|  |  |
| --- | --- |
| 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**

**Отчет**

**по лабораторной работе № 6**

**Название:** Работа с коллекциями

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

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Студент | ИУ6-21М |  |  | Д.Ю. Ермошин |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | П.В. Степанов |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2020

**Задание**

Научиться работать с Java Collections Framework.

**Выполнение работы**

**Вариант 1**

Ввести строки из файла, записать в список ArrayList. Выполнить сортировку строк.

|  |
| --- |
| package lab6;  import java.io.BufferedReader;  import java.io.IOException;  import java.io.InputStream;  import java.io.InputStreamReader;  import java.util.ArrayList;  import java.util.List;  /\*\*  \* 7. Ввести строки из файла, записать в список ArrayList.  \* Выполнить сортировку строк  \*/  public class Variant1 {  public void work() throws IOException {  final Class clazz = Variant1.class;  final InputStream inputStream = clazz.getResourceAsStream("variant1.txt");  final List<String> data = readFromInputStream(inputStream);  data.sort(String::compareTo);  data.forEach(System.out::println);  }  private List<String> readFromInputStream(final InputStream inputStream) throws IOException {  final List<String> result = new ArrayList<>();  try (final BufferedReader br = new BufferedReader(new InputStreamReader(inputStream))) {  String line;  while ((line = br.readLine()) != null) {  result.add(line);  }  }  return result;  }  } |

**Вариант 2**

На базе коллекций реализовать структуру хранения чисел с поддержкой следующих операций:

- добавление/удаление числа;

- поиск числа, наиболее близкого к заданному (т.е. модуль разницы минимален).

|  |
| --- |
| package lab6;  import java.util.HashSet;  import java.util.NoSuchElementException;  import java.util.Set;  /\*\*  \* 1. На базе коллекций реализовать структуру хранения чисел с поддержкой следующих операций:  \* - добавление/удаление числа;  \* - поиск числа, наиболее близкого к заданному (т.е. модуль разницы минимален).  \*/  public class Variant2 {  private final Set<Integer> values = new HashSet<>();  public boolean addNumber(final int number) {  return values.add(number);  }  public boolean removeNumber(final int number) {  return values.remove(number);  }  public Integer findNearest(final int number) {  if (values.isEmpty())  throw new NoSuchElementException("Collection is empty");  if (values.contains(number))  return number;  int delta = Integer.MAX\_VALUE;  Integer candidate = null;  for (final Integer value : values) {  final int currentDelta = Math.abs(number - value);  if (currentDelta <= delta) {  delta = currentDelta;  candidate = value;  }  }  return candidate;  }  } |

**Заключение**

В данной лабораторной работе мы научились работать с Java Collections Framework.