



Sử dụng Regex trong Java - Java Regex

[🕒 Lớp Object trong java](#)[Collection trong java 🕒](#)

Java Regex hoặc Regular Expression (biểu thức chính quy) là một API để định nghĩa một mẫu để tìm kiếm hoặc thao tác với chuỗi. Nó được sử dụng rộng rãi để xác định ràng buộc trên các chuỗi như xác thực mật khẩu, email, kiểu dữ liệu datetime, ...

REGULAR EXPRESSION 2 matches (~0ms)

`/\d{1,2}[-|\/]\d{1,2}[-|\/]\d{4}$/` / gm

TEST STRING SWITCH TO UNIT TESTS ▶

hello java 10/02/2018
hello world 2-12-2018

Nội dung chính

[Gói java.util.regex](#)[Lớp Matcher](#)[Lớp Pattern](#)[Ví dụ sử dụng Regex trong Java](#)[1. Ví dụ sử dụng Regex trong Java - tìm kiếm chuỗi con](#)[2. Ví dụ sử dụng Regex trong Java - xác thực email](#)[Test Regex Online](#)[Cú pháp Regex trong Java](#)[1. Các lớp ký tự Regex](#)[2. Số lượng ký tự trong Regex](#)

3. Ký tự đặc biệt trong Regex

4. Ký tự logic trong Regex

Gói java.util.regex

Java Regex API cung cấp 1 interface và 3 lớp trong gói java.util.regex.

Lớp Matcher và Pattern trong java cung cấp cơ sở của biểu thức chính quy. Gói java.util.regex cung cấp các lớp và giao diện sau cho các biểu thức chính quy.

1. Interface MatchResult
2. Lớp Matcher
3. Lớp Pattern
4. Lớp PatternSyntaxException

Lớp Matcher

Nó implements interface MatchResult, cung cấp bộ máy xử lý biểu thức chính quy để thao tác với chuỗi ký tự.

No.	Phương thức	Mô tả
1	boolean matches()	kiểm tra xem biểu thức chính quy có khớp với mẫu hay không.
2	boolean find()	tìm biểu thức tiếp theo khớp với mẫu.
3	boolean find(int start)	tìm biểu thức tiếp theo khớp với mẫu từ số bắt đầu đã cho.
4	String group()	trả về chuỗi con phù hợp.
5	int start()	trả về vị trí bắt đầu của chuỗi con phù hợp.
6	int end()	trả về vị trí kết thúc của chuỗi con phù hợp.
7	int groupCount()	trả về tổng số các chuỗi con phù hợp.

Lớp Pattern

Đây là phiên bản được biên dịch của một biểu thức chính quy. Nó được sử dụng để xác định một mẫu cho bộ máy regex.

No.	Phương thức	Mô tả
1	static Pattern	biên dịch regex đã cho và trả về thể hiện của Pattern.

	<code>compile(String regex)</code>	
2	Matcher <code>matcher(CharSequence input)</code>	tạo một matcher khớp với đầu vào đã cho với mẫu.
3	static boolean <code>matches(String regex, CharSequence input)</code>	Nó biên dịch biểu thức chính quy và tìm kiếm các chuỗi con từ chuỗi input phù hợp với mẫu regex.
4	<code>String[] split(CharSequence input)</code>	chia chuỗi input đã cho thành mảng các kết quả trùng khớp với mẫu đã cho.
5	<code>String pattern()</code>	trả về mẫu regex.

Ví dụ sử dụng Regex trong Java

1. Ví dụ sử dụng Regex trong Java - tìm kiếm chuỗi con

Ví dụ sau tìm tất cả các chuỗi ngày tháng có định dạng dd-mm-yyyy hoặc dd/mm/yyyy trong chuỗi văn bản text1 và xác minh xem chuỗi text2 và text3 có định dạng ngày tháng hay không.

Định nghĩa regex:

```
1 | \d{1,2}[-|/]\d{1,2}[-|/]\d{4}
```

`\d{1,2}`: nghĩa là một số có 1 hoặc 2 chữ số (ngày và tháng).

`[-|/]`: nghĩa là ký tự - hoặc /.

`\d{4}`: nghĩa là một số có 4 chữ số (năm).

File: `RegexExample1.java`

```

1 package vn.viettuts;
2
3 import java.util.regex.Matcher;
4 import java.util.regex.Pattern;
5
6 public class RegexExample1 {
7     public static void main(String[] args) {
8         String text1 = "Hello java regex 2-12-2018, hello world 12/12/2018";
9         Pattern pattern = Pattern.compile("\\d{1,2}[-|/]\\d{1,2}[-|/]\\d{4}");
10        Matcher matcher = pattern.matcher(text1);
11
12        System.out.println("Ngày tháng trong chuỗi text1: " + text1);
13        while (matcher.find()) {
14            System.out.println(text1.substring(matcher.start(), matcher.end()));
15        }
16
17        String text2 = "2/12/2018";
18        String text3 = "12/12/aaaa";
19        pattern = Pattern.compile("^\\d{1,2}[-|/]\\d{1,2}[-|/]\\d{4}$");
20        System.out.println("\nChuỗi " + text2 + " có định dạng ngày tháng: " + pattern.matcher(text2).matches());
21
22        System.out.println("Chuỗi " + text3 + " có định dạng ngày tháng: " + pattern.matcher(text3).matches());
23    }
24 }
25
26

```

Kết quả:

```

Ngày tháng trong chuỗi text1: Hello java regex 2-12-2018, hello world
12/12/2018
2-12-2018
12/12/2018

Chuỗi 2/12/2018 có định dạng ngày tháng: true
Chuỗi 12/12/aaaa có định dạng ngày tháng: false

```

2. Ví dụ sử dụng Regex trong Java - xác thực email

Định nghĩa email:

- Bắt đầu bằng chữ cái.
- Chỉ chứa chữ cái, chữ số và dấu gạch ngang (-).
- Chứa một ký tự @, sau @ là tên miền.
- Tên miền có thể là domain.xxx.yyy hoặc domain.xxx. Trong đó xxx và yyy là các chữ cái và có độ dài từ 2 trở lên.

Định nghĩa regex:

```

1 ^[a-zA-Z][\\w-]+@([\\w]+\\.([\\w]+)|[\\w]+\\.([\\w]{2,})\\.([\\w]{2,}))$

```

```

1  package vn.viettuts;
2
3  import java.util.regex.Pattern;
4
5  public class RegexExample2 {
6      public static void main(String[] args) {
7          String EMAIL_PATTERN =
8              "^[a-zA-Z][\\w-]+@([\\w]+\\. [\\w]+| [\\w]+\\. [\\w]{2,}\\ [\\w]
9
10         String email1 = "test1@gmail.com.vn";
11         String email2 = "123test@gmail.com.vn";
12         String email3 = "test2@gmail.com";
13         String email4 = "test3-1@gmail.com";
14         String email5 = "test4@@gmail.com";
15         String email6 = "test5@domain.com";
16         String email7 = "test6@gmail";
17
18         System.out.println(email1 + ": " + Pattern.matches(EMAIL_PATTERN));
19         System.out.println(email2 + ": " + Pattern.matches(EMAIL_PATTERN));
20         System.out.println(email3 + ": " + Pattern.matches(EMAIL_PATTERN));
21         System.out.println(email4 + ": " + Pattern.matches(EMAIL_PATTERN));
22         System.out.println(email5 + ": " + Pattern.matches(EMAIL_PATTERN));
23         System.out.println(email6 + ": " + Pattern.matches(EMAIL_PATTERN));
24         System.out.println(email7 + ": " + Pattern.matches(EMAIL_PATTERN));
25     }
26 }

```

Kết quả:

```

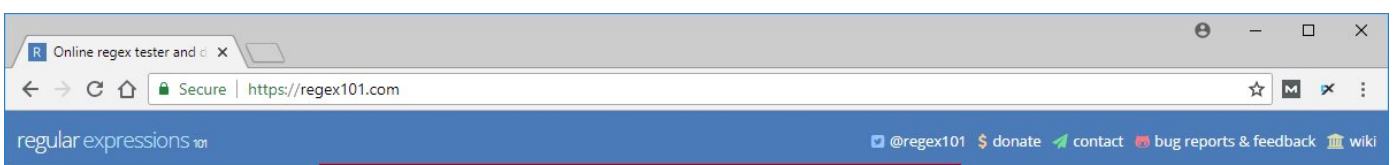
test1@gmail.com.vn: true
123test@gmail.com.vn: false
test2@gmail.com: true
test3-1@gmail.com: true
test4@@gmail.com: false
test5@domain.com: true
test6@gmail: false

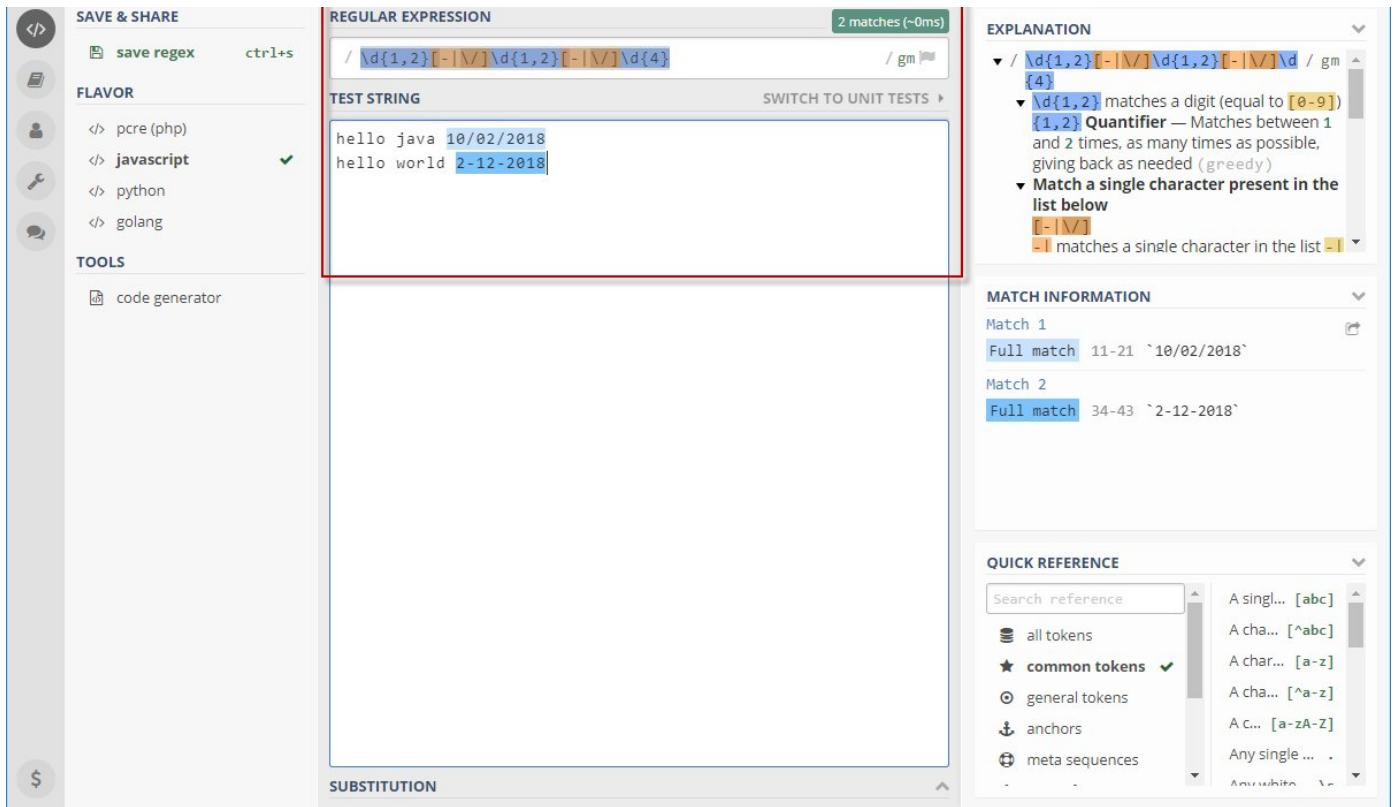
```

Test Regex Online

Bạn có thể test regex online bằng cách sử dụng trang web <https://regex101.com/>

Ví dụ: tìm tất cả các chuỗi ngày tháng có định dạng dd-mm-yyyy hoặc dd/mm/yyyy trong chuỗi văn bản. Ta sử dụng regex = `"\d{1,2}[-/]\d{1,2}[-/]\d{4}"`





Cú pháp của biểu thức chính quy sẽ được giải thích bên dưới.

Cú pháp Regex trong Java

Với các cú pháp và ví dụ sau bạn có thể kiểm tra kết quả bằng cách sử dụng trang web <https://regex101.com/>

1. Các lớp ký tự Regex

Regex	Mô tả
[...]	trả về một ký tự phù hợp
[abc]	a, b, hoặc c
[^abc]	Bất kỳ ký tự nào ngoại trừ a, b, hoặc c
[a-zA-Z]	a tới z hoặc A tới Z
[a-d[m-p]]	a tới d, hoặc m tới p: [a-dm-p]
[a-z&&[def]]	d, e, hoặc f
[a-z&&[^bc]]	a tới z, ngoại trừ b và c: [ad-z]
[a-z&&[^m-p]]	a tới z, ngoại trừ m tới p: [a-lq-z]
[0-9]	0 tới 9

Ví dụ:

```
1 package vn.viettuts;
2
3 import java.util.regex.Pattern;
4
5 public class RegexExample2 {
6     public static void main(String args[]) {
7         System.out.println(Pattern.matches("[a-z&&^bc]", "a"));
8         System.out.println(Pattern.matches("[a-z&&^bc]", "b"));
9         System.out.println(Pattern.matches("[a-z&&^m-p]", "a"));
10        System.out.println(Pattern.matches("[abc]", "c"));
11        System.out.println(Pattern.matches("[abc]", "abc"));
12        System.out.println(Pattern.matches("[0-9]", "8"));
13    }
14 }
```

Kết quả:

```
true
false
true
true
false
true
```

2. Số lượng ký tự trong Regex

Số lượng trong Regex chỉ định số lượng xảy ra của một ký tự.

Regex	Mô tả	Pattern	Ví dụ
X?	X xảy ra một hoặc không lần	hellos?	hello, hellos, helloss
X+	X xảy ra một hoặc nhiều lần	Version \w-\w+	Version A-b1_1
X*	X xảy ra không hoặc nhiều lần	A*B*C*	AAACC
X{n}	X chỉ xảy ra n lần	\d{4}	2018, 20189, 201
X{n,}	X xảy ra n hoặc nhiều lần	\d{4,}	2018, 20189, 201
X{y,z}	X xảy ra ít nhất y lần nhưng nhỏ hơn z lần	\d{2,3}	2018, 201

3. Ký tự đặc biệt trong Regex

Bảng sau đây liệt kê một số ký tự đặc biệt trong regex.

Regex	Mô tả
-------	-------

.	Bất kỳ ký tự nào
^	Có 2 cách sử dụng. 1. Đánh dấu bắt đầu của một dòng, một chuỗi. 2. Nếu sử dụng trong dấu [...] thì nó có nghĩa là phủ định.
\$	Đánh dấu Kết thúc của một dòng
\d	Bất kỳ chữ số nào, viết tắt của [0-9]
\D	Bất kỳ ký tự nào không phải chữ số, viết tắt của [^0-9]
\s	Bất kỳ ký tự trống nào (như dấu cách, tab, xuống dòng, ...), viết tắt của [\t\n\x0B\f\r]
\S	Bất kỳ ký tự trống nào không phải ký tự trống, viết tắt của [^\s]
\w	Bất kỳ ký tự chữ nào (chữ cái và chữ số), viết tắt của [a-zA-Z_0-9]
\W	Bất kỳ ký tự nào không phải chữ cái và chữ số, viết tắt của [^\w]
\b	Ranh giới của một từ
\B	Không phải ranh giới của một từ

4. Ký tự logic trong Regex

Bảng sau liệt kê một số ký tự logic trong Regex:

Regex	Mô tả	Pattern	Ví dụ
	Hoặc	22 33	33
(...)	Group các ký tự và chụp lại	A(nt pple)	Ant, Apple
\1	Nội dung của Group 1	r(\w)g\1x	regex
\2	Nội dung của Group 2	(\d\d)\+(\d\d)=\2\+\1	12+65=65+12
(?: ...)	Group không được chụp lại, bạn không thể sử dụng \1	A(?:nt pple)	Ant, Apple

Recent Updates

- ➔ Sử dụng Javascript trong HTML
- ➔ Toán tử dấu 2 chấm (::) trong Java 8
- ➔ Lambda Expression - Biểu thức Lambda trong java 8
- ➔ Bài tập Java - Sắp xếp nhanh (Quick Sort) trong Java
- ➔ Bài tập Java - Sắp xếp chèn (Insertion Sort) trong Java
- ➔ Bài tập Java - Sắp xếp nổi bọt (Bubble Sort) trong Java
- ➔ Bài tập quản lý sinh viên trong Java
- ➔ HashSet trong java
- ➔ HashMap trong java
- ➔ Lớp WindowAdapter trong Java Swing
- ➔ Lớp MouseMotionAdapter trong Java Swing
- ➔ Lớp MouseAdapter trong Java Swing

VietTuts on facebook

Danh sách bài học

Học java
Học servlet
Học jsp
Học Hibernate
Học Struts2
Học Spring
Học SQL

Câu hỏi phỏng vấn

[201 câu hỏi phỏng vấn java](#)

[25 câu hỏi phỏng vấn servlet](#)

[75 câu hỏi phỏng vấn jsp](#)

[52 câu hỏi phỏng vấn Hibernate](#)

[70 câu hỏi phỏng vấn Spring](#)

[57 câu hỏi phỏng vấn SQL](#)

About VietTuts.Vn

Hệ thống bài học trên VietTuts.Vn bao gồm các bài lý thuyết và thực hành về các công nghệ java và công nghệ web. Các bài lý thuyết trên hệ thống VietTuts.Vn được tham khảo và tổng hợp từ các trang <http://javatpoint.com>, <http://www.tutorialspoint.com>, <http://docs.oracle.com/en> ...