Міністерство освіти і науки України

Національний університет «Львівська політехніка»

Інститут комп’ютерних наук та інформаційних технологій

Кафедра автоматизованих систем управління



**ЗВІТ**

Про виконання лабораторної роботи **№ 2**

**«Робота з класами»**

з дисципліни **«Прикладне програмування»**

**Лектор:**

доц. кафедри АСУ

Скорохода О.В.

**Виконав:**

студ. групи КН-203

Репей А.В.

**Прийняв:**

асис. кафедри АСУ

Мельник Р. В.

«\_\_\_» \_\_\_\_\_\_ 2022 р.

∑ = \_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_

Львів – 2022

**Тема:** Робота з класами

**Мета:** Ознайомитись з класами в мові програмування Java.

**Посилання на репозиторій Github**: https://github.com/temerep/labs\_ap/tree/main/lab2

**Хід виконання лабораторної роботи:**

1. Створити проект, що складається з двох класів: основного (Main) та класу для представлення об’єкта відповідно специфікації, що наведена у варіанті індивідуального завдання. 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 ( використати метод toString() ). 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**

**Варіант 10:** **Train**: Пункт призначення, Номер поїзду, Час відправки, Число місць (загальних, купе, плацкарт, люкс).

Скласти масив об'єктів. Вивести:

a) список поїздів, які прямують до заданого пункту призначення;

b) список поїздів, які прямують до заданого пункту призначення та відправляються після заданої години;

c) список поїздів, які відправляються до заданого пункту призначення та мають загальні місця.

**Вихідний код програми**

**App.java**

package com.repei.train;

import java.text.DateFormat;

import java.text.SimpleDateFormat;

import java.util.ArrayList;

import java.util.Date;

import java.util.List;

import java.util.Scanner;

import com.repei.train.trainlist.Train;

public class App {

public static void main(String[] args) throws Exception {

List<Train> trains = new ArrayList<>();

trains.add(new Train(24, "Uzhgorod", "03:40", 25, 32, 15));

trains.add(new Train(36, "Lutsk", "08:55", 23, 60, 27));

trains.add(new Train(33, "Kyiv", "11:15", 37, 41, 18));

trains.add(new Train(131, "Lviv", "13:25", 0, 0, 0));

trains.add(new Train(127, "Lviv", "21:20", 33, 75, 36));

trains.add(new Train(121, "Odessa", "22:33", 34, 45, 32));

Scanner sc = new Scanner(System.in);

boolean checker = true;

while (checker) {

checker = false;

System.out.print("\033\143");

System.out.println(

"Choose the item:\n1) List of trains to your destination\n2) List of trains to your destination and departure\n3) List of trains to your destination and have places\n4) Show all trains\n5) Add train to list\n6) Exit");

System.out.print("=> ");

int choice = sc.nextInt();

switch (choice) {

case 1:

System.out.print("\033\143");

System.out.print("Enter destination: ");

sc.nextLine();

String dest = sc.nextLine();

System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

for (Train t : trains) {

if (dest.compareToIgnoreCase(t.getDestination()) == 0) {

System.out.print("\n#" + t.getTrainNumber() + "\tDestination: " + t.getDestination() + "\tDeparture: "

+ t.getStartTime() + "\tPlaces: " + t.getPlacesCount(0));

}

}

System.out.print("\n\n");

sc.nextLine();

checker = true;

break;

case 2:

System.out.print("\033\143");

System.out.print("Enter destination: ");

sc.nextLine();

String dest1 = sc.nextLine();

System.out.print("Enter time (HH:mm): ");

String time = sc.nextLine();

DateFormat sdf = new SimpleDateFormat("HH:mm");

Date user\_time = sdf.parse(time);

System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

for (Train t : trains) {

Date train\_time = sdf.parse(t.getStartTime());

if (dest1.compareToIgnoreCase(t.getDestination()) == 0 && user\_time.getTime() < train\_time.getTime()) {

System.out.print(t.toString());

}

}

System.out.print("\n\n");

sc.nextLine();

checker = true;

break;

case 3:

System.out.print("\033\143");

System.out.print("Enter destination: ");

sc.nextLine();

String dest2 = sc.nextLine();

System.out.println("\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_");

for (Train t : trains) {

if (dest2.compareToIgnoreCase(t.getDestination()) == 0 && t.getPlacesCount(0) > 0) {

System.out.print(t.toString());

}

}

System.out.print("\n\n");

sc.nextLine();

checker = true;

break;

case 4:

System.out.print("\033\143");

for (Train t : trains) {

System.out.print(t.toString());

}

sc.nextLine();

sc.nextLine();

checker = true;

break;

case 5:

System.out.print("\033\143");

System.out.print("Enter # of train: ");

int num\_of\_train = sc.nextInt();

System.out.print("Enter destination: ");

sc.nextLine();

String train\_dest = sc.nextLine();

System.out.print("Enter departure: ");

String train\_dep = sc.nextLine();

System.out.print("Enter count of compartment seats: ");

int num\_of\_coupe = sc.nextInt();

System.out.print("Enter count of econom-class seats: ");

int num\_of\_platsk = sc.nextInt();

System.out.print("Enter count of luxury seats: ");

int num\_of\_lux = sc.nextInt();

trains.add(new Train(num\_of\_train, train\_dest, train\_dep, num\_of\_coupe, num\_of\_platsk, num\_of\_lux));

System.out.print("\nTrain added successfully!");

sc.nextLine();

sc.nextLine();

checker = true;

break;

case 6:

System.out.print("\033\143");

checker = false;

break;

default:

checker = true;

}

}

sc.close();

}

}

**Train.java**

package com.repei.train.trainlist;

public class Train {

private int trainNumber;

private int[] placesCount = new int[4];

private String startTime;

private String destination;

public Train(int trainNumber, String destination, String startTime, int coupe, int platsk, int lux) {

this.trainNumber = trainNumber;

this.destination = destination;

this.startTime = startTime;

this.placesCount[0] = coupe + platsk + lux;

this.placesCount[1] = coupe;

this.placesCount[2] = platsk;

this.placesCount[3] = lux;

}

public int getTrainNumber() {

return trainNumber;

}

public int getPlacesCount(int i) {

return placesCount[i];

}

public String getStartTime() {

return startTime;

}

public String getDestination() {

return destination;

}

public void setTrainNumber(int trainNumber) {

this.trainNumber = trainNumber;

}

public void setDestination(String destination) {

this.destination = destination;

}

public void setStartDate(String startTime) {

this.startTime = startTime;

}

public void setPlacesCount(int coupe, int platsk, int lux) {

this.placesCount[0] = coupe + platsk + lux;

this.placesCount[1] = coupe;

this.placesCount[2] = platsk;

this.placesCount[3] = lux;

}

@Override

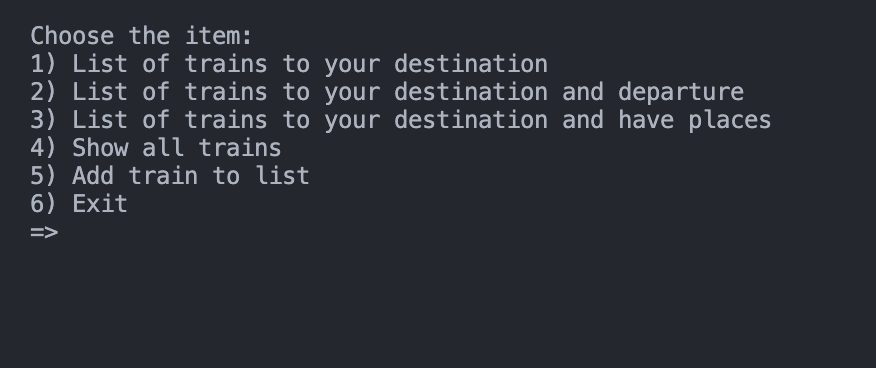
public String toString() {

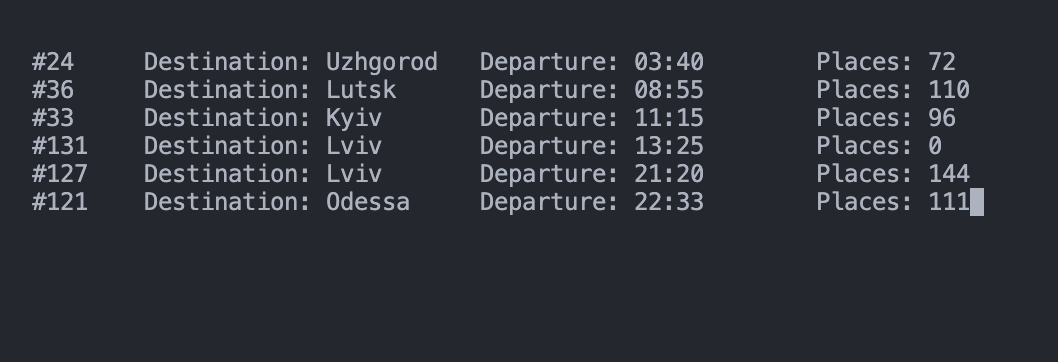
return ("\n#" + trainNumber + "\tDestination: " + destination + "\tDeparture: " + startTime + "\tPlaces: " + placesCount[0]);

}

}

**Результат виконання програми**

****

****

**Висновок**

На лабораторній роботі я ознайомився роботою класів в мові програмування Java та написав свою програму відносно мого індивідуального завдання.