



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

[Try it Yourself »](#)

## RegEx Functions

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

Function	Description
<a href="#">findall</a>	Returns a list containing all matches
<a href="#">search</a>	Returns a <a href="#">Match object</a> if there is a match anywhere in the string
<a href="#">split</a>	Returns a list where the string has been split at each match
<a href="#">sub</a>	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]"	<a href="#">Try it »</a>
\	Signals a special sequence (can also be used to escape special characters)	"\d"	<a href="#">Try it »</a>
.	Any character (except newline character)	"he..o"	<a href="#">Try it »</a>
^	Starts with	"^hello"	<a href="#">Try it »</a>
\$	Ends with	"planet\$"	<a href="#">Try it »</a>

[Dark mode](#)



+	One or more occurrences	"he.+o"	<a href="#">Try it »</a>
?	Zero or one occurrences	"he.?o"	<a href="#">Try it »</a>
{}	Exactly the specified number of occurrences	"he.{2}o"	<a href="#">Try it »</a>
	Either or	"falls stays"	<a href="#">Try it »</a>
()	Capture and group		

## 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
<code>\A</code>	Returns a match if the specified characters are at the beginning of the string	"\AThe"	<a href="#">Try it »</a>
<code>\b</code>	Returns a match where the specified characters are at the beginning or at the end of a word (the "r" in the beginning is making sure that the string is being treated as a "raw string")	r"\bain" r"ain\b"	<a href="#">Try it »</a> <a href="#">Try it »</a>
<code>\B</code>	Returns a match where the specified characters are present, but NOT at the beginning (or at the end) of a word (the "r" in the beginning is making sure that the string is being treated as a "raw string")	r"\Bain" r"ain\B"	<a href="#">Try it »</a> <a href="#">Try it »</a>
<code>\d</code>	Returns a match where the string contains digits (numbers from 0-9)	"\d"	<a href="#">Try it »</a>
<code>\D</code>	Returns a match where the string DOES NOT contain digits	"\D"	<a href="#">Try it »</a>



contains a white space character			
<code>\S</code>	Returns a match where the string DOES NOT contain a white space character	<code>"\S"</code>	<a href="#">Try it »</a>
<code>\w</code>	Returns a match where the string contains any word characters (characters from a to Z, digits from 0-9, and the underscore <code>_</code> character)	<code>"\w"</code>	<a href="#">Try it »</a>
<code>\W</code>	Returns a match where the string DOES NOT contain any word characters	<code>"\W"</code>	<a href="#">Try it »</a>
<code>\Z</code>	Returns a match if the specified characters are at the end of the string	<code>"Spain\Z"</code>	<a href="#">Try it »</a>

## Sets

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

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

Dark mode