



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 № 4

Название: stream API

Дисциплина: языки программирования для работы с большими данными

Студент

ИУ6-23М

(Группа)

(Подпись, дата)

А.А.Клушина

(И.О. Фамилия)

Преподаватель

П.В. Степанов

(Подпись, дата)

(И.О. Фамилия)

Москва, 2024

Цель: изучить stream API в java.

Задание 1: использовать ТОЛЬКО методы Stream API. Циклов и условий быть не должно. Задана коллекция чисел. Разделить числа на четные и нечетные.

Код класса Main:

```
import java.util.Arrays;
import java.util.List;
import java.util.stream.Collectors;

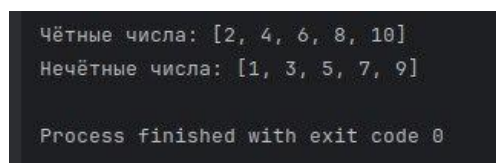
public class Main {
    public static void main(String[] args) {
        List<Integer> numbers = Arrays.asList(1, 2, 3, 4, 5,
        6, 7, 8, 9, 10);

        List<Integer> evenNumbers = numbers.stream()
            .filter(num -> num % 2 == 0)
            .collect(Collectors.toList());

        List<Integer> oddNumbers = numbers.stream()
            .filter(num -> num % 2 != 0)
            .collect(Collectors.toList());

        System.out.println("Чётные числа: " + evenNumbers);
        System.out.println("Нечётные числа: " + oddNumbers);
    }
}
```

Работа программы показана на рисунке 1.



```
Чётные числа: [2, 4, 6, 8, 10]
Нечётные числа: [1, 3, 5, 7, 9]

Process finished with exit code 0
```

Рисунок 1 – Работа программы

Задание 2: использовать ТОЛЬКО методы Stream API. Циклов и условий быть не должно. Задана коллекция строк. Вернуть список из коллекции без повторов.

Код класса Main:

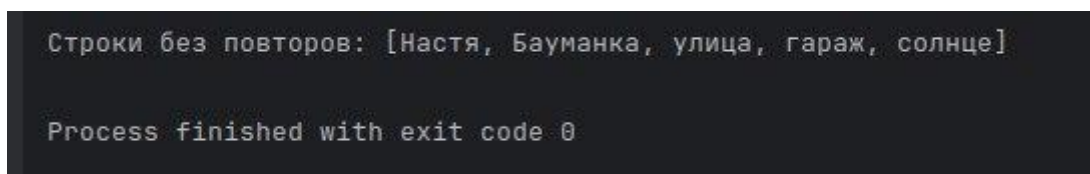
```
import java.util.Arrays;
import java.util.List;
import java.util.stream.Collectors;

public class Main {
    public static void main(String[] args) {
        List<String> strings = Arrays.asList("Настя", "Настя",
            "Бауманка", "улица", "Бауманка", "гараж", "солнце");

        List<String> distinctStrings = strings.stream()
            .distinct()
            .collect(Collectors.toList());

        System.out.println("Строки без повторов: " +
            distinctStrings);
    }
}
```

Работа программы показана на рисунке 2.

A screenshot of a terminal window with a dark background. It shows the output of the Java program: "Строки без повторов: [Настя, Бауманка, улица, гараж, солнце]" on the first line and "Process finished with exit code 0" on the second line.

```
Строки без повторов: [Настя, Бауманка, улица, гараж, солнце]
Process finished with exit code 0
```

Рисунок 2 – Работа программы

Задание 3: использовать ТОЛЬКО методы Stream API. Циклов и условий быть не должно. 2. Задана коллекция строк. Преобразовать в MAP, где первый символ ключ, второй – значение.

Код класса Main:

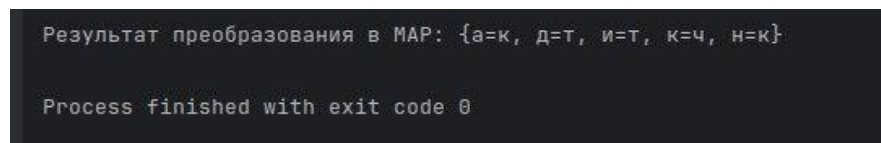
```
import java.util.Arrays;
import java.util.List;
import java.util.Map;
import java.util.stream.Collectors;

public class Main {
    public static void main(String[] args) {
        List<String> strings = Arrays.asList("нк", "ак", "дт",
"кч", "ит");

        Map<Character, Character> resultMap = strings.stream()
            .collect(Collectors.toMap(s -> s.charAt(0), s ->
s.charAt(1)));

        System.out.println("Результат преобразования в MAP: " +
resultMap);
    }
}
```

Работа программы показана на рисунке 3.



```
Результат преобразования в MAP: {a=к, д=т, и=т, к=ч, н=к}

Process finished with exit code 0
```

Рисунок 3 – Работа программы

Задание 4: использовать ТОЛЬКО методы Stream API. Циклов и условий быть не должно. Задана коллекция строк. Преобразовать в MAP, сгруппировав по первому символу строки.

Код класса Main:

```
import java.util.Arrays;
import java.util.List;
```

```

import java.util.Map;
import java.util.stream.Collectors;

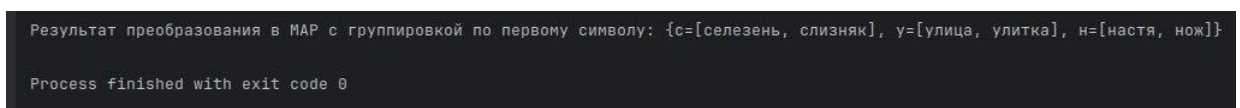
public class Main {
    public static void main(String[] args) {
        List<String> strings = Arrays.asList("настя", "нож",
"улица", "улитка", "селезень", "слизняк");

        Map<Character, List<String>> groupedByFirstChar =
strings.stream()
                .collect(Collectors.groupingBy(s ->
s.charAt(0)));

        System.out.println("Результат преобразования в MAP с
группировкой по первому символу: " + groupedByFirstChar);
    }
}

```

Работа программы показана на рисунке 4.



```

Результат преобразования в MAP с группировкой по первому символу: {с=[селезень, слизняк], у=[улица, улитка], н=[настя, нож]}
Process finished with exit code 0

```

Рисунок 4 – Работа программы

Вывод: во время выполнения лабораторной работы были изучена stream API в java.