



HTML

CSS

MORE ▼

Q

MOSCHOOLS.COM

Next >

a sequence of characters that forms a search

ing contains the specified search pattern.

re, which can be used to work with Regular

Import the re module:

import re

RegEx in Python

When you have imported the re module, you can start using regular expressions:

Example

Search the string to see if it starts with "The" and ends with "Spain":

```
txt = "The rain in Spain"
x = re.search("^The.*Spain$", txt)

Run example »
```

RegEx Functions

The re module offers a set of functions that allows us to search a string for a match:

Function	Description
findall	Returns a list containing all matches
<u>search</u>	Returns a Match object if there is a match anywhere in the string
<u>split</u>	Returns a list where the string has been split at each match
<u>sub</u>	Replaces one or many matches with a string

Metacharacters

Metacharacters are characters with a special meaning:

Character	Description	Example	Try it
[]	A set of characters	"[a-m]"	Try it »

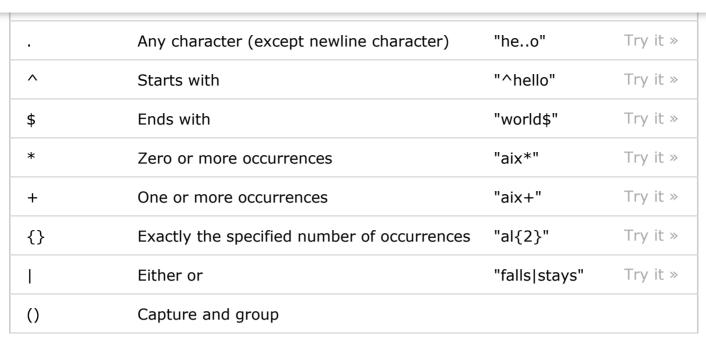




HTML

CSS

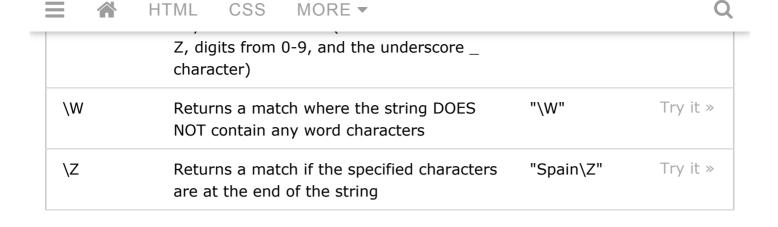




Special Sequences

A special sequence is a \ followed by one of the characters in the list below, and has a special meaning:

Character	Description	Example	Try it
\A	Returns a match if the specified characters are at the beginning of the string	"\AThe"	Try it »
\b	Returns a match where the specified characters are at the beginning or at the end of a word	r"\bain" r"ain\b"	Try it » Try it »
\B	Returns a match where the specified characters are present, but NOT at the beginning (or at the end) of a word	r"\Bain" r"ain\B"	Try it » Try it »
\d	Returns a match where the string contains digits (numbers from 0-9)	"\d"	Try it »
\D	Returns a match where the string DOES NOT contain digits	"\D"	Try it »
\s	Returns a match where the string contains a white space character	"\s"	Try it »
\S	Returns a match where the string DOES NOT contain a white space character	"\S"	Try it »



Sets

A set is a set of characters inside a pair of square brackets [] with a special meaning:

Set	Description	Try it
[arn]	Returns a match where one of the specified characters (${\tt a}$, ${\tt r}$, or ${\tt n}$) are present	Try it »
[a-n]	Returns a match for any lower case character, alphabetically between a and \ensuremath{n}	Try it »
[^arn]	Returns a match for any character EXCEPT $ a$, $ r$, and $ n$	Try it »
[0123]	Returns a match where any of the specified digits (0 , 1 , 2 , or 3) are present	Try it »
[0-9]	Returns a match for any digit between 0 and 9	Try it »
[0-5][0-9]	Returns a match for any two-digit numbers from 00 and 59	Try it »
[a-zA-Z]	Returns a match for any character alphabetically between ${\tt a}$ and ${\tt z}$, lower case OR upper case	Try it »
[+]	In sets, $+$, $*$, ., $ $, (), $$$, ${}$ } has no special meaning, so $[+]$ means: return a match for any $+$ character in the string	Try it »

The findall() Function

The findall() function returns a list containing all matches.





HTML

CSS MORE ▼

Print a list of all matches:

```
import re

str = "The rain in Spain"
x = re.findall("ai", str)
print(x)

Run example »
```

The list contains the matches in the order they are found.

If no matches are found, an empty list is returned:

Example

Return an empty list if no match was found:

```
import re

str = "The rain in Spain"
x = re.findall("Portugal", str)
print(x)

Run example »
```

The search() Function

The search() function searches the string for a match, and returns a <u>Match object</u> if there is a match.

If there is more than one match, only the first occurrence of the match will be returned:

Example

Search for the first white-space character in the string:



HTML CSS MORE ▼

```
str = "The rain in Spain"
x = re.search("\s", str)
print("The first white-space character is located in position:",
x.start())
Run example »
```

If no matches are found, the value None is returned:

Example

Make a search that returns no match:

```
import re
str = "The rain in Spain"
x = re.search("Portugal", str)
print(x)
Run example »
```

The split() Function

The split() function returns a list where the string has been split at each match:

Example

Split at each white-space character:

```
import re
str = "The rain in Spain"
x = re.split("\s", str)
print(x)
```





HTML CSS MORE ▼

You can control the number of occurrences by specifying the maxsplit parameter:

Example

Split the string only at the first occurrence:

```
import re
str = "The rain in Spain"
x = re.split("\s", str, 1)
print(x)
Run example »
```

The sub() Function

The sub () function replaces the matches with the text of your choice:

Example

Replace every white-space character with the number 9:

```
import re
str = "The rain in Spain"
x = re.sub("\s", "9", str)
print(x)
Run example »
```

You can control the number of replacements by specifying the count parameter:

Example

Replace the first 2 occurrences:

```
## HTML CSS MORE ▼

str = "The rain in Spain"
x = re.sub("\s", "9", str, 2)
print(x)

Run example »
```

Match Object

A Match Object is an object containing information about the search and the result.

Note: If there is no match, the value None will be returned, instead of the Match Object.

Example

Do a search that will return a Match Object:

```
import re

str = "The rain in Spain"
x = re.search("ai", str)
print(x) #this will print an object

Run example »
```

The Match object has properties and methods used to retrieve information about the search, and the result:

- .span () returns a tuple containing the start-, and end positions of the match.
- .string returns the string passed into the function
- .group () returns the part of the string where there was a match

Example

Print the position (start- and end-position) of the first match occurrence.

The regular expression looks for any words that starts with an upper case "S":





HTML CSS MORE ▼

C

```
str = "The rain in Spain"
x = re.search(r"\bS\w+", str)
print(x.span())
Run example »
```

Example

Print the string passed into the function:

```
import re

str = "The rain in Spain"

x = re.search(r"\bS\w+", str)
print(x.string)
```

Run example »

Example

Print the part of the string where there was a match.

The regular expression looks for any words that starts with an upper case "S":

```
import re

str = "The rain in Spain"

x = re.search(r"\bS\w+", str)
print(x.group())

Run example »
```

Note: If there is no match, the value None will be returned, instead of the Match Object.

 \equiv



HTML

CSS

MORE ▼

Q





HTML

CSS





COLOR PICKER



HOW TO

Tabs Dropdowns Accordions Side Navigation Top Navigation **Modal Boxes Progress Bars** Parallax Login Form **HTML Includes** Google Maps Range Sliders **Tooltips** Slideshow Filter List Sort List

SHARE









CERTIFICATES

HTML CSS JavaScript SQL Python





HTML

I CS

CSS

MORE ▼

Q

Bootstrap XML

Read More »

REPORT ERROR

PRINT PAGE

FORUM

ABOUT

Top Tutorials

JavaSCHPL TULUHAI

How To Tutorial

SQL Tutorial

Python Tutorial

W3.CSS Tutorial

Bootstrap Tutorial

PHP 5 Tutorial

PHP 7 Tutorial

jQuery Tutorial

Java Tutorial

Top References

HTML Reference

CSS Reference

JavaScript Reference

SQL Reference

Python Reference

W3.CSS Reference

Bootstrap Reference

PHP Reference

HTML Colors

jQuery Reference

Angular Reference

Java Reference

Top Examples

HTML Examples

CSS Examples

JavaScript Examples

How To Examples

SQL Examples

Python Examples

W3.CSS Examples

Bootstrap Examples

PHP Examples

jQuery Examples

Java Examples

XML Examples

Web Certificates

HTML Certificate

CSS Certificate

JavaScript Certificate

SQL Certificate

Python Certificate

iQuery Certificate

PHP Certificate

Bootstrap Certificate

XML Certificate

Get Certified »





HTML

CSS

MORE ▼

Q

warrant run correctness of an content. Wrine using this site, you agree to have read and accepted our terms of use, cookie and privacy policy. Copyright 1999-2019 by Refsnes Data. All Rights Reserved.

Powered by W3.CSS.

