Министерство науки и высшего образования Российской Федерации
Федеральное государственное бюджетное образовательное учреждение высшего образования
«Московский государственный технический университет имени Н.Э. Баумана
(национальный исследовательский университет)»
(МГТУ им. Н.Э. Баумана)

ФАКУЛЬТЕТ **ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ**

КАФЕДРА **КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)**

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.01 Информатика и вычислительная техника**

МАГИСТЕРСКАЯ ПРОГРАММА **09.04.01/12 Интеллектуальный анализ больших данных в системах поддержки принятия решений.**

О Т Ч Е Т

по лабораторной работе № 9

Вариант № 11

Название: Stream API

Дисциплина: Языки программирования для работы с большими данными

Студент ИУ6-23М (Группа) _____ (Подпись, дата) С.В.Мельников (И.О. Фамилия)

Преподаватель _____ (Подпись, дата) П.В. Степанов (И.О. Фамилия)

Москва, 2023

Цель работы

Изучить работу Stream API в языке программирования Java.

Задание 1 (Вариант 1, Задание 11):

Задана коллекция (Класс Student: имя и рейтинг):

```
Collection<Student> students = Arrays.asList(
        new Student("Ivan", 40),
        new Student("Petr", 60),
        new Student("Olga", 70)
);
```

Вернуть средний балл.

Листинг программы:

Код класса Student:

```
public class Student {
    private final String name;
    private int score;

    Student(String name, int score) {
        this.name = name;
        this.score = score;
    }

    public static void main(String[] args) {
        Collection<Student> students = Arrays.asList(
                new Student("Ivan", 40),
                new Student("Petr", 60),
                new Student("Olga", 70)
        );
        double avg_score =
                students.stream().map(student ->
student.score).reduce(Integer::sum).get() * 1.0 / students.size();
        System.out.println("Средний балл: " + avg_score);
    }
}
```

Работа программы представлена на рисунке 1.

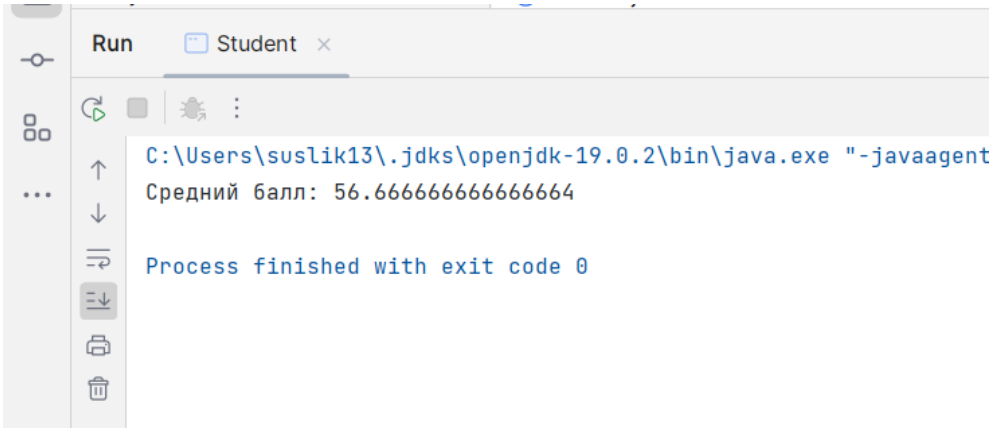

Рисунок 1 – Работа программы Student.java

Задание 2 (Вариант 1, Задание 12):

Задана коллекция строк. Вернуть первые два элемента.

Листинг программы:

Код класса StringCollection:

```
public class StringCollection {
    public static void main(String[] args) {
        Collection<String> strings = Arrays.asList(
                "Hello",
                "World",
                "Qwerty",
                "uiop",
                "ev132f23f5",
                "nmw934"
        );
        strings.stream().limit(2).forEach(System.out::println);
    }
}
```

Работа программы представлена на рисунке 2.

Рисунок 2 – Работа программы StringCollection.java

Задание 3 (Вариант 2, Задание 11):

Задана коллекция (Класс People: имя и возраст, пол (enum)):

```
Collection<People> peoples = Arrays.asList(
        new People("Ivan", 16, Sex.MAN),
        new People("Petr", 23, Sex.MAN),
        new People("Maria", 42, Sex.WOMAN)
);
```

Найти самого старшего человека мужского пола.

Задание 4 (Вариант 2, Задание 12):

Коллекция из 11 задания. Найти самый минимальны возраст человека, у которого есть буква “e” в имени.

Листинг программы:

Код класса People:

```
enum Sex { MAN, WOMAN }

public class People {
    private final String name;
    private final int age;
    private final Sex sex;

    public People(String name, int age, Sex sex) {
        this.name = name;
        this.age = age;
        this.sex = sex;
    }

    @Override
    public String toString() {
        return name + ": " + age + " years, " + (sex == Sex.MAN ? "male" :
"female");
    }

    public static void main(String[] args) {
        Collection<People> peoples = Arrays.asList(
                new People("Ivan", 16, Sex.MAN),
                new People("Petr", 23, Sex.MAN),
                new People("Maria", 42, Sex.WOMAN),
                new People("Vera", 5, Sex.WOMAN)
        );
        System.out.println(peoples.stream().filter(x -> x.sex ==
Sex.MAN).max(Comparator.comparingInt(x -> x.age)).get());
        System.out.println(peoples.stream().filter(x ->
x.name.contains("e")).min(Comparator.comparingInt((x -> x.age))).get());
    }
}
```

Работа программы представлена на рисунке 3.

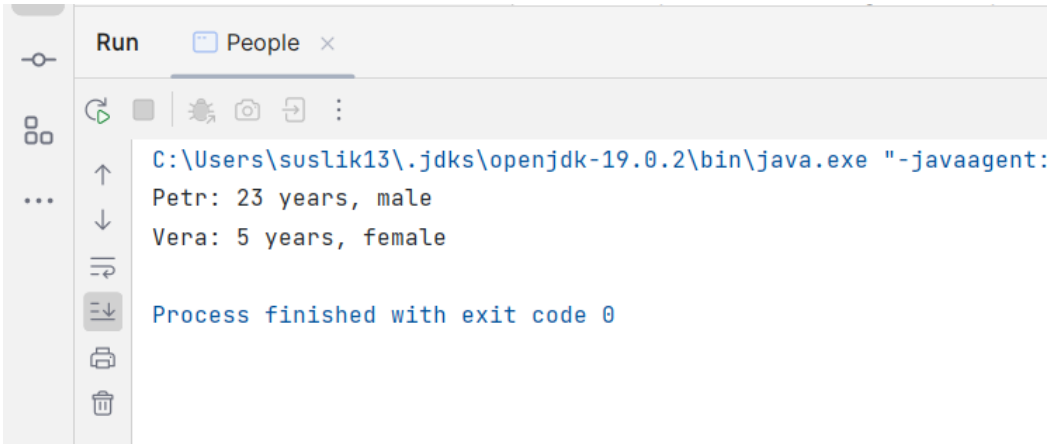

Рисунок 3 – Работа программы People.java

Вывод

В ходе выполнения лабораторной работы были получены навыки с работой в Stream API языка программирования Java.