Міністерство освіти і науки України

Національний університет „Львівська політехніка” Кафедра “Електронних обчислювальних машин”



Звіт з лабораторної роботи №3

з дисципліни «Кросплатформні засоби програмування» на тему:

«Спадкування та інтерфейси»

Виконала: ст.групи KI-305

Лесяк Х.В.

Перевірив:

Іванов Ю.С.

**Львів 2023**

**Мета:** ознайомитися з спадкуванням та інтерфейсами у мові Java.

**ЗАВДАННЯ :**

1. Написати та налагодити програму на мові Java, що розширює клас, що реалізований у лабораторній роботі №3, для реалізації предметної області заданої варіантом. Суперклас, що реалізований у лабораторній роботі №3, зробити абстрактним. Розроблений підклас має забезпечувати механізми свого коректного функціонування та реалізовувати мінімум один інтерфейс. Програма має розміщуватися в пакеті Група.Прізвище.Lab4 та володіти коментарями, які дозволять автоматично згенерувати документацію до розробленого пакету.

2. Автоматично згенерувати документацію до розробленого пакету.

3. Скласти звіт про виконану роботу з приведенням тексту програми, результату її виконання та фрагменту згенерованої документації.

4. Дати відповідь на контрольні запитання.

**Варіант: 18**



*Лістин*

*Scanner.java*

package KI305.Lesiak.Lab3;  
  
import java.io.File;  
import java.io.FileNotFoundException;  
import java.io.PrintWriter;  
  
public abstract class Scanner {  
 private String model;  
 private Resolution resolution; // Роздільна здатність сканування  
 private ScanningSpeed speed; // Швидкість сканування  
 private ConnectionType connectionType; // Тип з'єднання  
 PrintWriter fout; // Вивід до файлу  
  
  
 // Конструктор класу Scanner без параметрів  
 public Scanner() throws FileNotFoundException {  
 model = "Невідомо";  
 resolution = new Resolution(0, 0);  
 speed = new ScanningSpeed(0.0, "стр/хв");  
 connectionType = new ConnectionType("Невідомо");  
 fout = new PrintWriter(new File("ScannerLog.txt")); // Створення нового файлу  
 }  
  
 // Конструктор класу Scanner з параметрами  
 public Scanner(String model, Resolution resolution, ScanningSpeed speed, ConnectionType connectionType) throws FileNotFoundException {  
 this.model = model;  
 this.resolution = resolution;  
 this.speed = speed;  
 this.connectionType = connectionType;  
 fout = new PrintWriter(new File("ScannerLog.txt")); // Створення нового файлу  
 }  
  
 public String getModel() {  
 return model;  
 }  
  
 public void setModel(String model) {  
 this.model = model;  
 }  
  
 public Resolution getResolution() {  
 return resolution;  
 }  
  
 public void setResolution(Resolution resolution) {  
 this.resolution = resolution;  
 }  
  
 public ScanningSpeed getSpeed() {  
 return speed;  
 }  
  
 public void setSpeed(ScanningSpeed speed) {  
 this.speed = speed;  
 }  
  
 public ConnectionType getConnectionType() {  
 return connectionType;  
 }  
  
 public void setConnectionType(ConnectionType connectionType) {  
 this.connectionType = connectionType;  
 }  
  
 public void setFout(PrintWriter fout) {  
 this.fout = fout;  
 }  
  
 public abstract void scanDocument(String documentName);  
  
  
 // Додатковий метод для відображення інформації про сканер  
 public abstract void displayScannerInfo();  
  
  
 // Додатковий метод для закриття виводу до файлу  
 public void closeLogFile() {  
 fout.close();  
 }  
}

*CopierScanner.java*

package KI305.Lesiak.Lab3;  
  
import java.io.FileNotFoundException;  
  
// Інтерфейс для функціоналу сканера  
interface ScannerFunctionality {  
 void scanDocument(String documentName);  
 void displayScannerInfo();  
  
}  
  
// Інтерфейс для функціоналу копіювального апарату  
interface Copier {  
 void copyDocument(String documentName);  
}  
  
// Підклас, який реалізує функціонал копіювального апарату та сканера  
public class CopierScanner extends Scanner implements Copier, ScannerFunctionality {  
  
  
 // Конструктор  
 public CopierScanner(String model, Resolution resolution, ScanningSpeed speed, ConnectionType connectionType) throws FileNotFoundException {  
 super(model, resolution, speed, connectionType);  
  
 }  
 public CopierScanner() throws FileNotFoundException {  
 super();  
  
 }  
  
  
 // Реалізація методів з інтерфейсу Copier  
 @Override  
 public void copyDocument(String documentName) {  
 // Логіка копіювання документа  
 fout.println("Копіюємо документ '" + documentName + "'..."); // Запис у файл  
 System.*out*.println("Копіюємо документ '" + documentName + "'...");  
 fout.println("Копіювання завершено."); // Запис у файл  
 System.*out*.println("Копіювання завершено.");  
 }  
  
 // Реалізація методів з інтерфейсу ScannerFunctionality  
 @Override  
 public void scanDocument(String documentName) {  
 // Логіка сканування документа  
 fout.println("Скануємо документ '" + documentName + "'..."); // Запис у файл  
 System.*out*.println("Скануємо документ '" + documentName + "'...");  
 fout.println("Сканування завершено."); // Запис у файл  
 System.*out*.println("Сканування завершено.");  
 }  
  
 @Override  
 public void displayScannerInfo() {  
 // Логіка виводу інформації про сканер  
 fout.println("Інформація про сканер:"); // Запис у файл  
 System.*out*.println("Інформація про сканер:");  
 fout.println("Модель: " + getModel()); // Запис у файл  
 System.*out*.println("Модель: " + getModel());  
 fout.println("Роздільна здатність: " + getResolution()); // Запис у файл  
 System.*out*.println("Роздільна здатність: " + getResolution());  
 fout.println("Швидкість сканування: " + getSpeed()); // Запис у файл  
 System.*out*.println("Швидкість сканування: " + getSpeed());  
 fout.println("Тип з'єднання: " + getConnectionType()); // Запис у файл  
 System.*out*.println("Тип з'єднання: " + getConnectionType());  
 }  
}

*CopierScannerApp.java*

package KI305.Lesiak.Lab3;  
  
import java.io.FileNotFoundException;  
  
public class CopierScannerApp {  
 public static void main(String[] args) {  
 try {  
 // Створення об'єкту CopierScanner  
 CopierScanner copierScanner = new CopierScanner("Model1", new Resolution(1200, 800),  
 new ScanningSpeed(5.0, "стр/хв"), new ConnectionType("USB"));  
  
 // Виклик методу з інтерфейсу Copier  
 copierScanner.copyDocument("Document1");  
  
 // Виклик методів з інтерфейсу ScannerFunctionality  
 copierScanner.scanDocument("Document2");  
 copierScanner.displayScannerInfo();  
  
 // Закриття файлу  
 copierScanner.closeLogFile();  
  
 } catch (FileNotFoundException e) {  
 e.printStackTrace();  
 }  
 }  
}

**Результат**:

*Зображення, що містить текст, знімок екрана, Шрифт, дизайн

Автоматично згенерований опис*

**Висновок:** виконавши цю лабораторну роботу, я ознайомився з спадкуванням та інтерфейсами у мові Java.