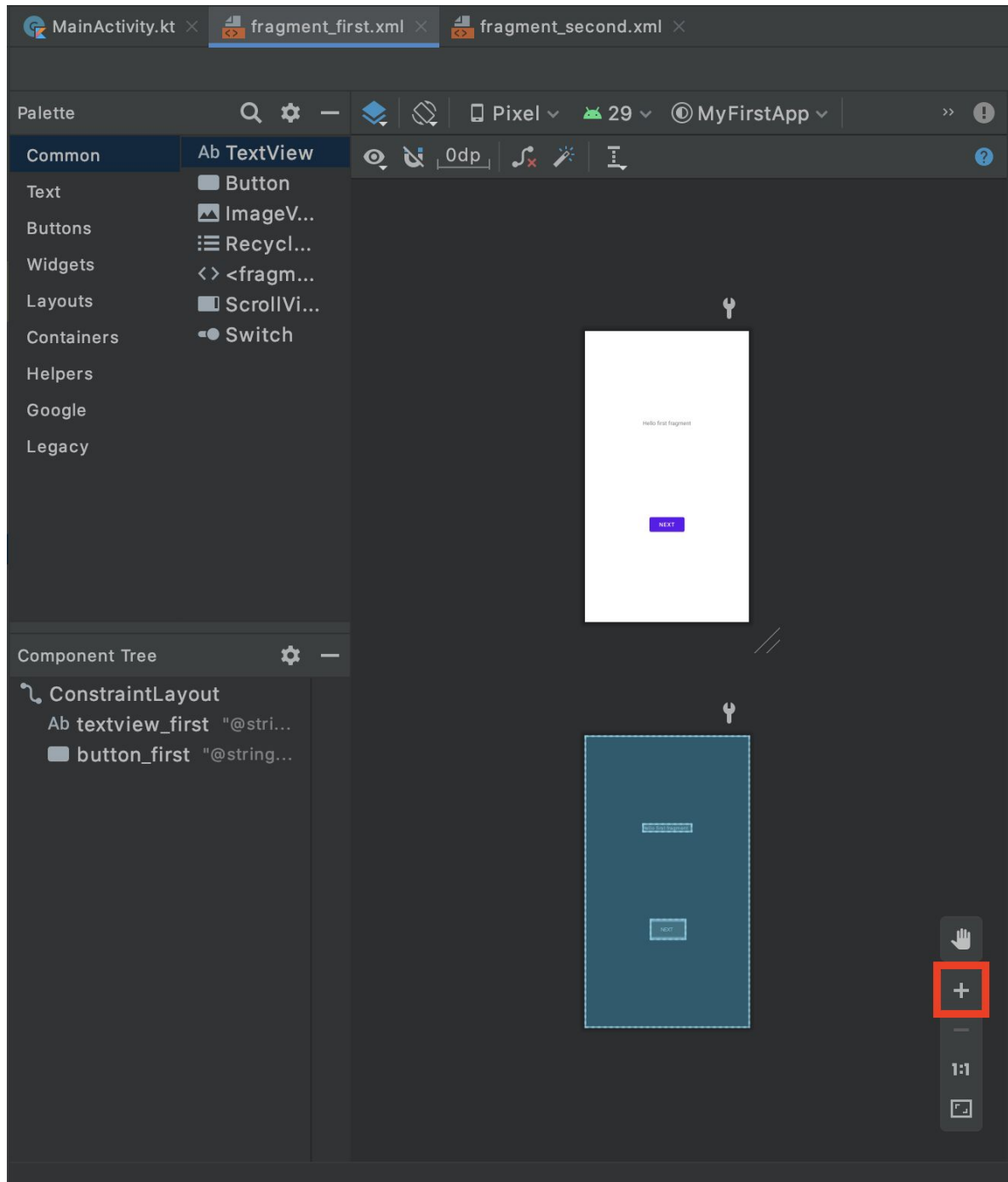
- 1.7. Остановите приложение кнопкой **Stop**, расположенной справа вверху:



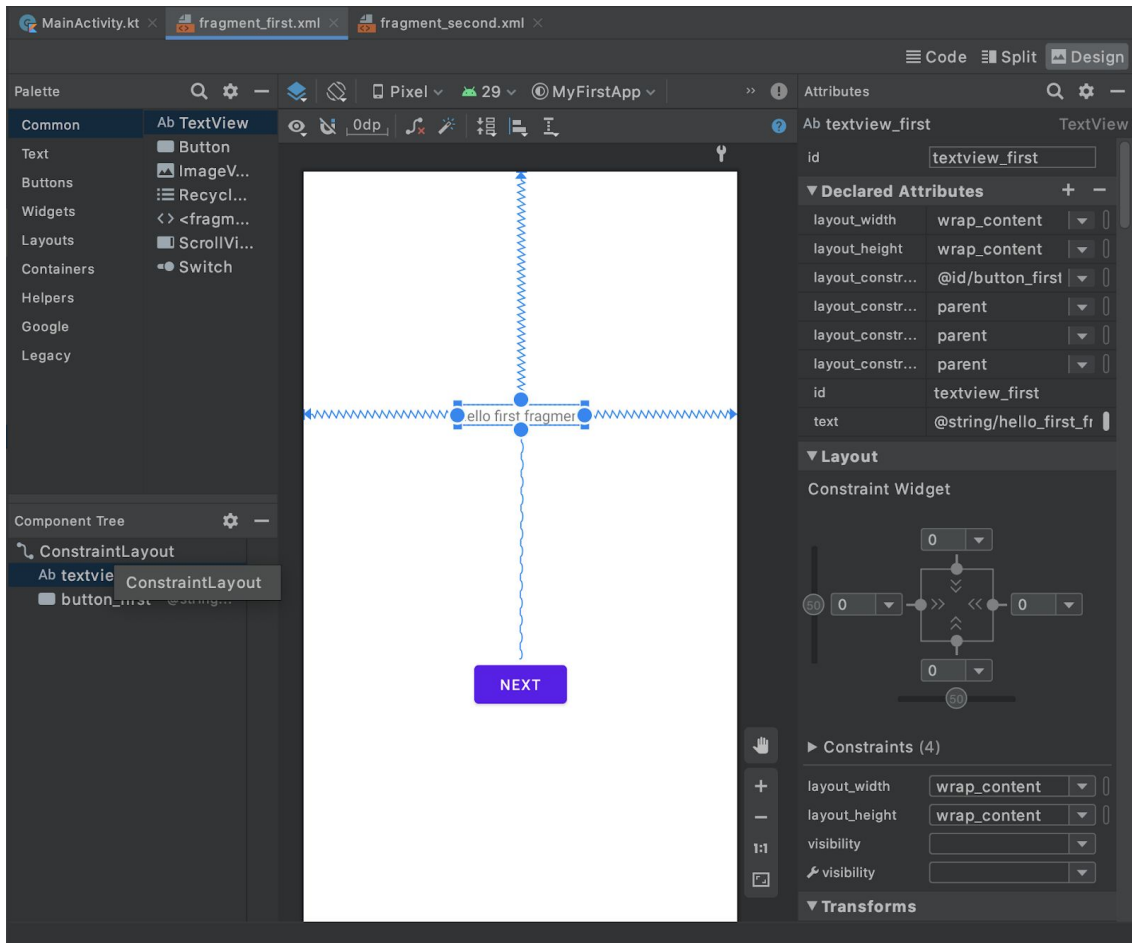
- 1.8. Откройте в списке файлов слева папку **res**, в ней папку **layout** и в ней кликните дважды по файлам **fragment_first.xml** и **fragment_second.xml**:



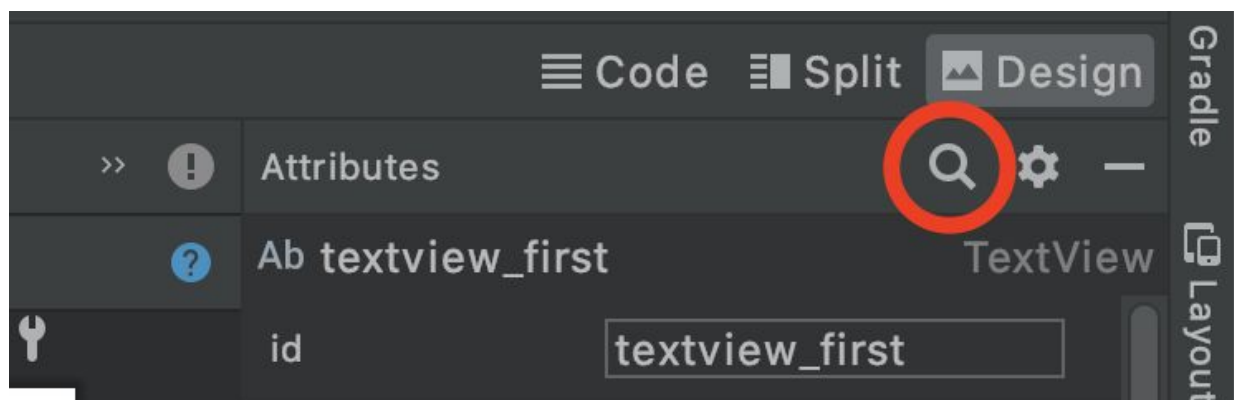
- 1.9. Справа откроются вкладки с редактором экранов приложения. Выберите вкладку **fragment_first.xml** и увеличьте изображение экрана, нажав на «+»:



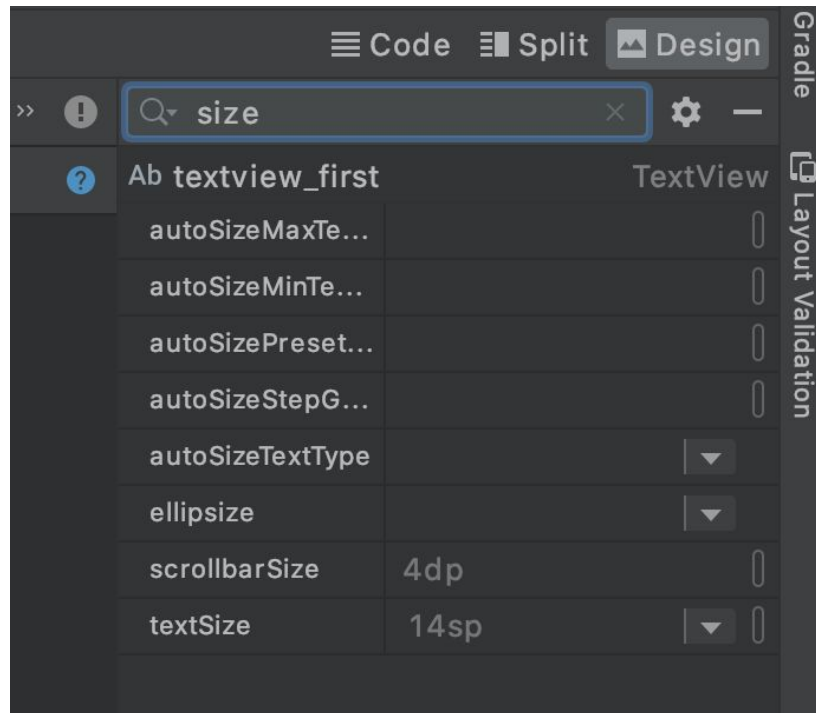
- 1.10. Нажмите дважды на надпись **Hello first fragment**, чтобы она выделилась, а справа открылся список её свойств:



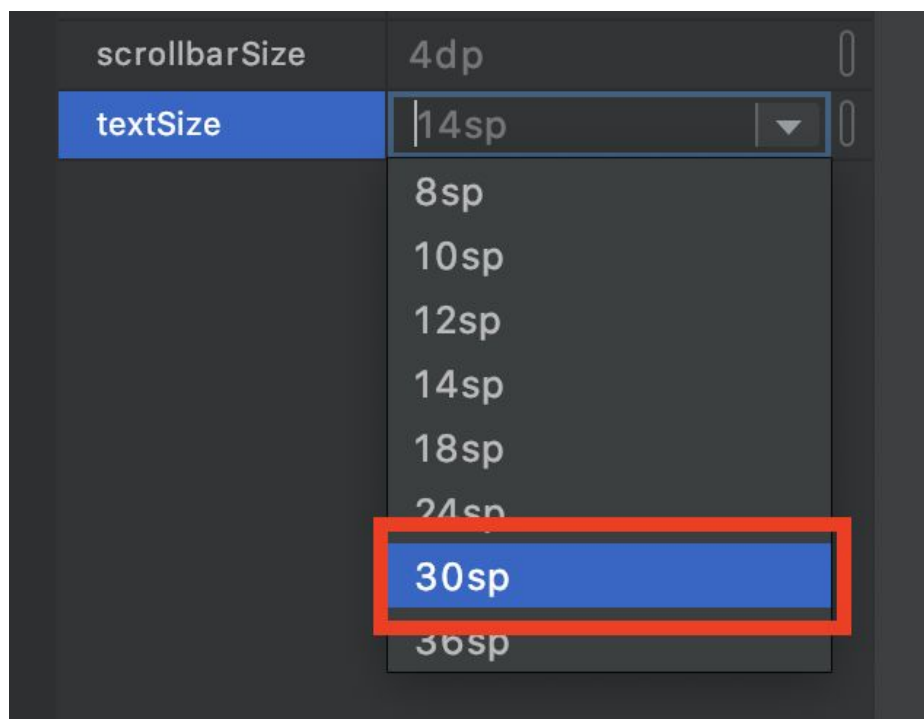
- 1.11. Нажмите справа вверху на иконку с лупой (кнопку **Search**):



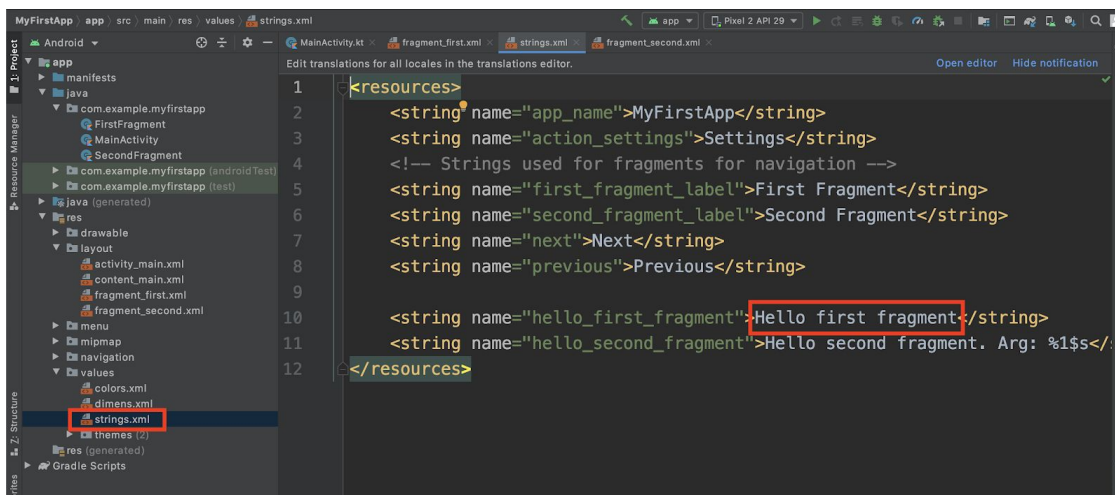
- 1.12. Введите в появившемся поле слово **size**. Отобразится список свойств, содержащих в своём названии слово **size**:



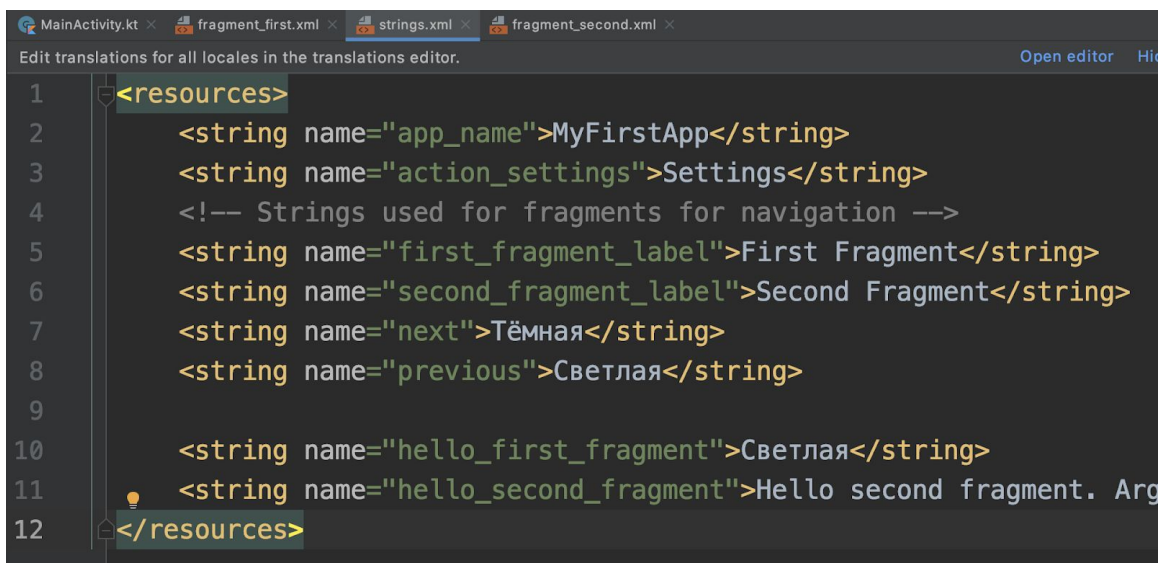
- 1.13. Измените свойство **textSize** на **30sp**, нажав на стрелку слева и выбрав соответствующее значение в выпадающем списке:



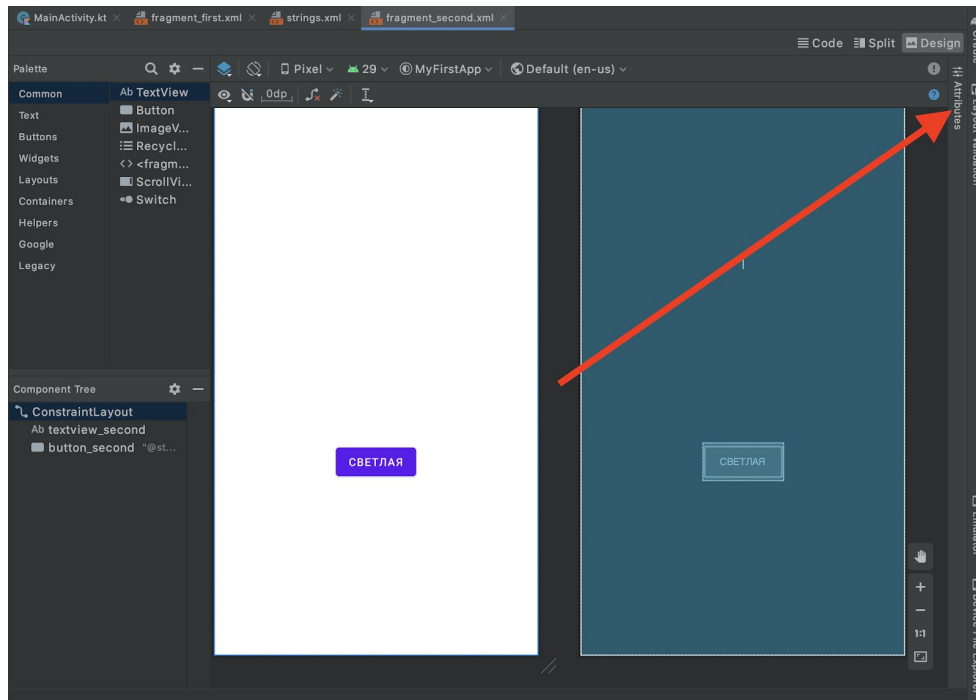
- 1.14. Надпись **Hello first fragment** на экране увеличится. Откройте слева в папке **values** файл **strings.xml** и измените эту надпись на любую другую в появившемся справа файле:



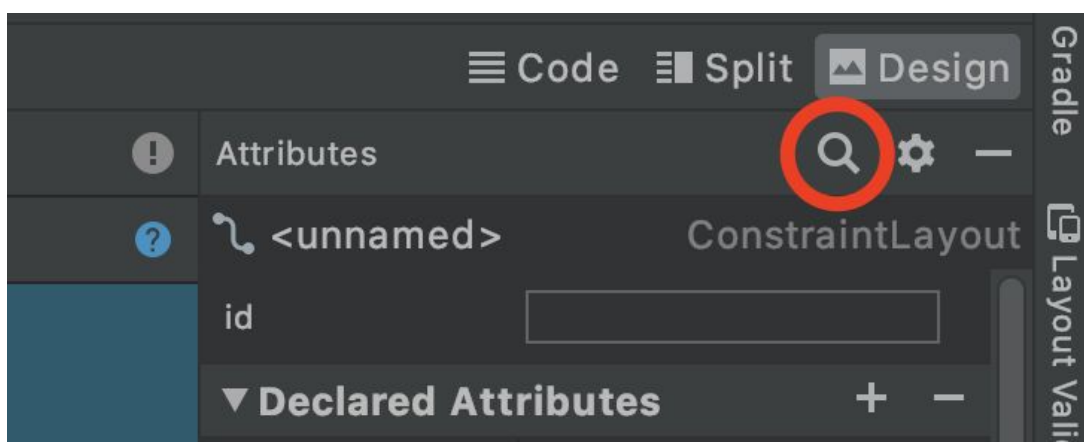
- 1.15. В этом же файле измените тексты, которые отображаются на кнопках **Next** и **Previous** — например, так:



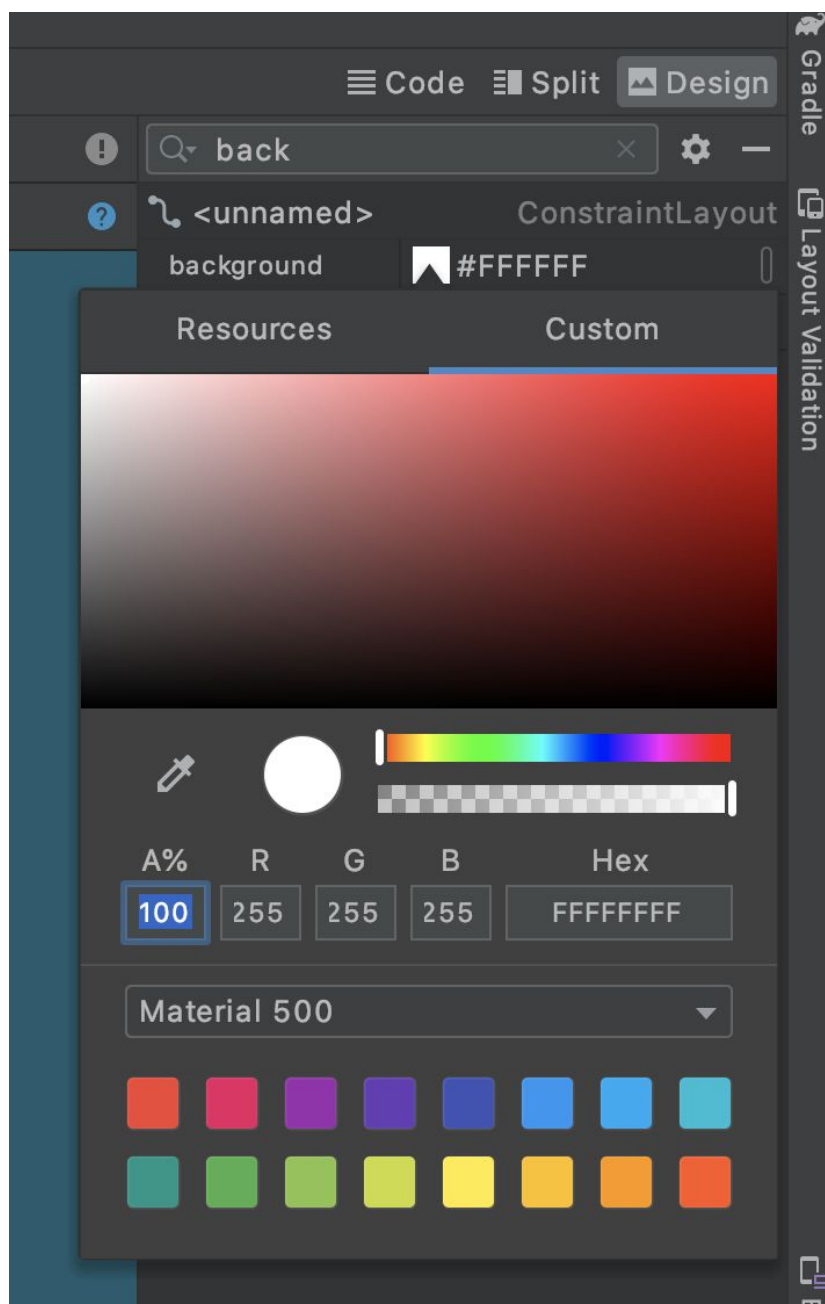
- 1.16. Перейдите во вкладку **fragment_second.xml**, увеличьте экран, нажав «+» справа внизу; нажмите на экран и откройте панель с его свойствами, нажав справа вверху на вертикальную надпись **Attributes**, если панель ещё не открыта:



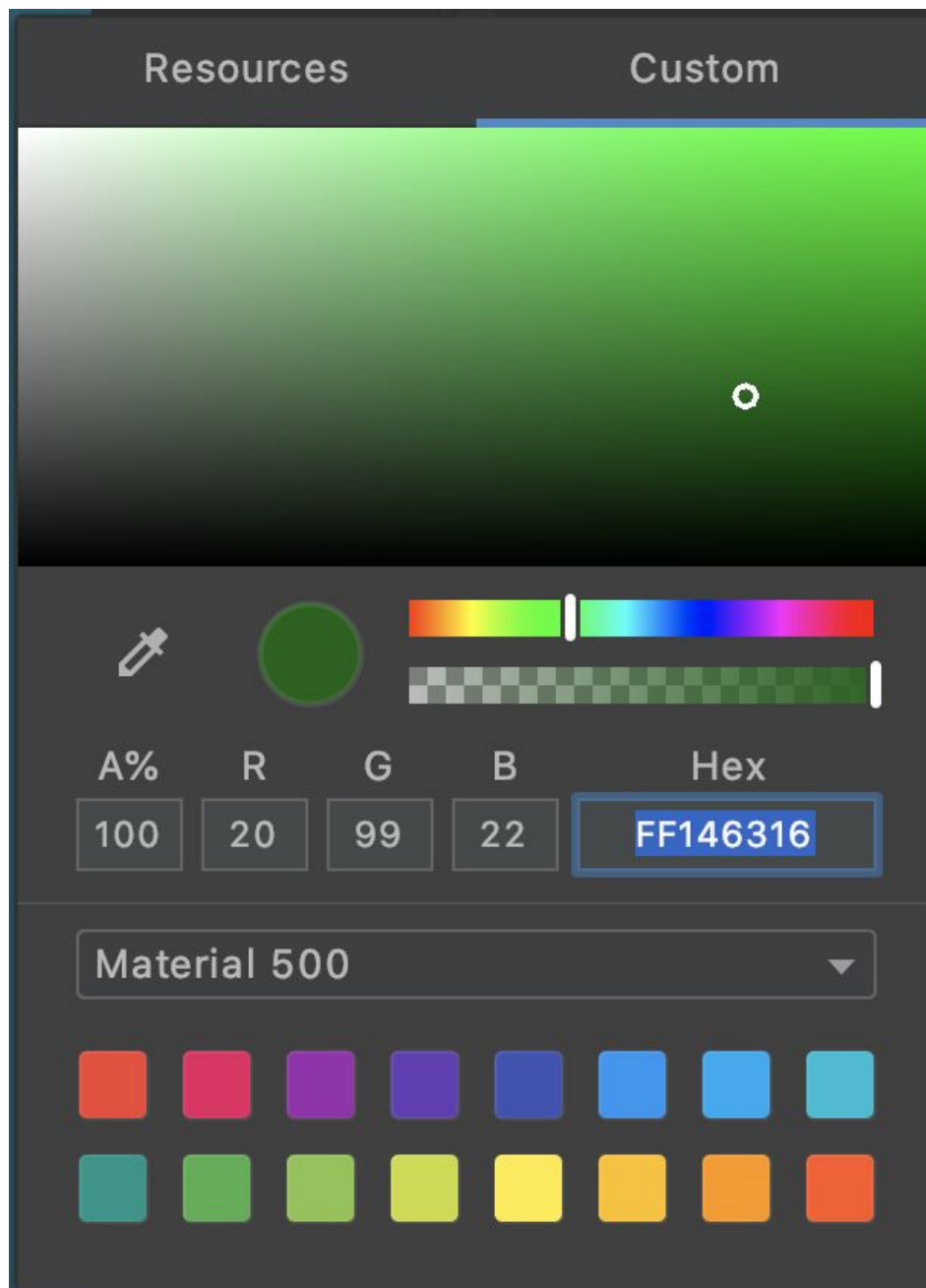
- 1.17. Нажмите справа на иконку с лупой (кнопку **Search**):



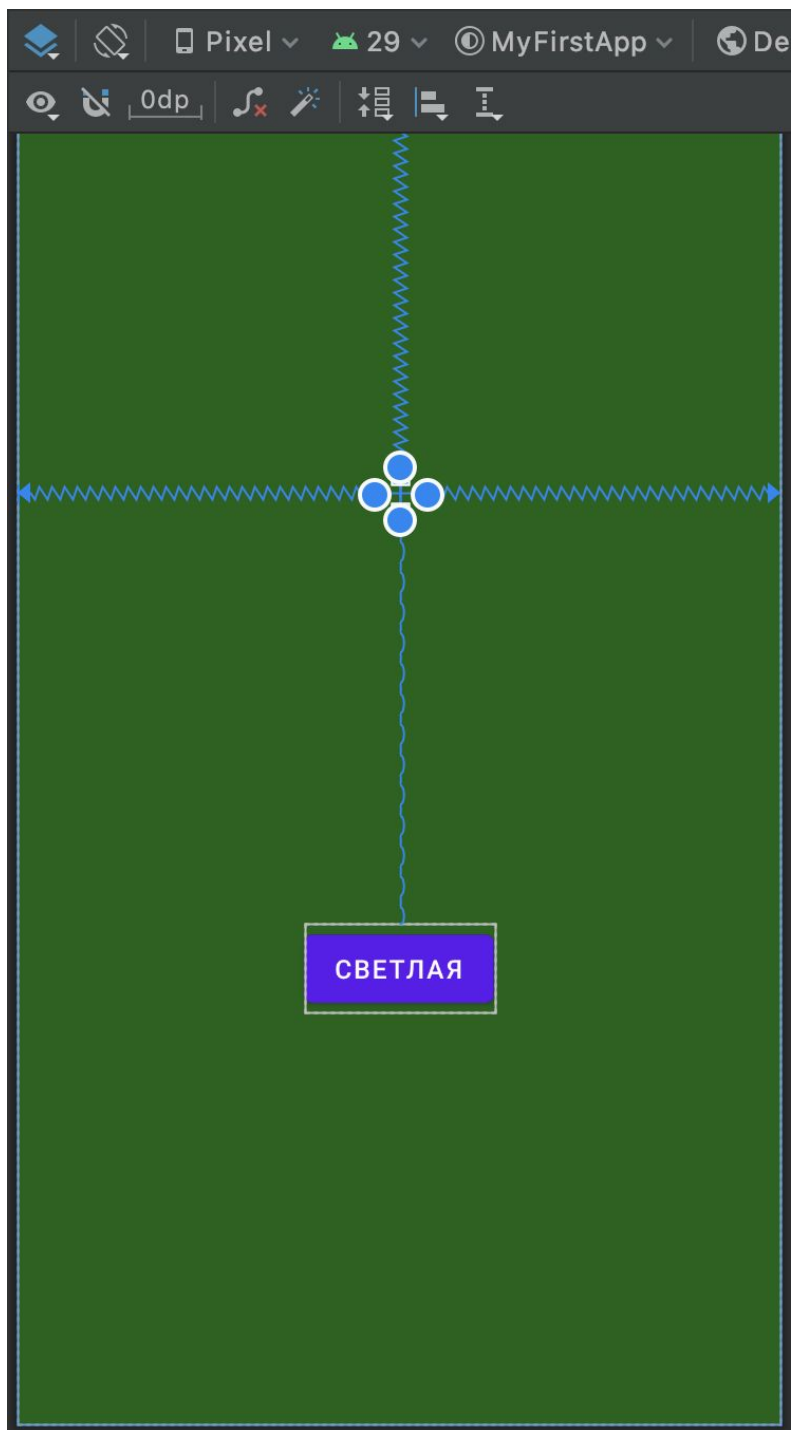
- 1.18. Введите в появившееся поле слово **back** и нажмите на пипетку рядом со свойством **background**. Откроется окно выбора цвета:



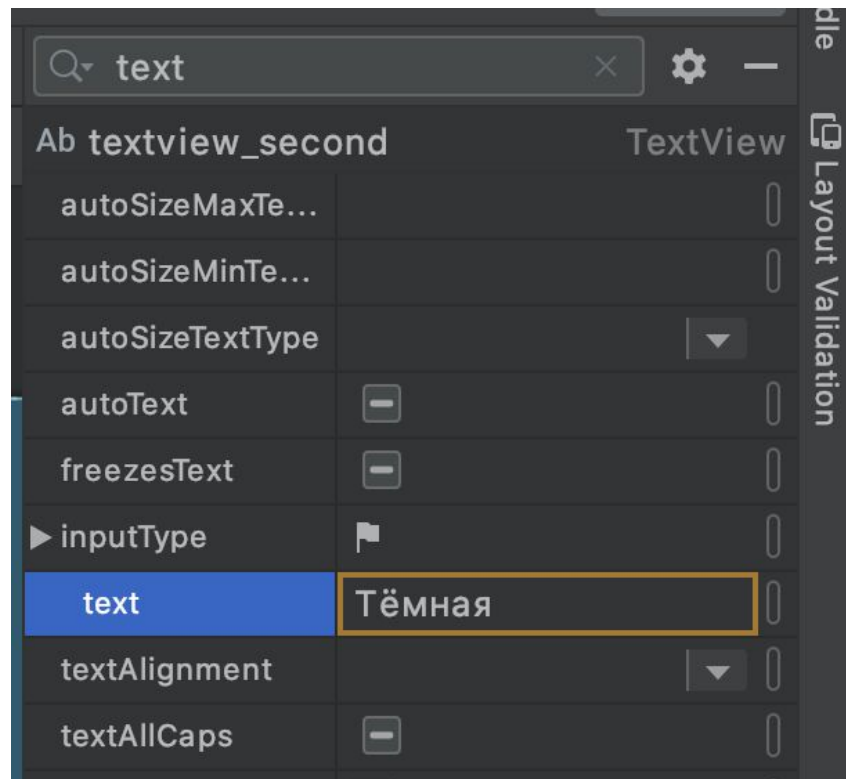
- 1.19. Выберите любой цвет фона экрана, который вам нравится, нажимая на цветное поле и двигая ползунок выбора цвета. Это будет условно тёмная тема. Запишите или скопируйте куда-нибудь код цвета, который вы выбрали. В примере ниже это цвет с кодом FF146316:



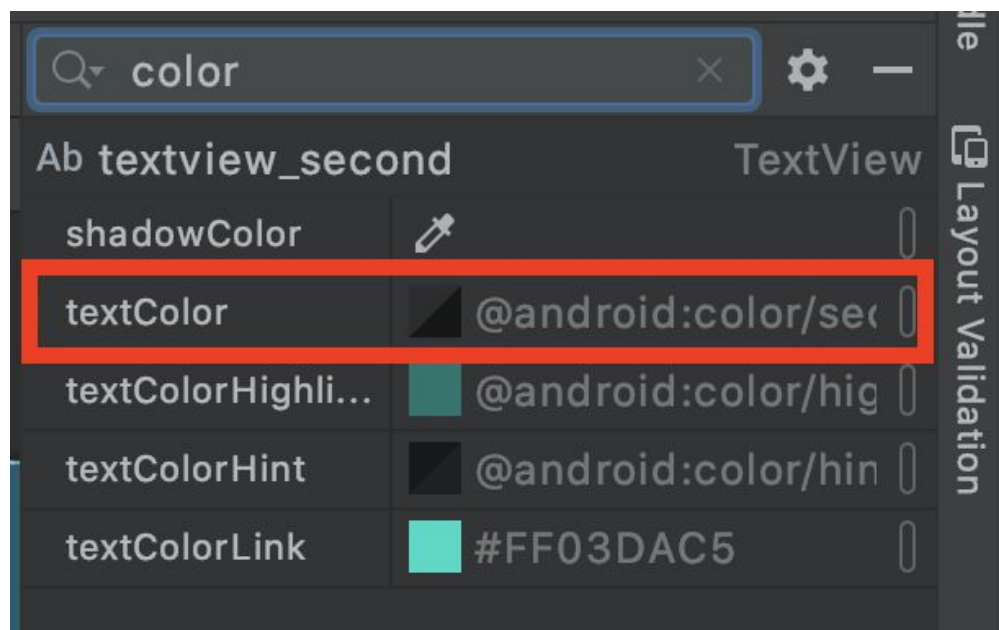
- 1.20. Выделите на экране элемент, на месте которого будет находиться надпись:



- 1.21. Справа найдите свойство **text** (также через поиск атрибутов, аналогично пунктам 1.11 и 1.17). В поле введите текст, который необходимо отобразить на надписи, например, «Тёмная»:



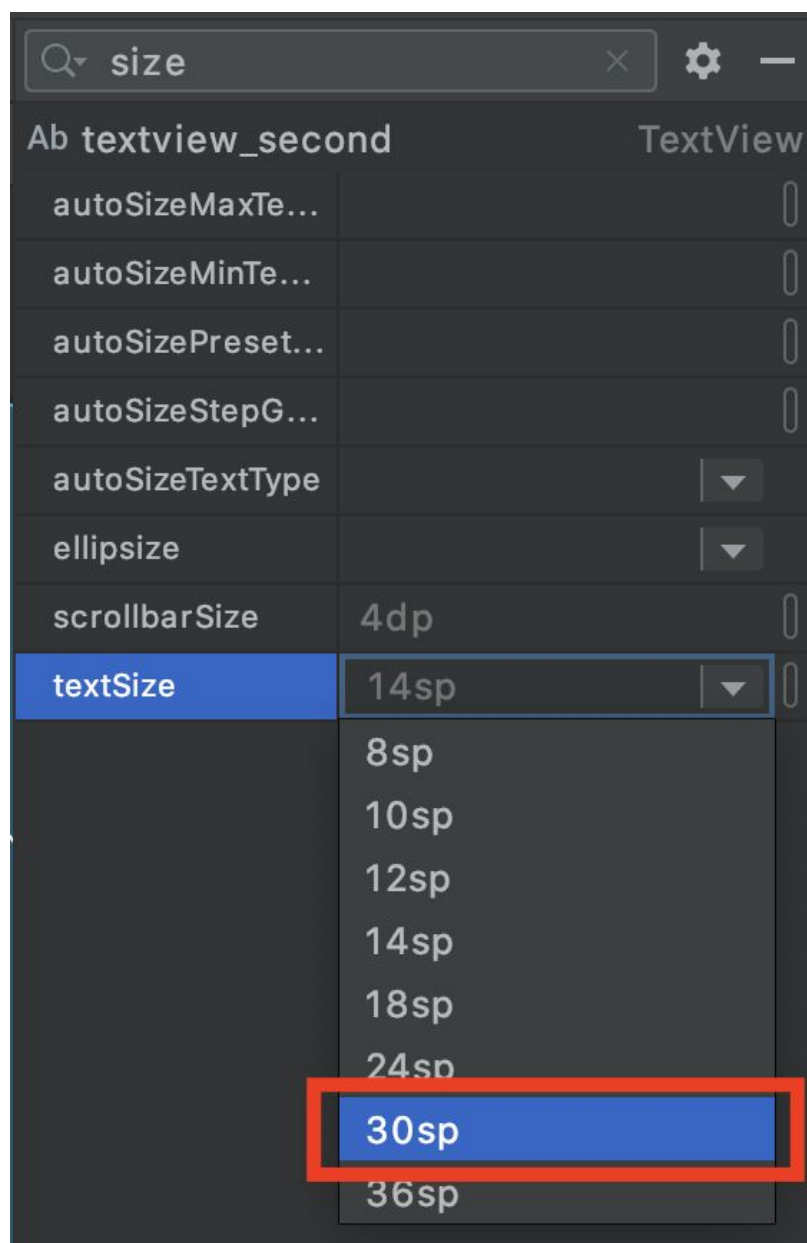
- 1.22. Введите в поиске свойств **color** и нажмите на поле выбора цвета:



- 1.23. Выпадет окно выбора цвета, и по умолчанию будет выбран белый. Нажмите на любой элемент за пределами окна выбора цвета, чтобы остался выбранным белый цвет надписи:



- 1.24. Наберите в строке поиска свойств **size**, найдите и увеличьте шрифт надписи (свойство **textSize**) до 30sp:



- 1.25. Запустите приложение (см. пункт 1.5) и убедитесь, что оно работает верно: кнопка на первом экране переключает на второй экран тёмного цвета, который вы выбрали, а кнопка на втором экране — возвращает обратно на первый экран; тексты на кнопках и тексты надписей на экранах соответствуют тому, что вы написали.
- 1.26. Поздравляем! Вы создали своё первое Android-приложение! Вы — молодец! :)
- 1.27. По вашему желанию вы можете поэкспериментировать с приложением: добавить на экран новые элементы, попробовать изменять их свойства и расположение.

Задание 2. Изменение кода на языке программирования Kotlin:

- 2.1. Откройте код по [ссылке](#).
- 2.2. Нажмите в левом верхнем углу ссылку **fork**:



- 2.3. Откроется ваша версия программного кода.
- 2.4. По [ссылке](#) — детально расписаны системы штрафов для Германии. Вам необходима **таблица со значениями штрафов для населённых пунктов**.

- 2.5. Измените в открытом коде переменные, в которых заданы суммы штрафов, таким образом, чтобы они соответствовали штрафам в Германии. Например, в России минимальный штраф за превышение скорости на 20–40 км/ч составляет 500 рублей. И в исходном коде сейчас написано:

```
var fineFor20to40 = 500
```

В Германии минимальный штраф составляет 15 евро и он выписывается при превышении скорости на 4–10 км/ч, следовательно, в коде вместо строки `var fineFor20to40 = 500` нужно написать:

```
var fineFor1to10 = 15
```

Измените не только суммы штрафов, но и переименуйте соответствующие переменные, а лучше удалите старые и напишите новые по аналогии.

- 2.6. Измените максимально разрешённую скорость в населённых пунктах (переменную `townSpeed`) на 50 км/ч.

- 2.7. Ниже в коде перепишите условия, проверяющие, на сколько превышена скорость. Например, условие, отвечающее за выписывание минимального штрафа, которое в изначальном коде выглядит так:

```
else if (overSpeed >= 20 && overSpeed < 40) {  
    print("Штраф: " + fineFor20to40)  
}
```

необходимо переписать так:

```
else if (overSpeed >= 1 && overSpeed <= 10) {  
    print("Штраф: " + fineFor1to10)  
}
```

Перепишите все остальные условия аналогичным образом. Обратите внимание, что условий для системы штрафов в Германии больше, чем для России.

- 2.8. После переписывания кода измените переменную **carSpeed** на 78 и запустите программу, нажав слева сверху кнопку **Run** (см. пункт 2.2).

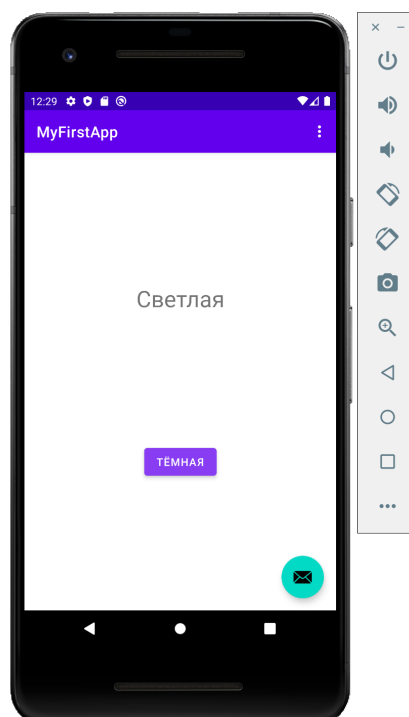


После запуска программы через несколько секунд внизу должно появиться сообщение с суммой штрафа 100 евро.

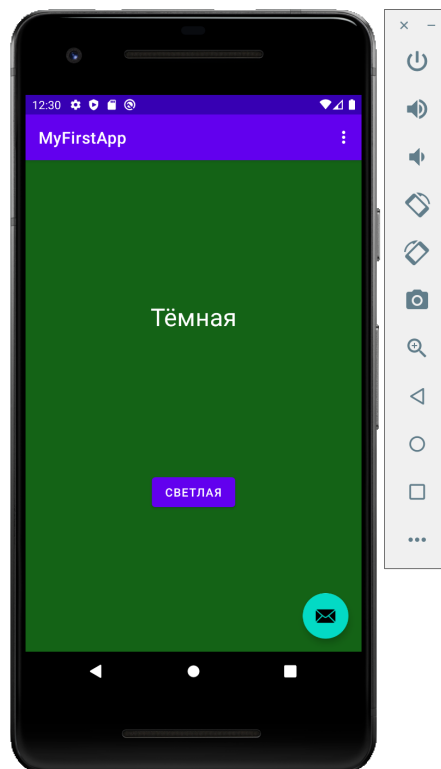
Как проверить результат

Проверьте результаты самостоятельно. Главное, чтобы вы сами видели, что приложение и код работают верно.

1. Приложение можно считать работающим верно, если при после выполнения инструкций выше оно выглядит так:



А второй — так:



2. Код на Kotlin можно проверить подстановкой различных значений переменных и сравнением конечного результата со значениями в таблице:

Значение переменной carSpeed	Штраф, евро
57	15
73	80
96	200
114	480
145	680