Міністерство освіти і науки України

Національний технічний університет України

«Київський політехнічний інститут імені Ігоря Сікорського»

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

Лабораторна робота №1

з дисципліни

«Об’єктно-орієнтоване програмування»

Виконав: Перевірив:

студент групи ІМ-32 Порєв В. М.

Король Олександр Володимирович

номер у списку групи: 14

Київ 2024

**Варіант завдання**

В1 = Ж mod 4 = 14 % 4 = 2

В2 = (Ж+1) mod 4 = 15 % 4 = 3

**Текст програми**

./Main.kt

package main

import module1.showFirstModuleDialogs

import module2.showSecondModuleDialog

fun main() {

*showFirstModuleDialogs* **{**

*showSecondModuleDialog*()

**}**

}

./module1/module1.kt

**package module1**

**import javax.swing.\***

**fun showFirstModuleDialogs(onComplete: () -> Unit) {**

**val frame = JFrame("Main Window First Module")**

**frame.*defaultCloseOperation* = JFrame.*EXIT\_ON\_CLOSE***

**frame.setSize(500, 500)**

**frame.*isVisible* = false**

***showFirstDialog*(frame, onComplete)**

**}**

**private fun showFirstDialog(frame: JFrame, onComplete: () -> Unit) {**

**val firstDialog = JDialog(frame, "First Dialog", true)**

**val nextButton = JButton("Next >")**

**val cancelButton = JButton("Cancel")**

**val firstPanel = JPanel()**

**nextButton.addActionListener {**

**firstDialog.dispose()**

***showSecondDialog*(frame, onComplete)**

**}**

**cancelButton.addActionListener {**

**firstDialog.dispose()**

**onComplete()**

**}**

**firstPanel.add(nextButton)**

**firstPanel.add(cancelButton)**

**firstDialog.add(firstPanel)**

**firstDialog.setSize(500, 500)**

**firstDialog.*isVisible* = true**

**}**

**private fun showSecondDialog(frame: JFrame, onComplete: () -> Unit) {**

**val secondDialog = JDialog(frame, "Second Dialog", true)**

**val backButton = JButton("< Back")**

**val yesButton = JButton("Yes")**

**val cancelButton = JButton("Cancel")**

**val secondPanel = JPanel()**

**backButton.addActionListener {**

**secondDialog.dispose()**

***showFirstDialog*(frame, onComplete)**

**}**

**yesButton.addActionListener {**

**JOptionPane.showMessageDialog(secondDialog, "You clicked Yes!")**

**}**

**cancelButton.addActionListener {**

**secondDialog.dispose()**

**onComplete()**

**}**

**secondPanel.add(backButton)**

**secondPanel.add(yesButton)**

**secondPanel.add(cancelButton)**

**secondDialog.add(secondPanel)**

**secondDialog.setSize(500, 500)**

**secondDialog.*isVisible* = true**

**}**

./module2/module2.kt

package module2

import javax.swing.\*

fun showSecondModuleDialog() {

val frame = JFrame("Main Window Second Module")

frame.*defaultCloseOperation* = JFrame.*EXIT\_ON\_CLOSE*

frame.setSize(500, 500)

frame.*isVisible* = false

val dialog = JDialog(frame, "Second Dialog", true)

val groupNames = *arrayOf*("IM", "ІП", "IС", "IА" , "IК", "IО")

val list = JList(groupNames)

val yesButton = JButton("Yes")

val cancelButton = JButton("Cancel")

val messageLabel = JLabel()

val panel = JPanel()

yesButton.addActionListener **{**

val selectedGroup = list.*selectedValue*

if (selectedGroup != null) {

messageLabel.*text* = "Selected: $selectedGroup"

} else {

messageLabel.*text* = "No group selected"

}

**}**

cancelButton.addActionListener **{**

dialog.dispose()

**}**

panel.add(JScrollPane(list))

panel.add(yesButton)

panel.add(cancelButton)

panel.add(messageLabel)

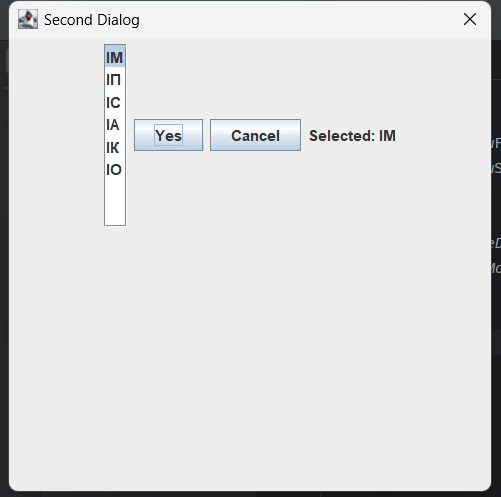
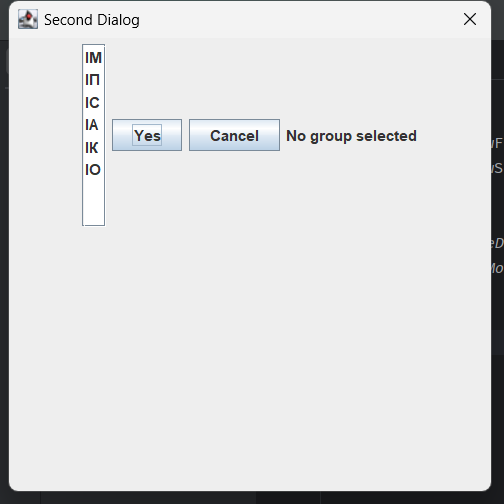
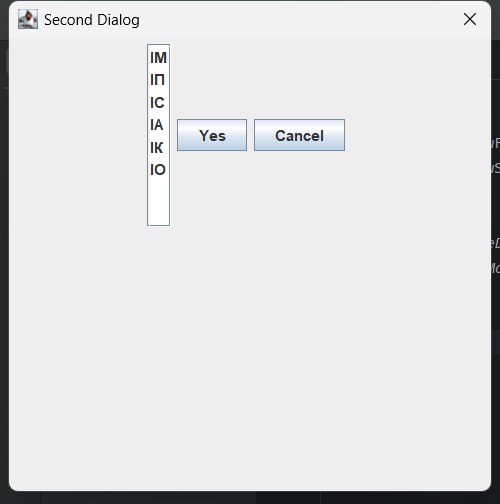
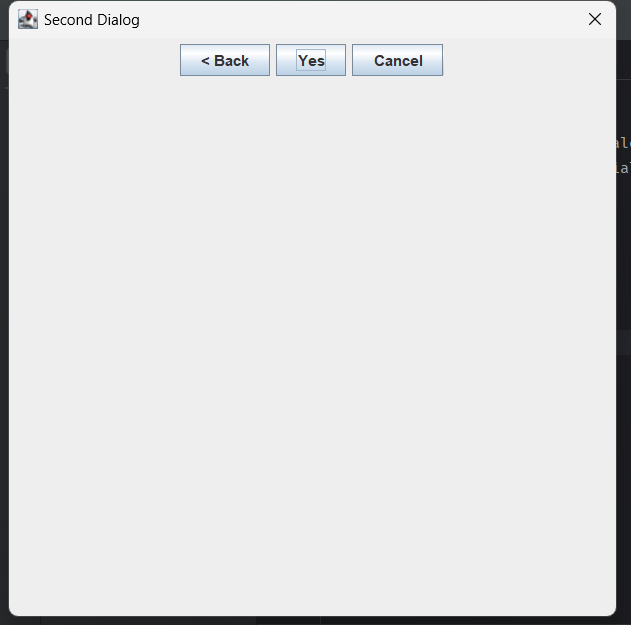
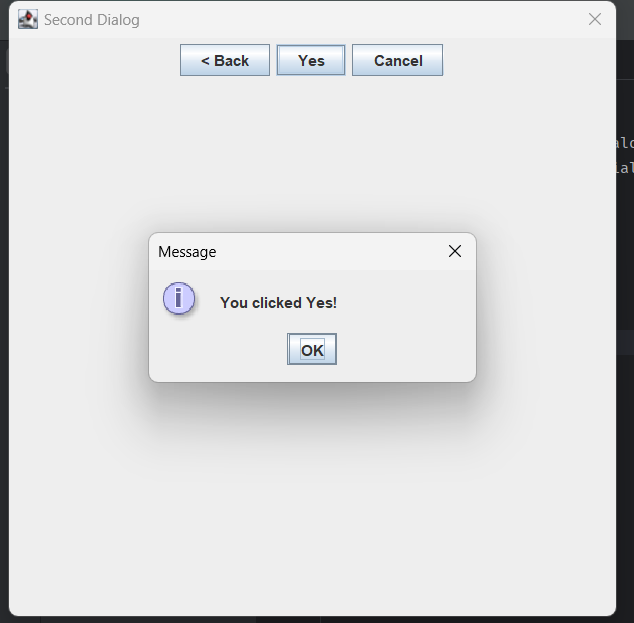
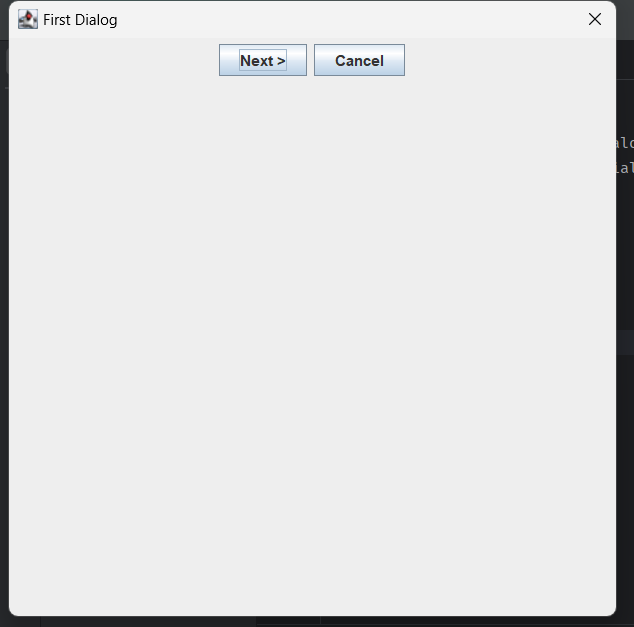
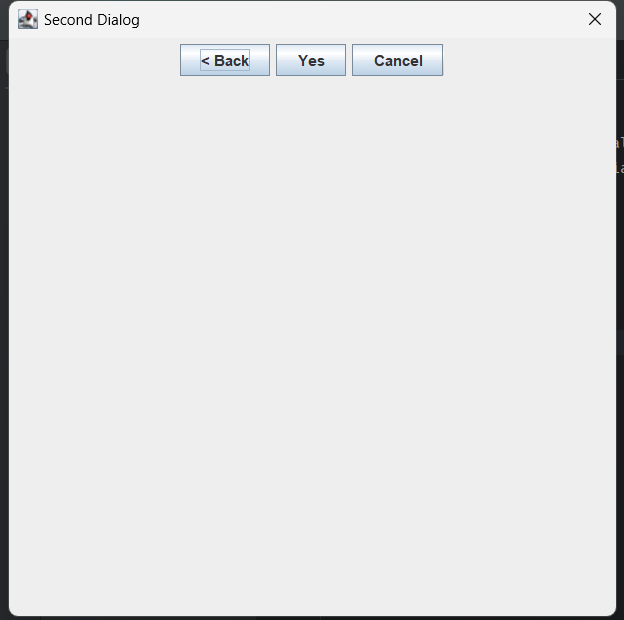
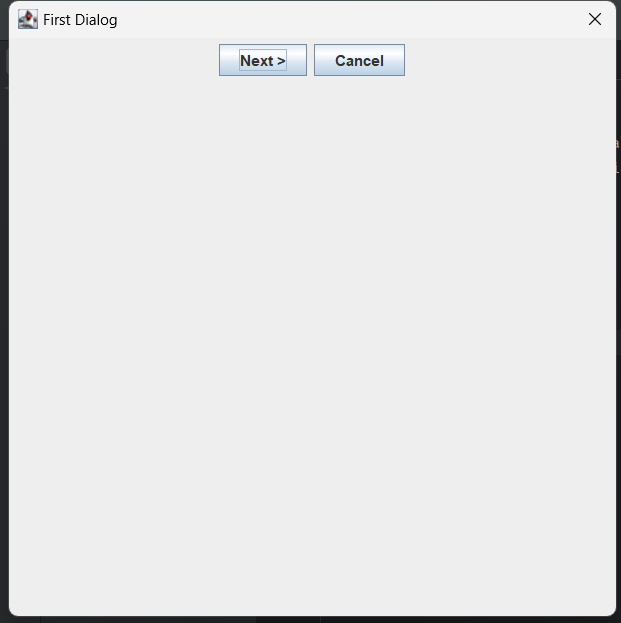
dialog.add(panel)

dialog.setSize(400, 400)

dialog.*isVisible* = true

}

**Результати тестування програми**



**Висновки**

У ході виконання цієї лабораторної роботи я здобув перші навички створення програм для Windows на основі проєктів з використанням Kotlin в середовищі Intellij IDEA. Я аналогічні задачі з використанням сучасних інструментів та мови Kotlin, що є популярною в екосистемі Java.

Основні здобутки включають:

1. Розуміння принципів модульного програмування: Я навчився розподіляти функціональність програми на окремі модулі, кожен з яких виконує свою роль. Це допомагає зробити програму більш структурованою, легко підтримуваною та зрозумілою.
2. Робота з графічним інтерфейсом користувача (GUI): Я отримав досвід створення діалогових вікон, роботи з елементами інтерфейсу (кнопками, списками) та обробки подій у програмах з графічним інтерфейсом.
3. Обробка подій: Навчився створювати динамічні взаємодії у програмі за допомогою слухачів подій та обробки натискань кнопок.
4. Моделювання поведінки вікон: Здобув навички роботи з модальними діалогами, які дозволяють блокувати взаємодію з іншими вікнами до закриття поточного.

Таким чином, виконання цієї роботи допомогло мені опанувати основи розробки графічних додатків з використанням Kotlin, що дозволяє ефективніше вирішувати завдання з розробки додатків для Windows.