

Regex – Match Any Character(s)

In regex, we can match any character using period "." character. To match only a given set of characters, we should use character classes.

 Lokesh Gupta

 August 27, 2023

 Java Regular Expressions

 Regular Expressions

In this Regular expression tutorial, learn to match a single character appearing once, a single character appearing multiple times, or a specific set of characters.

In regular expressions:

- To match any character, use the dot "." pattern.
- To match a single character (or multiple characters) zero or more times, use ".*" pattern.
- To match multiple characters or a given set of characters, use the character classes.



1. Match Any Character

By default, the '.' dot character in a regular expression matches a single character without regard to what character it is. The matched character can be an **alphabet**, a **number** or, any **special character**.

To create more meaningful patterns, we can combine the dot character with other regular expression constructs.

Pattern	Description
.	Matches only a single character.
A.B	Matches only a single character at second place in a 3 character long string where the string starts with 'A' and ends with 'B'.

Matching a single character

```
Pattern.compile(".").matcher("a").matches();      //true
Pattern.compile(".").matcher("ab").matches();     //false

Pattern.compile("A.B").matcher("AIB").matches();   //true
Pattern.compile("A.B").matcher("ABI").matches();   //false

Pattern.compile("A[abc]B").matcher("AaB").matches(); //true
Pattern.compile("A[abc]B").matcher("AkB").matches(); //false
```



2. Match Any Character from the Specified Range

If we want to match a range of characters at any place, we need to use character classes with a hyphen between the ranges. e.g. '[a-f]' will match a single character which can be either of 'a', 'b', 'c', 'd', 'e' or 'f'.

Pattern	Description
[abc]	Matches only a single character from a set of given characters.
[aA]	Matches only a single character 'a', case-insensitive.
[a-f]	Matches only a single character in the range from 'a' to 'f'.
[a-z]	Matches only a single lowercase character in the range from 'a' to 'z'.
[A-Z]	Matches only a single uppercase character in the range from 'A' to 'Z'.
[a-zA-Z]	Matches only a single character in the range from 'a' to 'z', case-insensitive .

Pattern	Description
[0-9]	Matches only a single number in the range from '0' to '9'.

```
Pattern.compile("[a-f]").matcher("b").matches();      //true
Pattern.compile("[a-f]").matcher("g").matches();      //false
```

```
Pattern.compile("[a-zA-Z]").matcher("a").matches(); //true
Pattern.compile("[a-zA-Z]").matcher("B").matches(); //true
Pattern.compile("[a-zA-Z]").matcher("4").matches(); //false
```

```
Pattern.compile("[0-9]").matcher("9").matches();      //true
Pattern.compile("[0-9]").matcher("91").matches();     //false
```

3. Match Any Character: Zero or More Occurrences

The asterisk (*) is used with any regex pattern for matching zero or more occurrences within strings.

Pattern	Description
.*	Matches any number of characters including special characters.
[0-9]*	Matches any number of digits.
[a-zA-Z]*	Matches any number of alphabets.
[a-zA-Z0-9]*	Matches any number of alphanumeric characters.
https?://.*?(?:\s \$)	Matches URLs from Text
\(.*\)	Matches text between parentheses

```
Pattern.compile(".*").matcher("abcd").matches();          //true
Pattern.compile("[a-zA-Z]*").matcher("abcd").matches();  //true
```

```
Pattern.compile("[0-9]*").matcher("01234").matches();           //true
Pattern.compile("[a-zA-Z0-9]*").matcher("a1b2c3").matches();     //true
```

Happy Learning !!

Weekly Newsletter

Stay Up-to-Date with Our Weekly Updates. Right into Your Inbox.

SUBSCRIBE

Comments

[Subscribe](#) ▾

Join the discussion

B I U S , </> {} [+]



2 COMMENTS



[Most Voted](#) ▾

Josie Hanschmann

⌚ 3 years ago

I want to write a regex that matches any of the exact names (case insensitive) i provide. For example: the answer to a question could be any of the following "tom, peter, nancy, bill, novak". So if the response is "Tom" or "tom" or "tOm" or "PETER" or "Peter", the validation should come out as true. But if the response is "Tomas" or "peterson" or "billboard", then the response is invalid.

 1  Reply**Lokesh Gupta** Reply to [Josie Hanschmann](#)  1 year ago`"^(?i)(tom|peter|nancy|bill|novak)$"` 1  Reply

Java Regex Tutorial

[Regex Introduction](#)[Alphanumeric](#)[Credit Card Numbers](#)[Canadian Postcodes](#)[Currency Symbol](#)[Date Format](#)[Email Address](#)[Password](#)[Greek Characters](#)[ISBNs](#)[Min/Max Length](#)[No. of Lines](#)[No. of Words](#)[SSN](#)[UK Postal Codes](#)[US Postal Codes](#)[Trademark Symbol](#)[Intl Phone Numbers](#)[North American Phone Numbers](#)

Lokesh Gupta

A fun-loving family man, passionate about computers and problem-solving, with over 15 years of experience in Java and related technologies. An avid Sci-Fi movie enthusiast and a fan of Christopher Nolan and Quentin Tarantino.

 Follow on Twitter

 Portfolio

[PREVIOUS](#)

Regex – Match Start or End of String (Line Anchors)

[NEXT](#)

Java – Sorting a String Array in Alphabetical Order

About Us

HowToDoInJava provides tutorials and how-to guides on Java and related technologies.

It also shares the best practices, algorithms & solutions and frequently asked interview questions.

Tutorial Series

[OOP](#)[Regex](#)[Maven](#)[Logging](#)[TypeScript](#)

Python

Meta Links

[About Us](#)

[Advertise](#)

[Contact Us](#)

[Privacy Policy](#)

Our Blogs

[REST API Tutorial](#)

Follow On:

[Copyright © 2025 | Sitemap](#)

[Update Privacy Settings](#)