# 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 |
| search | Returns a Match object if there is a match anywhere in the string |
| split | Returns a list where the string has been split at each match |
| sub | 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 » |
| \ | Signals a special sequence (can also be used to escape special characters) | "\d" | Try it » |
| . | Any character (except newline character) | "he..o" | Try it » |
| ^ | Starts with | "^hello" | Try it » |
| $ | Ends with | "world$" | Try it » |

| | | | |
|---|---|---|---|
| * | Zero or more occurrences | "aix*" | Try it » |
| + | One or more occurrences | "aix+" | Try it » |
| {} | Exactly the specified number of occurrences | "al{2}" | Try it » |
| \| | Either or | "falls\|stays" | Try it » |
| () | 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 |
|---|---|---|---|
| \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 (the "r" in the beginning is making sure that the string is being treated as a "raw string") | r"\bain"<br>r"ain\b" | Try it »<br>Try it » |
| \B | Returns a match where the specified characters are present, but NOT at the beginning (or at the end) of a word<br>(the "r" in the beginning is making sure that the string is being treated as a "raw string") | r"\Bain"<br>r"ain\B" | Try it »<br>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 » |
| \w | Returns a match where the string contains any word characters (characters from a to Z, digits | "\w" | Try it » |

| | | | |
|---|---|---|---|
| | from 0-9, and the underscore _ character) | | |
| \W | Returns a match where the string DOES NOT contain any word characters | "\W" | Try it » |
| \Z | Returns a match if the specified characters are at the end of the string | "Spain\Z" | 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 ( `a` , `r` , or `n` ) are present | Try it » |
| [a-n] | Returns a match for any lower case character, alphabetically between `a` and `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 `a` and `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 » |