

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ імені
ТАРАСА ШЕВЧЕНКА**



ФАКУЛЬТЕТ ІНФОРМАЦІЙНИХ ТЕХНОЛОГІЙ

Кафедра прикладних інформаційних систем

Звіт до лабораторної роботи №3

з курсу

«Архітектура мобільних операційних систем і пристроїв»

*Студентки 4 курсу
групи ПП-41*

спеціальності 122 «Комп'ютерні науки»

ОП «Прикладне програмування»

Штось Софії Максимівни

Викладач:

Краснощок В.М.

Київ – 2023

Тема роботи: Робота із активностями, кольором та локалізацією в ОС Android.

Мета роботи: Отримати досвід роботи із процесами відстеження станів активності, використанням значень рядків і кольорів та зміною локалізації у додатку.

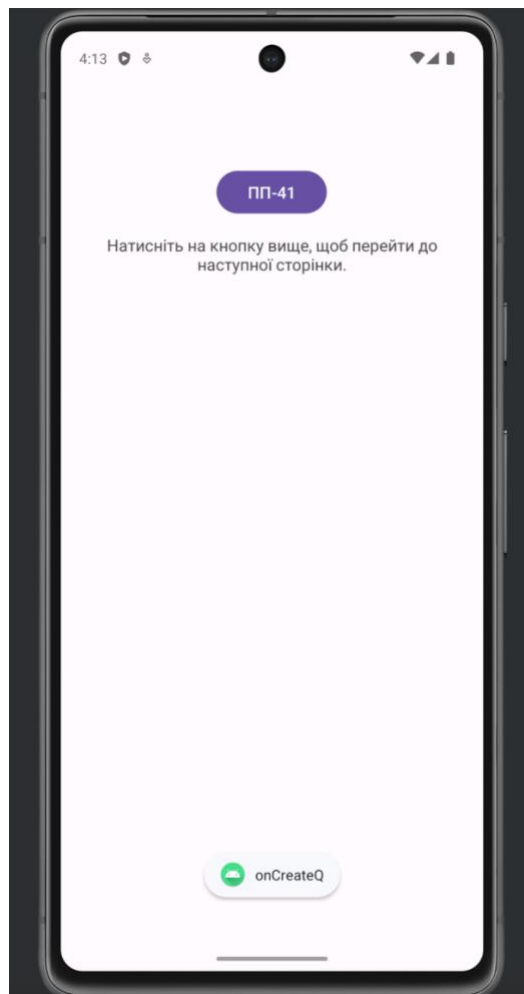
Завдання для виконання (варіант 23)

Використовуючи програму створену в 2й практичній роботі додайте до неї необхідні елементи для виконання в ній зміни у відповідності до завдань які розглянуті у пунктах теоретичних відомостей поданих вище.

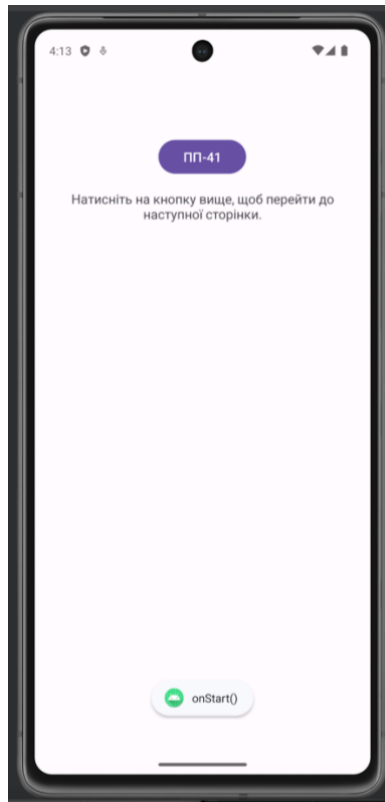
Хід роботи

Завдання 1. «Відстеження станів Активності»

Обробник подій onCreate, який викликається за умови *першого* запуску програми:



Обробник подій `onStart`, який викликається за умови запуску програми (не залежно від того, чи це перший запуск, чи ні):



Обробник подій `onPause`, який викликається за умови виходу з програми (але не її повного закриття):



Обробник подій onRestart, який викликається за умови відкриття програми заново після виходу з неї:



Завдання 2. «Використання значень рядків і кольорів»

Colors.xml:

```
<?xml version="1.0" encoding="utf-8"?>
<resources>
    <color name="black">#FF000000</color>
    <color name="white">#FFFFFFFF</color>
    <color name="view_bkg_color">#FF0</color>
    <color name="screen_bkg_color">#F88</color>
    <color name="text_color">#8004</color>
</resources>
```

Strings.xml:

```
<resources>
    <string name="app_name">Shton_Lab3</string>
    <string name="action_settings">Налаштування</string>
    <!-- Strings used for fragments for navigation -->
    <string name="first_fragment_label">Лабораторна робота №3</string>
    <string name="second_fragment_label">Ім'я студента</string>
    <string name="next">Next</string>
    <string name="group">ПП-41</string>
    <string name="previous">Назад</string>
```

```

<string name="paragraph_text">
    Натисніть на кнопку вище, щоб перейти до наступної сторінки.
</string>

<string name="student_full_name">Штонь Софія Максимівна</string>

<string name="hello">Hello, Android World!</string>
</resources>

```

Main.xml:

```

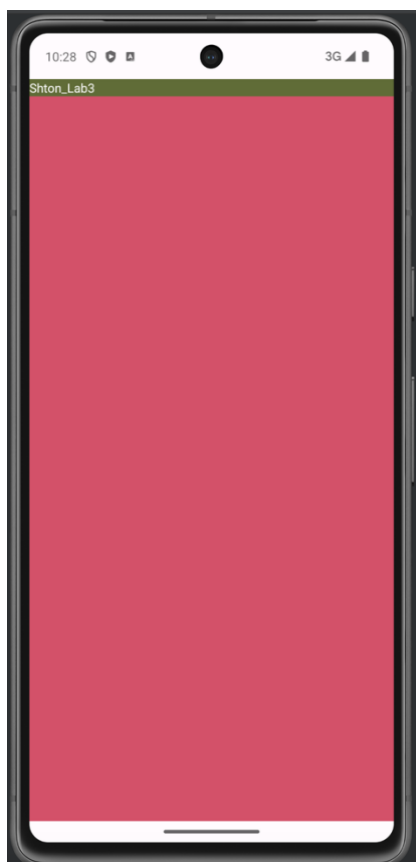
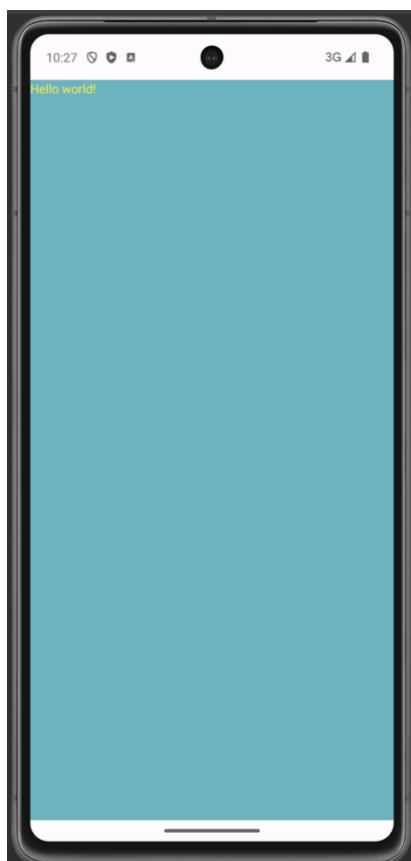
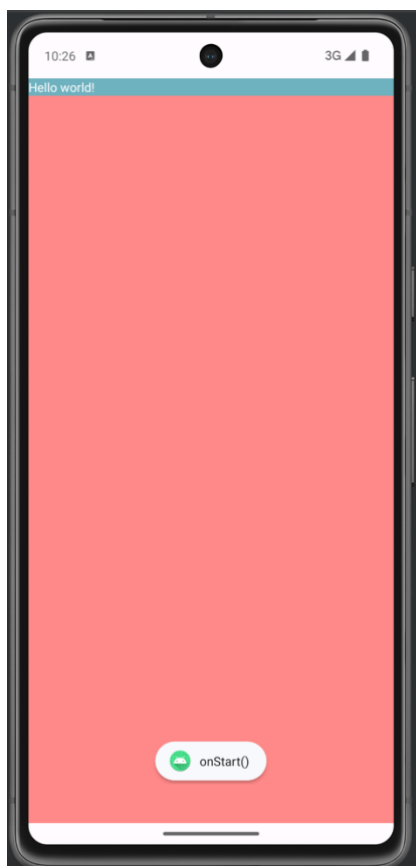
<?xml version="1.0" encoding="utf-8"?>
<LinearLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    android:layout_width="fill_parent"
    android:layout_height="fill_parent"
    android:orientation="vertical"
    android:background="@color/screen_bkg_color">
    <TextView
        android:layout_width="fill_parent"
        android:layout_height="wrap_content"
        android:text="@string/hello"
        android:background="@color/view_bkg_color"
        android:textColor="@color/text_color" />
</LinearLayout>

```

Результат роботи програми:

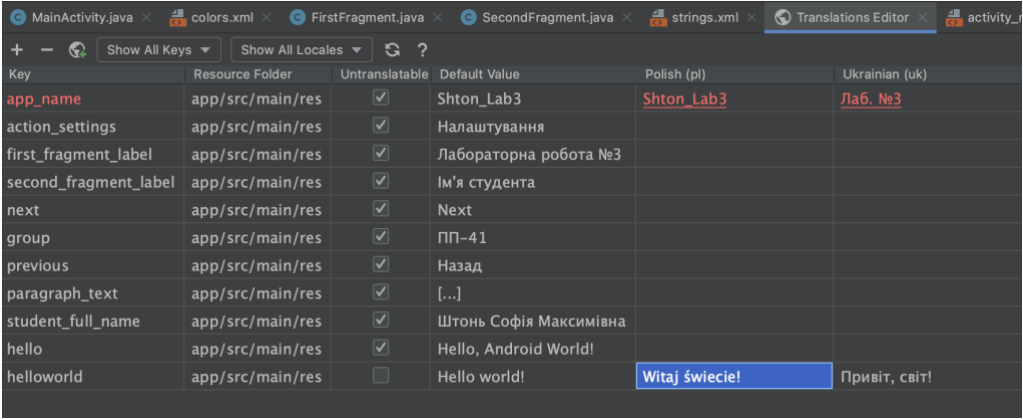
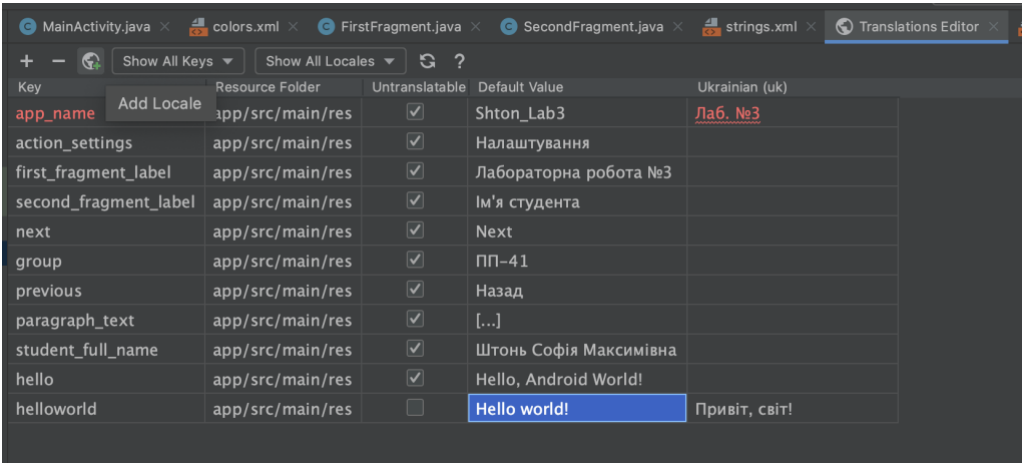


Введення різних значень кольорів та строкових констант:

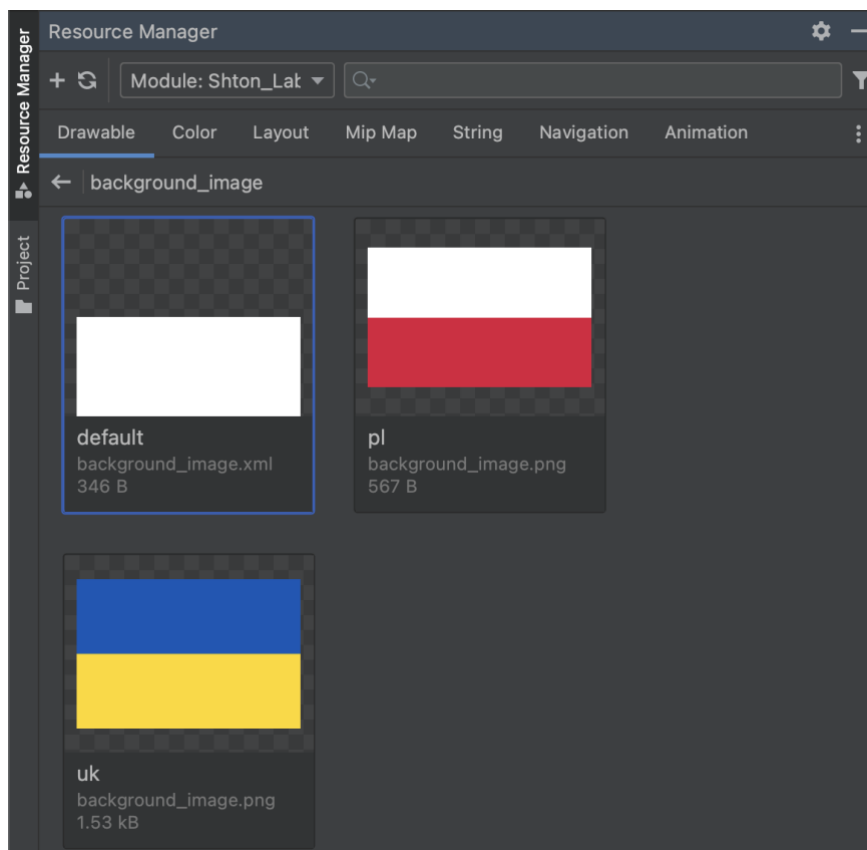


Завдання 3. «Локалізація додатку»

Додавання локалізації для hello world:



Додавання локалізації для фону:



Висновок: Отже, у ході цієї лабораторної роботи було проведено роботу з активностями, кольором та локалізацією в ОС Android. Було розроблено функціонал, який дає змогу відстежувати стани активності. Також було додано локалізацію до додатку, що змінює значення рядків та фону.

Контрольні запитання

1. Стани активностей та їх відстеження.

Активності можуть знаходитися в одному з чотирьох станів:

- **Активна (Active).** Активність знаходиться на передньому плані (на вершині стека) і має можливість взаємодіяти з користувачем. Android намагатиметься зберегти її працездатність, при необхідності перериваючи роботу інших активних, що знаходяться на нижчих позиціях в стеку для надання необхідних ресурсів. При виході на передній план іншої активності робота даної активності буде припинена або зупинена.

- **Призупинена (Paused).** Активність може бути видима на екрані, але не може взаємодіяти з користувачем: в цей момент вона призупинена. Це трапляється, коли на передньому плані знаходяться напівпрозорі або плаваючі (наприклад, діалогові) вікна. Робота призупиненої активності може бути припинена, якщо ОС необхідно виділити ресурси активності переднього плану. Якщо активність повністю зникає з екрану, вона зупиняється.

- **Зупинена (Stopped).** Активність невидима, вона знаходиться в пам'яті, зберігаючи інформацію про свій стан. Така активність стає кандидатом на передчасне закриття, якщо системі буде потрібно пам'ять для чогось іншого. При зупинці Активності розробнику важливо зберегти дані і поточний стан користувача інтерфейсу (стан полів введення, позицію курсора і т.д.). Якщо активність завершує свою роботу або закривається, вона стає неактивною.

- **Неактивна (Inactive).** Коли робота активності завершена, і перед тим, як вона буде запущена, дана активність знаходиться в неактивному стані. Такі активності видаляються з стека і повинні бути (пере) запущені, щоб їх можна було використовувати.

2. *Особливості використання ресурсів у коді програми.*

Незалежно від використовуваного середовища розробки, досить розумно відокремлювати використовувані додатком ресурси від коду. Зовнішні ресурси легше підтримувати, оновлювати і контролювати. Така практика також дозволяє описувати альтернативні ресурси для підтримки додатком різних пристроїв та реалізовувати локалізацію додатків.

Додатки Android використовують різноманітні ресурси із зовнішніх (по відношенню до коду) файлів, від простих (рядки і кольору) до більш складних (зображення, анімації, візуальні стилі). Надзвичайно корисно також відокремлювати від коду такі важливі ресурси, як розмітки екранів (Layout), використовувані в активності. Android автоматично вибирає найбільш підходящі варіанти з дерева ресурсів додатків, що містять різні значення для різних апаратних конфігурацій, мов і регіонів, не вимагаючи при цьому жодного рядка коду.

Ресурси програми зберігаються в каталозі `res` в дереві каталогів проекту. Плагін ADT автоматично створює каталог `res` з підкаталогами `values`, `layout` і `drawable-*`, в яких зберігаються, відповідно: строкові константи, розмітка за замовчуванням іконка програми.

Для дев'яти головних типів ресурсів використовуються різні підкаталоги каталогу `res`, це:

- прості значення,
- зображення,
- розмітка,
- анімація,
- стилі,
- меню,
- налаштування пошуку,
- XML,
- «сирі» дані.

При збірці пакету `.apk` ці ресурси максимально ефективно компілюються і

включаються в пакет.

3. *Робота із анімацією в Android.*

Android підтримує два види анімації: покрокову анімацію, послідовно виводить на екран зображення з заданою тривалістю, і анімацію, що базується на розрахунку проміжних кадрів, в цьому випадку застосовуються різні перетворення - обертання, розтягування, переміщення і затемнення. Всі ці трансформації описуються в XML-файлі в каталозі `res / anim`. Ресурс, що описує покрокову анімацію, зберігається в каталозі `res / drawable`.

4. *Робота із кольорами та колірні моделі в Android.*

Для опису кольорів використовується тег `<color>`. Значення кольору вказуються в шістнадцятковому вигляді в одному з наступних форматів:

- `# RGB`
- `# ARGB`
- `# RRGGBB`
- `# AARRGGBB`

У прикладі показано опис напівпрозорого червоного кольору і непрозорого зеленого:

```
<color name="transparentred">#77FF0000</color>
```

```
<color name="opaquegreen">#0F0</color>
```

5. *Життєвий цикл активності.*

Для забезпечення повноцінного інтерфейсу додатку, зміни його стану повинні бути непомітні для користувача. Міняючи свій стан з призупиненого на зупинений або з неактивного на активний, активність не повинна зовні змінюватися. При зупинці або призупиненні роботи активності розробник повинен забезпечити збереження стану активності, щоб її можна було відновити при виході активності на передній план. Для цього в класі `Activity` міститься обробники подій, перевизначення яких дозволяє розробнику відслідковувати зміну станів активності.

Обробники подій класу `Activity` дозволяють відслідковувати зміни станів відповідного об'єкта `Activity` під час усього життєвого циклу, що відображає рисунок

поданий нижче.

