Лабораторна робота 4.

Розмітка екрану, основи XML.

**Завдання**

Створити проект з трьома активностями, використовуючи різні макети для кожної з активностей. Додати кнопки для переходу між активностями. Додати на кожну активність елемент TextView та Button.

Створити проект з трьома активностями, використовуючи різні макети для кожної з активностей. Додати кнопки для переходу між активностями. На активностях додати декілька елементів (View) з лекції. Змінити їх базові властивості: розмір, кольори тексту, та фоновий колір. Змінити налаштування, що знаходяться у ресурсних файлах. Зробити градієнтний колір для будь-якого View.

Додаткові завдання:

1. Змінити форму кнопки або будь-якого іншого елементу

2. Змінити іконку застосунку

**Хід роботи**

Для створення проекту з трьома активностями в Android Studio необхідно виконати декілька кроків, починаючи від створення нового проекту до налаштування переходів між активностями та зміни базових властивостей елементів.

Спершу відкриваємо Android Studio та створюємо новий проект. Вибираємо шаблон Empty Activity і задаємо необхідні параметри, такі як ім'я проекту, місце збереження та мову програмування (наприклад, Kotlin).

Далі, створюємо три активності: MainActivity, ActivitySecond та ActivityThird. Кожна з активностей має свій XML файл для макету: activity\_main.xml, activity\_second.xml та activity\_third.xml. В кожному з цих XML файлів використовуються різні макети (ConstraintLayout, LinearLayout, RelativeLayout, FrameLayout) для розміщення елементів на екрані.

На кожну активність додаємо елементи TextView для відображення тексту та Button для виконання певних дій, таких як перехід до іншої активності. Наприклад, в MainActivity додаємо дві кнопки для переходу до ActivitySecond та ActivityThird. Відповідні методи для обробки натискання кнопок реалізовуються в класах активностей.

Після цього додаємо декілька View елементів на кожну активність та змінюємо їх базові властивості, такі як розмір, кольори тексту та фоновий колір. Наприклад, один з View елементів може мати градієнтний фон, інший - простий колір. Також змінюємо налаштування у ресурсних файлах, таких як colors.xml та styles.xml, щоб відобразити ці зміни.

Для створення градієнтного кольору для будь-якого View елемента додаємо новий ресурсний файл в папку drawable, де визначаємо градієнтні кольори. Цей файл застосовуємо до фону відповідного View елемента.

Додаткові завдання включають зміну форми кнопки або іншого елемента та зміну іконки застосунку. Для зміни форми кнопки створюємо новий XML файл в папці drawable, де визначаємо форму кнопки (наприклад, круглу) та застосовуємо цей файл до фону кнопки. Для зміни іконки застосунку замінюємо стандартну іконку в папці res/drawable на нову іконку, створену або завантажену користувачем.

Завершивши всі налаштування, запускаємо проект на емуляторі або фізичному пристрої для перевірки роботи активностей та їх переходів. Переконуємось, що всі зміни відображаються коректно та переходи між активностями працюють належним чином.

Повний код програми:

package com.lab\_4

import android.content.Intent

import androidx.appcompat.app.AppCompatActivity

import android.os.Bundle

import android.widget.Button

class MainActivity : AppCompatActivity() {

private lateinit var buttonToSecond: Button

private lateinit var buttonToThird: Button

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_main)

buttonToSecond = findViewById(R.id.button\_to\_second)

buttonToThird = findViewById(R.id.button\_to\_third)

buttonToSecond.setOnClickListener {

val intent = Intent(this, ActivitySecond::class.java)

startActivity(intent)

}

buttonToThird.setOnClickListener {

val intent = Intent(this, ActivityThird::class.java)

startActivity(intent)

}

}

}

package com.lab\_4

import androidx.appcompat.app.AppCompatActivity

import android.os.Bundle

import android.widget.Button

class ActivitySecond : AppCompatActivity() {

private lateinit var buttonBackToMain: Button

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_second)

buttonBackToMain = findViewById(R.id.button\_back\_to\_main)

buttonBackToMain.setOnClickListener {

finish() // Navigates back to the previous activity (MainActivity)

}

}

}

package com.lab\_4

import androidx.appcompat.app.AppCompatActivity

import android.os.Bundle

import android.widget.Button

class ActivityThird : AppCompatActivity() {

private lateinit var buttonBackToMain: Button

override fun onCreate(savedInstanceState: Bundle?) {

super.onCreate(savedInstanceState)

setContentView(R.layout.activity\_third)

buttonBackToMain = findViewById(R.id.button\_back\_to\_main)

buttonBackToMain.setOnClickListener {

finish() // Navigates back to the previous activity (MainActivity)

}

}

}

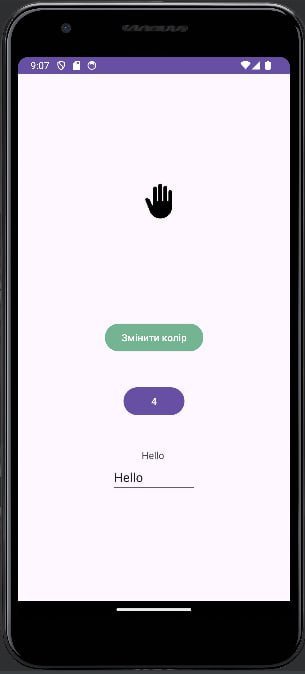
**Результати роботи**



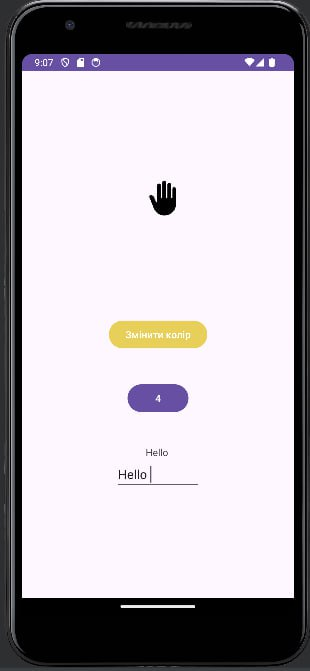
Додаток після запуску



Додаток після введення тексту



Натискання на кнопку з цифрою



Натискання на кнопку з цифрою



Переміщення елементу