**ЛАБОРАТОРНАЯ РАБОТА № 4**

**СТРОКОВЫЕ И ЦВЕТОВЫЕ РЕСУРСЫ**

**Цель работы:** Создать приложение с применением строковых и цветовых ресурсов.

Изменим файл res/values/strings.xml следующим образом:

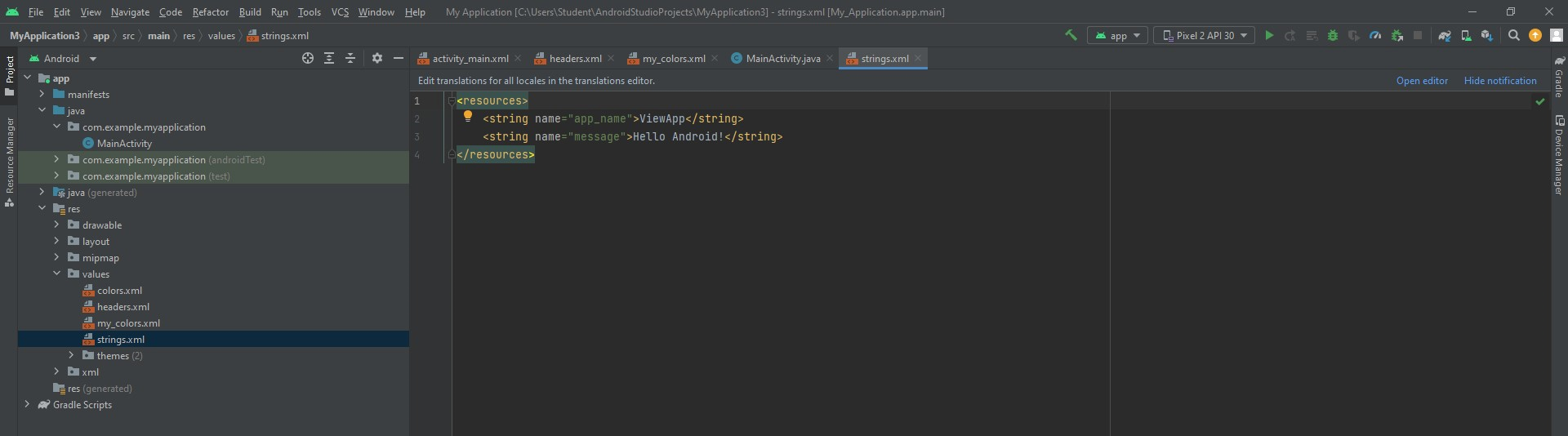


Рисунок 4.1 - изменение файла string.xml

С помощью выражения @string/message передаем атрибуту android:text значение из ресурса.

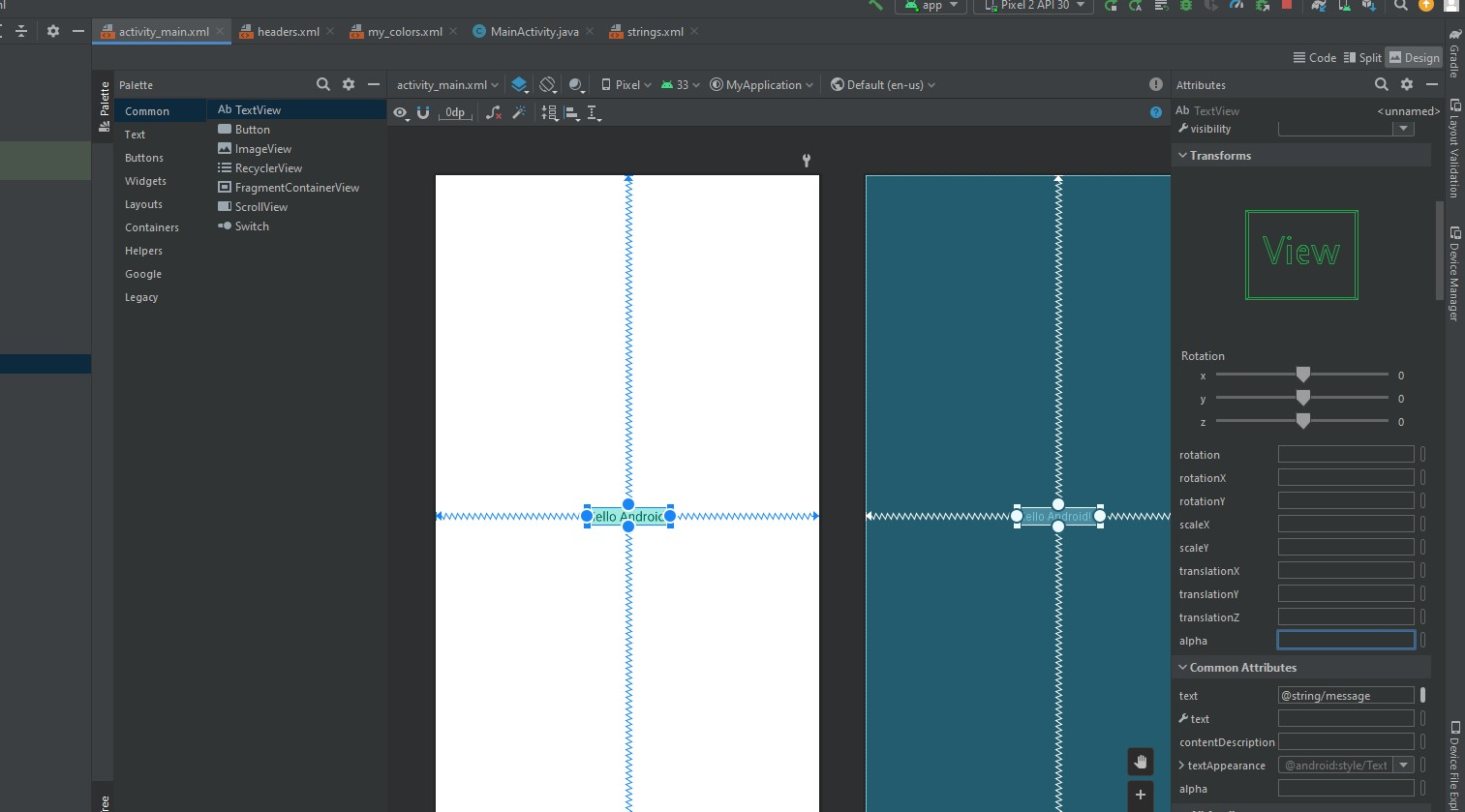


Рисунок 4.2 - передача атрибуту значения из ресурса

Так, нажмем на папку res/values правой кнопкой мыши и в появившемся списке выберем пункт New -> Value Resource File.

Данный тип файлов будет характерен для любого типа ресурсов, который добавляется в папку res/values. После этого определяем для файла имя: Назовем файл headers(заголовки для страниц). Для всех остальных полей оставляем значения по умолчанию. И в папку res/values будет добавлен новый файл headers.xml.

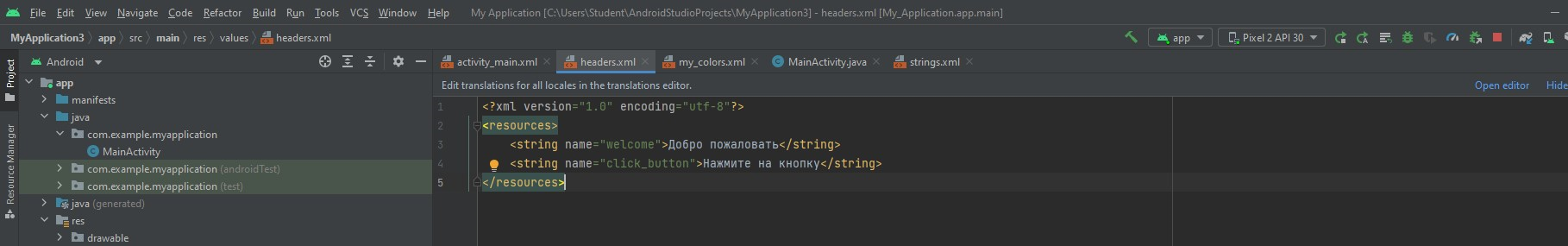


Рисунок 4.3 - создание и изменение файла headers.xml

После этого мы сможем использовать определенные здесь ресурсы в коде Activity или в файле layout.

В приложении Android также можно определять ресурсы цветов (Color). Они должны храниться в файле по пути res/values и, также, как и ресурсы строк, заключены в тег . Так, по умолчанию при создании самого простого проекта в папку res/values добавляется файл colors.xml.

Цвет определяется с помощью элемента . Его атрибут name устанавливает название цвета, которое будет использоваться в приложении, а шестнадцатеричное число - значение цвета.

Определим свой новый файл ресурсов и для этого добавим в папку res/values новый файл ресурсов, который назовем my\_colors.xml.

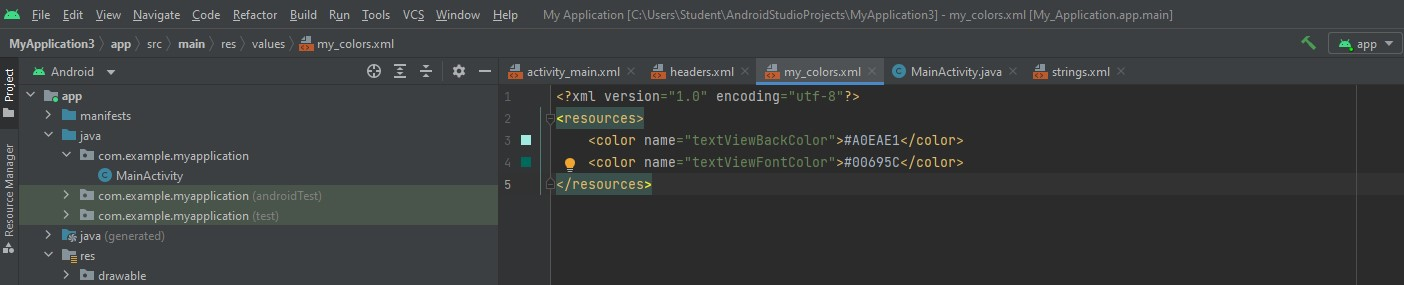


Рисунок 4.4 - создание и изменение файла my\_colors.xml

С помощью атрибута android:textColor устанавливается цвет текста в TextView, а атрибут android:background устанавливает фон TextView. В качестве значения они используют цвет, например, в том же шестнадцатеричном формате. Для получения самого цвета после "@color/" указывается имя ресурса.

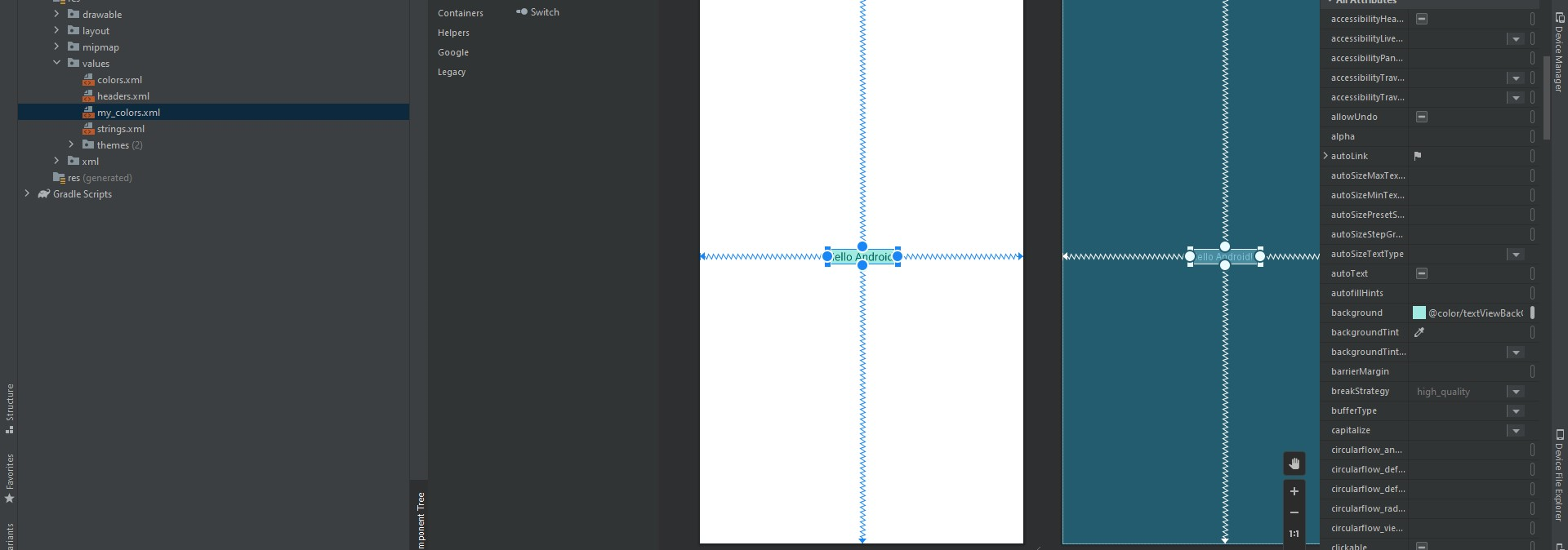


Рисунок 4.5 - изменение цвета фона и текста

Все цвета из папки res проекта подгрузятся автоматически и будут доступны в окне выбора цвета.

Также можно использовать цветовые ресурсы в коде MainActivity:

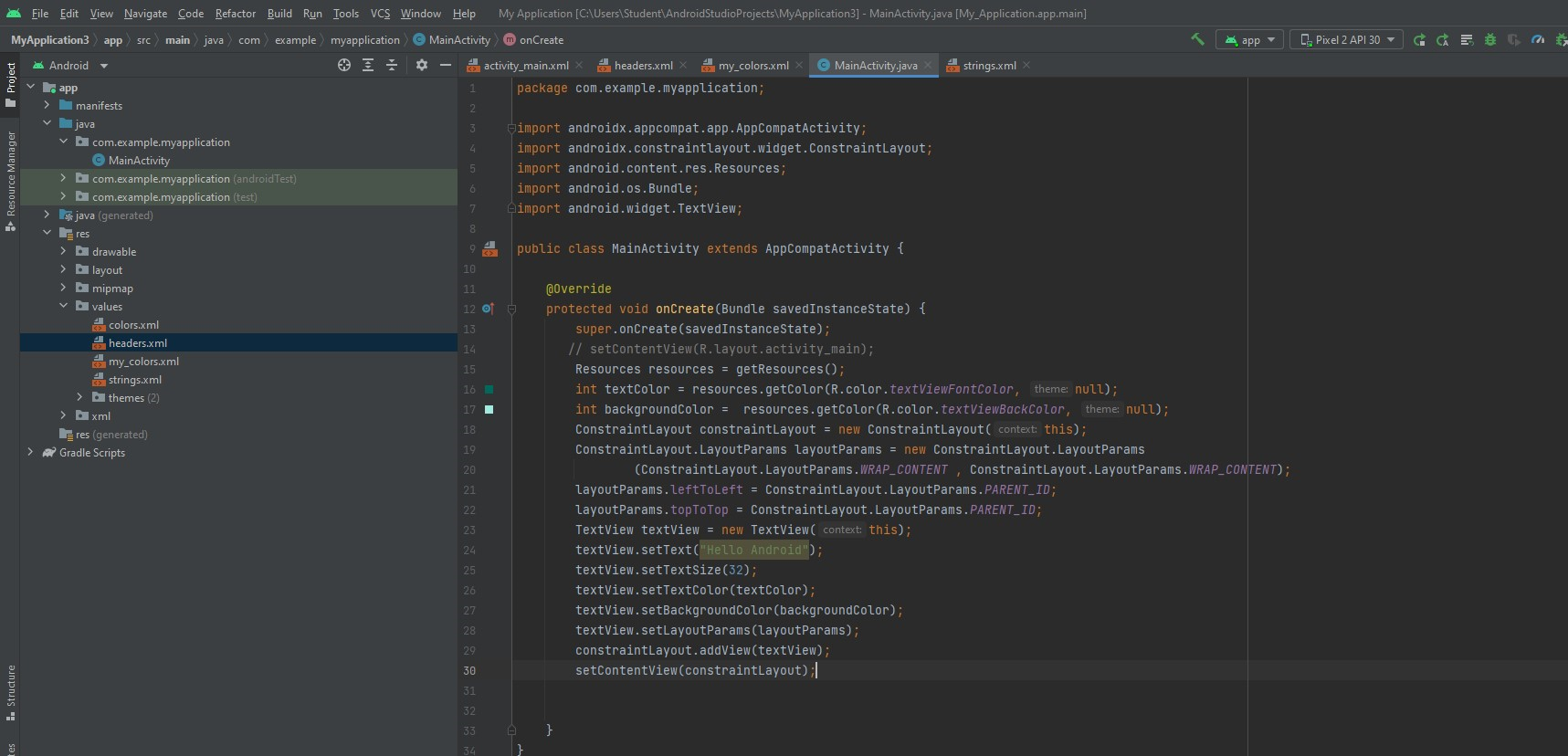


Рисунок 4.6 - использование цветовых ресурсов в MainActivity.java

Для получения цвета из ресурсов применяется метод resources.getColor(), который принимает два параметра. Первый параметр - идентификатор ресурса, цвет которого надо получить. Второй параметр представляет тему. Но поскольку в данном случае тема не важна, для этого параметра передаем значение null.

Следует учитывать, что метод resources.getColor() с двумя параметрами, который использован выше, доступен, если для минимальная версия Android не ниже Android 6.0 (или Android 23).

Итоговый файл activity\_main.xml:

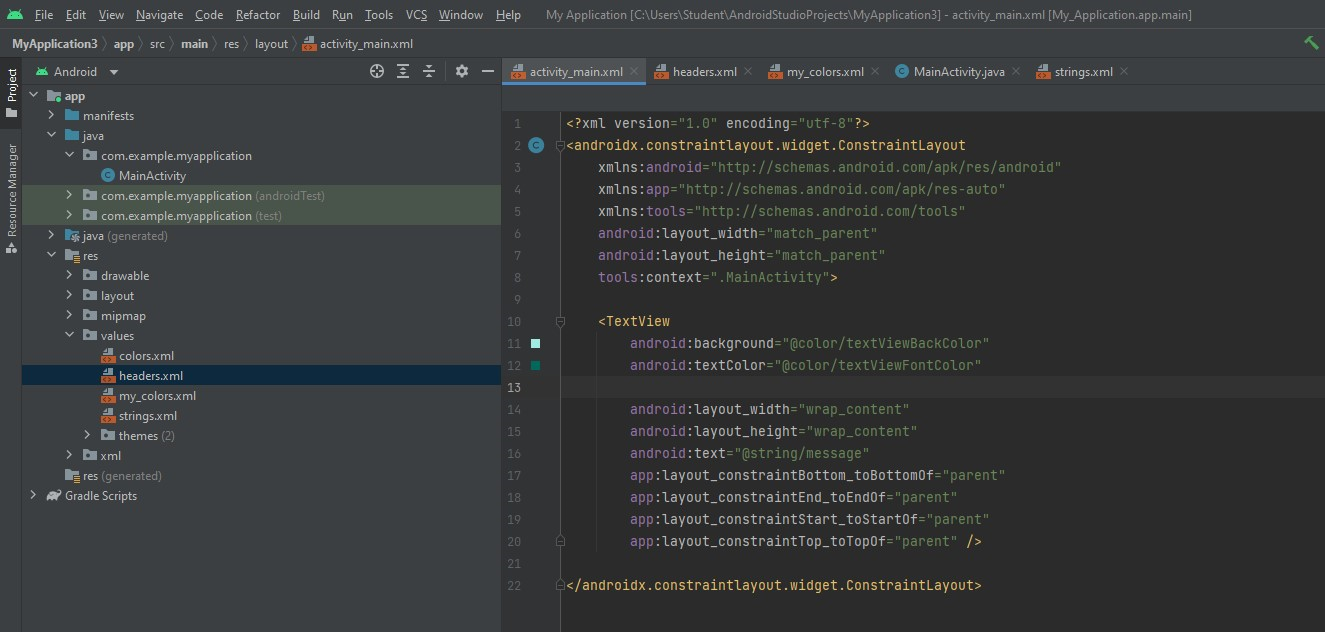


Рисунок 4.7 - изменение файла activity\_main.xml

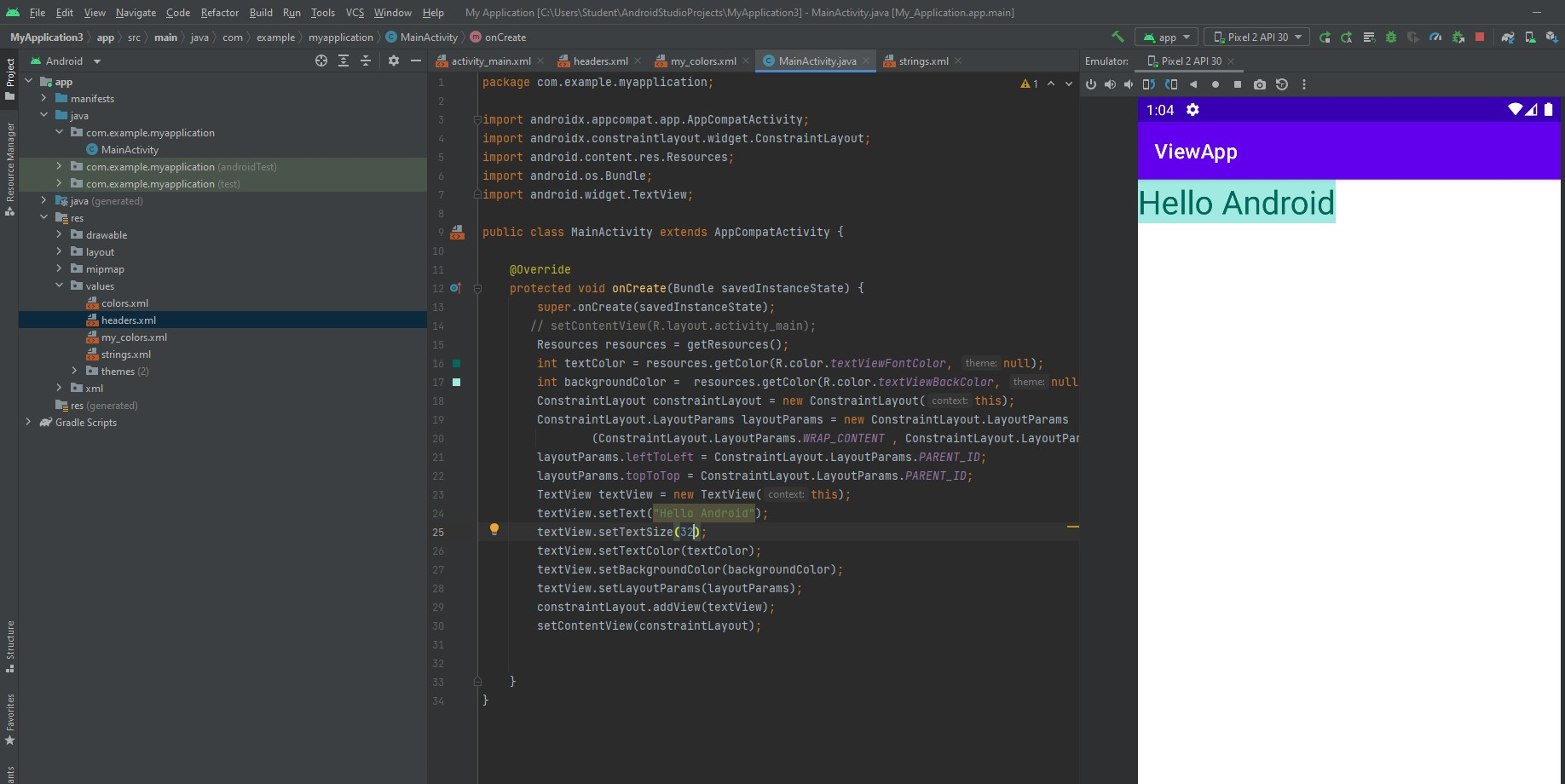


Рисунок 4.8 - результат

Вывод: в ходе выполнения лабораторной работы было создано приложение с применением строковых и цветовых ресурсов.

|  |  |
| --- | --- |
| Выполнил | Семеняк В.А. |
| Проверил | Елкин Н.С. |