Міністерство освіти і науки України

Національний технічний університет

«Харківський політехнічний інститут»

Кафедра «Обчислювальна техніка та програмування»

**ЗВІТ**

Про виконання лабораторної роботи № 8

«[Основи введення/виведення Java SE](https://oop-khpi.gitlab.io/#tasks)»

Керівник: Жилін Володимир Анатолійович

Виконавець: студентка гр. КІТ-320 Шпилька Д. О.

Харків 2021

Тема роботи: [Основи введення/виведення Java SE](https://oop-khpi.gitlab.io/#tasks)

Мета:

1. Забезпечити можливість збереження і відновлення масива об'єктів рішення завдання [лабораторної роботи №7](https://oop-khpi.gitlab.io/%D0%B7%D0%B0%D0%B2%D0%B4%D0%B0%D0%BD%D0%BD%D1%8F/task07/).
2. Забороняється використання [стандартного протокола серіалізації](https://docs.oracle.com/javase/8/docs/platform/serialization/spec/serialTOC.html).
3. Продемонструвати використання моделі [Long Term Persistence](https://docs.oracle.com/javase/tutorial/javabeans/advanced/longpersistence.html).
4. Забезпечити діалог з користувачем у вигляді простого текстового меню.
5. При збереженні та відновленні даних забезпечити діалоговий режим вибору директорії з відображенням вмісту і можливістю переміщення по підкаталогах.

Код:

**package** ua.khpi.oop.Shpylka08;

**import** java.io.\*;

**import** java.util.\*;

**public** **class** main {

**public** **static** **void** main(String[] args) {

Scanner in = **new** Scanner(System.***in***);

String filename = "lab08.txt";

System.***out***.println("Меню:" + "\n1)Вывести информацию" + "\n2)Режим диалога" + "\n3)Записать в файл");

String display = in.nextLine();

**switch** (display){

**case** "1":

Route one = **new** Route("№6", 40, "12:35", "17:20", "Пятница", 5, 300, "Харьков", "Киев", "Полтава, Лубня, Борисполь");

one.display1();

**break**;

**case** "2":

**while** (**true**) {

System.***out***.println("Человек:");

String human = in.nextLine();

**if** (human.toLowerCase().equals("end")) {

**break**;

}

System.***out***.println("Консультант:");

String consultant = in.nextLine();

}

**case** "3":

**try**(ObjectOutputStream oos = **new** ObjectOutputStream(**new** FileOutputStream(filename))){

**var** a = **new** Route("№6", 40, "12:35", "17:20", "Пятница", 5, 300, "Харьков", "Киев", "Полтава, Лубня, Борисполь");

oos.writeObject(a);

oos.close();

System.***out***.println("Файл был написан");

}

**catch**(Exception ex){

System.***out***.println(ex.getMessage());

}

}

}

}

**class** Bus\_station {

**private** String name\_station;

**private** **int** number\_bus;

**private** String arrival;

**private** String departure;

**public** Bus\_station (String name\_station, **int** number\_bus, String arrival, String departure) {

**this**.name\_station = name\_station;

**this**.number\_bus = number\_bus;

**this**.arrival = arrival;

**this**.departure = departure;

}

**public** String getName\_station() {

**return** name\_station;

}

**public** **void** setName\_station(String name\_station) {

**this**.name\_station = name\_station;

}

**public** **int** getNumber\_bus() {

**return** number\_bus;

}

**public** **void** setNumber\_bus(**int** number\_bus) {

**this**.number\_bus = number\_bus;

}

**public** String getArrival() {

**return** arrival;

}

**public** **void** setArrival(String arrival) {

**this**.arrival = arrival;

}

**public** String getDeparture() {

**return** departure;

}

**public** **void** setDeparture(String departure) {

**this**.departure = departure;

}

}

**class** Timetable **extends** Bus\_station{

String day;

**int** places;

**int** price;

**public** Timetable(String name\_station, **int** number\_bus, String arrival, String departure, String day, **int** places, **int** price) {

**super**(name\_station, number\_bus, arrival, departure);

**this**.day = day;

**this**.places = places;

**this**.price = price;

}

}

**class** Route **extends** Timetable{

String start\_town;

String intermediate\_cities;

String last\_town;

**public** Route(String name\_station, **int** number\_bus, String arrival, String departure, String day, **int** places,

**int** price, String start\_town, String last\_town, String intermediate\_cities) {

**super**(name\_station, number\_bus, arrival, departure, day, places, price);

**this**.start\_town = start\_town;

**this**.intermediate\_cities = intermediate\_cities;

**this**.last\_town = last\_town;

}

**void** display1 () {

System.***out***.println("Автостанция: " + getName\_station() + "\nНомер автобуса: " + getNumber\_bus() + "\nВремя прибытия: " + getArrival()

+ "\nВремя отъезда: " + getDeparture() + "\n----------------------------------" + "\nДата: " + day + "\nСвободные места: " + places

+ "\nЦена в грн: " + price + "\n----------------------------------" + "\nНачало маршрута: " + start\_town + "\nПромежуточные остановки: " +

intermediate\_cities + "\nКонечная станция: " + last\_town);

}

}

***Результати роботи програми:***

Меню:

1)Вывести информацию

2)Режим диалога

3)Записать в файл

1

Автостанция: №6

Номер автобуса: 40

Время прибытия: 12:35

Время отъезда: 17:20

----------------------------------

Дата: Пятница

Свободные места: 5

Цена в грн: 300

----------------------------------

Начало маршрута: Харьков

Промежуточные остановки: Полтава, Лубня, Борисполь

Конечная станция: Киев

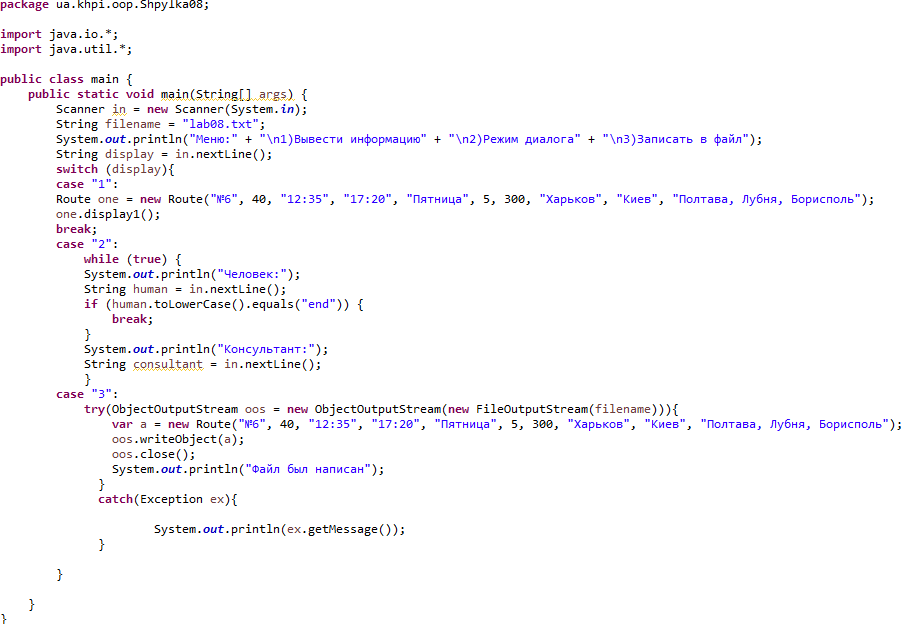


Рис.1

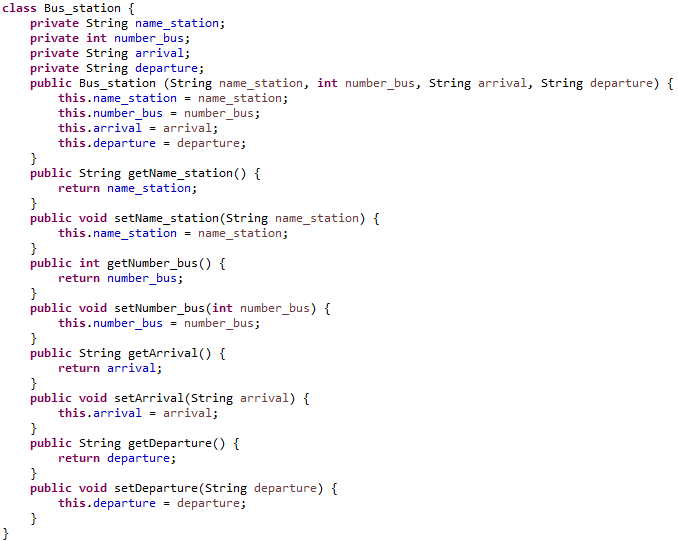


Рис.2

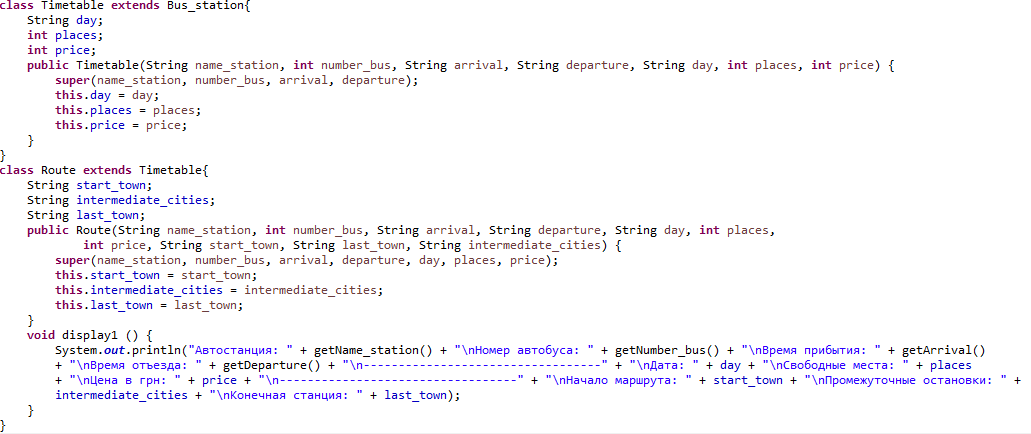


Рис.3

Висновок: Було набуло навичок управління введенням/виведенням даних з використанням класів платформи Java SE.