|  |  |
| --- | --- |
| 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 |

**Название:** Stream API

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

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Студент | ИУ6-22М |  |  | Н.Б. Гендина | |
|  | (Группа) |  | (Подпись, дата) | | (И.О. Фамилия) |
|  |  |  |  | |  |
| Преподаватель |  |  |  | | П.В. Степанов |
|  |  |  | (Подпись, дата) | | (И.О. Фамилия) |

Москва, 2024

**Цель работы:** освоить базовые принципы работы со Stream API на языке Java.

**Задание 1 вариант 4**

Использовать ТОЛЬКО методы Stream API. Циклов и условий быть не должно. Задана коллекция чисел. Разделить числа на четные и нечетные.

|  |
| --- |
| import java.util.List;  import java.util.Map;  import java.util.stream.Collectors;  public class Main {  public static void main(String[] args) {  List<Integer> array = List.of(1, 2, 3, 4, 5, 6, 7, 8, 9, 10);  Map<Boolean, List<Integer>> result = array.stream().collect(  Collectors.partitioningBy(x -> x%2 == 0)  );  System.out.println(result.get(true));  System.out.println(result.get(false));  }  } |

**Задание 1 вариант 5**

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

|  |
| --- |
| import java.util.Arrays;  import java.util.List;  public class Main {  public static void main(String[] args) {  List<String> array = List.of("one", "two", "one", "three", "one", "five", "four", "four", "five");  System.out.println(Arrays.toString(array.stream().distinct().toArray()));  }  } |

**Задание 2 вариант 4**

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

|  |
| --- |
| import java.util.List;  import java.util.stream.Collectors;  public class Main {  public static void main(String[] args) {  List<String> array = List.of("one", "two", "one", "three", "one", "five", "four", "four", "five");  System.out.println(array.stream().collect(Collectors.groupingBy(  x -> x.charAt(0),  Collectors.mapping(x -> x.charAt(1), Collectors.toSet())  )));  }  } |

**Задание 2 вариант 5**

Задана коллекция строк. Преобразовать в MAP, сгруппировав по первому символу строки.

|  |
| --- |
| import java.util.List;  import java.util.stream.Collectors;  public class Main {  public static void main(String[] args) {  List<String> array = List.of("one", "two", "one", "three", "one", "five", "four", "four", "five");  System.out.println(array.stream().collect(Collectors.groupingBy(  x -> x.charAt(0),  Collectors.mapping(x -> x, Collectors.toSet())  )));  }  } |

**Вывод:** в ходе лабораторной работы были освоены базовые принципы работы со Stream API на языке Java.