



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 18**

**Название:** Stream API в Java

**Дисциплина:** Языки программирования для работы с большими данными

Студент

ИУ6-23М

(Группа)

\_\_\_\_\_  
(Подпись, дата)

Д.Н. Хныкин

(И.О. Фамилия)

Преподаватель

\_\_\_\_\_  
(Подпись, дата)

П.В. Степанов

(И.О. Фамилия)

Москва, 2022

## Цель работы

Получение навыков работы со Stream API в Java.

## Выполнение

### Задание 1 и 2:

1. Задана коллекция чисел. Разделить числа на четные и нечетные.
2. Задана коллекция строк. Вернуть список из коллекции без повторов.
3. Задана коллекция строк. Преобразовать в MAP, где первый символ ключ, второй – значение.
4. Задана коллекция строк. Преобразовать в MAP, сгруппировав по первому символу строки

Листинг выполнения подзадачи 1

```
#Main.java

package com.aranei;

import java.util.Arrays;
import java.util.Collection;
import java.util.stream.Collectors;

public class Main {
    public static void main(String[] args) {
        n9v1n4();
        n9v1n5();
        n9v2n4();
        n9v2n5();
    }

    public static void n9v1n4() {
        Collection<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5, 6, 7, 8, 9, 10);
        System.out.println("Дано: " + Arrays.toString(numbers.toArray()) +
            "\nНечетные: " + numbers.stream().filter((e) -> e % 2 !=
0).collect(Collectors.toList()) +
            "\nЧетные: " + numbers.stream().filter((e) -> e % 2 ==
0).collect(Collectors.toList()));
    }

    public static void n9v1n5() {
        Collection<String> strings = Arrays.asList("a", "b", "aa", "bb", "aa", "b", "aAa",
"B", "aaa");
        System.out.println("\nДано: " + Arrays.toString(strings.toArray()) +
            "\nУникальные значения: " +
strings.stream().distinct().collect(Collectors.toList()));
    }

    public static void n9v2n4() {
        Collection<String> strings = Arrays.asList("aa", "ab", "Aa", "AC", "tHx", "thQ");
        System.out.println("\nДано: " + Arrays.toString(strings.toArray()) +
            "\nMAP: " + strings.stream().collect(Collectors.toMap(e ->
e.charAt(0),
e ->
e.charAt(1), (e1, e2) -> e2)));
    }

    public static void n9v2n5() {
        Collection<String> strings = Arrays.asList("aa", "ab", "Aa", "AC", "tHx", "thQ");
        System.out.println("\nДано: " + Arrays.toString(strings.toArray()) +
            "\nMAP: " +
strings.stream().collect(Collectors.groupingBy(e -> e.charAt(0))));
    }
}
```

### **Ссылка на программное решение**

Программное решение представлено в репозитории распределённой системы управления версиями Git:

<https://github.com/Aranei99/labsJava/tree/main/LB9>

### **Вывод**

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