|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  |  | | --- | --- | --- | |  |  |  | | МИНОБРНАУКИ РОССИИ | | | | Федеральное государственное бюджетное образовательное учреждение  высшего образования  **«МИРЭА – Российский технологический университет»**  **РТУ МИРЭА** | | |   Институт Информационных технологий | |
|  | |
| Кафедра Математического обеспечения и стандартизации информационных технологий | |
|  | |
|  | |

|  |  |
| --- | --- |
| **ОТЧЕТ ПО ЛАБОРАТОРНОЙ РАБОТЕ № 2** | |
| **по дисциплине** | |
| **«**Объектно-ориентированное программирование**»**  **Тема: «Строки в Java. Применение регулярных выражений для поиска**  **заданных слов.»** | |
|  | |
| Выполнил студент группы ИКБО-17-18 | Мисаилиди А.А. |
| Принял преподаватель | Баранова И.А. |

|  |  |  |
| --- | --- | --- |
| Лабораторная работа выполнена | «\_\_»\_\_\_\_\_\_\_201\_\_ г. | *(подпись студента)* |
|  |  |  |
| «Зачтено» | «\_\_»\_\_\_\_\_\_\_201\_\_ г. | *(подпись руководителя)* |

Москва 2019

Задание 1. Применение класса String.

1. Условие задачи

Дан текст, который содержит информацию о продажах в магазине и о каждой продаже известно: название товара (одно слово строчными буквами) и сумма. Продаж может быть несколько и информация по одной продаже завершается символом ;(точка с запятой).

* Найти сумму всех продаж.
* Отредактировать текст: название товара должно начинаться с прописной буквы.
* Сформировать рейтинг товаров, учтенных в продажах (по количеству продаж). Результат представить строкой.
* Показать информацию по заданной по номеру продаже.

2. Постановка задачи

Дано: строки с данными о продажах.

Ограничения на решаемую задачу: вводимые данные должны быть корректными.

3. Определение входных и выходных данных

Входные данные – строка с данными о продажах в формате «товар цена; …»

Результат –

* Сумма всех продаж
* Отредактированный текст
* Ранжированный список
* Информация по заданному номеру

4. Разработка тестов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Номер теста* | *Исходные данные* | *Эталон результата* | *Результат программы* | *Отметка о правильном прохождении теста* |
| 1. | Ирис 500; клен 400 | 900 | 900 | Пройден |
| 2. | Ирис 500; клен 400 | ирис 500; клен 400 | ирис 500; клен 400 | Пройден |
| 3. | Ирис 300; клен 400 | клен 400; Ирис 300 | клен 400; Ирис 300 | Пройден |
| 4. | Ирис 300; клен 400  2 | Клен 400 | Клен 400 | Пройден |

5. Исходный код программы

package com.company;

import java.lang.System;

import java.util.Scanner;

public class Task1 {

private String txt;

private String[] list;

private String[] prices;

Task1() {

txt = new String("");

}

void consoleInput() {

Scanner in = new Scanner(System.in);

txt = in.nextLine();

}

void consoleInputProcessing() {

System.out.print("Введите информацию о продажах в формате(товар цена; ...): ");

this.consoleInput();

list = txt.split("; ");

prices = new String[list.length];

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

prices[i] = list[i].split(" ")[1];

}

}

int soldSum() {

int res = 0;

for (String el: prices) {

res += Integer.parseInt(el);

}

return res;

}

void soldSumProcessing() {

System.out.print("Сумма всех продаж: ");

System.out.println(this.soldSum());

}

void edit() {

this.txt.toLowerCase();

}

String rating() {

String buf;

String[] listCopy = list;

String[] pricesCopy = prices;

for (int i = 0; i < prices.length - 1; i++) {

for (int j = 0; j < prices.length - 1 - i; j++) {

if (Integer.parseInt(pricesCopy[j]) < Integer.parseInt(pricesCopy[j + 1])) {

buf = pricesCopy[j];

pricesCopy[j] = pricesCopy[j + 1];

pricesCopy[j + 1] = buf;

buf = listCopy[j];

listCopy[j] = listCopy[j + 1];

listCopy[j + 1] = buf;

}

}

}

String res = String.join(";", listCopy);

return res;

}

void ratingProcessing() {

System.out.println("Список продаж, отсортированный по количеству продаж: " + this.rating());

}

void indexOutput(int index) {

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

if (i == index - 1) {

System.out.println("Наименование товара: " + list[i].split(" ")[0]);

System.out.println("Сумма продаж: " + list[i].split(" ")[1]);

}

}

}

void indexOutputProcessing() {

System.out.print("Введите номер продажи по списку: ");

Scanner in = new Scanner(System.in);

int index = in.nextInt();

indexOutput(index);

}

void consoleOutput() {

System.out.print("Список продаж: " + String.join("; ", list));

}

}

Задание 2. Применение класса StringBuilder

1. Условие задачи

Дан текст, который содержит информацию о продажах в магазине и о

каждой продаже известно: название товара (одно слово строчными

буквами) и сумма. Продаж может быть несколько и информация по

одной продаже завершается символом ;(точка с запятой).

* Удалить из текста сведения о продажах с суммой равной 0
* Найти информацию о продаже, имеющей наименьшую длину. Заменить эту продажу на продажу наибольшего размера.
* Добавить новую продажу в текст.

2. Постановка задачи

Дано: строки с данными о продажах.

Ограничения на решаемую задачу: вводимые данные должны быть корректными.

3. Определение входных и выходных данных

Входные данные – строка с данными о продажах в формате «товар цена; …»

Результат –

* Текст без нулевых продаж
* Информация о продажах наименьшей длины заменяется на продажу наибольшей длины
* Добавление новой продажи

4. Разработка тестов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Номер теста* | *Исходные данные* | *Эталон результата* | *Результат программы* | *Отметка о правильном прохождении теста* |
| 1. | Ирис 500; клен 0 | Ирис 500 | Ирис 500 | Пройден |
| 2. | Ирис 500; язь 4 | Ирис 500; Ирис 500 | Ирис 500; Ирис 500 | Пройден |
| 3. | Ирис 300; клен 400  Язь 4 | Ирис 300; клен 400; Язь 4 | Ирис 300; клен 400; Язь 4 | Пройден |

5. Исходный код программы

package com.company;

import java.util.Scanner;

public class Task2 {

private StringBuffer txt;

private String[] list;

private String[] prices;

Task2() {

txt = new StringBuffer("");

}

void consoleInput() {

Scanner in = new Scanner(System.in);

txt = new StringBuffer(in.nextLine());

}

void consoleInputProcessing() {

System.out.print("Введите информацию о продажах в формате(товар цена; ...): ");

this.consoleInput();

list = txt.toString().split("; ");

prices = new String[list.length];

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

prices[i] = list[i].split(" ")[1];

}

}

void zeroDelete() {

String[] listCopy = list;

String[] pricesCopy = prices;

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

if (prices[i] == "0") {

for (int j = 0; j < list.length; j++) {

if (j < i) {

listCopy[j] = list[j];

pricesCopy[j] = prices[j];

}

if (j > i) {

listCopy[j] = list[j + 1];

pricesCopy[j] = prices[j + 1];

}

}

}

}

list = listCopy;

prices = pricesCopy;

txt = new StringBuffer(String.join("; ", list));

}

void zeroDeleteProcessing() {

this.zeroDelete();

System.out.println("Сведения о продажах с суммой равной 0 удалены.");

}

void minMaxReplacement() {

int min = 0;

int max = 0;

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

if (list[i].length() < list[min].length()) {

min = i;

}

if (list[i].length() < list[max].length()) {

max = i;

}

}

list[min] = list[max];

prices[min] = list[max];

txt = new StringBuffer(String.join("; ", list));

}

void minMaxReplacementProcessing() {

this.minMaxReplacement();

System.out.println("Данные минимальной длины заменены на данные максимальной длины.");

}

void newInfoAppend(String name, String val) {

String[] listCopy = new String[list.length + 1];

String[] pricesCopy = new String[list.length + 1];

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

listCopy[i] = list[i];

pricesCopy[i] = prices[i];

}

listCopy[listCopy.length] = name + " " + val;

pricesCopy[pricesCopy.length] = val;

}

void newInfoAppendProcessing() {

Scanner in = new Scanner(System.in);

System.out.print("Введите название товара: ");

String name = in.nextLine();

System.out.print("Введите сумму: ");

String val = in.nextLine();

newInfoAppend(name, val);

txt = new StringBuffer(String.join("; ", list));

}

void indexOutput(int index) {

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

if (i == index - 1) {

System.out.println("Наименование товара: " + list[i].split(" ")[0]);

System.out.println("Сумма продаж: " + list[i].split(" ")[1]);

}

}

}

void indexOutputProcessing() {

System.out.print("Введите номер продажи по списку: ");

Scanner in = new Scanner(System.in);

int index = in.nextInt();

indexOutput(index);

}

void consoleOutput() {

System.out.print("Список продаж: " + String.join("; ", list));

}

}

Задание 3. Регулярные выражения

1. Условие задачи

Определите, что переданная строка является корректным временем вида '12:59', '23:41', '00:12', '00:00', '09:15'. Время '24.00', '25.00', '12.60', '12.93', '41.93' является некорректным.

Удалите из текста все значения времени из промежутка 00:00 до 02:00.

2. Постановка задачи

Дано: время в формате «ЧЧ:ММ».

Ограничения на данные: время должно быть введено корректно.

3. Определение входных и выходных данных

Входные данные – три целых числа, вводятся с клавиатуры.

Результат – сообщение, выводимое на монитор, указывающее на вид треугольника или диагностическое сообщение о том, что результат получен быть не может.

4. Разработка тестов

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Номер теста* | *Исходные данные* | *Эталон результата* | *Результат программы* | *Отметка о правильном прохождении теста* |
| 1. | 23:58 | Корректно | Корректно | Пройден |
| 2. | 00:00 | Корректно | Корректно | Пройден |
| 3. | 24:72 | Некорректно | Некорректно | Пройден |
| 4. | Приходи с 00:00 до 1:30 | Приходи с до | Приходи с до | Пройден |

5. Исходный код программы

package com.company;

import java.util.Scanner;

import java.util.regex.\*;

public class Task3 {

private String txt1;

private String txt2;

Task3() {

txt1 = new String("");

txt2 = new String("");

}

void consoleInput() {

Scanner in = new Scanner(System.in);

txt1 = new String(in.nextLine());

}

void consoleInputProcessing() {

System.out.print("Введите время в формате \"ЧЧ:ММ\": ");

this.consoleInput();

}

boolean correctnessCheck(String time) {

return Pattern.matches("(([0-1]{1}[0-9]{1})|(2[0-3])):[0-5]{1}[0-9]{1}", time);

}

void correctnessCheckProcessing() {

if (correctnessCheck(txt1)) {

System.out.println("Ведено корректное время");

} else {

System.out.println("Введено некорректное время");

}

}

void consoleTextInput() {

Scanner in = new Scanner(System.in);

txt2 = new String(in.nextLine());

}

void textTimeEdit() {

txt2.replaceAll("(0[0-1]:[0-5][0-9])|(02:00)", "");

}

}