*Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение высшего образования*

|  |  |
| --- | --- |
| **Gerb-BMSTU_01** | ***«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

ДИСЦИПЛИНА Языки программирования для работы с большими данными

**Отчет**

**по лабораторной работе № 7**

**Вариант 5**

Студент гр. ИУ6-21М **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_**Дейкин И. И.\_\_\_

(Подпись, дата) (И.О. Фамилия)

Преподаватель  **\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

(Подпись, дата) (И.О. Фамилия)

Москва, 2020

**ЗАДАНИЕ 1**

*Условие*

В тексте после k-го символа вставить заданную подстроку.

*Решение*

Код решения приведен ниже:

|  |
| --- |
| package com.company;  import java.util.Scanner;  public class Main {  public static void main(String[] args) {  // write your code here  Scanner scan = new Scanner(System.in);  System.out.println("Enter the string:");  String str1 = scan.nextLine();  System.out.println("Enter the substring:");  String str2 = scan.nextLine();  System.out.println("Enter the position:");  int k = scan.nextInt();  String stres = str1.substring(0, k) + str2 + str1.substring(k, str1.length());  System.out.println(stres);  }  } |

**ЗАДАНИЕ 2**

*Условие*

В стихотворении найти количество слов, начинающихся и заканчивающихся гласной буквой.

*Решение*

Описание решения:

|  |
| --- |
| package com.company;  import java.io.File;  import java.io.FileNotFoundException;  import java.util.Scanner;  import java.util.regex.Matcher;  import java.util.regex.Pattern;  public class Main {  public static void main(String[] args) {  // write your code here  int res = 0;  File input = new File("Input.txt");  try (Scanner scan = new Scanner(input)) {  while (scan.hasNextLine()) {  String buf = scan.nextLine();  String[] words = buf.split("[\\s,.:!?()\"—]+");  Pattern pattern = Pattern.compile("^[AEYUIOaeyuio].\*[aeyuio]$");  for (String word : words) {  Matcher matcher = pattern.matcher(word);  if (matcher.find()) {  System.out.println(word);  res++;  }  }  }  System.out.println(res);  }  catch (FileNotFoundException exp) {  System.out.println("No such file found!");  }  }  } |

**ЗАДАНИЕ 3**

*Условие*

В каждом предложении текста поменять местами первое слово с последним, не изменяя длины предложения.

*Решение*

Код решения приведен ниже:

|  |
| --- |
| package com.company;  import java.io.File;  import java.io.FileNotFoundException;  import java.util.Scanner;  import java.util.regex.Matcher;  import java.util.regex.Pattern;  public class Main {  public static void main(String[] args) {  // write your code here  File input = new File("Input.txt");  try (Scanner scan = new Scanner(input)) {  String res\_big = "";  while (scan.hasNextLine()) {  String buf = scan.nextLine();  String[] sentences = buf.split("[\\!|\\.|\\?]\\s?");  int p = 0;  //Pattern pattern = Pattern.compile("^[AEYUIOaeyuio].\*[aeyuio]$");  for (String sentence : sentences) {  String[] words = sentence.split("[\\s,.:!?()\"—]+");  String w1 = words[0].toLowerCase(); // Переводим первое слово в lowercase  String w2 = words[words.length - 1];  String w3 = w2.substring(0, 1).toUpperCase() + w2.substring(1); // Увелич первую букву последнего слова  int pos = sentence.indexOf(w2);  String res = w3 + sentence.substring(w1.length(), pos) + w1;  int pos2 = buf.indexOf(res);  p += res.length();  res\_big += res + buf.charAt(p) + " "; //Добавляем знак в конце предложения  p += 2;  }  }  System.out.println(res\_big);  }  catch (FileNotFoundException exp) {  System.out.println("No such file found!");  }  }  } |

**ЗАДАНИЕ 4**

*Условие*

Заменить все одинаковые рядом стоящие символы в тексте одним символом.

*Решение*

Код решения приведен ниже:

|  |
| --- |
| package com.company;  import java.io.File;  import java.io.FileNotFoundException;  import java.util.Scanner;  public class Main {  public static void main(String[] args) {  File input = new File("Input.txt");  try (Scanner scan = new Scanner(input)) {  while (scan.hasNextLine()) {  String buf = scan.nextLine();  System.out.println(buf.replaceAll("([A-Za-z])\\1+", "$1"));  }  }  catch (FileNotFoundException exp) {  System.out.println("No such file found!");  }  }  } |

**ВЫВОД**

В ходе выполнения данной работы была проведена работа со строками, применялись регулярные выражения.