|  |  |
| --- | --- |
|  | **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\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

НАПРАВЛЕНИЕ ПОДГОТОВКИ **09.04.01 Информатика и вычислительная техника**

МАГИСТЕРСКАЯ ПРОГРАММА **09.04.01/12 Интеллектуальный анализ больших**

**данных в системах поддержки принятия решений.**

**Отчет**

**по лабораторной работе № 9**

**Название: S**tream\_API

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

Студент \_\_\_ИУ6-23М\_\_\_\_ **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_**А.Д. Зеленский**\_\_\_**

(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_**П. В. Степанов**\_\_\_\_**

(Подпись, дата) (И.О.Фамилия)

**Цель:** ознакомиться с базовыми принципами языка Java для работы с большими данными.

**Вариант 1:**

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

1. Задана коллекция строк. Получить список без дубликатов с сохранением порядка.
2. Задана коллекция строк. Вернуть количество вхождений строки.

**Решение:**

Ниже представлен код класса первой программы, а на рисунке 1 – результат работы программы.

import java.util.Arrays;

import java.util.List;

import java.util.stream.Collectors;

public class Main {

// Задана коллекция строк. Получить список без дубликатов с сохранением порядка.

public static void main(String[] args) {

List<String> strings = Arrays.asList("Панки", "хой", "Панки", "хой", "свет", "Свет");

List<String> Strings1 = strings.stream().distinct().collect(Collectors.toList());

System.out.println(Strings1);

}

}

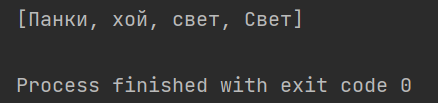


Рисунок 1 – Результат работы программы

Ниже представлен код класса второй программы, а на рисунке 2 – результат работы программы.

import java.util.Arrays;

import java.util.List;

public class Main {

// Задана коллекция строк. Вернуть количество вхождений строки.

public static void main(String[] args) {

List<String> strings = Arrays.asList("Панки", "хой", "Панки", "хой", "свет", "Свет");

String String1 = "хой";

long count = strings.stream()

.filter(s -> s.equals(String1))

.count();

System.out.println("Count: " + count);

}

}

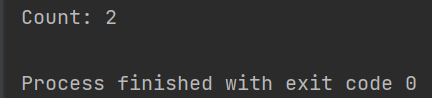


Рисунок 2 – Результат работы программы

**Вариант 2:**

1. Задана коллекция чисел. Получить сумму всех кратных 7.
2. Задана коллекция чисел. С помощью метода reduce вернуть максимум и минимум.

**Решение:**

Ниже представлен кусок кода первой программы, а на рисунке 3 – результат работы программы.

import java.util.Arrays;

import java.util.List;

public class Main {

//Задана коллекция чисел. Получить сумму всех кратных 7.

public static void main(String[] args) {

List<Integer> numbers = Arrays.asList(7, 1, 15, 14, 4, 3, -7);

int sum = numbers.stream()

.filter(n -> n % 7 == 0)

.mapToInt(Integer::intValue)

.sum();

System.out.println("Sum: " + sum);

}

}

}

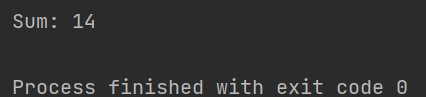


Рисунок 3 – Результат работы программы

Ниже представлен часть кода второй программы, а на рисунке 4 – результат работы программы.

import java.util.Arrays;

import java.util.List;

import java.util.Optional;

public class Main {

//Задана коллекция чисел. С помощью метода reduce вернуть максимум и минимум.

public static void main(String[] args) {

List<Integer> numbers = Arrays.asList(5, 10, 3, 8, 15, 2);

Optional<Integer> max = numbers.stream()

.reduce(Integer::max);

Optional<Integer> min = numbers.stream()

.reduce(Integer::min);

System.out.println("Max: " + max.get());

System.out.println("Min: " + min.get());

}

}}

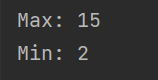


Рисунок 4 – Результат работы программы

**Вывод:** были получены базовые представления о работе языка java с STREAM\_API. Были написаны программы согласно вариантам.