

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Інститут комп'ютерних наук та інформаційних технологій
Кафедра автоматизованих систем управління



Звіт
до лабораторної роботи №2
з дисципліни
Прикладне програмування
на тему:
«Робота з класами»

Виконав: студент КН-206
Филипчук Богдан

Прийняв: Скрибайло-Леськів Д. Ю.

Лабораторна робота № 2

Варіант №6

Тема роботи: Робота з класами

Завдання лабораторної роботи

1. Створити проект, що складається з двох класів: основного (Main) та класу для представлення об'єкта відповідно специфікації. Кожний клас повинен бути розміщений у окремому пакеті. У створеному класі визначити приватні поля для зберігання указаних даних, конструктори для створення об'єктів та відкриті методи setValue(), getValue(), toString() для доступу до полів об'єкту.

2. В основному класі програми визначити методи, що створюють масив об'єктів. Задати критерії вибору даних та вивести ці дані на консоль. Для кожного критерію створити окремий метод.

3. Виконати програму, та пересвідчитись, що дані зберігаються та коректно виводяться на екран відповідно до вказаних критеріїв.

Індивідуальне завдання:

6	<p>Phone: id, Прізвище, Ім'я, По батькові, Номер рахунку, Час міських розмов, Час міжміських розмов.</p> <p>Скласти масив об'єктів. Вивести:</p> <ul style="list-style-type: none">a) відомості про абонентів, у яких час міських розмов перевищує заданий;b) відомості про абонентів, які користувались міжміським зв'язком;c) відомості про абонентів чий номер рахунку знаходиться у вказаному діапазоні.
----------	---

```

package main;

import myclass.Phone;

import java.util.ArrayList;
import java.util.Scanner;

public class Main {
    private static ArrayList<Phone> phones = new ArrayList<>();

    /**
     * Заповнює поля об'єкту класу Phone даними від користувача
     *
     * @return Об'єкт з введеними даними
     */
    public static Phone CreateObject() {
        Phone input = new Phone();
        Scanner scan = new Scanner(System.in);
        input.setId(scan.nextInt());
        input.setLName(scan.next());
        input.setFName(scan.next());
        input.setFatherName(scan.next());
        input.setAccountNumber(scan.nextInt());
        input.setDurationWithinCity(scan.nextInt());
        input.setDurationOutCity(scan.nextInt());

        return input;
    }

    /**
     * Створює і заповнює масив об'єктів класу Phone
     *
     * @return динамічний масив об'єктів Phone
     */
    public static ArrayList<Phone> CreateArray() {
        System.out.print("Кількість користувачів: ");
        Scanner scan = new Scanner(System.in);
        int numb = scan.nextInt();

        ArrayList<Phone> phones = new ArrayList<>();
        System.out.println("id Прізвище Ім'я По батькові Номер рахунку Час міських розмов Час міжміських розмов");
        for (int i = 0; i < numb; i++) {
            phones.add(CreateObject());
        }

        return phones;
    }

    /**
     * Виводить відомості про абонентів, у яких час міських розмов перевищує заданий
     */
    public static void GetMoreThanInputed() {
        System.out.print("\nЗадайте час міських розмов: ");
        Scanner scan = new Scanner(System.in);
        int asked = scan.nextInt();

        System.out.println("Абоненти, у яких час міських розмов перевищує заданий:");
        for (Phone curr : phones) {
            if (curr.getDurationWithinCity() > asked)
                System.out.println(curr);
        }
    }
}

```

```

/**
 * Виводить відомості про абонентів, які користувались міжміським зв'язком
 */
public static void CheckOutCityUsage() {
    System.out.println("\nАбоненти, які користувались міжміським зв'язком: ");
    for (Phone curr: phones) {
        if (curr.getDurationOutCity() > 0)
            System.out.println(curr);
    }
}

/**
 * Виводить відомості про абонентів чий номер рахунку знаходиться вказаному
діапазоні
 */
public static void InRange() {
    Scanner scan = new Scanner(System.in);
    System.out.print("\nЗадайте діапазон номера рахунку. Від: ");
    int from = scan.nextInt();
    System.out.print("До: ");
    int to = scan.nextInt();

    for (Phone curr: phones) {
        if (curr.getAccountNumber() > from && curr.getAccountNumber() < to)
            System.out.println(curr);
    }
}

public static void main(String[] args) {
    phones = CreateArray();

    GetMoreThanInputed();
    CheckOutCityUsage();
    InRange();
}
}

```

```
package myclass;

public class Phone {
    /**
     * ID абонента
     */
    private int id;
    /**
     * Прізвище абонента
     */
    private String LName;
    /**
     * Ім'я абонента
     */
    private String FName;
    /**
     * По батькові абонента
     */
    private String fatherName;
    /**
     * Номер рахунку
     */
    private int accountNumber;
    /**
     * Час міських розмов
     */
    private int durationWithinCity;
    /**
     * Час міжміських розмов
     */
    private int durationOutCity;

    public Phone() {}

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }

    public String getLName() {
        return LName;
    }

    public void setLName(String LName) {
        this.LName = LName;
    }

    public String getFName() {
        return FName;
    }

    public void setFName(String FName) {
        this.FName = FName;
    }

    public String getFatherName() {
        return fatherName;
    }

    public void setFatherName(String fatherName) {
        this.fatherName = fatherName;
    }
}
```

```

public int getAccountNumber() {
    return accountNumber;
}

public void setAccountNumber(int accountNumber) {
    this.accountNumber = accountNumber;
}

public int getDurationWithinCity() {
    return durationWithinCity;
}

public void setDurationWithinCity(int durationWithinCity) {
    this.durationWithinCity = durationWithinCity;
}

public int getDurationOutCity() {
    return durationOutCity;
}

public void setDurationOutCity(int durationOutCity) {
    this.durationOutCity = durationOutCity;
}

@Override
public String toString() {
    return id + " - " + LName + " " + FName + " " + fatherName +
        "; рахунок - " + accountNumber +
        "; час міських розмов - " + durationWithinCity +
        "; час міжміських розмов - " + durationOutCity;
}
}

```

Кількість користувачів: 10

id Прізвище Ім'я По батькові Номер рахунку Час міських розмов Час міжміських розмов

```

21 Филипчук Богдан Володимирович 14785 20 0
64 Корнійчук Олександр Петрович 36842 98 75
32 Степанюк Володимир Богданович 19638 15 130
49 Либа Роман Віталійович 98468 51 48
65 Гладюк Юрій Олександрович 56842 200 0
98 Фіцай Вадим Юрійович 64483 15 172
17 Тараненко Назар Олексійович 51543 180 0
72 Нескорожена Карина Віталіївна 65698 0 176
01 Мітнічук Віталій Ігорович 47864 250 300
82 Демчина Тарас Богданович 74461 87 13

```

Задайте час міських розмов: 100

Абоненти, у яких час міських розмов перевищує заданий:

```

65 - Гладюк Юрій Олександрович; рахунок - 56842; час міських розмов - 200; час міжміських розмов - 0
17 - Тараненко Назар Олексійович; рахунок - 51543; час міських розмов - 180; час міжміських розмов - 0
1 - Мітнічук Віталій Ігорович; рахунок - 47864; час міських розмов - 250; час міжміських розмов - 300

```

Абоненти, які користувались міжміським зв'язком:

```

64 - Корнійчук Олександр Петрович; рахунок - 36842; час міських розмов - 98; час міжміських розмов - 75
32 - Степанюк Володимир Богданович; рахунок - 19638; час міських розмов - 15; час міжміських розмов - 130
49 - Либа Роман Віталійович; рахунок - 98468; час міських розмов - 51; час міжміських розмов - 48
98 - Фіцай Вадим Юрійович; рахунок - 64483; час міських розмов - 15; час міжміських розмов - 172
72 - Нескорожена Карина Віталіївна; рахунок - 65698; час міських розмов - 0; час міжміських розмов - 176
1 - Мітнічук Віталій Ігорович; рахунок - 47864; час міських розмов - 250; час міжміських розмов - 300
82 - Демчина Тарас Богданович; рахунок - 74461; час міських розмов - 87; час міжміських розмов - 13

```

Задайте діапазон номера рахунку. Від: 50000

До: 80000

65 - Гладюк Юрій Олександрович; рахунок - 56842; час міських розмов - 200; час міжміських розмов - 0

98 - Фіцай Вадим Юрійович; рахунок - 64483; час міських розмов - 15; час міжміських розмов - 172

17 - Тараненко Назар Олексійович; рахунок - 51543; час міських розмов - 180; час міжміських розмов - 0

72 - Нескорожена Карина Віталіївна; рахунок - 65698; час міських розмов - 0; час міжміських розмов - 176

82 - Демчина Тарас Богданович; рахунок - 74461; час міських розмов - 87; час міжміських розмов - 13

Висновок

Метою даної лабораторної роботи було навчитися працювати з класами на мові програмування Java. Створив два класи Main та Phone, та помістив їх у відповідні пакети. У класі Phone описав відкриті методи setValue(), getValue() для кожного із закритих полів. Перевантажив метод toString() для даного класу.

Програма працює коректно, помилок не виникає.