|  |
| --- |
|  |
| МИНОБРНАУКИ РОССИИ |
| *Федеральное государственное бюджетное образовательное учреждение высшего образования* ***«МИРЭА – Российский технологический университет»***  **РТУ МИРЭА** |

**Институт информационных технологий (ИТ)**

**Кафедра инструментального и прикладного программного обеспечения (ИиППО)**

**Дисциплина «Программирование на языке Джава»**

**ОТЧЕТ**

**ПО ПРАКТИЧЕСКОМУ ЗАНЯТИЮ №12**

Выполнил студент группы ИНБО-02-20 Баринов.И.В.

Принял Степанов П.В.

Практические работы выполнены «\_\_\_»\_\_\_\_\_\_\_2021г.

«\_\_\_\_\_\_\_\_\_\_\_\_\_\_» «\_\_\_»\_\_\_\_\_\_\_2021г.

Отметка о выполнении

**Москва – 2021 г.**

## **Задание**

Закрепить знания в области обработки строк, научиться применять методы класса String и других классов для обработки строк.

## **Ход Работы**

В ходе выполнения работы были получены следующие исходные коды:

package Practise12;  
  
public class Test {  
 public static void main(String[] args) {  
 Person person = new Person("Баринов", "Иван", "Владиславович");  
 System.*out*.println(person.getSNP());  
  
 Person person1 = new Person("Баринов", "Иван");  
 System.*out*.println(person1.getSNP());  
  
 Person person2 = new Person("Баринов");  
 System.*out*.println(person2.getSNP());  
 }  
}

----

package Practise12;  
  
public class Person {  
 private String surname;  
 private String name = "Не определено";  
 private String patronymic = "Не определено";  
  
 public Person(String surname) {  
 this.surname = surname;  
 }  
  
 public Person(String surname, String name) {  
 this.surname = surname;  
 this.name = name;  
 }  
  
 public Person(String surname, String name, String patronymic) {  
 this.surname = surname;  
 this.name = name;  
 this.patronymic = patronymic;  
 }  
  
 public StringBuilder getSNP() {  
 return new StringBuilder("Person{" +  
 "surname='" + surname + '\'' +  
 ", name='" + name + '\'' +  
 ", patronymic='" + patronymic + '\'' +  
 '}');  
 }  
}

----

package Practise12.Exercise5;  
  
import java.io.BufferedReader;  
import java.io.FileReader;  
import java.io.IOException;  
import java.io.InputStreamReader;  
import java.util.ArrayList;  
import java.util.Collections;  
import java.util.Scanner;  
  
public class Solution {  
 public static void main(String[] args) throws IOException  
 {  
 BufferedReader reader = new BufferedReader(new InputStreamReader(System.*in*));  
 Scanner scanner = new Scanner(new FileReader(reader.readLine()));  
 String cities = "";  
 while (scanner.hasNextLine())  
 {  
 cities = scanner.nextLine();  
 }  
 StringBuilder result = *getLine*(cities.split(" "));  
 System.*out*.println(result.toString());  
 }  
 public static StringBuilder getLine(String... words) {  
 if (words == null || words.length == 0)  
 { return new StringBuilder(); }  
 if ("".equals(words[0]) || words.length == 1)  
 {return new StringBuilder(words[0]); }  
  
 StringBuilder result = new StringBuilder();  
  
 ArrayList<String> list = new ArrayList<>();  
 for (String word : words)  
 if (!word.equals(""))  
 list.add(word);  
 while (!*isOK*(list))  
 {  
 Collections.*shuffle*(list);  
 }  
 for (String s : list)  
 result.append(s).append(" ");  
 result.deleteCharAt(result.length()-1);  
 return result;  
 }  
 public static boolean isOK(ArrayList<String> list)  
 {  
 for (int i = 0; i < list.size()-1; i++)  
 {  
 String first = list.get(i);  
 String second = list.get(i+1);  
 first = first.toLowerCase();  
 second =second.toLowerCase();  
 if (first.charAt(first.length()-1)!=second.charAt(0))  
 return false;  
 }  
 return true;  
 }  
}

----

package Practise12.Exercise4;  
  
public class Test {  
 public static void main(String[] args) {  
 TelephoneNumber telephoneNumber = new TelephoneNumber("+3249670376973");  
 System.*out*.println(telephoneNumber);  
 TelephoneNumber telephoneNumber1 = new TelephoneNumber("89670376973");  
 System.*out*.println(telephoneNumber1);  
 }  
}

----

package Practise12.Exercise4;  
  
public class TelephoneNumber {  
 private String countryCode;  
 private String firstTriad;  
 private String secondTriad;  
 private String foursome;  
  
 public TelephoneNumber(String telephoneNumber) {  
 if (telephoneNumber.startsWith("+")) {  
 char[] charArray = telephoneNumber.toCharArray();  
 foursome = (charArray[telephoneNumber.length() - 4] + "") + (charArray[telephoneNumber.length() - 3] + "") +  
 (charArray[telephoneNumber.length() - 2] + "") + (charArray[telephoneNumber.length() - 1] + "");  
 secondTriad = (charArray[telephoneNumber.length() - 7] + "") + (charArray[telephoneNumber.length() - 6] + "")  
 + (charArray[telephoneNumber.length() - 5] + "");  
 firstTriad = (charArray[telephoneNumber.length() - 10] + "") + (charArray[telephoneNumber.length() - 9] + "") +  
 (charArray[telephoneNumber.length() - 8] + "");  
 countryCode = telephoneNumber.substring(0, telephoneNumber.length()-10);  
 } else if (telephoneNumber.startsWith("8")) {  
 char[] charArray = telephoneNumber.toCharArray();  
 foursome = (charArray[telephoneNumber.length() - 4] + "") + (charArray[telephoneNumber.length() - 3] + "") +  
 (charArray[telephoneNumber.length() - 2] + "") + (charArray[telephoneNumber.length() - 1] + "");  
 secondTriad = (charArray[telephoneNumber.length() - 7] + "") + (charArray[telephoneNumber.length() - 6] + "")  
 + (charArray[telephoneNumber.length() - 5] + "");  
 firstTriad = (charArray[telephoneNumber.length() - 10] + "") + (charArray[telephoneNumber.length() - 9] + "") +  
 (charArray[telephoneNumber.length() - 8] + "");  
 countryCode = "+7";  
 }  
 }  
  
 @Override  
 public String toString() {  
 return "TelephoneNumber{" +  
 "countryCode=" + countryCode +  
 ", firstTriad=" + firstTriad +  
 ", secondTriad=" + secondTriad +  
 ", foursome=" + foursome +  
 '}';  
 }  
}

----

package Practise12.Exercise3;  
  
import java.util.Arrays;  
  
public class Test {  
 public static void main(String[] args) {  
 String[] shirts = {  
 "S001,Black Polo Shirt,Black,XL", "S002,Black Polo Shirt,Black,L",  
 "S004,Blue Polo Shirt,Blue,M", "S005,Tan Polo Shirt,Tan,XL",  
 "S006,Black T-Shirt,Black,XL", "S007,White T-Shirt,White,XL",  
 "S008,White T-Shirt,White,L", "S009,Green T-Shirt,Green,S",  
 "S010,Orange T-Shirt,Orange,S", "S011,Maroon Polo Shirt,Maroon,S"};  
 Shirt[] newShirts = new Shirt[10];  
 for (int i = 0; i < 10; i++) {  
 newShirts[i] = new Shirt(shirts[i]);  
 }  
 for (int i = 0; i < 10; i++) {  
 System.*out*.println(newShirts[i]);  
 }  
 }  
}

----

package Practise12.Exercise3;  
  
public class Shirt {  
 private String code;  
 private String name;  
 private String color;  
 private String size;  
  
 public Shirt(String str) {  
 String[] specifications = str.split(",");  
 this.code = specifications[0];  
 this.name = specifications[1];  
 this.color = specifications[2];  
 this.size = specifications[3];  
 }  
  
 @Override  
 public String toString() {  
 return "Shirt{" +  
 "code='" + code + '\'' +  
 ", name='" + name + '\'' +  
 ", color='" + color + '\'' +  
 ", size='" + size + '\'' +  
 '}';  
 }  
}

----

package Practise12.Exercie2;  
  
public class Test {  
 public static void main(String[] args) {  
 Address address = new Address();  
  
 address.setAddress1("Россия,Московская область,Химки,Горшина,3,2,1");  
 System.*out*.println(address);  
  
 address.setAddress2("Россия;Московская область;Химки;Горшина;3;2;1");  
 System.*out*.println(address);  
 }  
}

----

package Practise12.Exercie2;  
  
import java.util.StringTokenizer;  
  
public class Address {  
 private String country;  
 private String region;  
 private String city;  
 private String street;  
 private String house;  
 private String building;  
 private String flat;  
  
  
 public void setAddress1(String address) {  
 String[] partsOfAddress = getAddressDefault(address);  
 this.country = partsOfAddress[0];  
 this.region = partsOfAddress[1];  
 this.city = partsOfAddress[2];  
 this.street = partsOfAddress[3];  
 this.house = partsOfAddress[4];  
 this.building = partsOfAddress[5];  
 this.flat = partsOfAddress[6];  
 }  
  
 public void setAddress2(String address) {  
 StringTokenizer tokenizer = getAddressSeparator(address);  
 int i = 0;  
 String[] partsOfAddress = new String[7];  
 while (tokenizer.hasMoreTokens()) {  
 String token = tokenizer.nextToken();  
 partsOfAddress[i] = token;  
 i++;  
 }  
 this.country = partsOfAddress[0];  
 this.region = partsOfAddress[1];  
 this.city = partsOfAddress[2];  
 this.street = partsOfAddress[3];  
 this.house = partsOfAddress[4];  
 this.building = partsOfAddress[5];  
 this.flat = partsOfAddress[6];  
 }  
  
 public String[] getAddressDefault(String address) {  
 return address.split(",");  
 }  
  
 public StringTokenizer getAddressSeparator(String address) {  
 return new StringTokenizer(address, ",.;");  
 }  
  
 @Override  
 public String toString() {  
 return "Address{" +  
 "country='" + country + '\'' +  
 ", region='" + region + '\'' +  
 ", city='" + city + '\'' +  
 ", street='" + street + '\'' +  
 ", house='" + house + '\'' +  
 ", building='" + building + '\'' +  
 ", flat='" + flat + '\'' +  
 '}';  
 }  
}

## **Вывод**

Получены знания по обработке строк

GitHub:

https://github.com/BarinovIvan/RuPractises/tree/master/src/Practise12