



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

О Т Ч Е Т

по лабораторной работе № 9

Название: Stream API

Дисциплина: Языки программирования для работы с большими
данными

Студент

ИУ6-23М

(Группа)

(Подпись, дата)

А.А. Аветисян

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

(И.О. Фамилия)

Москва, 2022

Лабораторная работа № 9

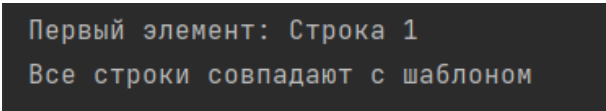
Задание:

Использовать ТОЛЬКО методы Stream API. Циклов и условий быть не должно. 2. Задана коллекция строк. Вернуть первый элемент коллекции, а также существуют ли все совпадения с шаблоном. Шаблон можно выбрать произвольно.

Ход работы:

Код программы:

```
public static void main(String[] args){
    Collection<String> collection = Arrays.asList("Строка 1", "Строка 2", "Строка 3");
    System.out.println("Первый элемент: " + collection.stream().findFirst().get());
    boolean flag = collection.stream().allMatch(str-> str.matches("Строка [1-3]"));
    if (flag) {
        System.out.println("Все строки совпадают с шаблоном");
    } else {
        System.out.println("Есть строки не совпадающие с шаблоном");
    }
}
```



```
Первый элемент: Строка 1
Все строки совпадают с шаблоном
```

Рисунок 1 – Результат работы программы

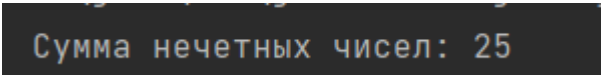
Задание:

3. Задана коллекция чисел. Вернуть сумму нечетных чисел.

Ход работы:

Код программы:

```
public static void main(String[] args){
    Collection<Integer> collection = Arrays.asList(1, 2, 3, 4, 5, 6, 7, 8, 9);
    int sum = collection.stream().filter(num-> num % 2 != 0).reduce(Integer::sum).orElse(0);
    System.out.println("Сумма нечетных чисел: " + sum);
}
```



```
Сумма нечетных чисел: 25
```

Рисунок 2 – Результат работы программы

Задание:

2. Задана коллекция:

(Класс People: имя и возраст)

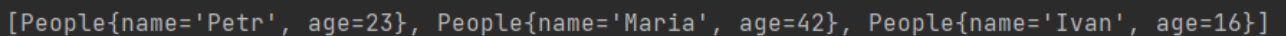
```
Collection<People> peoples = Arrays.asList(  
    new People("Ivan", 16),  
    new People("Petr", 23),  
    new People("Maria", 42)  
);
```

Отсортировать по имени в обратном алфавитном порядке.

Ход работы:

Код программы:

```
public static void main(String[] args){  
    Collection<People> peoples = Arrays.asList(  
        new People("Ivan", 16),  
        new People("Petr", 23),  
        new People("Maria", 42)  
    );  
    System.out.println(peoples.stream().sorted((first, second)->-first.getName().compareTo(second.getName()))).toList());  
}
```



```
[People{name='Petr', age=23}, People{name='Maria', age=42}, People{name='Ivan', age=16}]
```

Рисунок 3 – Результат работы программы

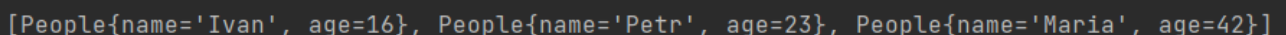
Задание:

3. Коллекция из 2 задания. Отсортировать по возрасту.

Ход работы:

Код программы:

```
public static void main(String[] args){  
    Collection<People> peoples = Arrays.asList(  
        new People("Ivan", 16),  
        new People("Petr", 23),  
        new People("Maria", 42)  
    );  
    System.out.println(peoples.stream().sorted(Comparator.comparingInt(People::getAge)).toList());  
}
```



```
[People{name='Ivan', age=16}, People{name='Petr', age=23}, People{name='Maria', age=42}]
```

Рисунок 4 – Результат работы программы

Вывод: лабораторная работа выполнена в соответствии с заданием и вариантом.