



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 №17-18

Выполнил студент группы ИНБО-02-20

Лукияненко Д.В.

Принял

Степанов П.В.

Практическая работа выполнена «__» _____ 2021 г.

«_____»
Отметка о выполнении

«__» _____ 2021 г.

Москва – 2021 г.

СОДЕРЖАНИЕ

Цель работы	3
Задание	3
Репозиторий	3
Выполнение работы	3
Код выполненной работы	4
Вывод.....	6

Цель работы

Цель данной практической работы – Научиться реализовывать регулярные выражения и применять их на практике.

Задание

Задание 1

Написать регулярное выражение, определяющее является ли данная строка строкой "abcdefghijklmnpqrstuv18340" или нет.

– пример правильных выражений: abcdefghijklmnpqrstuv18340.

– пример неправильных выражений: abcdefghijklmnoasdfsdpqrstuv18340.

Задание 2

Написать регулярное выражение, определяющее является ли заданная строка правильным MAC-адресом.

– пример правильных выражений: aE:dC:cA:56:76:54.

– пример неправильных выражений: 01:23:45:67:89:Az.

Задание 3

Есть текст со списками цен. Извлечь из него цены в USD, RUR, EU.

– пример правильных выражений: 23.78 USD.

– пример неправильных выражений: 22 UDD, 0.002 USD.

Репозиторий

Ссылка: <https://github.com/neluckoff/mirea-java-lessons/tree/master/src/ru/luckoff/mirea/exercies17and18>

Выполнение работы

Решение по всем заданиям

Задания по сути являются одинаковыми и даны нам для знакомства с Pattern и Matcher для того, чтобы научиться писать регулярные выражения.

Все выражения были реализованы лично мной и не были просмотрены где-либо, посмотреть их можно ниже, в коде выполненной работы.

Код выполненной работы

Здесь в нескольких скриншотах можно увидеть как выглядит код выполненного задания и результат его работы.

Полученный код для задания №1

```
public class Main {  
    public static void main(String[] args) {  
        String testStr = "abcdefghijklmnopqrstuv18340";  
        System.out.println("Является строка в testStr строкой 'abcdefghijklmnopqrstuv18340': "  
            + testStr.matches(regex: "abcdefghijklmnopqrstuv18340"));  
  
        Pattern pattern = Pattern.compile("^abcdefghijklmnopqrstuv18340$");  
        Matcher matcher = pattern.matcher(testStr);  
        if(matcher.find()) {  
            System.out.println("2 Является строка в testStr строкой 'abcdefghijklmnopqrstuv18340': да");  
        } else  
            System.out.println("2 Является строка в testStr строкой 'abcdefghijklmnopqrstuv18340': нет");  
    }  
}
```

Рисунок 1 – Первое регулярное выражение

Полученный код для задания №2

```
public class Main {  
    public static void main(String[] args) {  
        String testStr = "aE:dC:cA:56:76:54";  
  
        Pattern pattern = Pattern.compile("^([a-zA-Z1-9]{2}[:]){5}([a-zA-Z1-9]{2})$");  
        Matcher matcher = pattern.matcher(testStr);  
        if (matcher.find()) {  
            System.out.println("Является ли МАК адресом: Да");  
        } else  
            System.out.println("Является ли МАК адресом: Нет");  
    }  
}
```

Рисунок 2 – Второе регулярное выражение

Полученный код для задания №3

```
public class Main {  
    public static void main(String[] args) {  
        List<String> list = Arrays.asList("25.65 USD", "5000 RUR", "34.72 BNB", "17.020 USD");  
        Pattern pattern = Pattern.compile("^([0-9]+.[0-9]{2}) (USD|RUR|EU)$");  
  
        StringBuilder all = new StringBuilder();  
        for (String str : list) {  
            if (pattern.matcher(str).find()) {  
                all.append(str).append(", ");  
            }  
        }  
        System.out.println(all.substring(0, all.length()-2));  
    }  
}
```

Рисунок 3 – Третье регулярное выражение

Вывод

В результате выполнения данной практической работы я научился реализовывать регулярные выражения, а также применять их на практике в языке программирования Java.