

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
Федерального государственного бюджетного образовательного учреждения
высшего образования
«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»

Факультет _____ ИТР _____

Кафедра _____ ПИН _____

ЛАБОРАТОРНАЯ РАБОТА №4

Руководитель

Кульков Я.Ю.

(фамилия, инициалы)

(подпись)

(дата)

Студент _____ ПИН - 121 _____

(группа)

Ермилов М.В.

(фамилия, инициалы)

(подпись)

(дата)

Муром 2023

Лабораторная работа №2

Тема: Ввод и вывод информации в файлы.

Цель: изучение методов работы с текстовыми и бинарными файлами.

Задачи:

Разработать программу работы с данными, хранящимися в текстовом файле. Организовать чтение и обработку данных из файла в соответствии с индивидуальным заданием. Сохранить полученные результаты в новый текстовый файл. В таблице представлена структура одной записи в файле. Одной записи соответствует одна строка файла. Таких записей в файле может быть несколько.

Вариант	Задание
«Покупатель»	фамилия; имя; отчество; пол; национальность; рост; вес; дата рождения (год, месяц число); номер телефона; домашний адрес (почтовый индекс, страна, область, район, город, улица, дом, квартира); номер кредитной карточки; банковского счета. Вывести данные о покупателях с города Муром

					МИ ВлГУ 09.03.04		
Изм.	Лист	№ докум.	Подпись	Дата			
Разраб.							
Провер.	.						
Реценз.							
Н. Контр.							
Утверд.	.						
					Лит.	Лист	Листов
						2	11
					ПИН-121		

Код программы:

```
import java.io.BufferedWriter;
import java.io.IOException;
import java.nio.charset.StandardCharsets;
import java.nio.file.Files;
import java.nio.file.Path;
import java.nio.file.Paths;
import java.nio.file.StandardOpenOption;

public class Main {

    public static void main(String[] args) throws IOException {
        Path p = Paths.get("file1.txt");
        write(p,
"Петр;Петров;Петрович;170;80;1;1;1970;88005553535;Россия;Московская область;Московский
район;Москва;Большая садовая;302-бис;50;111;2222000022220000;123456789");
        write(p,
"Михаил;Михайлов;Михайлович;170;80;1;1;1970;88005553535;Россия;Владимирская
область;Муромский район;Муром;Большая садовая;302-
бис;50;111;2222000022220000;123456789");
        write(p,
"Василий;Васильев;Васильевич;170;80;1;1;1970;88005553535;Россия;Владимирская
область;Муромский район;Муром;Большая садовая;302-
бис;50;111;2222000022220000;123456789");
        write(p,
"Иван;Иванов;Иванович;170;80;1;1;1970;88005553535;Россия;Владимирская область;Муромский
район;Муром;Большая садовая;302-бис;50;111;2222000022220000;123456789");
        Item[] arr = read(p);
        for(Item i : arr) {
            if(i.getAddress() != null && i.getAddress().getCity().equals("Муром"))
            {
                System.out.println(i.getName() + " " + i.getSurname());
            }
        }
    }

    static Item[] read(Path path) throws IOException {
        String[] arr1 = Files.readString(path).split("\n");
        Item[] arr2 = new Item[arr1.length];
        for(int i = 0; i < arr1.length; i++) {
            arr2[i] = Item.parse(arr1[i]);
        }
        return arr2;
    }

    static void write(Path path, String str) throws IOException {
        if(!Files.exists(path)) {
            Files.createFile(path);
        }
        else
        {
            str = "\n" + str;
        }
        BufferedWriter writer = Files.newBufferedWriter(path,
StandardCharsets.UTF_8, StandardOpenOption.APPEND);
        writer.write(str);
        writer.close();
    }
}

import java.util.Date;
```

					МИ ВлГУ 09.03.04	Лист
						3
Изм.	Лист	№ докум.	Подпись	Дата		

```

public class Item {
    private String name = "";
    private String surname = "";
    private String patronymic = "";
    private double Height = 0;
    private double Width = 0;
    private Date birth = null;
    private long phone = 0;
    private Address address = null;
    private long card = 0;
    private long bank = 0;

    public String getPatronymic() {
        return patronymic;
    }
    public void setPatronymic(String patronymic) {
        this.patronymic = patronymic;
    }
    public String getName() {
        return name;
    }
    public void setName(String name) {
        this.name = name;
    }
    public String getSurname() {
        return surname;
    }
    public void setSurname(String surname) {

```

					МИ ВлГУ 09.03.04	Лист
						4
Изм.	Лист	№ докум.	Подпись	Дата		

```

        this.surname = surname;
    }
    public double getHeight() {
        return Height;
    }
    public void setHeight(double height) {
        if(height > 0) {
            Height = height;
        }
    }
    public double getWidth() {
        return Width;
    }
    public void setWidth(double width) {
        if(width > 0) {
            Width = width;
        }
    }
    public long getPhone() {
        return phone;
    }
    public void setPhone(long phone) {
        if(phone > 0) {
            this.phone = phone;
        }
    }
    public Date getBirth() {
        return birth;
    }
    public void setBirth(Date birth) {

```

```

        this.birth = birth;
    }
    public Address getAddress() {
        return address;
    }
    public void setAddress(Address address) {
        this.address = address;
    }
    public long getCard() {
        return card;
    }
    public void setCard(long card) {
        if(card > 0) {
            this.card = card;
        }
    }
    public long getBank() {
        return bank;
    }
    public void setBank(long bank) {
        if(bank > 0) {
            this.bank = bank;
        }
    }
}

```

```

@SuppressWarnings("deprecation")
public static Item parse(String str)
{
    String[] arr = str.split(";");
}

```

```

Item i = new Item();
int j = 0;
if(arr.length > j) {
    i.setName(arr[j]);
    j++;
}
if(arr.length > j) {
    i.setSurname(arr[j]);
    j++;
}
if(arr.length > j) {
    i.setPatronymic(arr[j]);
    j++;
}
if(arr.length > j) {
    try {
        double a = Double.parseDouble(arr[j]);
        i.setHeight(a);
    } catch (Exception e) { }
    j++;
}
if(arr.length > j) {
    try {
        double a = Double.parseDouble(arr[j]);
        i.setWidth(a);
    } catch (Exception e) { }
    j += 3;
}
if(arr.length > j) {
    try {

```

```

        int d = Integer.parseInt(arr[j - 2]);
        int m = Integer.parseInt(arr[j - 1]);
        int y = Integer.parseInt(arr[j]);
        i.setBirth(new Date(y, m, d));
    } catch (Exception e) { }
    j++;
}
if(arr.length > j) {
    try {
        long a = Long.parseLong(arr[j]);
        i.setPhone(a);
    } catch (Exception e) { }
    j += 8;
}
if(arr.length > j) {
    Address a = new Address();
    a.setCountry(arr[j - 7]);
    a.setArea(arr[j - 6]);
    a.setDistrict(arr[j - 5]);
    a.setCity(arr[j - 4]);
    a.setStreet(arr[j - 3]);
    a.setHome(arr[j - 2]);
    try {
        int b = Integer.parseInt(arr[j - 1]);
        a.setFlat(b);
        int c = Integer.parseInt(arr[j]);
        a.setPostalCode(c);
    } catch (Exception e) { }
    i.setAddress(a);
    j++;
}

```



```

    }
    if(arr.length > j) {
        try {
            long a = Long.parseLong(arr[j]);
            i.setCard(a);
        } catch (Exception e) { }
        j++;
    }
    if(arr.length > j) {
        try {
            long a = Long.parseLong(arr[j]);
            i.setBank(a);
        } catch (Exception e) { }
    }
    return i;
}
}

public class Address {
    private String country = "";
    private String area = "";
    private String district = "";
    private String city = "";
    private String street = "";
    private String home = "";
    private int flat = 0;
    private int postalCode = 0;
    public String getCountry() {
        return country;
    }
    public void setCountry(String country) {

```

```

        this.country = country;
    }
    public String getArea() {
        return area;
    }
    public void setArea(String area) {
        this.area = area;
    }
    public String getDistrict() {
        return district;
    }
    public void setDistrict(String district) {
        this.district = district;
    }
    public String getCity() {
        return city;
    }
    public void setCity(String city) {
        this.city = city;
    }
    public String getHome() {
        return home;
    }
    public void setHome(String home) {
        this.home = home;
    }
    public int getPostalCode() {
        return postalCode;
    }
    public void setPostalCode(int postalCode) {

```

```

        if(postalCode > 0) {
            this.postalCode = postalCode;
        }
    }

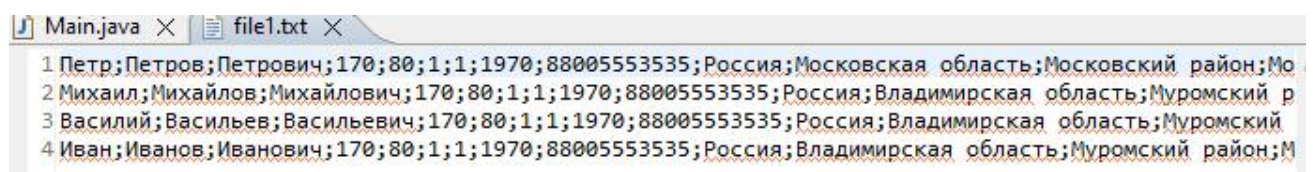
    public int getFlat() {
        return flat;
    }

    public void setFlat(int flat) {
        if(flat > 0) {
            this.flat = flat;
        }
    }

    public String getStreet() {
        return street;
    }

    public void setStreet(String street) {
        this.street = street;
    }
}

```

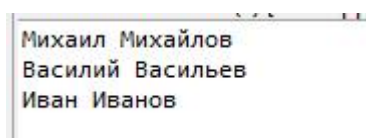


```

1 Петр;Петров;Петрович;170;80;1;1;1970;88005553535;Россия;Московская область;Московский район;Мо
2 Михаил;Михайлов;Михайлович;170;80;1;1;1970;88005553535;Россия;Владимирская область;Муромский р
3 Василий;Васильев;Васильевич;170;80;1;1;1970;88005553535;Россия;Владимирская область;Муромский
4 Иван;Иванов;Иванович;170;80;1;1;1970;88005553535;Россия;Владимирская область;Муромский район;М

```

Рисунок 1 - пример сохраненных данных, с помощью программы



```

Михаил Михайлов
Василий Васильев
Иван Иванов

```

Рисунок 2 - пример вывода необходимой информации