**Лабораторна робота №8**

**Основи введення/виведення Java SE**

**Мета:** Оволодіння навичками управління введенням/виведенням даних з використанням класів платформи Java SE.

**1 ВИМОГИ**

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.

**1.1 Розробник**

***Виконавець:*** Сиромятников Марк

***Група:*** КІТ-119а

**Варіант:** 19

**2 ОПИС ПРОГРАМИ**

Було використано наступні засоби:

* *java.beans.XMLDecoder* – вивід з XML файлу
* java.beans.XMLEncoder– введення у файл
* .close(); - закриття потоку вводу
* import java.util.NoSuchElementException – обробка відсутності елементу.
* java.io.FileInputStream – введеня інформації до файлу
* java.io.FileOutputStream – виведення інформації з файлу

**2.1 Ієрархія та структура класів**

Було створено 1 клас MainClas, який містить у собі реалізацію меню та десеріалізацію з серіалізацією, об’єкт класу AdressBook, що містить у собі масив об’єктів Person з відповідними методами, гетерами та сетерами.

**2.2 Важливі фрагменти програми**

dir = new File(sb.toString());

try {

XMLDecoder decoder = new XMLDecoder(new BufferedInputStream(new FileInputStream(dir)));

ab.array = (Person[]) decoder.readObject();

decoder.close();

for (**var** hotelRoom : ab.array) {

System.***out***.println("Full name: "+ hotelRoom.getFullName() + "\nDate of birth: " + hotelRoom.getDateOfBirth() + "\nTelephone numbers: "+ hotelRoom.showNumbers() + "\nAdress: " + hotelRoom.getAdress() + "\nEditing date and time: " + hotelRoom.getEditingDateAndTime() + "\n");

}

} catch (FileNotFoundException e) {

e.printStackTrace();

}

break;

case 5:

System.***out***.println("Enter number of element which you want to delete: ");

number = option.nextInt();

if (number < ab.getSize() && number > -1) {

ab.remove(number);

System.***out***.println("Removing complete\n");

ab.printAll();

} else {

System.***out***.println("There is no such element in array");

}

break;

case 6:

menu = false;

option.close();

someData.close();

break;

default:

**3 ВАРІАНТИ ВИКОРИСТАННЯ**

Нижче наведено опис пунктів меню та прикріплено відповідні зображення задля пояснення роботи. Пункти меню, відповідно, можуть бути викликані користувачем

1. Пункт меню “Show current data” Викликає метод виведення на екран вмісту класу з об’єктами прекладної галузі (рис 1).

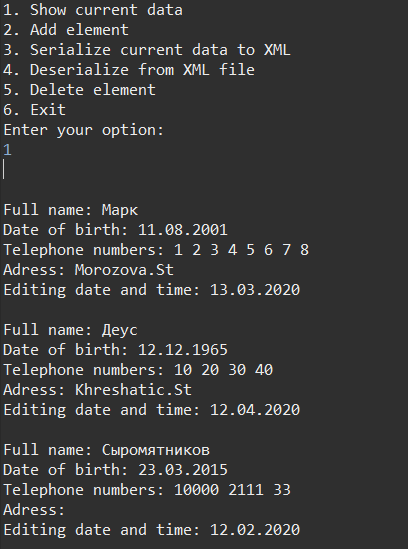


Рисунок 1 - Перший пункт меню

1. Пункт меню “Add element” викликає метод додавання нового елементу до до масиву об’єктів (рис 2).

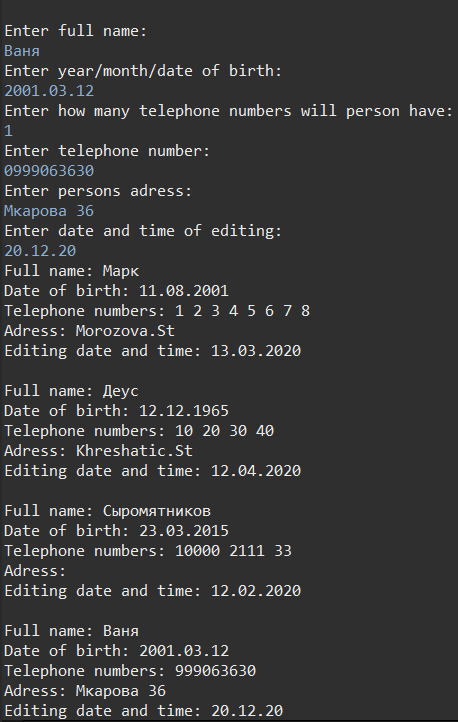


Рисунок 2 - Другий пункт меню

1. Пункт меню “Serialize current data to XML” викликає меню серіалізації поточних даних до XML файлу, де необхідно обрати дерик торію серіалізації (рис 3).

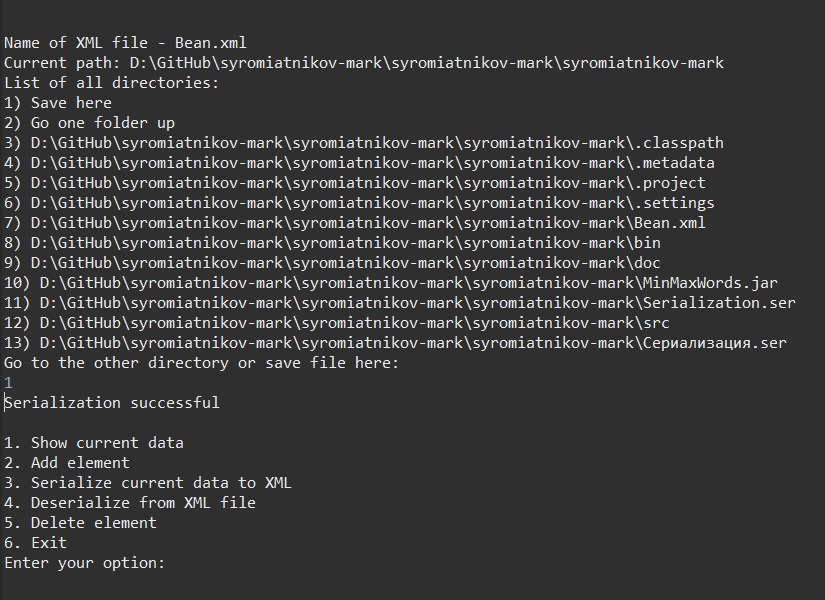


Рисунок 3 - Третій пункт меню

1. Пункт меню “Deserialize from XML file” викликає меню десеріалізації даних, де необхідно обрати файл, з якого буде проводитися десеріалізація (рис 4).

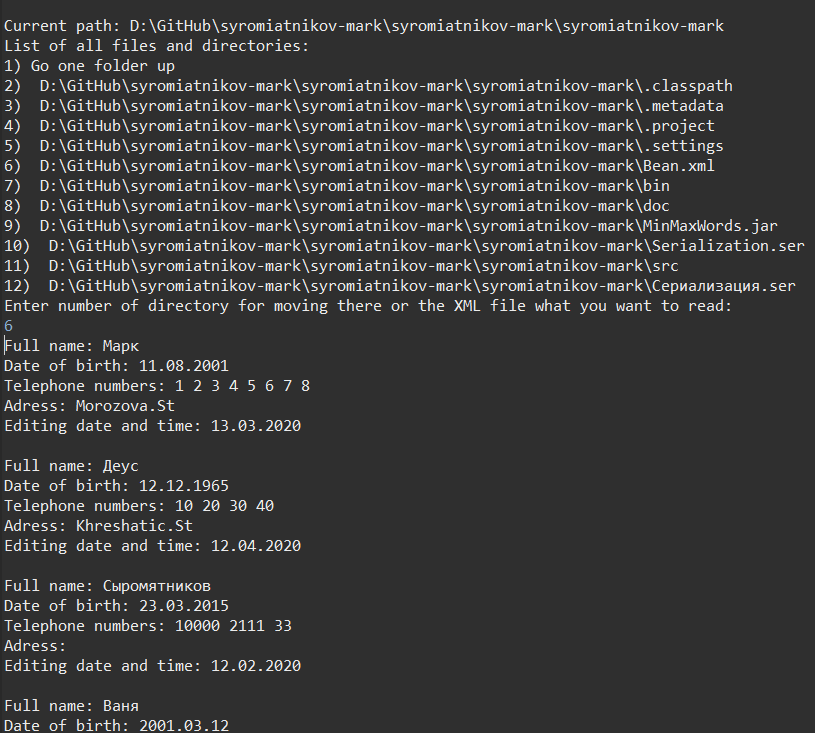


Рисунок 4 - Четвертий пункт меню

1. Пункт меню “Delete element” викликає метод видалення певного об’єкту з масиву з виводом результату на екран (рис 5).

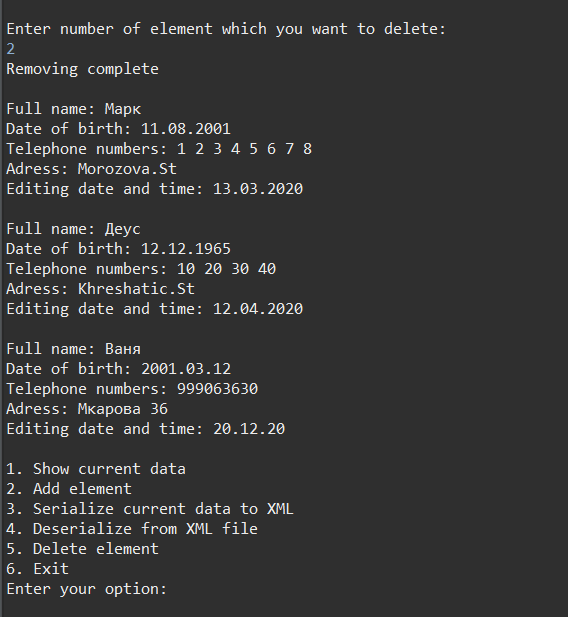


Рисунок 5 – П’ятий пункт меню

**ВИСНОВКИ**

У результаті виконання лабораторної роботи було розроблено методи серіалізації та десеріалізації даних до XML файлу, закріплено навички введення/виведення у Java SE. Розроблено меню для діалогу с користувачем.