МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ

ВІДОКРЕМЛЕНИЙ СТРУКТУРНИЙ ПІДРОЗДІЛ

«ПОЛТАВСЬКИЙ ПОЛІТЕХНІЧНИЙ ФАХОВИЙ КОЛЕДЖ

НАЦІОНАЛЬНОГО ТЕХНІЧНОГО УНІВЕРСИТЕТУ

«ХАРКІВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ»

Циклова комісія дисциплін програмної інженерії

**ЗВІТ**

з навчальної практики

«Практика з об’єктно-орієнтованого програмування»

Виконав: здобувач освіти 3 курсу, групи 35 напрям підготовки: 121 «Інженерія програмного забезпечення» спеціалізації «Розробка програмного забезпечення»

Любченко О. В.

(прізвище та ініціали)

Керівник

Зелінська А. О.

(прізвище та ініціали)

(підпис)

Полтава – 2024

**ЗМІСТ**

[ВСТУП 3](#_Toc161613903)

[1. ІНДИВІДУАЛЬНЕ ЗАВДАННЯ №1 4](#_Toc161613904)

[1.1. Постановка завдання 4](#_Toc161613905)

[1.2. Код програми 4](#_Toc161613906)

[1.3. Результат роботи програми 6](#_Toc161613907)

[2. ІНДИВІДУАЛЬНЕ ЗАВДАННЯ №2 7](#_Toc161613908)

[2.1. Постановка завдання 7](#_Toc161613909)

[2.2. Код програми 7](#_Toc161613910)

[2.3. Результат роботи програми 9](#_Toc161613911)

[3. ІНДИВІДУАЛЬНЕ ЗАВДАННЯ №3 11](#_Toc161613912)

[3.1. Постановка завдання 11](#_Toc161613913)

[3.2. Код програми 11](#_Toc161613914)

[3.3. Результат роботи програми 13](#_Toc161613915)

[4. ІНДИВІДУАЛЬНЕ ЗАВДАННЯ №4 15](#_Toc161613916)

[4.1. Постановка завдання 15](#_Toc161613917)

[4.2. Код програми 15](#_Toc161613918)

[4.3. Результат роботи програми 16](#_Toc161613919)

[5. ІНДИВІДУАЛЬНЕ ЗАВДАННЯ №5 18](#_Toc161613920)

[5.1. Постановка завдання 18](#_Toc161613921)

[5.2. Код програми 18](#_Toc161613922)

[5.3. Результат роботи програми 19](#_Toc161613923)

[ВИСНОВКИ 21](#_Toc161613924)

[СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ 22](#_Toc161613925)

# **ВСТУП**

Практика з об'єктно-орієнтованого програмування є частиною навчального процесу і сприяє саморозвитку студентів з метою розвитку професійних навичок та набуття студентами першого досвіду у сфері професійної діяльності. Метою даної практики є не тільки розвиток професійних навичок, а й підвищення майбутньої професійної кваліфікації. Практична підготовка дає можливість ознайомитися з обраною спеціалізацією та набути практичних навичок для виконання основних завдань за цією спеціалізацією, а також закріпити теоретичні знання та сформувати потребу в систематичному оновленні знань і творчому застосуванні їх на практиці.

Завданнями практики об'єктно-орієнтованого програмування є наступні:

* Поглиблення та закріплення теоретичних знань з об'єктно-орієнтованого програмування, теорії алгоритмів та структур даних, технологій розробки програмного забезпечення та інших спеціалізованих областей;
* Набуття практичних навичок об'єктно-орієнтованого програмування;
* Вивчення основних інструментів об'єктно-орієнтованого програмування та їх застосування для розробки реальних програмних продуктів;
* Аналіз об'єктно-орієнтованого програмного коду;
* Практичні навички написання об'єктно-орієнтованого програмного коду;
* Ознайомлення з життєвим циклом розробки програмного продукту.
* Написання звіту - набору проектних документів, що відображають основні результати реалізації;
* Розробка програмного продукту.

# **1. ІНДИВІДУАЛЬНЕ ЗАВДАННЯ №1**

## **1.1. Постановка завдання**

* Розробити клас, що серіалізується, для зберігання параметрів і результатів обчислень.
* Використовуючи агрегування, розробити клас для знаходження рішення задачі.
* Розробити клас для демонстрації в діалоговому режимі збереження та відновлення стану об'єкта, використовуючи серіалізацію. Показати особливості використання transient полів.
* Розробити клас для тестування коректності результатів обчислень та серіалізації/десеріалізації.

## **1.2. Код програми**

|  |
| --- |
| **package ex01;**  **import java.io.BufferedReader;**  **import java.io.IOException;**  **import java.io.InputStreamReader;**  **public class Main {**  **private Calc calc = new Calc();**  **private void menu() {**  **String s = null;**  **BufferedReader in = new BufferedReader(new InputStreamReader(System.in));**  **do {**  **do {**  **System.out.println("Enter command...");**  **System.out.print("'q'uit, 'v'iew, 'g'enerate, 's'ave, 'r'estore: ");**  **try {**  **s = in.readLine();**  **} catch (IOException e) {**  **System.out.println("Error: " + e);**  **System.exit(0);**  **}**  **} while (s.length() != 1);**  **switch (s.charAt(0)) {**  **case 'q':**  **System.out.println("Exit.");**  **break;**  **case 'v':**  **System.out.println("View current.");**  **calc.show();**  **break;**  **case 'g':**  **System.out.println("Random generation.");**  **calc.init(Math.random() \* 360.0, Math.random() \* 100.0);**  **calc.show();**  **break;**  **case 's':**  **System.out.println("Save current.");**  **try {**  **calc.save();**  **} catch (IOException e) {**  **System.out.println("Serialization error: " + e);**  **}**  **calc.show();**  **break;**  **case 'r':**  **System.out.println("Restore last saved.");**  **try {**  **calc.restore();**  **} catch (Exception e) {**  **System.out.println("Serialization error: " + e);**  **}**  **calc.show();**  **break;**  **default:**  **System.out.print("Wrong command. ");**  **}**  **} while (s.charAt(0) != 'q');**  **}**  **public static void main(String[] args) {**  **Main main = new Main();**  **main.menu();**  **}**  **}** |

Таблиця 1.1 – Код файлу Main.java

Повний код програми доступний на веб-сервісі GitHub за посиланням, що зображене на рисунку 1.1.



Рисунок 1.1 – Посилання на теку програми до завдання 1

## **1.3. Результат роботи програми**

|  |
| --- |
| **Windows PowerShell**  **Copyright (C) Microsoft Corporation. All rights reserved.**  **Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows**  **PS D:\Work folder\College\3-PPPC-Practic-OOP\Task-2\demo> & 'C:\Users\Uzer\.vscode\extensions\redhat.java-1.28.1-win32-x64\jre\17.0.10-win32-x86\_64\bin\java.exe' '@C:\Users\Uzer\AppData\Local\Temp\cp\_1bgwho2ntawrad6hrz9g780wv.argfile' 'ex01.Test'**  **Restoring Result from file:**  **Result restored successfully.**  **Restored Result:**  **Mass: 198.96328154670874 kg, Energy: 19498.401591577458 J**  **Testing Calculation:**  **Calculated Energy: 19498.401591577458 J**  **Mass: 198.96328154670874 kg, Energy: 19498.401591577458 J**  **Testing Serialization:**  **Calculation result saved successfully.**  **Calculation result restored successfully.**  **Restored Result:**  **Mass: 198.96328154670874 kg, Energy: 19498.401591577458 J**  **PS D:\Work folder\College\3-PPPC-Practic-OOP\Task-2\demo>** |

Таблиця 1.2 – Результат роботи програми

# **2. ІНДИВІДУАЛЬНЕ ЗАВДАННЯ №2**

## **2.1. Постановка завдання**

* Як основа використовувати вихідний текст проекту попередньої лабораторної роботи. Забезпечити розміщення результатів обчислень у колекції з можливістю збереження/відновлення.
* Використовуючи шаблон проектування Factory Method (Virtual Constructor), розробити ієрархію, що передбачає розширення рахунок додавання нових відображуваних класів.
* Розширити ієрархію інтерфейсом "фабрикованих" об'єктів, що представляє набір методів для відображення результатів обчислень.
* Реалізувати ці методи виведення результатів у текстовому виде.
* Розробити тареалізувати інтерфейс для "фабрикуючого" методу.

## **2.2. Код програми**

|  |
| --- |
| **package ex03;**  **import java.io.BufferedReader;**  **import java.io.IOException;**  **import java.io.InputStreamReader;**  **public class Main {**  **private Calc calc = new Calc();**  **private void menu() {**  **String s = null;**  **BufferedReader in = new BufferedReader(new InputStreamReader(System.in));**  **do {**  **do {**  **System.out.println("Enter command...");**  **System.out.print("'q'uit, 'v'iew, 'g'enerate, 's'ave, 'r'estore: ");**  **try {**  **s = in.readLine();**  **} catch (IOException e) {**  **System.out.println("Error: " + e);**  **System.exit(0);**  **}**  **} while (s.length() != 1);**  **switch (s.charAt(0)) {**  **case 'q':**  **System.out.println("Exit.");**  **break;**  **case 'v':**  **System.out.println("View current.");**  **Displayable displayable = calc.createDisplayable();**  **displayable.display();**  **break;**  **case 'g':**  **System.out.println("Random generation.");**  **calc.init(Math.random() \* 360.0, Math.random() \* 100.0);**  **calc.show();**  **break;**  **case 's':**  **System.out.println("Save current.");**  **try {**  **calc.save();**  **} catch (IOException e) {**  **System.out.println("Serialization error: " + e);**  **}**  **calc.show();**  **break;**  **case 'r':**  **System.out.println("Restore last saved.");**  **try {**  **calc.restore();**  **} catch (Exception e) {**  **System.out.println("Serialization error: " + e);**  **}**  **calc.show();**  **break;**  **default:**  **System.out.print("Wrong command. ");**  **}**  **} while (s.charAt(0) != 'q');**  **}**  **public static void main(String[] args) {**  **Main main = new Main();**  **main.menu();**  **}**  **}** |

Таблиця 2.1 – Код файлу Main.java

Повний код програми доступний на веб-сервісі GitHub за посиланням, що зображене на рисунку 2.1.



Рисунок 2.1 – Посилання на теку програми до завдання 2

## **2.3. Результат роботи програми**

|  |
| --- |
| **Windows PowerShell**  **Copyright (C) Microsoft Corporation. All rights reserved.**  **Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows**  **PS D:\Work folder\College\3-PPPC-Practic-OOP\Task-3\demo> & 'C:\Users\Uzer\.vscode\extensions\redhat.java-1.28.1-win32-x64\jre\17.0.10-win32-x86\_64\bin\java.exe' '@C:\Users\Uzer\AppData\Local\Temp\cp\_9xuxezt5ch2lr7g8ia62itw29.argfile' 'ex03.Test'**  **Choose an option:**  **1. Initialize and view**  **2. Save and restore**  **3. Exit**  **1**  **Enter mass:**  **3434**  **Enter height:**  **324234**  **Mass: 3434.0 kg, Energy: 1.0911511648800001E10 J**  **Choose an option:**  **1. Initialize and view**  **2. Save and restore**  **3. Exit**  **2**  **Saving...**  **Restoring...**  **Mass: 3434.0 kg, Energy: 1.0911511648800001E10 J**  **Choose an option:**  **1. Initialize and view**  **2. Save and restore**  **3. Exit**  **3**  **Exiting...**  **PS D:\Work folder\College\3-PPPC-Practic-OOP\Task-3\demo>** |

Таблиця 2.2 – Результат роботи програми

# **3. ІНДИВІДУАЛЬНЕ ЗАВДАННЯ №3**

## **3.1. Постановка завдання**

* За основу використовувати вихідний текст проекту попередньої лабораторної роботи Використовуючи шаблон проектування Factory Method (Virtual Constructor), розширити ієрархію похідними класами, реалізують методи для подання результатів у вигляді текстової таблиці. Параметри відображення таблиці мають визначатися користувачем.
* Продемонструвати заміщення (перевизначення, overriding), поєднання (перевантаження, overloading), динамічне призначення методів (Пізнє зв'язування, поліморфізм, dynamic method dispatch).
* Забезпечити діалоговий інтерфейс із користувачем.
* Розробити клас для тестування основної функціональності.

## **3.2. Код програми**

|  |
| --- |
| **package ex04;**  **import java.io.BufferedReader;**  **import java.io.IOException;**  **import java.io.InputStreamReader;**  **public class Main {**  **public static void main(String[] args) {**  **Calc calc = new Calc();**  **BufferedReader in = new BufferedReader(new InputStreamReader(System.in));**  **String input;**  **try {**  **do {**  **System.out.println("Choose an option:");**  **System.out.println("1. Initialize and view");**  **System.out.println("2. Save and restore");**  **System.out.println("3. Exit");**  **input = in.readLine();**  **switch (input) {**  **case "1":**  **System.out.println("Enter mass:");**  **double mass = Double.parseDouble(in.readLine());**  **System.out.println("Enter height:");**  **double height = Double.parseDouble(in.readLine());**  **calc.init(mass, height);**  **System.out.println("Enter display format (default or table):");**  **String format = in.readLine();**  **calc.show(format);**  **break;**  **case "2":**  **System.out.println("Saving...");**  **calc.save();**  **System.out.println("Restoring...");**  **try {**  **calc.restore();**  **} catch (Exception e) {**  **System.out.println("Error restoring: " + e.getMessage());**  **}**  **calc.show("default");**  **break;**  **case "3":**  **System.out.println("Exiting...");**  **break;**  **default:**  **System.out.println("Invalid option");**  **break;**  **}**  **} while (!input.equals("3"));**  **} catch (IOException e) {**  **e.printStackTrace();**  **}**  **}**  **}** |

Таблиця 3.1 – Код файлу Main.java

Повний код програми доступний на веб-сервісі GitHub за посиланням, що зображене на рисунку 3.1.



Рисунок 3.1 – Посилання на теку програми до завдання 3

## **3.3. Результат роботи програми**

|  |
| --- |
| **Windows PowerShell**  **Copyright (C) Microsoft Corporation. All rights reserved.**  **Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows**  **PS D:\Work folder\College\3-PPPC-Practic-OOP\Task-4> & 'C:\Users\Uzer\.vscode\extensions\redhat.java-1.28.1-win32-x64\jre\17.0.10-win32-x86\_64\bin\java.exe' '@C:\Users\Uzer\AppData\Local\Temp\cp\_4cat7q7h2xdzmduqvtymj4eja.argfile' 'ex04.Test'**  **Choose an option:**  **1. Initialize and view**  **2. Save and restore**  **3. Test**  **4. Exit**  **1**  **Enter mass:**  **454334**  **Enter height:**  **435345435**  **Enter display format (default or table):**  **table**  **| Mass (kg) | Energy (J) |**  **|-----------|-------------|**  **| 454334,00 | 1938363882079842,00 |**  **Choose an option:**  **1. Initialize and view**  **2. Save and restore**  **3. Test**  **4. Exit**  **2**  **Saving...**  **Restoring...**  **Mass: 454334.0 kg, Energy: 1.938363882079842E15 J**  **Choose an option:**  **1. Initialize and view**  **2. Save and restore**  **3. Test**  **4. Exit**  **3**  **Testing...**  **Choose an option:**  **1. Initialize and view**  **2. Save and restore**  **3. Test**  **4. Exit**  **4**  **Exiting...**  **PS D:\Work folder\College\3-PPPC-Practic-OOP\Task-4>** |

Таблиця 3.2 – Результат роботи програми

# **4. ІНДИВІДУАЛЬНЕ ЗАВДАННЯ №4**

## **4.1. Постановка завдання**

* Реалізувати можливість скасування (undo) операцій (команд).
* Продемонструвати поняття "макрокоманда".
* При розробці програми використовувати шаблон Singletone.
* Забезпечити діалоговий інтерфейс із користувачем.
* Розробити клас для тестування функціональності програми.

## **4.2. Код програми**

|  |
| --- |
| **package ex05;**  **import java.util.Scanner;**  **public class Main {**  **public static void main(String[] args) {**  **Scanner scanner = new Scanner(System.in);**  **CommandManager commandManager = new CommandManager();**  **MacroCommand macroCommand = new MacroCommand();**  **ColorCommand colorCommand = new ColorCommand();**  **boolean running = true;**  **while (running) {**  **System.out.println("Enter 'generate' to generate a color, 'undo' to undo last operation, 'exit' to exit:");**  **String input = scanner.nextLine();**  **switch (input) {**  **case "generate":**  **commandManager.executeCommand(colorCommand);**  **macroCommand.addCommand(colorCommand);**  **break;**  **case "undo":**  **commandManager.undoLastCommand();**  **break;**  **case "exit":**  **running = false;**  **break;**  **default:**  **System.out.println("Invalid command.");**  **break;**  **}**  **}**  **scanner.close();**  **}**  **}** |

Таблиця 4.1 – Код файлу Main.java

Повний код програми доступний на веб-сервісі GitHub за посиланням, що зображене на рисунку 4.1.



Рисунок 4.1 – Посилання на теку програми до завдання 4

## **4.3. Результат роботи програми**

|  |
| --- |
| **Windows PowerShell**  **Copyright (C) Microsoft Corporation. All rights reserved.**  **Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows**  **PS D:\Work folder\College\3-PPPC-Practic-OOP\Task-5> & 'C:\Users\Uzer\.vscode\extensions\redhat.java-1.28.1-win32-x64\jre\17.0.10-win32-x86\_64\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'D:\Work folder\College\3-PPPC-Practic-OOP\Task-5\demo\target\classes' 'ex05.Main'**  **Enter 'generate' to generate a color, 'undo' to undo last operation, 'exit' to exit:**  **generate**  **Generated color: Purple**  **Enter 'generate' to generate a color, 'undo' to undo last operation, 'exit' to exit:**  **generate**  **Generated color: Yellow**  **Enter 'generate' to generate a color, 'undo' to undo last operation, 'exit' to exit:**  **undo**  **Undoing last operation. Previous color: Purple**  **Enter 'generate' to generate a color, 'undo' to undo last operation, 'exit' to exit:**  **exit**  **PS D:\Work folder\College\3-PPPC-Practic-OOP\Task-5>** |

Таблиця 4.2 – Результат роботи програми

# **5. ІНДИВІДУАЛЬНЕ ЗАВДАННЯ №5**

## **5.1. Постановка завдання**

* Продемонструвати можливість паралельної обробки елементів колекції (пошук мінімуму, максимуму, обчислення середнього значення, відбір за критерієм, статистична обробка тощо).
* Управління чергою завдань (команд) реалізувати за допомогою шаблону Worker Thread.

## **5.2. Код програми**

|  |
| --- |
| **package ex06;**  **import java.util.ArrayList;**  **import java.util.Collections;**  **import java.util.List;**  **import java.util.concurrent.\*;**  **public class Main {**  **public static void main(String[] args) {**  **List<Integer> data = new ArrayList<>();**  **for (int i = 0; i <= 100; i++) {**  **data.add(i);**  **}**  **ExecutorService executor = Executors.newFixedThreadPool(4);**  **Future<Integer> minFuture = executor.submit(() -> Collections.min(data));**  **Future<Integer> maxFuture = executor.submit(() -> Collections.max(data));**  **Future<Double> avgFuture = executor.submit(() -> data.stream().mapToInt(Integer::intValue).average().orElse(0));**  **Future<Integer> sumFuture = executor.submit(() -> data.stream().filter(x -> x > 50).mapToInt(Integer::intValue).sum());**  **try {**  **int min = minFuture.get();**  **int max = maxFuture.get();**  **double avg = avgFuture.get();**  **int sum = sumFuture.get();**  **System.out.println("Мінімум: " + min);**  **System.out.println("Максимум: " + max);**  **System.out.println("Середнє: " + avg);**  **System.out.println("Сума елементів більших за 50: " + sum);**  **} catch (InterruptedException | ExecutionException e) {**  **e.printStackTrace();**  **}**  **executor.shutdown();**  **}**  **}** |

Таблиця 5.1 – Код файлу Main.java

Повний код програми доступний на веб-сервісі GitHub за посиланням, що зображене на рисунку 5.1.



Рисунок 5.1 – Посилання на теку програми до завдання 5

## **5.3. Результат роботи програми**

|  |
| --- |
| **Windows PowerShell**  **Copyright (C) Microsoft Corporation. All rights reserved.**  **Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows**  **PS D:\Work folder\College\3-PPPC-Practic-OOP\Task-6> & 'C:\Users\Uzer\.vscode\extensions\redhat.java-1.28.1-win32-x64\jre\17.0.10-win32-x86\_64\bin\java.exe' '-XX:+ShowCodeDetailsInExceptionMessages' '-cp' 'D:\Work folder\College\3-PPPC-Practic-OOP\Task-6\demo\target\classes' 'ex06.Main'**  **Мінімум: 0**  **Максимум: 100**  **Середнє: 50.0**  **Сума елементів більших за 50: 3775**  **PS D:\Work folder\College\3-PPPC-Practic-OOP\Task-6>** |

Таблиця 4.2 – Результат роботи програми

# **ВИСНОВКИ**

# **СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ**

1. Методичні вказівки до виконання дипломних робіт для здобувачів освіти спеціальності 121 Інженерія програмного забезпечення спеціальності «Розробка програмного забезпечення» / Укл.: Бабич Ол-др. В., Бабич Олена В. – Полтава: ВСП «ППФК НТУ “ХПІ”, 2020. – 51 с.
2. Academy i. Java Programming: Your Step by Step Guide to Easily Learn Java in 7 Days. Independently Published, 2017.
3. Empowering the world to develop technology through collective knowledge - Stack Overflow. Empowering the world to develop technology through collective knowledge - Stack Overflow. URL: https://try.stackoverflow.co/ (дата звернення: 17.03.2024).
4. GitHub: Let’s build from here. GitHub. URL: https://github.com/ (дата звернення: 17.03.2024).
5. Grunitzky G. Java Programming: Java. Independently Published, 2017.
6. Lassoff M. Java Programming for Beginners: Learn the fundamentals of programming with Java. Packt Publishing, 2017. 318 с.
7. McBride P. K. Java Made Simple (Made Simple Programming). 2-ге вид. Made Simple, 2002. 208 с.
8. Norton W. Java Programming for Beginners: A Piratical Beginners Guide to Learn Programming, Fundamentals and Code. Independently Published, 2020. 145 с.