



Міністерство освіти і науки України КПІ ім. Ігоря Сікорського
Факультет інформатики та обчислювальної техніки
Кафедра інформаційних систем та технологій

ЗВІТ

до лабораторної роботи №2 з дисципліни
«Розробка мобільних застосунків під Android»

Перевірив:

Орленко С. П.

Виконала:

студентка групи ІК-22

Шелкович А. В.

Київ – 2025

Тема: Дослідження роботи з компонентом Fragment.

Мета роботи: дослідити створення та взаємодію з компонентом Фрагмент (Fragment) компоненту Діяльність та набути практичні навички з використання фрагментів для інтерфейсу користувача.

Завдання

Написати програму під платформу Андроїд, яка має інтерфейс, побудований з декількох фрагментів згідно варіанту. Перший фрагмент представляє з себе форму для введення даних та кнопку підтвердження («ОК»), а інший фрагмент відображає результат взаємодії. Тобто другий фрагмент містить тестове поле з результатом та кнопкою «Cancel», яка очищає або приховує (або видаляє) другий фрагмент та очищає форму введення з першого фрагменту. Зверніть увагу, що робота з фрагментами відбувається в рамках однієї Діяльності.

Варіант 10. Вікно містить групу опцій (радіо-батони) для вибору розміру шрифту, текстове поле для введення інформації та кнопки «ОК» і «Cancel». Вивести введений рядок обраним розміром при натисканні на кнопку «ОК» у деяке текстове поле та очистити при натисканні кнопки «Cancel».

Хід роботи

1. Створення головного активності (MainActivity)

- Файл MainActivity.kt:
 - Наслідується від AppCompatActivity.
 - В методі onCreate встановлюється activity_main.xml як макет.
 - При першому запуску в fragmentManager додається InputFragment.

2. Створення першого фрагмента InputFragment

- Файл InputFragment.kt:
 - Використано EditText для введення тексту.

- RadioGroup із трьома RadioButton для вибору розміру шрифту.
- Кнопка ОК, яка передає введені дані у ResultFragment.
- Додано перевірку на заповненість полів перед переходом до ResultFragment.
- При натисканні ОК передаються введені дані через Bundle.

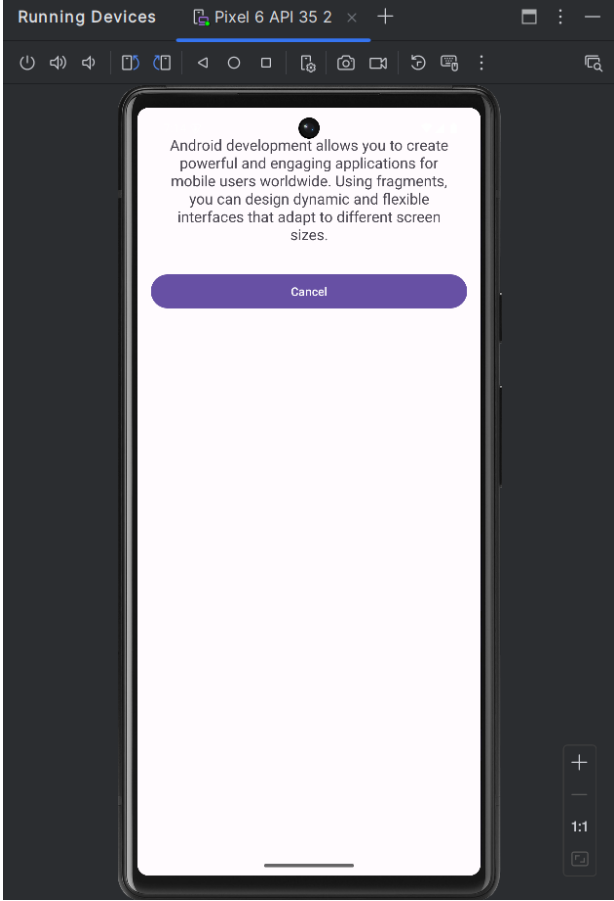
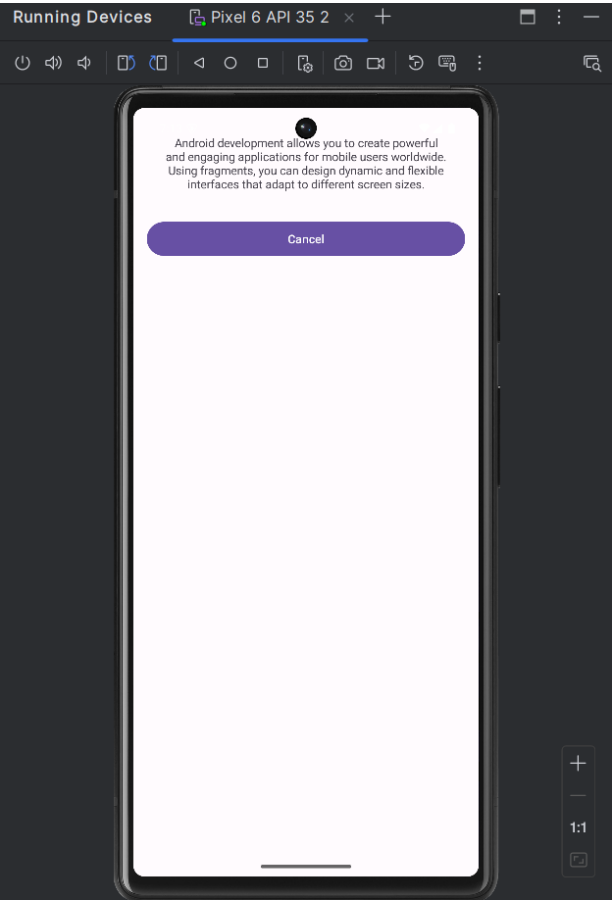
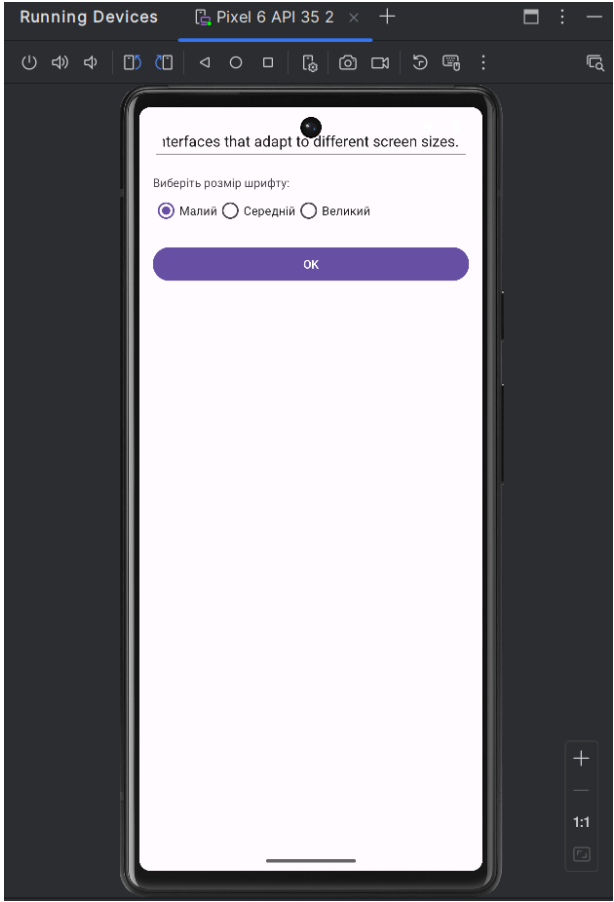
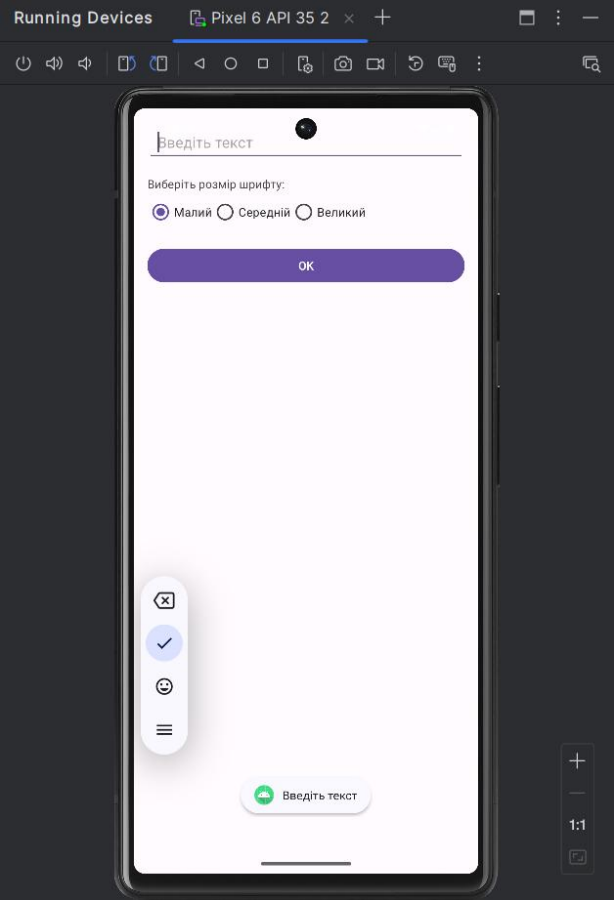
3. Створення другого фрагмента ResultFragment

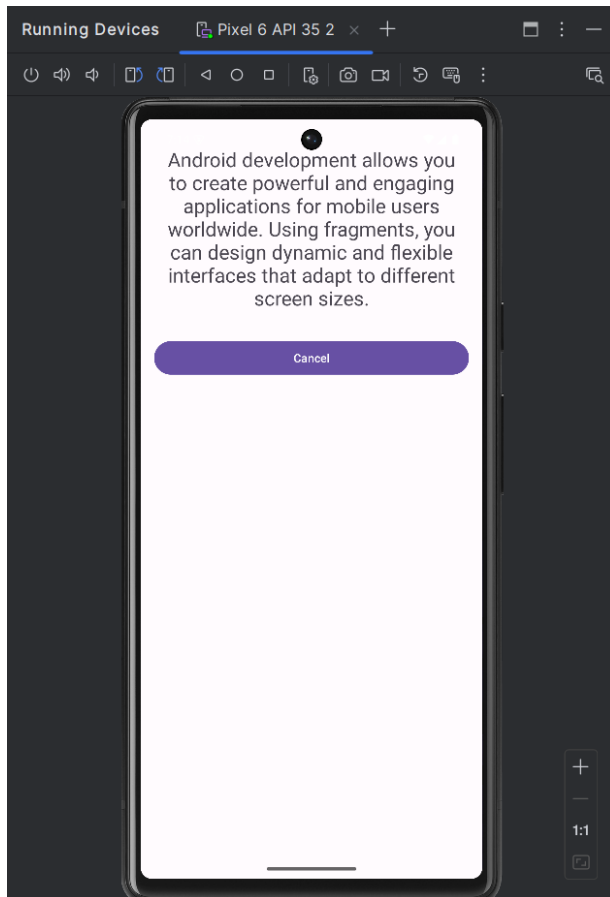
- Файл ResultFragment.kt:
 - Отримує передані дані (text та size).
 - Відображає текст у TextView із відповідним розміром шрифту.
 - Кнопка Cancel, яка закриває фрагмент і повертає користувача назад.

4. Макети інтерфейсу

- activity_main.xml: містить FrameLayout для розміщення фрагментів.
- fragment_input.xml:
 - EditText із hint та inputType.
 - RadioGroup для вибору розміру шрифту.
 - Кнопка ОК для переходу до ResultFragment.
- fragment_result.xml:
 - TextView для відображення тексту із змінним розміром шрифту.
 - Кнопка Cancel для закриття фрагмента.

Результат:





Висновок: В результаті виконання лабораторної роботи створено застосунок, що містить два фрагменти та дозволяє користувачеві вводити текст, змінювати розмір шрифту та очищати введені дані. Робота з фрагментами виконувалася в межах однієї активності.

Контрольні питання

1. Призначення та можливості компонента Фрагмент

Фрагмент (Fragment) — це незалежний компонент інтерфейсу користувача, який можна додати до Activity. Основні можливості:

- Може бути багаторазово використаний у різних Activity.
- Дозволяє створювати адаптивний інтерфейс для різних розмірів екрана.
- Може бути доданий або видалений динамічно під час роботи застосунку.
- Має власний життєвий цикл, схожий на Activity.

2. Життєвий цикл компонента Фрагмент

Життєвий цикл Fragment містить такі ключові методи:

- `onAttach()` – фрагмент прикріплюється до Activity.
- `onCreate()` – ініціалізація фрагмента.
- `onCreateView()` – створення інтерфейсу користувача.
- `onViewCreated()` – викликається після створення View.
- `onStart()` – фрагмент стає видимим.
- `onResume()` – фрагмент активний та взаємодіє з користувачем.
- `onPause()` – фрагмент більше не активний.
- `onStop()` – фрагмент не відображається.
- `onDestroyView()` – знищення інтерфейсу користувача.
- `onDestroy()` – фіналізація фрагмента.
- `onDetach()` – від'єднання від Activity.

3. Способи створення компонента Фрагмент

- Статичний спосіб – додається у XML-файл `activity_main.xml` за допомогою `<fragment>`.
- Динамічний спосіб – додається програмно через `FragmentManager`.

4. Способи управління компонентом Фрагмент

- Використання `FragmentManager` для додавання (`add()`), заміни (`replace()`) або видалення (`remove()`) фрагментів.
- Використання `FragmentTransaction` для анімацій, додавання в стек (`addToBackStack()`).
- Передача даних через `Bundle`, інтерфейси або `ViewModel`.

5. Способи взаємодії між Фрагментами

- Використання `setArguments(Bundle bundle)` для передачі даних при створенні фрагмента.
- Використання `ViewModel`, який спільно використовується між `Activity` та `Fragment`.
- Використання колбек-методів або інтерфейсів для передачі даних між фрагментами.

6. Наведіть поняття системи, малої системи та мобільної платформи

- Система – сукупність взаємопов'язаних компонентів, що виконують певні функції.
- Мала система – частина системи, що має меншу кількість компонентів та взаємодіє з іншими частинами.
- Мобільна платформа – це сукупність апаратного забезпечення та операційної системи, що використовується для мобільних пристроїв.

7. Типи мобільних застосунків

- Нативні (Native) – розроблені спеціально для конкретної платформи (Android, iOS).
- Кросплатформні (Cross-platform) – працюють на кількох платформах (React Native, Flutter).
- Веб-застосунки (Web apps) – відкриваються через браузер (PWA).
- Гібридні (Hybrid) – поєднують нативні та веб-технології (Ionic, Cordova).

8. Класифікація та загальна характеристика середовищ розробки мобільних застосунків

- Android Studio – основне середовище для Android (Java/Kotlin).
- Xcode – використовується для розробки під iOS (Swift/Objective-C).
- React Native, Flutter – для кросплатформних додатків.

- Unity – для розробки ігор.

9. Класифікація та загальна характеристика мобільних платформ

- Android – гнучка, підтримує багато пристроїв, має відкритий код.
- iOS – замкнута екосистема, висока продуктивність, строгі вимоги до застосунків.
- HarmonyOS – платформа Huawei, орієнтована на розумні пристрої.
- KaiOS – легка ОС для бюджетних кнопочових телефонів.