|  |  |
| --- | --- |
| **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 **09.04.01/12 Интеллектуальный анализ больших**

**данных в системах поддержки принятия решений**

**Отчет**

**по лабораторной работе № 9**

**Название:** Stream API

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

Студент гр. ИУ6-23М **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_** Г. Е. Горский **\_\_\_**

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_**П.В. Степанов **\_\_**

(Подпись, дата) (И.О. Фамилия)

Москва, 2023

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

**Вариант 1:**

1. Задана коллекция строк. Получить массив уникальных значений.
2. Задана коллекция строк. Получить список без дубликатов с сохранением порядка.

**Решение:**

import java.util.ArrayList;  
import java.util.Collection;  
import java.util.Random;  
public class Main {  
 //7. Задана коллекция строк. Получить массив уникальных значений.  
 //8. Задана коллекция строк. Получить список без дубликатов с сохранением порядка.  
 public static void main(String[] args) {  
 System.out.println("Hello world!");  
 Collection<String> stringCollection = new ArrayList<>();  
 stringCollection.add("czc");  
 stringCollection.add("fsdfdsfds");  
 stringCollection.add("czc");  
 stringCollection.add("czcdwdwa");  
 stringCollection.stream().distinct().forEach(System.out::println);

Результат выполнения представлен на рисунке 1.

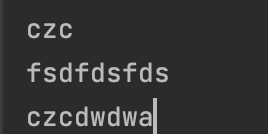
****

Рисунок 1 — пример выполнения

**Вариант 2:**

1. Задана коллекция чисел. Вычесть из каждого значения 10 и получить среднее.
2. Задана коллекция чисел. Получить сумму всех кратных 7.

**Решение:**

//7. Задана коллекция чисел. Вычесть из каждого значения 10 и получить среднее.  
 Collection<Integer> intCollection = new ArrayList<>();  
 Collection<Integer> tempCollection = new ArrayList<>();  
 Random rnd =new Random();  
 for (int i=0; i<10; i++){  
 intCollection.add(rnd.nextInt(0,21));  
 }  
 intCollection.stream().forEach((o)-> {  
 o-= 10;  
 tempCollection.add(o);  
 });  
 System.out.println(intCollection);  
 System.out.println(tempCollection);  
 System.out.println(intCollection.stream().mapToInt(Integer::intValue).average());  
 System.out.println(tempCollection.stream().mapToInt(Integer::intValue).average());  
 //8. Задана коллекция чисел. Получить сумму всех кратных 7.  
 System.out.println(intCollection.stream().filter(x-> x%7==0).mapToInt(Integer::intValue).sum());  
 }  
}

Результат выполнения представлен на рисунке 2.

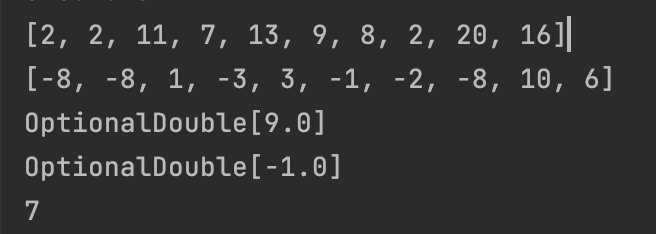


Рисунок 2 — пример выполнения

**Вывод:** были разработаны классы и методы согласно вариантам.