|  |  |
| --- | --- |
| 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, ИСКУССТВЕННЫЙ ИНТЕЛЛЕКТ 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 |  |  | И.Л. Баришпол |
|  | (Группа) |  | (Подпись, дата) | (И.О. Фамилия) |
|  |  |  |  |  |
| Преподаватель |  |  |  | П.В. Степанов |
|  |  |  | (Подпись, дата) | (И.О. Фамилия) |

Москва, 2023

**Задания**

2. Задана коллекция строк. Вернуть первый элемент коллекции, а также существуют ли все совпадения с шаблоном. Шаблон можно выбрать произвольно.

package lab9

//2. Задана коллекция строк. Вернуть первый элемент коллекции, а также существуют ли все совпадения с шаблоном.

// Шаблон можно выбрать произвольно.

fun main(args: Array<String>) {

val list: MutableList<String> = ArrayList()

list.add("loremol")

list.add("ipsumol")

list.add("dolol")

list.add("molol")

list.add("colol")

list.add("sitol")

list.add("ametol")

println(list.stream().findFirst().get())

val pattern = ".\*ol"

println(list.stream().allMatch { s: String ->

s.matches(

pattern.toRegex()

)

})

}

3. Задана коллекция чисел. Вернуть сумму нечетных чисел.

package lab9

//3. Задана коллекция чисел. Вернуть сумму нечетных чисел.

fun main(args: Array<String>) {

val list: MutableList<Int> = ArrayList()

list.add(5)

list.add(10)

list.add(3)

list.add(8)

list.add(12)

println(list.stream()

.filter { n: Int -> n % 2 != 0 }

.mapToInt { obj: Int -> obj }

.sum())

}

1. Задана коллекция строк. Отсортировать значения по алфавиту и убрать повторы.

package lab9

/\*\*

\* Вариант 2. Задача 1.

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

\* Задана коллекция строк. Отсортировать значения по алфавиту и убрать повторы.

\*/

fun main(args: Array<String>) {

// Обявляем коллекцию строк

val list: MutableList<String> = ArrayList()

list.add("lorem")

list.add("ipsum")

list.add("dolor")

list.add("sit")

list.add("amet")

list.add("sit")

list.add("dolor")

list.add("dolor")

// Сортируем значения по алфавиту и убираем повторы

list.stream().sorted().distinct().forEach { x: String? ->

println(x)

}

}

2. Задана коллекция:

(Класс People: имя и возраст)

Collection<People> peoples = Arrays.asList(

new People("Ivan", 16),

new People("Petr", 23),

new People("Maria", 42)

);

Отсортировать по имени в обратном алфавитном порядке.

package lab9

/\*\*

\* Вариант 2. Задача 2.

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

\* Задана коллекция:

\* (Класс People: имя и возраст)

\* Collection<People> peoples = Arrays.asList(

\* new People("Ivan", 16),

\* new People("Petr", 23),

\* new People("Maria", 42)

\* );

\* Отсортировать по имени в обратном алфавитном порядке.

</People> \*/

fun main(args: Array<String>) {

// Объявляем коллекуию пиплов

val peoples: Collection<People> = listOf(

People("Ivan", 16),

People("Petr", 23),

People("Maria", 42)

)

// Сортируем по имени в обратном алфавитном порядке через Stream API

peoples.stream().sorted { p1: People, p2: People ->

p2.name.compareTo(

p1.name

)

}.forEach { x: People? -> println(x) }

}

// Класс People

internal class People(val name: String, val age: Int) {

// Оверрайд для строкового представления объекта

override fun toString(): String {

return "People{" +

"name='" + name + '\'' +

", age=" + age +

'}'

}

}

**Вывод:** в ходе выполнения данной лабораторной работы были освоены основы работы с Stream API в языке программирования Kotlin.