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 **09.04.01/07 Интеллектуальные системы анализа, обработки и интерпретации больших данных**

О Т Ч Е Т

по лабораторной работе № 9

Вариант 10

Название: Stream API в Java

Дисциплина: Языки программирования для работы с большими данными

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

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

Москва, 2022

Цель работы:

Получение навыков работы со Stream API в Java.

Выполнение:

Задание 1:

Использовать ТОЛЬКО методы Stream API. Циклов и условий быть не должно.

1. 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 список студентов имя которых начинается на P и рейтинг находится в интервале 40-60.

2. Задана коллекция строк. Вернуть средний балл.

Листинг файла Var1.java

```
package com.company;

import java.util.Arrays;
import java.util.Collection;

class Student
{
    private String name;  // Название
    private int ball;  // Балл
    public Student(String nm,int bl)
    {
        this.ball=bl;
        this.name=nm;
    }
    public String getName()
    {
        return this.name;
    }
    public  int getBall()
    {
        return  ball;
    }
}
public class Var1 {

    public static void main(String[] args) {
     ///STREAM    APIIII
        Collection<Student> students= Arrays.asList(
```

```
                new Student("Ivan",40),
                new Student("Petr",60),
                new Student("Olga",70)
        );
        // Через StreamAPI находим средний (AveragingInt используем)
      double ans=students.stream().mapToInt(x->x.getBall()).average().getAsDouble();
      System.out.println(ans);
    }
}
```

Листинг файла Var2.java

```
package com.company;

import java.util.Arrays;
import java.util.Collection;


enum Sex{
    MAN,
    WOMAN;
}
class People
{
    String name;  // Имя
    int age;  // Возраст
    Sex sex;
    public People(String nm,int ag,Sex s)
    {
        this.name=nm;
        this.age=ag;
        this.sex=s;
    }

    public String getName() {
        return name;
    }

    public int getAge() {
        return age;
    }

    public Sex getSex() {
        return sex;

    }
// еше нам нужен компаратор для нахождения возраста
public static int compare (People p1, People p2){
    if(p1.getAge() > p2.getAge())
        return 1;
    return -1;
}
    @Override
    public String toString() {
        return "People{" +
                "name='" + name + '\'' +
                ", age=" + age +
                ", sex=" + sex +
                '}';
    }
}
public class Var2 {

    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)
```

```
        );
        People p=peoples.stream().filter(x->x.sex==Sex.MAN).max(People::compare).get();
        System.out.println("Самый старший человек мужского пола");
        System.out.println(p);
    }
}
```

```
56.666666666666664

Process finished with exit code 0
```

Рисунок 1 - Результат выполнения кода решения задачи 1

```
Самый старший человек мужского пола
People{name='Petr', age=23, sex=MAN}

Process finished with exit code 0
```

Рисунок 2 - Результат выполнения кода решения задачи 2

Ссылка на программное решение:

Программное решение представлено в репозитории распределённой системы управления версиями Git:

https://github.com/Wingo11/BigDataLanguages/tree/Lab9/src

Вывод:

При выполнении лабораторной работы были получены навыки работы со Stream API в Java.