|  |  |
| --- | --- |
|  | **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\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

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 № 7**

**Название:** Строки. Регулярные выражения.

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

Студент \_\_\_ИУ6-23М\_\_\_\_ **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_**А.Д. Зеленский**\_\_\_**

(Группа) (Подпись, дата) (И.О.Фамилия)

Руководитель **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_**П. В. Степанов**\_\_\_\_**

(Подпись, дата) (И.О.Фамилия)

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

**Вариант 1:**

1. Из небольшого текста удалить все символы, кроме пробелов, не являющиеся буквами. Между последовательностями подряд идущих букв оставить хотя бы один пробел.
2. Из текста удалить все слова заданной длины, начинающиеся на согласную букву.

**Решение:**

Ниже представлен код класса первой программы, а на рисунке 1 – результат работы программы.

public class Main {

public static void main(String[] args) {

String input = "Теккст, содеррржащий различнные симвооолы, такие как !@#$%^&\*()\_-+=[]{},.<>/?\\|~`;:\'\", а также цифры и буквы.";

String output = input.replaceAll("[^\\p{L} ]", "")

.replaceAll("\\s+", " ")

.replaceAll("(.)\\1+", "$1 $1");

System.out.println(output);

}

}

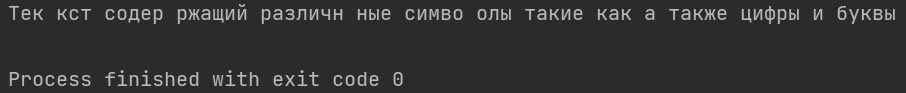


Рисунок 1 – Результат работы программы

Ниже представлен код класса второй программы, а на рисунке 2 – результат работы программы.

public class Main {

public static void main(String[] args) {

String input = "Удалить все слова длиной пять, начинающиеся на согласную букву.";

int wordLength = 5;

// Разбиваем строку на слова

String[] words = input.replaceAll("[^\\p{L} ]", "").split("\\s+");

// Перебираем все слова и заменяем те, которые соответствуют условию

for (String word : words) {

// Проверяем длину слова и первую букву

if (word.length() == wordLength && word.matches("^[бвгджзйклмнпрстфхцчшщ].\*")) {

input = input.replaceAll( word , "");

}

}

System.out.println(input);

}

}

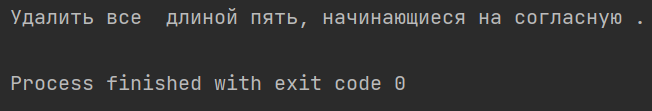


Рисунок 2 – Результат работы программы

**Вариант 2:**

1. Напечатать квитанцию об оплате телеграммы, если стоимость одного слова задана.
2. В стихотворении найти одинаковые буквы, которые встречаются во всех словах.

**Решение:**

Ниже представлен кусок кода первой программы, а на рисунке 3 – результат работы программы.

public class Main {

public static void main(String[] args) {

String input = "Удалить все слова длиной пять, начинающиеся на согласную букву.";

int wordprice = 5;

// Разбиваем строку на слова

String[] words = input.replaceAll("[^\\p{L} ]", "").split("\\s+");

System.out.println("С вам "+ words.length \* wordprice +" рублей \nза "

+ words.length + " слов \n" + wordprice + " рублей за слово.");

}

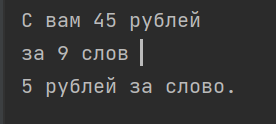
}

Рисунок 3 – Результат работы программы

Ниже представлен часть кода второй программы, а на рисунке 4 – результат работы программы.

public class Main {

public static void main(String[] args) {

String text = "Над розоавым мораем пурапурный закат," +

"Ва темноате искарятся золаотые звезады," +

"Спаит лаес таинственный, иа толаько сверачок спат" +

"Нае хочеат, над зеленым цвеатет дубрава.";

String[] words = text.replaceAll("[^\\p{L} ]", "").toLowerCase().split("\\s+");

Set<Character> commonLetters = new HashSet<>(words[0].length());

for (char letter : words[0].toCharArray()) {

boolean isCommon = true;

for (int i = 1; i < words.length && isCommon; i++) {

if (words[i].indexOf(letter) == -1) {

isCommon = false;

}

}

if (isCommon) {

commonLetters.add(letter);

}

}

System.out.println( commonLetters);

}

}

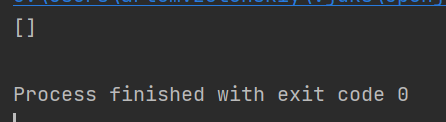


Рисунок 4 – Результат работы программы

**Вариант 3:**

1. На основании правила кодирования, описанного в предыдущем примере, расшифровать заданный набор символов.
2. Напечатать слова русского текста в алфавитном порядке по первой букве. Слова, начинающиеся с новой буквы, печатать с красной строки.

**Решение:**

Ниже представлен код класса первой программы, а на рисунке 5 – результат работы программы.

public class Main {

public static void main(String[] args) {

StringBuilder stringBuilder1= new StringBuilder();

StringBuilder stringBuilder2= new StringBuilder();

StringBuilder stringBuilder3= new StringBuilder();

String str = "Hello world!";

//шифрую

for (int i=3;i<str.length()+3; i+=3){

stringBuilder1.append(str.charAt(i-3));

stringBuilder2.append(str.charAt(i-2));

stringBuilder3.append(str.charAt(i-1));

}

System.out.println(str.length()+3);

System.out.println("1: "+ stringBuilder1);

System.out.println("2: "+ stringBuilder2);

System.out.println("3: "+ stringBuilder3);

//расшифровка

for (int i=0; i<(stringBuilder1.length()+stringBuilder2.length()+stringBuilder3.length())/3; i++){

System.out.print(""+stringBuilder1.charAt(i)+ stringBuilder2.charAt(i) + stringBuilder3.charAt(i));

}

}

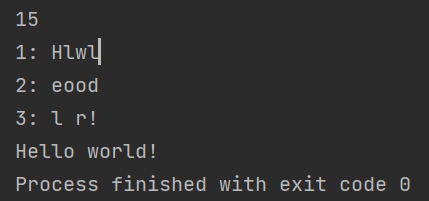
}

Рисунок 5 – Результат работы программы

Ниже представлен код класса второй программы, а на рисунке 6 – результат работы программы.

import java.util.\*;

public class Main {

public static void main(String[] args) {

String text = "Сегодня яблоки покупала в магазине,\n" +

"Клубнику и апельсины тоже купила мне.\n" +

"Но вот дома я поняла, что забыла мандарины,\n" +

"Пришлось второй раз идти в магазин, прости Господины.";

String[] words = text.replaceAll("[^\\p{L} ]", "").toLowerCase().split("\\s+");

TreeMap<Character, ArrayList<String>> map = new TreeMap<>();

for (String word : words) {

if (!word.isEmpty()) {

char firstLetter = word.charAt(0);

ArrayList<String> list = map.getOrDefault(firstLetter, new ArrayList<>());

list.add(word);

map.put(firstLetter, list);

}

}

// Выводим слова на экран в алфавитном порядке с красной строки при изменении первой буквы

char prevLetter = ' ';

for (Map.Entry<Character, ArrayList<String>> entry : map.entrySet()) {

char letter = entry.getKey();

if (letter != prevLetter) {

System.out.println();

System.out.print(Character.toUpperCase(letter) + ": ");

} else {

System.out.print(", ");

}

ArrayList<String> wordsList = entry.getValue();

Collections.sort(wordsList);

System.out.print(String.join(", ", wordsList));

prevLetter = letter;

}

}

}

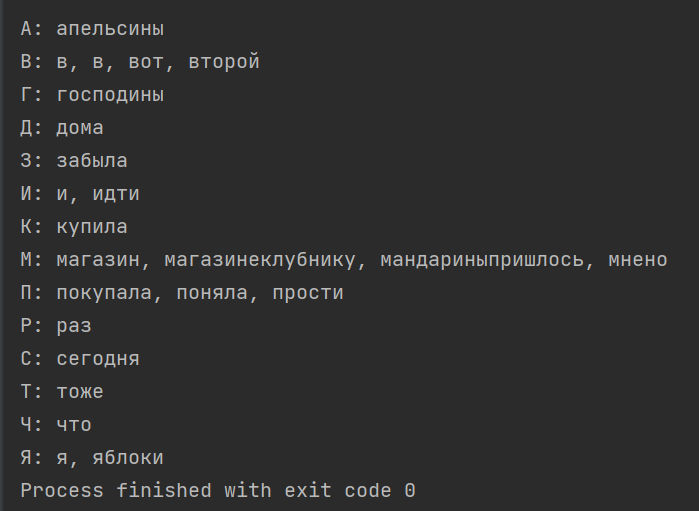


Рисунок 6 – Результат работы программы

**Вариант 4:**

1. Подсчитать, сколько раз заданное слово входит в текст.
2. Преобразовать каждое слово в тексте, удалив из него все последующие (предыдущие) вхождения первой (последней) буквы этого слова.

**Решение:**

Ниже представлен кусок кода первой программы.

public class Main {

public static void main(String[] args) {

String poem = "Над розоавым цвет пурапурный закат," +

"Ва темноате цвет золаотые звезады," +

"Спаит лаес таинственный, иа цвет сверачок спат" +

"Нае хочеат, над цвет цвет дубрава.";

// Разбиваем строку на слова и перебираем их

String[] words = poem.replaceAll("[^\\p{L} ]", "").toLowerCase().split(" ");

String word="цвет";

int count=0;

for (int i=0;i<words.length;i++){

if (words[i].equals(word)){

count++;

}

}

System.out.println("Слово "+ word +" встречается в тексте " + count +" раз!");

}

}

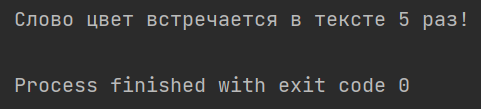


Рисунок 7 – Результат работы программы

Ниже представлен часть кода второй программы.

public class Main {

public static void main(String[] args) {

String str = "Раз два дядя мама папа два";

// Разбиваем строку на слова и перебираем их

String[] words = str.replaceAll("[^\\p{L} ]", "").toLowerCase().split(" ");

for (String word: words){

char ch=word.charAt(0);

System.out.print(ch+word.replaceAll(String.valueOf(ch),"")+ " ");

}

System.out.println();

System.out.println("Hello world!");

}

}

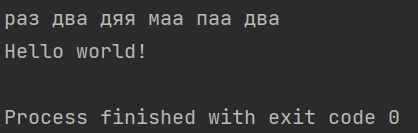


Рисунок 8 – Результат работы программы

**Вывод:** были получены базовые представления о работе языка java с строками, регулярными выражениями. Были написаны программы согласно вариантам.