|  |  |
| --- | --- |
| Gerb-BMSTU_01 | **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.**

**Отчет**

|  |  |
| --- | --- |
| **по лабораторной работе №** | 9 |
| **Вариант №** 8 |  |

**Название:**

Использование Stream API

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Студент | ИУ6-23М | |  |  | Н.М. Иванюк |
|  | | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  | |  |  |  |  |
| Преподаватель | |  |  |  | П.В. Степанов |
|  | |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2022

## Цель работы

Получение навыков работы с Stream API языка программирования Java.

## Ход работы

*Задание 1.*

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

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

9. Задана коллекция строк. Вернуть количество вхождений строки.

Листинг 1 – Код программы первого задания

import java.util.Arrays;

import java.util.List;

import java.util.stream.Collectors;

public class App {

public static void main(String[] args) throws Exception {

List<String> list = Arrays.asList("A", "B", "C", "D", "A", "B", "C");

// Get list without duplicates

List<String> distinctItems = list.stream()

.distinct()

.collect(Collectors.toList());

// Let's verify distinct elements

System.out.println(distinctItems);

List<String> list2 = Arrays.asList("A", "B", "C", "D", "A", "B", "C");

long count = list2.stream().filter(c -> c == "A").count();

System.out.println(count);

}

}

Приведем результаты выполнения данного кода.

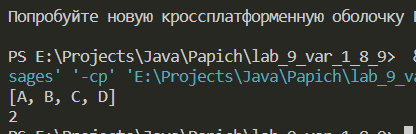


Рисунок 1 – Результат выполнения варианта задания 1

*Задание 2.*

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

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

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

Листинг 3 – Код выполнения задания 2

import java.util.Arrays;

import java.util.List;

public class App {

public static void main(String[] args) throws Exception {

List<Integer> integers = Arrays.asList(1, 2, 3, 4, 5, 7, 14);

Integer sum = integers.stream()

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

.reduce(0, Integer::sum);

System.out.println(sum);

Integer min = integers.stream().reduce(Math::min).orElse(0);

Integer max = integers.stream().reduce(Math::max).orElse(0);

System.out.println(min);

System.out.println(max);

}

}

Результаты выполнения задания приведем на рисунке далее.

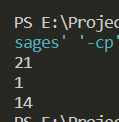


Рисунок 2 – Выполненный код задания 2

## Вывод

По итогам выполнения лабораторной работы были получены навыки программирования с использованием Stream API языка Java.