

Министерство цифрового развития, связи и массовых коммуникаций Российской Федерации  
Ордена Трудового Красного Знамени  
федеральное государственное бюджетное образовательное учреждение высшего образования  
МОСКОВСКИЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ СВЯЗИ И ИНФОРМАТИКИ

Кафедра «Математической кибернетики и информационных технологий»

Информационные технологии и программирование  
Лабораторная работа №5

Выполнил: студент группы

БВТ2306

Кесслер Алексей Сергеевич

Москва, 2024 г.

Цель работы: Изучение основ регулярных выражений в языке Java.

Задача работы: Применение регулярных выражений для выполнения прикладных задач.

### Выполнение

Задание 1: Поиск всех чисел в тексте

Необходимо написать программу, которая будет искать все числа в заданном тексте и выводить их на экран.

Для использования регулярных выражений в Java используются два класса - Pattern и Matcher. Класс Pattern задает сам шаблон регулярных выражений, а класс Matcher занимается поиском в тексте по этому шаблону.

```
public static void main(String[] args) {  
    String text = "The price of the product is $19. 6. 8. 1.3 -5";  
    Pattern pattern = Pattern.compile( regex: "-?(\\d)+(\\.\\d+)?");  
    Matcher matcher = pattern.matcher(text);  
    while (matcher.find()) {  
        System.out.println(matcher.group());  
    }  
}
```

Рисунок 1 - Первое задание

Задание 2: Проверка корректности ввода пароля

Необходимо написать программу, которая будет проверять корректность ввода пароля.

```
String[] pass = {pass1, pass2, pass3, pass4, pass5};  
Pattern pattern = Pattern.compile( regex: "(?=.*[0-9])(?=.*[a-z])(?=.*[A-Z])[0-9a-zA-Z]{8,16}");  
  
for (int i = 0; i < 5; i++) {  
    Matcher matcher = pattern.matcher(pass[i]);  
    if (matcher.matches()) {  
        System.out.println(pass[i]);  
    }  
}
```

Рисунок 2 - Второе задание

Задание 3: Замена всех ссылок на гиперссылки

Необходимо написать программу, которая будет заменять все ссылки в заданном тексте на гиперссылки.

```

Pattern pattern = Pattern.compile(reg);
Matcher matcher = pattern.matcher(string);

ArrayList<String> xs = new ArrayList<>();
while (matcher.find()) {
    System.out.println("Full match: " + matcher.group());
    xs.add(matcher.group());
}
System.out.println(xs);

String result = string;
for (int i = 0; i < xs.size(); i++){
    Pattern pattern1 = Pattern.compile( regex: "\\b(?

```

Рисунок 3 - Третье задание

Задание 4: Проверка корректности ввода IP-адреса

Необходимо написать программу, которая будет проверять корректность ввода IP-адреса.

```

String[] pass = {pass1, pass2, pass3, pass4, pass5};
Pattern pattern = Pattern.compile( regex: "^(([01]?\\d\\d?|2[0-4]\\d|25[0-5])\\.){3}" +
    "([01]?\\d\\d?|2[0-4]\\d|25[0-5])$" );

for (int i = 0; i < 5; i++) {
    Matcher matcher = pattern.matcher(pass[i]);
    if (matcher.matches()) {
        System.out.println(pass[i]);
    }
}

```

Рисунок 4 - Четвертое задание

Задание 5: Поиск всех слов, начинающихся с заданной буквы

Необходимо написать программу, которая будет искать все слова в заданном тексте, начинающиеся с заданной буквы, и выводить их на экран.

```

String firstLetter = "a";
Pattern pattern = Pattern.compile( regex: "(?<=[\\s\\._,`';:~]" + firstLetter + "[\\w]+(?=[\\s\\._,`';:~])?", Pattern.
Matcher matcher = pattern.matcher(text);
while (matcher.find()) {
    System.out.println(matcher.group());
}

```

Рисунок 5 - Пятое задание

