

Regular Expressions

Regular Expressions

Packages: `java.util.regex`

Class:

`Matcher`

`Pattern`

`PatternSyntaxException`

Regular Expressions

```
String pattern = "pattern";
```

```
Pattern regEx = Pattern.compile(pattern); // Create a Pattern object
```

```
// Now create matcher object.
```

```
Matcher m = regEx.matcher("Just a string have a pattern");
```

```
while (matcher.find()) {
```

```
    System.out.println(m.group());
```

```
}
```

Regular Expressions (pattern)

x A character, ex: a, b, c

[abc] a, b, or c (simple class)

[^abc] Any character except a, b, or c (negation)

[a-zA-Z] a through z or A through Z, inclusive (range)

Regular Expressions (pattern)

. Any character

\d A digit: [0-9]

\D A non-digit: [^0-9]

\s A whitespace character: [\t\n\x0B\f\r]

\S A non-whitespace character: [^\s]

\w A word character: [a-zA-Z_0-9]

\W A non-word character: [^\w]

Regular Expressions (pattern)

^ The beginning of a line

\$ The end of a line

Regular Expressions (pattern)

$X?$ X , once or not at all

X^* X , zero or more times

X^+ X , one or more times

$X\{n\}$ X , exactly n times

$X\{n,\}$ X , at least n times

$X\{n,m\}$ X , at least n but not more than m times

Regular Expressions (pattern)

Logical operators

XY X followed by Y

$X|Y$ Either X or Y

(X) X, as a capturing group

Regular Expressions (pattern)

Cách đánh số nhóm (group): từ trái qua phải.

VD: ((A)(B(C)))

1. ((A)(B(C)))

2. (A)

3. (B(C))

4. (C)