

Регулярные выражения

Метасимволы для поиска символьных классов

Метасимвол	Назначение
<code>\d</code>	цифровой символ
<code>\D</code>	нецифровой символ
<code>\s</code>	символ пробела
<code>\S</code>	непробельный символ
<code>\w</code>	буквенно-цифровой символ или знак подчёркивания
<code>\W</code>	любой символ, кроме буквенного, цифрового или знака подчёркивания
<code>.</code>	любой символ

Метасимволы для группировки символов

Метасимвол	Назначение
<code>[абв]</code>	любой из перечисленных (а, б, или в)
<code>[^абв]</code>	любой, кроме перечисленных (не а, б, в)
<code>[a-zA-Z]</code>	слияние диапазонов (латинские символы от а до z без учета регистра)
<code>[a-d[m-p]]</code>	объединение символов (от а до d и от m до p)

<code>[a-z&&[def]]</code>	пересечение символов (символы d, e, f)
<code>[a-z&&[^bc]]</code>	вычитание символов (символы a, d-z)

Метасимволы для обозначения количества символов – квантификаторы. Квантификатор всегда следует после символа или группы символов.

Метасимвол	Назначение
<code>?</code>	один или отсутствует
<code>*</code>	ноль или более раз
<code>+</code>	один или более раз
<code>{n}</code>	n раз
<code>{n, }</code>	n раз и более
<code>{n,m}</code>	не менее n раз и не более m раз

Метасимволы для поиска символов редактирования текста

Метасимвол	Назначение
<code>\t</code>	символ табуляции
<code>\n</code>	символ новой строки
<code>\r</code>	символ возврата каретки
<code>\f</code>	переход на новую страницу
<code>\u 0085</code>	символ следующей строки

<code>\u 2028</code>	символ разделения строк
<code>\u 2029</code>	символ разделения абзацев

Метасимволы для поиска совпадений границ строк или текста

Метасимвол	Назначение
<code>^</code>	начало строки
<code>\$</code>	конец строки
<code>\b</code>	граница слова
<code>\B</code>	не граница слова
<code>\A</code>	начало ввода
<code>\G</code>	конец предыдущего совпадения
<code>\Z</code>	конец ввода
<code>\z</code>	конец ввода

Граница слова `\b` - это точка между двумя символами, у которой `\w` с одной стороны от нее и `\W` с другой стороны (в любом порядке).

Задания

1. Напишите по три примера строк, для которых регулярные выражения из приведенной ниже таблицы, возвращают истину и ложь соответственно. Выполните проверку.

```
String text = "AAABBBCCC";
Pattern pattern = Pattern.compile("A|B");
Matcher matcher = pattern.matcher(text);
System.out.println(matcher.find());
```

#	Выражение
1.	<code>aa bb cc AA BB CC</code>
2.	<code>a (a b) b</code>
3.	<code>[a-cA-C] { 2 }</code>
4.	<code>\b\w+\b</code>
5.	<code>[a-d[m-p]] { 3 }</code>

2. Напишите регулярные выражения для форматов из приведенной ниже таблицы. Выполните проверку.

#	Формат
1.	<code>nn-nnnnnn</code> (n – цифра от 1 до 9)
2.	<code>#XXXXXX</code> (X = 0 – 9, или A-F, или a-f)
3.	<code>И.О.Фамилия</code>
4.	<code>nnn.nnn.nnn.nnn</code> (n – цифра от 1 до 9)
5.	<code>D:\\folder\\filename.ext</code>

3) Дана строка, содержащая задание цветов в двух различных форматах.

Например: **"#fff rgb(0.2 0.5 0.4) #12ABCD"**

Заменить все вхождения в формате rgb на цвет в шестнадцатеричном формате.

4) Большие целые числа удобно читать, когда цифры в них разделены на тройки запятыми. Переформатируйте целые числа в строке. Запятую нужно поставить во всякое место, после которого идёт число цифр, кратное 3.

5) Дана строка, состоящая из целых чисел, разделенных пробелами. Записать числа в столбик, выполнив выравнивание по правому краю.