## Работа с графическими элементами

### Цель работы

Освоить работу с базовыми визуальными элементами Android Studio и способы организации архитектуры приложения на Android.

### Задание

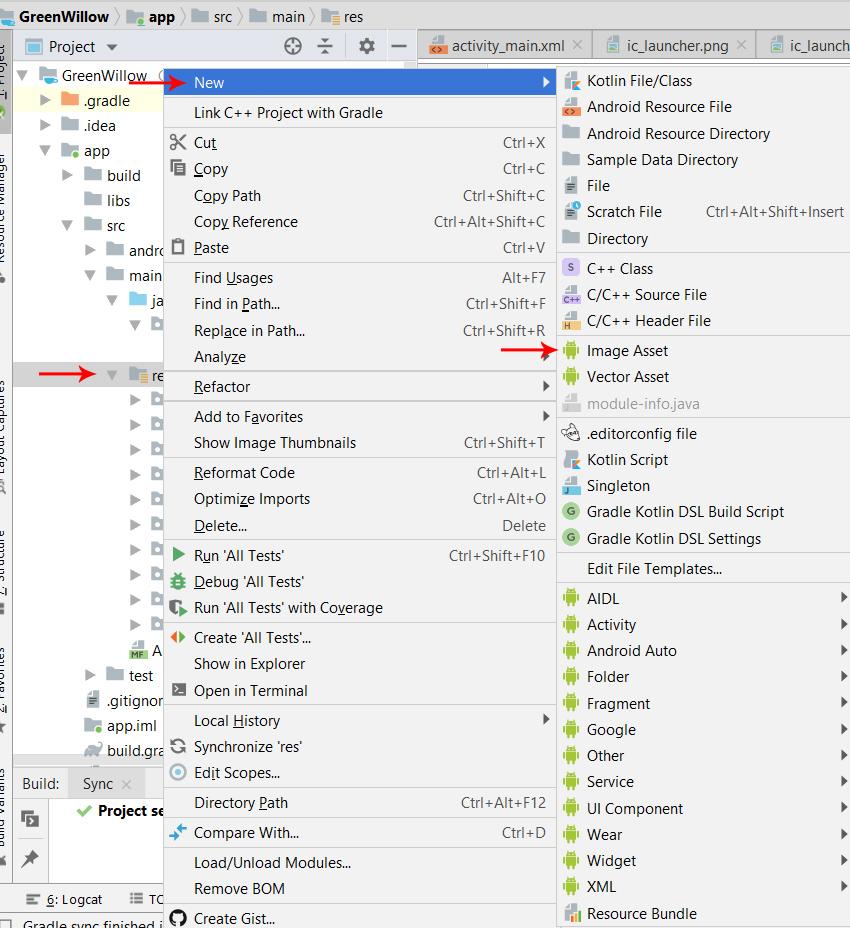
На основе лабораторных работ «MD1.1 Основные элементы» и «MD2.2 Фильтры намерений» реализуйте приложение «Список дел», состоящее из нескольких экранов. На главном экране реализована навигация по существующим категориям. При переходе на каждую категорию реализовать возможность добавления новых дел.

При оформлении приложения не оставляйте стандартного оформления, поменяйте цвета навигации.

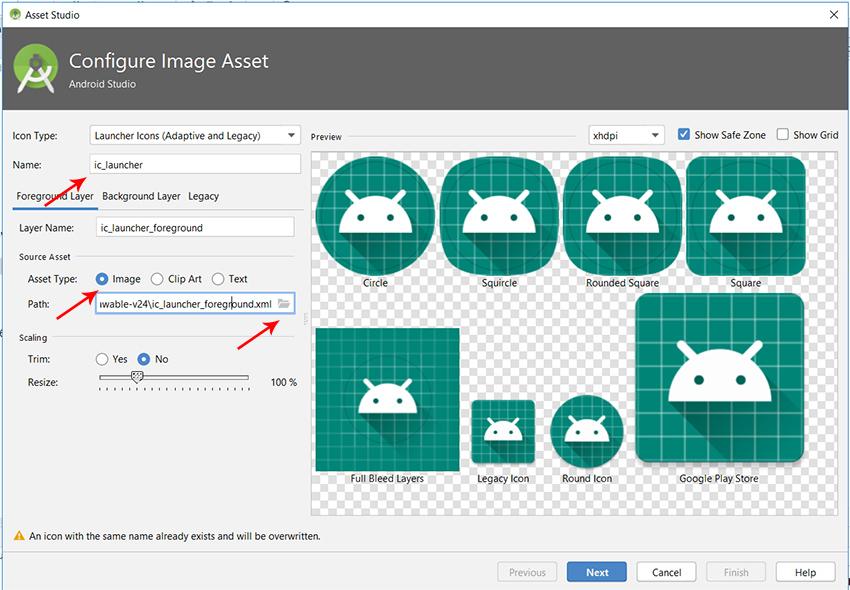
Также поменяйте иконку приложения.

Создать иконку можно в Figma (512 х 512 px): <https://www.figma.com/>

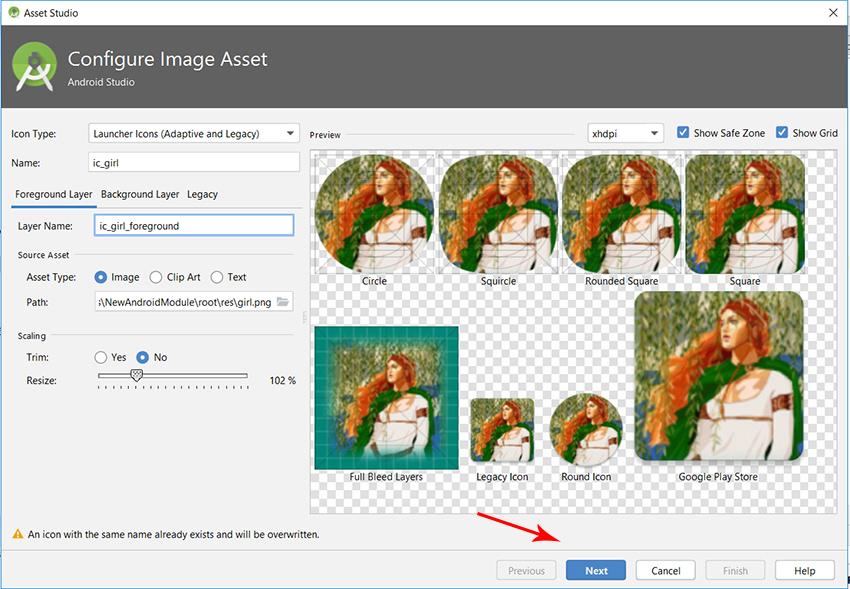
Для того чтобы добавить своё изображение в проект, в программе Android Studio, в каталоге Вашего проекта, найдите путь: app > src > res и вызовите контекстное меню (правой кнопкой мыши):



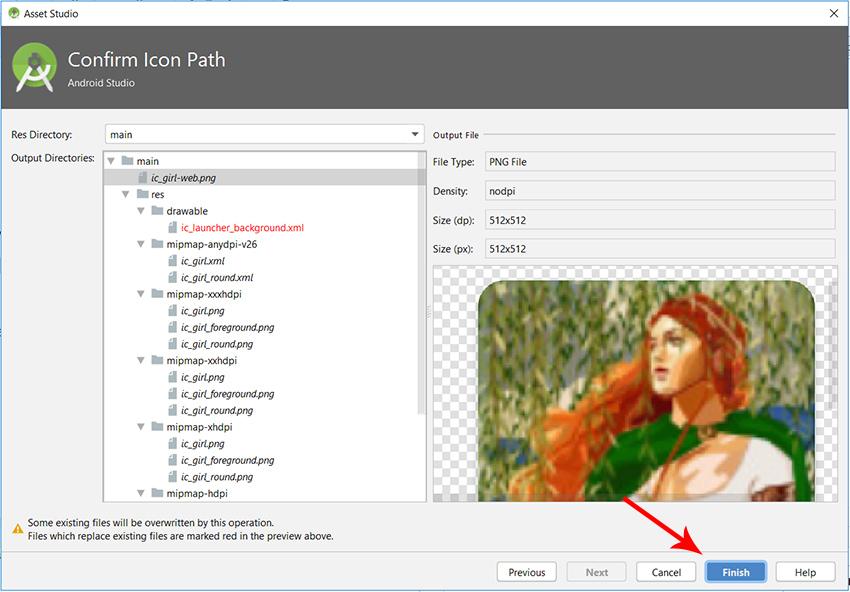
Здесь, как можно увидеть на скриншоте, мы выбираем: New > Image Asset. И попадаем во вкладку, где необходимо указать путь к изображению, задать ему имя и выбрать тип (в нашем случае Image):



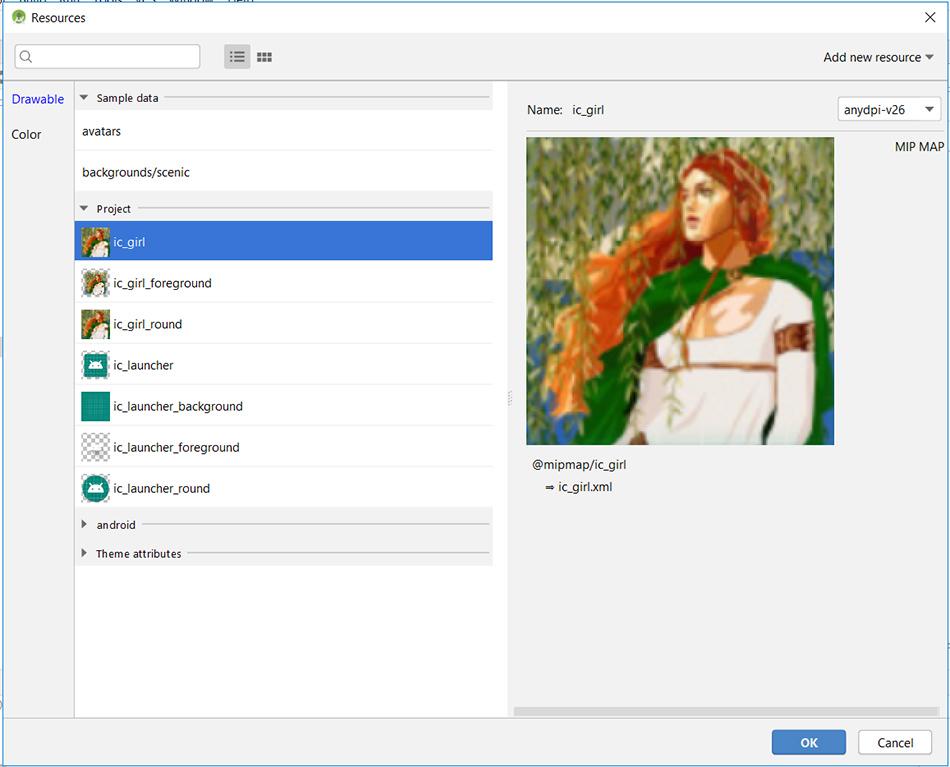
Изображение, это файл с расширением \*.png, с разрешением 512х512 точек. Для примера, я взял png с разрешением 256х256, чтобы можно было почувствовать разницу. После чего нажимаем Next.

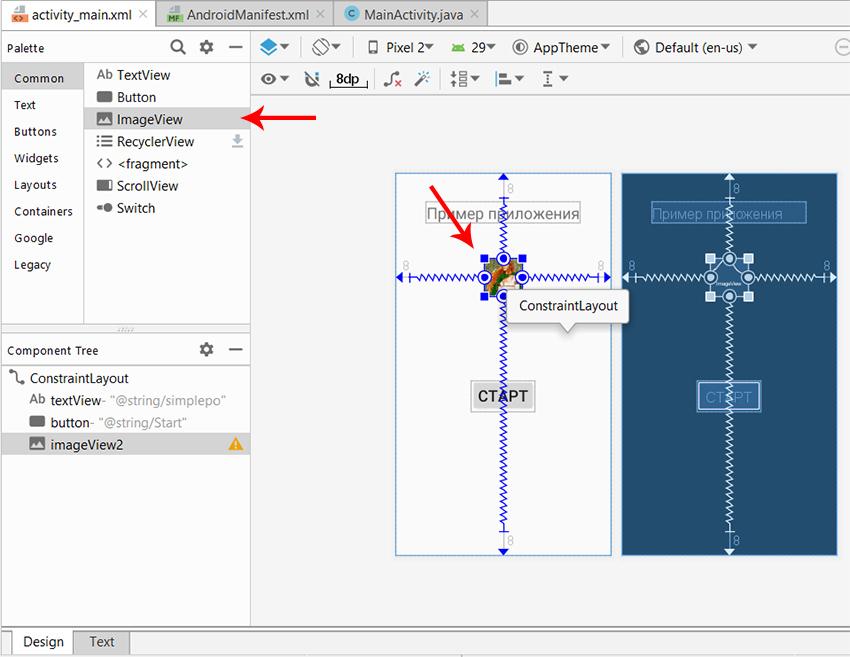


После чего мы видим сообщение о том, что будет заменено текущее изображение загрузчика.

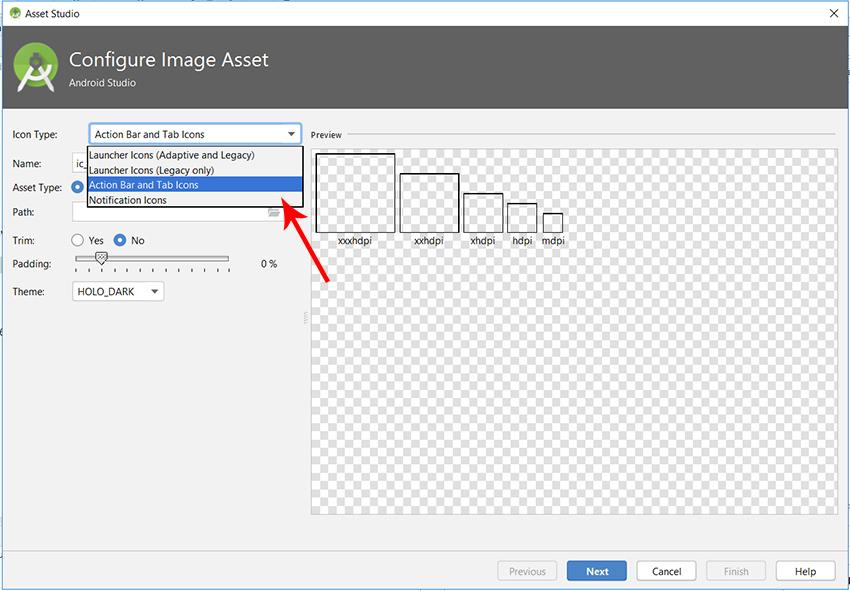


Кстати текущее изображение, можно загрузить и на рабочую область.





Здесь как раз отчётливо видна разница между изображениями 256х256 и 512х512. И кстати, если Вам необходимо добавить изображение для Вашего проекта, то выбрав **New > Image Asset**, укажите панель **Action Bar and Tab Icons**.

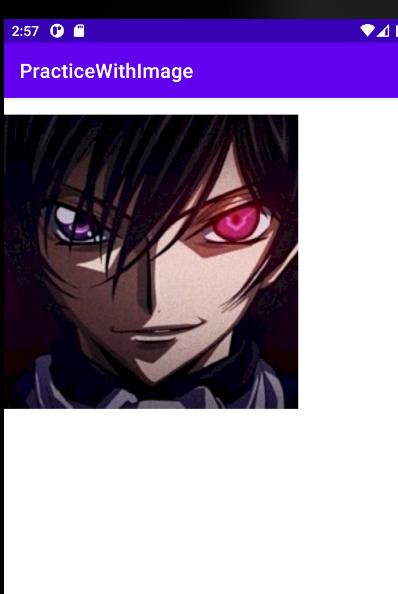
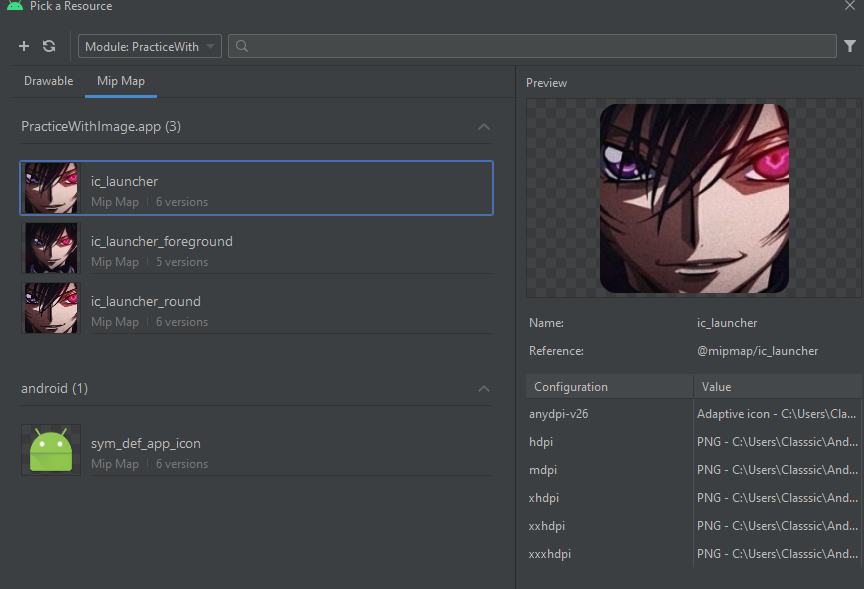


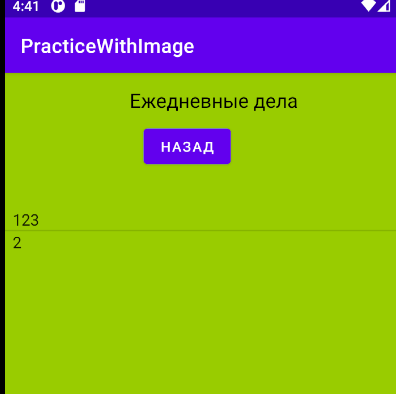
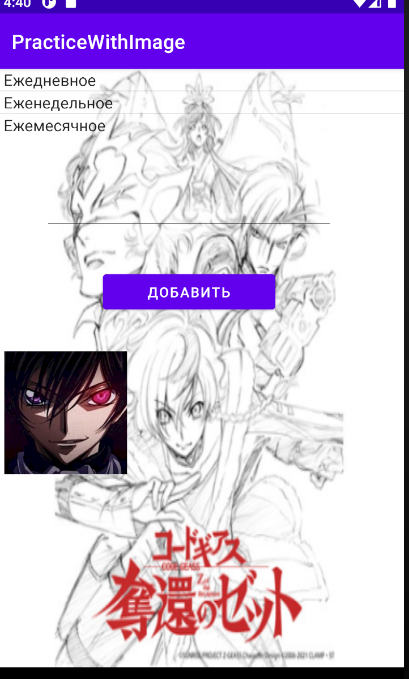
### Контрольные вопросы

1. Что такое событийно-ориентированное программирование?
2. Какие события вы использовали в своем приложении?
3. С помощью какого класса предоставляется доступ к ресурсам из кода Java?
4. Какие основные квалификаторы ресурсов вы знаете?
5. Какие основные XML-атрибуты используются для задания расположения виджета на экране?
6. Какие основные XML-атрибуты используются для задания отображения виджета на экране?
7. Какие существуют соглашения в порядке наименования действий?
8. Как передать информацию в активность используя неявный вызов?
9. Какие еще параметры можно задавать при создании неявного интента?
10. Зачем нужна категория в интент-фильтрах? Какие существуют категории?
11. Зачем нужен элемент <requestFocus>?
12. Зачем нужны аргументы requestCode и resultCode в обратном интенте?
13. Зачем делить приложение на несколько окно? Почему нельзя использовать разные расположения?
14. Что такое интент и зачем он нужен?
15. Как вызвать определенное окно своего приложение? А другого?
16. Что такое таск? Почему при перемещении между окнами работает кнопка “Назад”?

### Дополнительные задания

1. Создайте приложение, на главном окне которого будет расположено поле ввода текста и при нажатии на кнопку “перейти” будет запускаться браузер по введенному пользователем адресу.
2. Создайте приложение, отвечающее на какое-либо стандартное системное действие. Проверьте его работоспособность.
3. Создайте приложение, которое выводит текстовую надпись и предлагает выбрать цвет и выравнивание надписи. Выбор должен производиться в двух разных активностях. При возврате в основную активность форматирование надписи должно меняться.
4. (\*) Создайте приложение, запускающее приложение камеры. Когда пользователь делает снимок, он должен вернуться в наше приложение, и оно должно отобразить его в виде миниатюры на экране.





Изображение выглядит как текст

Автоматически созданное описание