

## **Лабораторная работа №13**

### **Создание нового проекта**

#### **1 Цель работы**

1.1 Изучить процесс создания приложения в Android Studio.

#### **2 Литература**

2.1 Jetpack Compose | Руководство. metanit.com – Текст : электронный // metanit.com, 2023. – URL: <https://metanit.com/kotlin/jetpack/> – гл.1.

#### **3 Подготовка к работе**

3.1 Повторить теоретический материал (см.п.2).

3.2 Изучить описание лабораторной работы.

#### **4 Основное оборудование**

4.1 Персональный компьютер.

#### **5 Задание**

5.1 Создание проекта Jetpack Compose

5.1.1 Создать проект типа Empty Activity. Включить режим отображения приложения Split.

5.1.2 Изменить вызов функции и код функции Greeting, чтобы в нее передавалось название группы и на экран выводился «Привет, название группы» (вместо названия — параметр функции).

5.2 Добавление в проект composable-функции

5.2.1 Добавить в приложение composable-функцию Task2

5.2.2 Добавить в функцию метку (Text), изменить ее размер, текст и цвет фона, добавить отступ от границ по 20.

5.2.3 Реализовать предпросмотр внешнего вида результата работы composable-функции при ее вызове в функции Task2Preview.

5.2.4 Вызвать функцию Task2 вместо функции Greeting.

5.3 Использование контейнеров

5.3.1 Добавить в приложение composable-функцию Task3.

5.3.2 Разместить в ней метку с текстом «Добро пожаловать» и кнопку с текстом «ОК». Для описания кнопки используется следующий код: `Button(onClick = {}){ Text("подпись") }`

5.3.3 Для того, чтобы элементы отображались, не перекрывая друг друга, нужно поместить их в контейнере, выделив элементы и в меню выбрав Show Context Action – Surround with widget – выбрать нужный контейнер.

5.3.4 Для каждого контейнера создать свою функцию с соответствующим названием: Task3Column Task3Row Task3Box

5.3.5 Реализовать вызов созданных функций.

5.4 Позиционирование компонентов в контейнерах

5.4.1 С использованием средств контейнера Box (другие контейнеры не использовать) разработать макет интерфейса в соответствии с рисунком 1

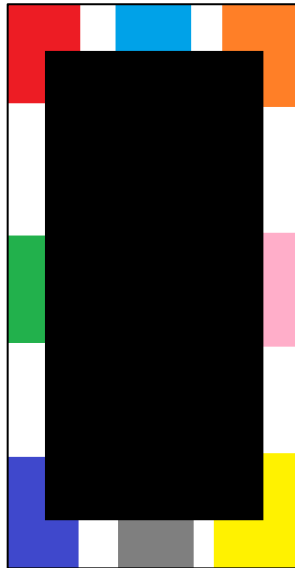


Рисунок 1

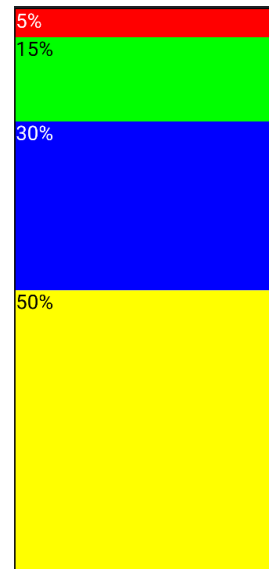


Рисунок 2

5.4.1 С использованием средств контейнера Column (другие контейнеры не использовать) разработать макет интерфейса в соответствии с рисунком 2

5.5 Создание элементов управления в цикле

5.5.1 Добавить в приложение composable-функцию Task4.

5.5.2 Добавить в нее список с пятью разными строковыми значениями.

5.5.3 Используя цикл, вывести на экран значения элементов списка, используя метки Text. Все элементы должны выводиться друг под другом.

5.5.4 Реализовать вызов созданной функции.

5.6 Создание пакета для экранов.

5.6.1 Добавить в папку ui.theme новый пакет с названием screens.

5.6.2 Добавить в пакет новый файл Kotlin.

5.6.3 В нем описать composable-функцию для настройки окна авторизации. В окне авторизации должны быть: поля для ввода логина и пароля (TextField), метки для комментариев к полям ввода (Text), кнопка с подписью «Авторизоваться».

5.6.4 Реализовать вызов созданной функции экрана авторизации.

## 6 Порядок выполнения работы

6.1 Выполнить все задания из п.5.

6.2 Ответить на контрольные вопросы.

## 7 Содержание отчета

7.1 Титульный лист

7.2 Цель работы

7.3 Ответы на контрольные вопросы

7.4 Вывод

## 8 Контрольные вопросы

8.1 Что такое Jetpack Compose?

8.2 Какой тип проекта нужно выбрать, чтобы создать проект с использованием Jetpack Compose?

8.3 Что такое Activity?

8.4 Для чего используется MainActivity?

8.5 Для чего используется функция onCreate?

8.6 Что делает функция setContentView?

8.7 Для чего используется функция @Composable?

8.8 Для чего указывается @Preview?