Національний технічний університет України

‘Київський політехнічний інститут’

Факультет інформатики та обчислювальної техніки

Кафедра обчислювальної техніки

***Лабораторна робота №6***

*“ Розширення можливостей графічного інтерфейсу користувача на основі SWING”*

*Виконав: Мроць Ю.Б.*

*Група: ІО-12*

*Номер зк: 1219*

*Перевірив: Абу Усбах О.Н.*

*Київ — 2013*

***Мета***

Ознайомлення з реалізацією MVC-патерну в бібліотеці SWING. Отримання навичок з побудови інтерактивного інтерфейсу користувача. Поглиблене вивчення функціональних можливостей бібліотек SWING та Java2D.

***Завдання***

1. Ознайомитись з концепцією подій (events) в Java. Вільно володіти механізмом обробників (event handlers).

2. Ознайомитись з реалізацією MVC-патерну в бібліотеці SWING. Вільно володіти механізмом створення і модифікації моделі даних.

3. Ознайомитись з концепцією менеджерів зовнішнього вигляду LookAndFeel. Вміти змінювати зовнішній вигляд інтерфейсу користувача.

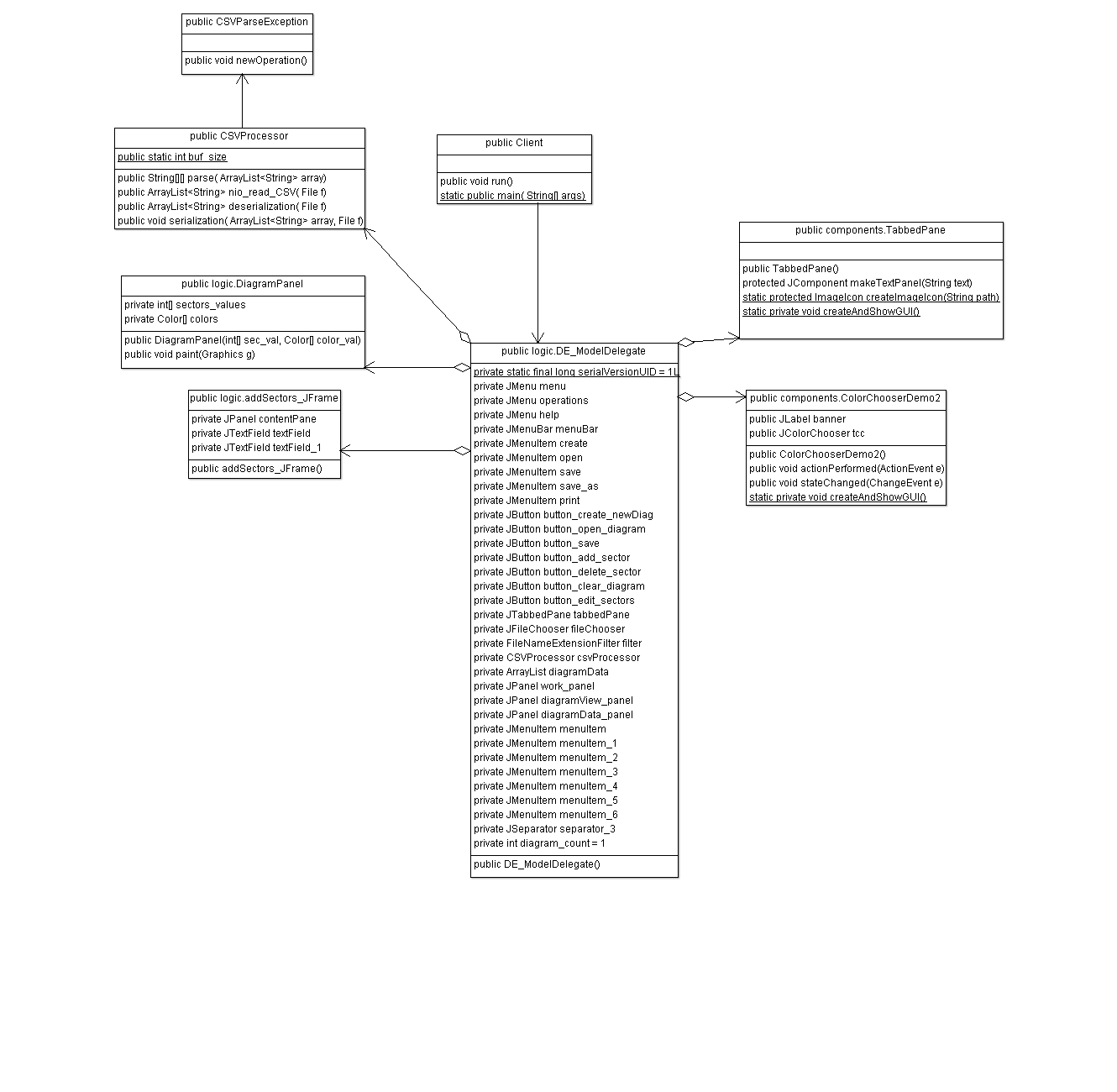
4. Вивчити механізм використання багатопотоковості при побудові додатків з графічним інтерфейсом на основі компонентів SWING.

5. Ознайомитись з функціональними можливостями бібліотеки Java2D. Вміти завантажувати зображення з файлу, модифікувати його за допомогою перетворень та зберігати у файлі (формати GIF, PNG, JPEG).

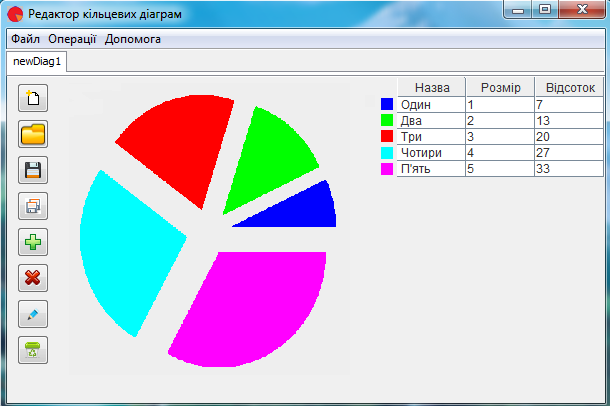
6. Модифікувати додаток з ЛР-6 таким чином, щоб забезпечити:

* можливість модифікації завантажених даних згідно варіанту (T-у табличній частині, G-у графічній частині);
* синхронізацію графічної і табличної частин додатку;
* можливість збереження діаграми у файлі формату JPEG.

***Діаграма класів***



***Дизайн графічного інтерфейсу***



***Таблиця відповідності елементів дизайну***

|  |  |
| --- | --- |
| ***Елемент інтерфейсу*** | ***Класи пакету java.swing*** |
| Головне вікно | java.swing .JFrame |
| Панель «Меню» | java.swing .JMenuBar |
| Панель «Вкладки» | java.swing .JTabbedPane |
| Область відображення діаграми | java.swing .JPanel |
| Діаграма | java.awt.Graphics |
| Список секторів | java.swing .JTable |
| Кнопка «Файл» | java.swing .JMenu |
| Кнопка «Операції» | java.swing .JMenu |
| Кнопка «Про програму» | java.swing .JMenu |
| Кнопка «Новий файл» | java.swing .JMenuItem |
| Кнопка «Зберегти файл» | java.swing .JMenuItem |
| Кнопка «Зберегти як…» | java.swing .JMenuItem |
| Кнопка «Відкрити файл» | java.swing .JMenuItem |
| Кнопка «Закрити файл» | java.swing .JMenuItem |
| Кнопка «Додати сектор» | java.swing .JMenuItem |
| Кнопка «Видалити сектор» | java.swing .JMenuItem |
| Кнопка «Змінити параметри сектора» | java.swing .JMenuItem |
| Кнопка «Очистити діаграму» | java.swing .JMenuItem |
| Кнопка «Автор» | java.swing .JMenuItem |
| Кнопка «Про програму» | java.swing .JMenuItem |
| Бічна панель кнопок | java.swing .JPanel |
| Кнопка «Новий файл» | java.swing .JButton |
| Кнопка «Зберегти файл» | java.swing .JButton |
| Кнопка «Зберегти як…» | java.swing .JButton |
| Кнопка «Відкрити файл» | java.swing .JButton |
| Кнопка «Додати сектор» | java.swing .JButton |

***Лістинг коду***

**package logic;**

**import java.io.File;**

**import java.io.FileInputStream;**

**import java.io.FileNotFoundException;**

**import java.io.FileOutputStream;**

**import java.io.IOException;**

**import java.io.ObjectInputStream;**

**import java.io.ObjectOutputStream;**

**import java.nio.ByteBuffer;**

**import java.nio.channels.FileChannel;**

**import java.nio.charset.Charset;**

**import java.util.ArrayList;**

**/\*\***

**\* У класі CSVProcessor реалізувані методи завантаження/вивантаження CSV-файлу в/з агрегований об’єкт класу ArrayList,**

**\* в якому знаходяться рядки CSV-файлу. Реалізувати методи серіалізації/десеріалізації агрегованого об’єкту класу ArrayList**

**\* в файлі serialized.dat.**

**\* @author Aruy**

**\*/**

**public class CSVProcessor {**

**/\*\***

**\* Розмір буфера.**

**\*/**

**private static int buf\_size;**

**/\*\***

**\* Конструктор**

**\* @param buf\_size розмір буфера.**

**\*/**

**public CSVProcessor(int buf\_size) {**

**setBuf\_size(buf\_size);**

**}**

**/\*\***

**\* Метод зчитує з об'єкту типу ArrayList<String> дані і записує їх в об'єкт типу String[][].**

**\* @param array об'єкт звідки треба зчитати дані**

**\* @return об'єкт, типу String[][]**

**\* @throws CSVParseException**

**\*/**

**public ArrayList<String[]> parse(ArrayList<String> array) throws CSVParseException{**

**ArrayList<String[]> mass = null;**

**int j = 0;**

**String k;**

**String[] tmp = new String[8];**

**for (int i = 0; i < array.size(); i++) {**

**k = array.get(i);**

**if (k != null) {**

**if (!k.equals("\n")) {**

**tmp[j] = k;**

**j++;**

**} else {**

**mass.add(tmp);**

**tmp = null;**

**j = 0;**

**}**

**}**

**}**

**return mass;**

**}**

**/\*\***

**\* Метод зчитує інформацію з CSV-файлу за допомогою потоків вводу пакету java.nio**

**\* @param f CSV-файл, звідки зчитуються дані**

**\* @return масив елементів типу String, що містяться в класі ArrayList**

**\* @throws IOException**

**\*/**

**public ArrayList<String> read\_CSV (File f) throws CSVParseException, IOException{**

**ArrayList<String> array = new ArrayList<String>();**

**FileChannel in\_channel = null;**

**ByteBuffer buf;**

**try {**

**in\_channel = new FileInputStream(f).getChannel();**

**buf = ByteBuffer.allocate(getBuf\_size());**

**while ((in\_channel.read(buf)) != -1 ) {**

**buf.flip();**

**for (int i = 0; i < buf.limit(); i++)**

**buf.clear();**

**}**

**} catch (FileNotFoundException e) {**

**e.printStackTrace();**

**} catch (IOException e) {**

**e.printStackTrace();**

**} finally {**

**try {**

**if (out\_channel != null)**

**out\_channel.close();**

**} catch (IOException e) {**

**e.printStackTrace();**

**}**

**}**

**}**

**/\*\***

**\* @return the buf\_size**

**\*/**

**public static int getBuf\_size() {**

**return buf\_size;**

**}**

**/\*\***

**\* @param buf\_size the buf\_size to set**

**\*/**

**public static void setBuf\_size(int buf\_size) {**

**CSVProcessor.buf\_size = buf\_size;**

**}**

**/\*\***

**\* Метод серіалізує дані об'єкту array у файл**

**\* @param array об'єкт, що серіалізуватиметься**

**\* @param f об'єкт типу File, що містить шлях до файлу, в який серіалізується об'єкт**

**\*/**

**public void serialization (ArrayList<String> array, File f) {**

**ObjectOutputStream oos = null;**

**try {**

**oos = new ObjectOutputStream(new FileOutputStream(f));**

**oos.writeObject(array);**

**} catch (IOException e) {**

**e.printStackTrace();**

**} finally {**

**try {**

**if (oos != null)**

**oos.close();**

**} catch (IOException e) {**

**e.printStackTrace();**

**}**

**}**

**}**

**/\*\***

**\* Метод десеріалізує дані з файлу і повертає у вигляді об'єкту типу ArrayList<String>**

**\* @param f об'єкт типу File, що містить шлях до файлу, в якому міститься серіалізований об'єкт**

**\* @return десеріалізований об'єкт типу ArrayList<String>**

**\*/**

**@SuppressWarnings("unchecked")**

**public ArrayList<String> deserialization (File f) {**

**ArrayList<String> array = null;**

**ObjectInputStream ois = null;**

**try {**

**ois = new ObjectInputStream(new FileInputStream(f));**

**array = (ArrayList<String>) ois.readObject();**

**} catch (FileNotFoundException e) {**

**e.printStackTrace();**

**} catch (IOException e) {**

**e.printStackTrace();**

**} catch (ClassNotFoundException e) {**

**e.printStackTrace();**

**} finally {**

**try {**

**if (ois != null)**

**ois.close();**

**} catch (IOException e) {**

**e.printStackTrace();**

**}**

**}**

**return array;**

**}**

**}**

***JavaDoc***

## C

[**ColorChooser**](file:///D:\--==--\Workspace\Kursova\doc\components\ColorChooser.html) - Клас в пакеті [components](file:///D:\\--==--\\Workspace\\Kursova\\doc\\components\\package-summary.html)

Клас, що реалізує вікно для вибору кольору сектора

[**ColorChooser()**](file:///D:\--==--\Workspace\Kursova\doc\components\ColorChooser.html#ColorChooser()) – Конструктор класу [ColorChooser](file:///D:\--==--\Workspace\Kursova\doc\components\ColorChooser.html)

Конструктор класу, в якому ініціалізується вікно

[**createAndShowGUI(JFrame)**](file:///D:\--==--\Workspace\Kursova\doc\logic\Sector_adding_Panel.html#createAndShowGUI(javax.swing.JFrame)) – Метод класу logic.[Sector\_adding\_Panel](file:///D:\--==--\Workspace\Kursova\doc\logic\Sector_adding_Panel.html)

Створення і відображення вікна

[**CSVParseException**](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVParseException.html) – Клас виключень пакету [logic](file:///D:\\--==--\\Workspace\\Kursova\\doc\\logic\\package-summary.html)

Клас виключень, що викидається при помилці

[**CSVParseException()**](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVParseException.html#CSVParseException()) – Конструктор для класу виключень logic.[CSVParseException](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVParseException.html)

Конструктор, реалізовує виведення помилки на екран

[**CSVProcessor**](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html) – Класу пакету [logic](file:///D:\--==--\Workspace\Kursova\doc\logic\package-summary.html)

У класі CSVProcessor реалізувані методи завантаження/вивантаження CSV-файлу в/з агрегований об’єкт класу ArrayList, в якому знаходяться рядки CSV-файлу.

[**CSVProcessor(int)**](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html#CSVProcessor(int)) – Конструктор для класу logic.[CSVProcessor](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html)

Конструктор, в якому задається значення буфера

## D

[**DE\_ModelDelegate**](file:///D:\--==--\Workspace\Kursova\doc\logic\DE_ModelDelegate.html) – Клас пакету [logic](file:///D:\\--==--\\Workspace\\Kursova\\doc\\logic\\package-summary.html)

Клас в якому реалізоване головне вікно

[**DE\_ModelDelegate()**](file:///D:\--==--\Workspace\Kursova\doc\logic\DE_ModelDelegate.html#DE_ModelDelegate()) – Конструктор класу logic.[DE\_ModelDelegate](file:///D:\--==--\Workspace\Kursova\doc\logic\DE_ModelDelegate.html)

Конструктор, де будується вікно програмного додатку

[**deserialization(File)**](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html#deserialization(java.io.File)) - Метод класу logic.[CSVProcessor](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html)

Метод десеріалізує дані з файлу і повертає у вигляді об'єкту типу ArrayList

[**DiagramPanel**](file:///D:\--==--\Workspace\Kursova\doc\logic\DiagramPanel.html) – Клас пакету  [logic](file:///D:\\--==--\\Workspace\\Kursova\\doc\\logic\\package-summary.html)

Клас, в якому реалізована панель для рисування діаграми

**[DiagramPanel(int[], Color[])](file:///D:\\--==--\\Workspace\\Kursova\\doc\\logic\\DiagramPanel.html" \l "DiagramPanel(int[], java.awt.Color[]))** Конструктор класу logic.[DiagramPanel](file:///D:\--==--\Workspace\Kursova\doc\logic\DiagramPanel.html)

Конструктор, який прорисовує діаграму на основі вхідних параметрів

## M

[**main(String[])**](file:///D:\--==--\Workspace\Kursova\doc\logic\TestIO.html#main(java.lang.String[]))  Статичний метод класу.[TestIO](file:///D:\\--==--\\Workspace\\Kursova\\doc\\logic\\TestIO.html" \o "class in logic)

Метод, в якому здійснюється запуск програми

## P

[**paint(Graphics)**](file:///D:\--==--\Workspace\Kursova\doc\logic\DiagramPanel.html#paint(java.awt.Graphics)) - метод класу logic.[DiagramPanel](file:///D:\--==--\Workspace\Kursova\doc\logic\DiagramPanel.html)

 Метод, що рисує діаграму, з кількістю секторів, що задана в полі count\_sec.

[**parse(ArrayList<String>)**](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html#parse(java.util.ArrayList)) – Метод класу logic.[CSVProcessor](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html)

Метод зчитує з об'єкту типу ArrayList дані і записує їх в об'єкт типу String[][].

## R

[**read\_CSV(File)**](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html#read_CSV(java.io.File)) – Метод класу logic.[CSVProcessor](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html)

Метод зчитує інформацію з CSV-файлу за допомогою потоків вводу пакету java.nio

## S

[**Sector\_adding\_Panel**](file:///D:\--==--\Workspace\Kursova\doc\logic\Sector_adding_Panel.html) - Class in [logic](file:///D:\\--==--\\Workspace\\Kursova\\doc\\logic\\package-summary.html)

Клас, що реалізує вікно додавання секторів

[**Sector\_adding\_Panel(int, JFrame)**](file:///D:\--==--\Workspace\Kursova\doc\logic\Sector_adding_Panel.html#Sector_adding_Panel(int, javax.swing.JFrame)) - Конструктор класу logic.[Sector\_adding\_Panel](file:///D:\--==--\Workspace\Kursova\doc\logic\Sector_adding_Panel.html)

Конструктор вікна, в якому проводиться додавання секторів

[**serialization(ArrayList<String>, File)**](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html#serialization(java.util.ArrayList, java.io.File)) - метод класу logic.[CSVProcessor](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html)

Метод серіалізує дані об'єкту array у файл

[**setBuf\_size(int)**](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html#setBuf_size(int)) – Статичний метод класу logic.[CSVProcessor](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html)

 Метод для задання розміру буфера

[**stateChanged(ChangeEvent)**](file:///D:\--==--\Workspace\Kursova\doc\components\ColorChooser.html#stateChanged(javax.swing.event.ChangeEvent)) - метод класу components.[ColorChooser](file:///D:\--==--\Workspace\Kursova\doc\components\ColorChooser.html)

Метод, який реагує на зміну вибору кольору у палітрі

## T

[**TabbedPane**](file:///D:\--==--\Workspace\Kursova\doc\components\TabbedPane.html) - Class in [components](file:///D:\\--==--\\Workspace\\Kursova\\doc\\components\\package-summary.html)

Клас, у якому реалізовано панель вкладок

**[TabbedPane()](file:///D:\\--==--\\Workspace\\Kursova\\doc\\components\\TabbedPane.html" \l "TabbedPane())** – Конструктор класу components.[TabbedPane](file:///D:\--==--\Workspace\Kursova\doc\components\TabbedPane.html)

Конструктор класу, в якому ініціалізується панель вкладок

## W

[**write\_CSV(ArrayList<String>, File)**](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html#write_CSV(java.util.ArrayList, java.io.File)) - метод класу logic.[CSVProcessor](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html)

Метод записує дані з заданого масиву типу ArrayList в файл за допомогою потоків виводу пакету java.nio

[**write\_CSV\_modif(ArrayList<String[]>, File)**](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html#write_CSV_modif(java.util.ArrayList, java.io.File)) - метод класу logic.[CSVProcessor](file:///D:\--==--\Workspace\Kursova\doc\logic\CSVProcessor.html)

Метод записує дані з заданого масиву типу String[][] в файл за допомогою потоків виводу пакету java.nio