

Молдавский Государственный Университет
Факультет Математики и Информатики
Департамент Информатики

Лабораторная работа Nr.1

Тема: “Создание программ использующих activity”

Проверил; Георгий Латул

Выполнил: Константин Рунтов

Группа: i 1802

Содержание

Содержание	2
Условие задачи	3
Выполнение задачи	3
1. Создание приложения	3
1.1 Создание скелета приложения	3
1.2 Добавление активных элементов	4
1.3 Написание кода приложения	5
2. Работа приложения	7
Вывод	9

Условие задачи

Создать приложение “Светофор”, содержащее кнопки, отвечающие за смену фона. Создаются 3 кнопки и при нажатии меняем фон приложения на соответствующий.

Выполнение задачи

1. Создание приложения

1.1 Создание скелета приложения

Первым делом созданием скелет нашего приложения на основе макета с пустой активностью. На основе данного макета будем выстраивать вид и логику нашего приложения.

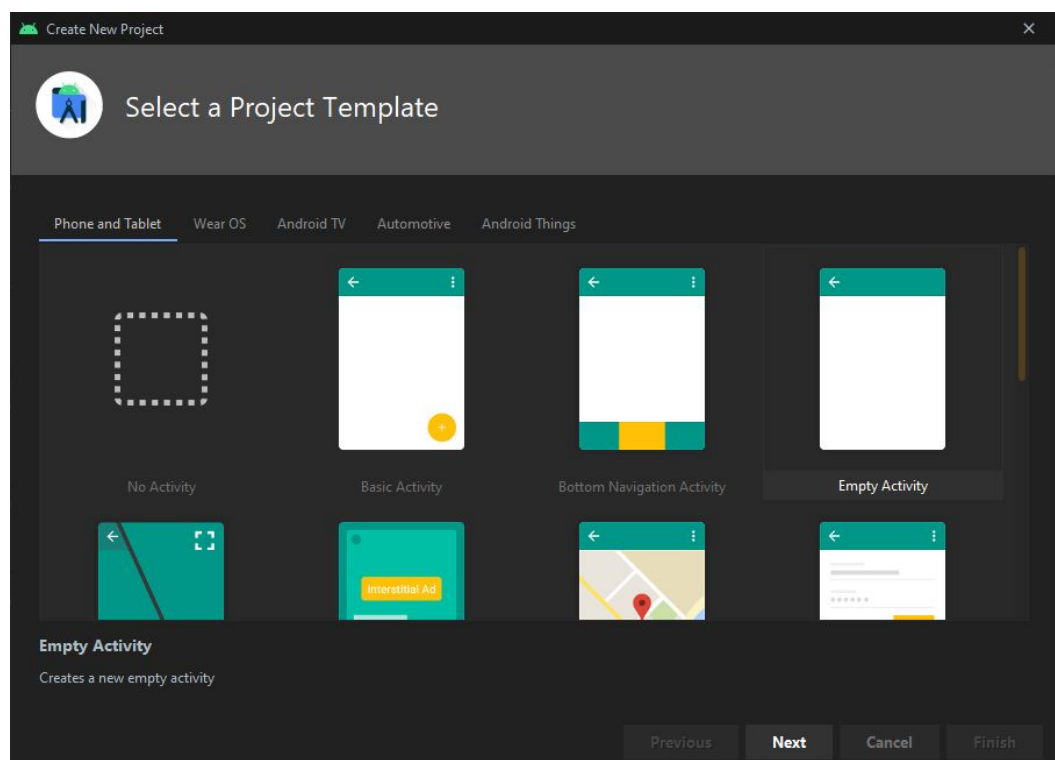


Рис. 1.1 Окно выбора шаблона приложения

1.2 Добавление активных элементов

Редактируем основное окно приложения путём добавления следующих элементов в файл *activity_main.xml* :

- Текстовое поле с названием приложения — *TextView*
- Трёх кнопок, расположенных по центру окна приложения — *Button*
- Плавающего окна с иконкой — *FloatingActionButton*

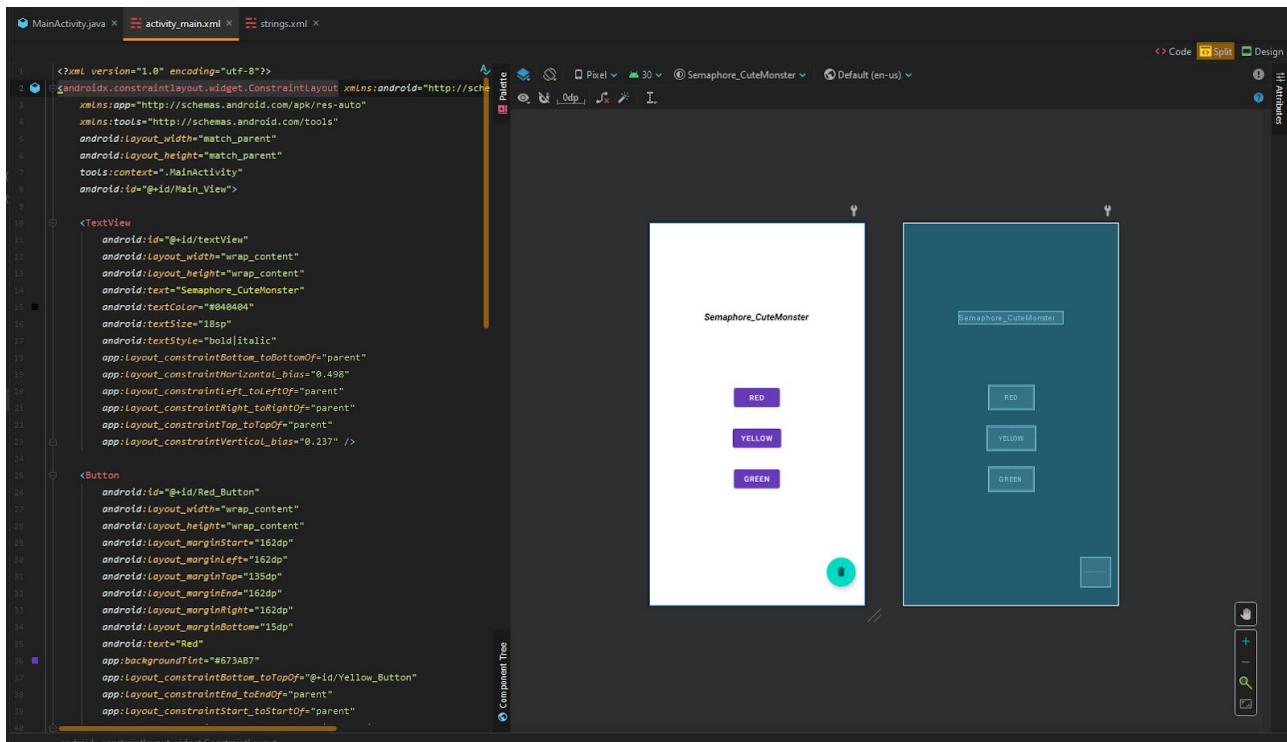


Рис. 1.2 Конструктор окна приложения

Для корректного объявления отображаемого текста на кнопках и текстовом поле, добавляем соответствующие элементы в файл *strings.xml*, который содержит все ресурсы данного типа. Использование данного подхода облегчает создание и редактирование текстов, используемых приложением, так как при необходимости изменить какую-нибудь часть текста не нужно будет искать каждый элемент, который содержит данный текст, а просто изменить данный объект и изменения автоматически будут применены ко всем использующим объектам.

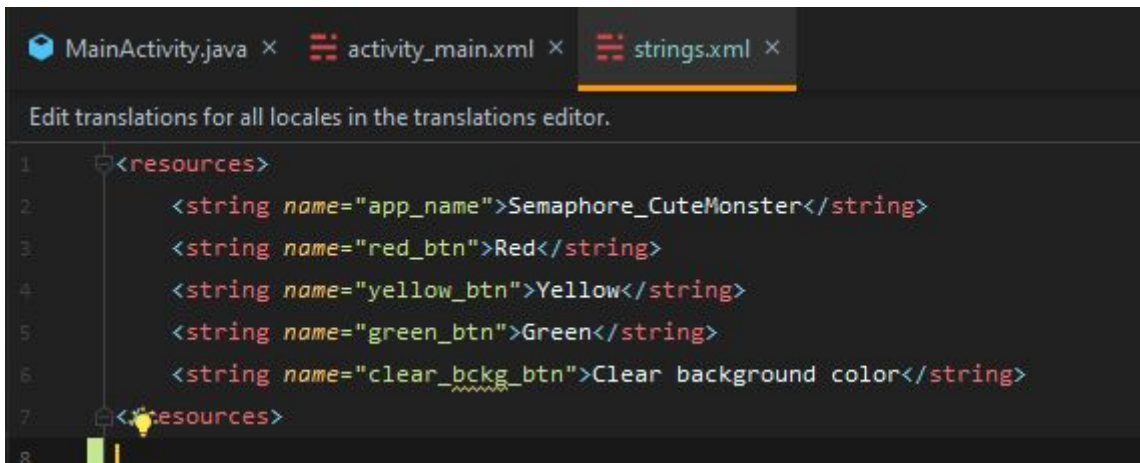


Рис. 1.3 Содержимое strings.xml

Получение текста осуществляется путём вызова “@string/<name_value>” в качестве значения для параметра объекта содержащего текстовое значение.

1.3 Написание кода приложения

После создания вида нашего основного окна приложения следует написать код, для корректного поведения наших добавленных элементов. Поведение наших объектов будем описывать в файле *MainActivity.java*, который отвечает за основное окно нашего приложения.

Первым делом добавим пару параметров в метод *onCreate*. Данные параметры будут отвечать за создание объекта представления и вызов метода инициализации слушателей на кнопки.

```
@Override
protected void onCreate(Bundle savedInstanceState) {
    super.onCreate(savedInstanceState);
    setContentView(R.layout.activity_main);

    // Find Main_View by id
    this.view = findViewById(R.id.Main_View);

    // Initialize button listeners on click
    this.init_btn_click_listeners();
}
```

```

/**
 * Method which initializes button listeners
 * */
public void init_btn_click_listeners(){

    // Changing button background color
    findViewById(R.id.Red_Button).setOnClickListener(
        view -> view.setBackgroundColor(Color.RED)
    );

    // Changing Main_View background color when button is pressed
    findViewById(R.id.Yellow_Button).setOnClickListener(
        view -> this.view.setBackgroundColor(Color.YELLOW)
    );

    findViewById(R.id.Green_Button).setOnClickListener(
        view -> this.view.setBackgroundColor(Color.GREEN)
    );

    findViewById(R.id.Clear_Background_Color_Button).setOnClickListener(
        view -> this.view.setBackgroundColor(Color.WHITE)
    );
}

```

Данный метод отвечает за поиск кнопок по их *id* и инициализации слушателей на каждой из них. Действие, исполняемое данным слушателем задаётся путем использования лямбда выражения, так как в нашем случае выполняется лишь одно действие.

Слушатель *Красной кнопки*, меняет цвет фона самой кнопки, так как вызывается параметр лямбды.

Слушатели остальных кнопок вызывают параметр, отвечающий за основной экран нашего приложения. Данной разграничение осуществляется путём добавления *this.*, указывающего на то, что данный объект является параметром нашего класса, значение которого мы объявили в методе *onCreate*.

Код желаемого цвета передаём путем вызова класса *Color*, который содержит в себе переменные с названием основных цветов и их числовыми значениями.

2. Работа приложения

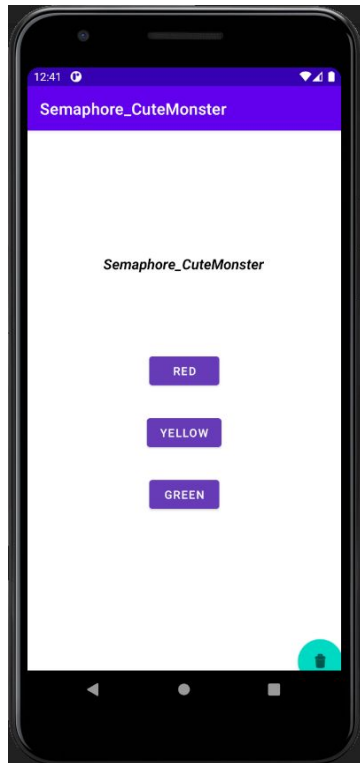


Рис. 1.4 Начальное состояние

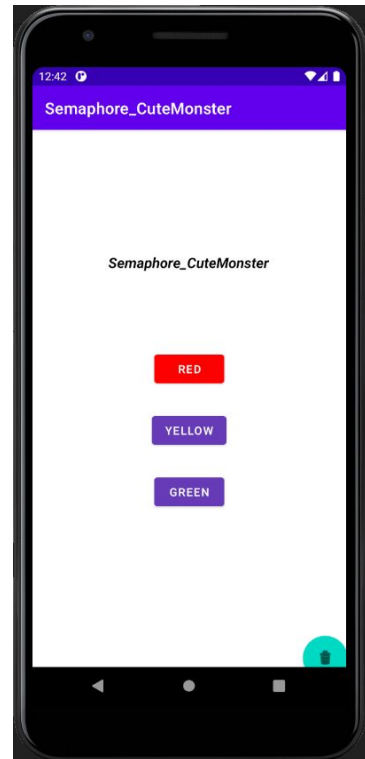


Рис. 1.5 Нажатие красной кнопки

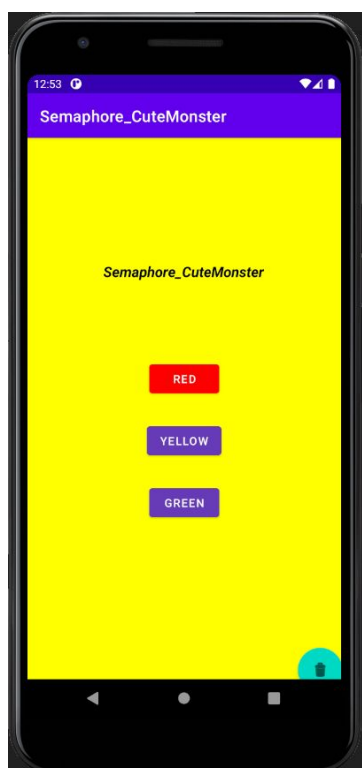


Рис. 1.6 Нажатие желтой кнопки

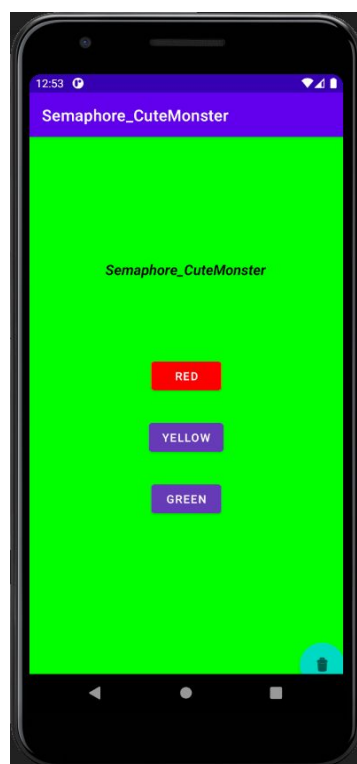


Рис. 1.7 Нажатие зеленой кнопки

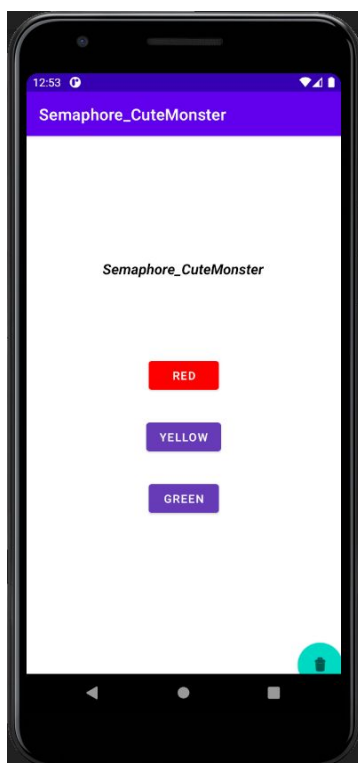


Рис. 1.8 Нажатие очищающей кнопки

Вывод

Выполнение данной работы дало понимание базовых действий над главным окном приложения, работой с его компонентами. Также был ознакомлен с функционалом ide Android Studio.