ПРАКТИЧНЕ ЗАНЯТТЯ № 5

Тема: Засосування бібліотек Java для створення найпростішого графічного інтерфейсу.

Мета: Закріпити навички та прийоми підключення бібліотек java.awt, java.awt.event та javax.swing.

Обладнання: комп'ютери Pentium, Celeron.

Програмне забезпечення: MS Windows XP, пакет Java SDK.

Хід роботи Задача 1.

1. Створити просте візуальне застосування, використовувати пакет javax.swing, а також java.awt i java.awt.event (файл Dialog1.java).

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
public class Dialog1 {
    public static void main(String[] args) {
// фрагмент as is (1)
UIManager.setLookAndFeel(UIManager.getSystemLookAndFeelClassName());
        catch(Exception e) {
        JFrame frm = new JFrame("Перше візуальне застосування");
        frm.setSize(300, 200);
        Container c = frm.getContentPane();
        c.add(new JLabel("Hello, привіт"));
// dparment as is (2)
        WindowListener wndCloser = new WindowAdapter() {
            public void windowClosing(WindowEvent e) {
                System.exit(0);
        };
        frm.addWindowListener(wndCloser);
        frm.setVisible(true);
    }
}
```

- 2. У прикладі використані наступні класи стандартної бібліотеки Java
 - UIManager розглянемо пізніше;
 - JFrame дозволяє сформувати основне вікно додатку. Усі інші візуальні компоненти поміщаються всередину цього вікна.
 - Container клас для візуальних класів-контейнерів, тобто візуальних компонент, які можуть усередині себе містити інші візуальні компоненти.
 - JLabel клас для створення міток.
 - WindowListener i WindowAdapter as is.
- 3. Фрагмент "as is (1)" забезпечує Windows Look&Fill. Закоментуйте його і подивіться, що
- 4. Фрагмент "as is (2)" забезпечує завершення усього додатку у разі, якщо закрилося головне вікно додатку. Теж корисно закоментувати і подивитися, що трапиться.
- 5. Рядки

```
Container c = frm.getContentPane();
c.add(new JLabel("Hello, привіт"));
```

вимагають пояснення. Головне вікно (JFrame) влаштоване складним чином. Воно складається з ряду елементів. Той елемент, в який слід додавати інші візуальні компоненти, може бути отриманий за допомогою методу getContentPane(). Ми запам'ятовуємо посилання на нього в

змінній 'c' і потім використовуємо його в другому рядку фрагмента, який просто створює і додає (метод add) мітку на вікно.

При додаванні візуальних компонент в Java не вказується ні їх позиція, ні розмір. Це пов'язано з тим, що усі візуальні застосування Java масштабовані і для забезпечення масштабованості використовуються інші принципи розміщення візуальних елементів, а саме, використовуються різні Layout.

6. Рядок frm.setVisible(true); – виводить вікно на екран і активізує діалог з користувачем.

Задача 2.

- 1. Імпортувати в клас DoubleVector пакет algebra, який містить метод класу TestVector, що виводить на консоль строки.
- 2. Переробити DoubleVector в діалогову програму. Тобто замість виводу на консоль операторами System.out.println(..) треба забезпечити вивід на екран.

Пропонується такий простий варіант. Мають бути три операції виведення результату. Робимо на екрані не одну, а три мітки і заповнюємо їх потрібними значеннями або в конструкторі, або методом setText(..) класу JLabel. При простому додаванні трьох міток виникне невелика проблема - вони будуть розміщені одна поверх іншої - і видна буде тільки одна з них. Щоб уникнути цього ефекту, досить додати в конструктор рядок

c.setLayout(new FlowLayout());
перед рядками додавання міток.