

. [RegEx]*

Regular Expression (Python) Cheat Sheet

<https://learnbyexample.github.io/python-regex-cheatsheet/>

V2020.10.11

(Dr Yan Xu)

metacharacters

- ^
- \$
- \
- prefix a \ character to remove the special meaning and match such characters literally. e.g. \^

Start/End

Anchors	Description
\A	restricts the match to the start of string
\Z	restricts the match to the end of string
^	restricts the match to the start of line
\$	restricts the match to the end of line
\n	newline character is used as line separator
re.MULTILINE or re.M	flag to treat input as multiline string
\b	restricts the match to the start/end of words,
\B	matches wherever \b doesn't match

Quantifiers

*	Match zero or more times
+	Match one or more times
?	Match zero or one times
{m,n}	Match m to n times (inclusive)
{m,}	Match at least m times
{,n}	Match up to n times (including 0 times)
{n}	Match exactly n times
pat1.*pat2	any number of characters between pat1 and pat2

- Quantifiers matching is greedy

Characters

[a-f]	- defines a range, so this matches any of abcdef characters
\d	Match a digit, same as [0-9]
\D	Match non-digit, same as [^0-9] or [^\d]
\w	Match word character, same as [a-zA-Z0-9_]
\W	Match non-word character, same as [^a-zA-Z0-9_] or [^\w]
\s	Match whitespace character, same as [\ \t\n\r\f\v]
\S	Match non-whitespace character, same as [^\ \t\n\r\f\v] or [^\s]

Special Flags

re.IGNORECASE or re.I	flag to ignore case
-----------------------	---------------------

Python Functions (re module)

re.search	Check if given pattern is present anywhere in input string, Output is a re.Match object.
re.findall	returns all the matches as a list. if 1 capture group is used, only its matches are returned. 1+, each element will be tuple of capture groups
re.sub	search and replace
re.compile	Compile a pattern for reuse, outputs re.Pattern object
re.split	split a string based on RE

Examples

- re.search(r'is','This is a string')
- re.search(r'this','This is a string', flags=re.I)
- re.search(r'\Ahi', 'hi hello\ntop spot')
- re.findall(r'\b0*[1-9]\d{2,}\b', '0501035 154 12 26 98234')
 - ['0501', '154', '98234']
- re.findall(r'([^\s]+)/([^\s,]+),?', '2020/04,1986/Mar')
 - [('2020', '04'), ('1986', 'Mar')]
- re.split(r'\d+', 'Sample123string42')
 - ['Sample', 'string']
- re.split(r'(\d+)', 'Sample123string42')
 - ['Sample', '123', 'string', '42']
- re.sub(r'\bpar', r'X', 'par spar apparent spare part')
 - 'X spar apparent spare Xt'
- wd = re.compile(r'\w+')
wd.search('They bought a dog')