PYTHON CHEAT SHEET

STXNEXT

import datetime

%%	Literal%
%a	Sun
%a	Sunday
%b	Jan
%B	January
%d	Day (01)
%H	Hour 24
%I	Hour 12
%j	Day of the year
%m	Month (01)
%M	Minute
%p	AM/PM
%S	Second
%w	Weekday
%x	dd.mm.yyyy
%X	hh.mm.ss
%y	Year (00 to 99)
%Y	Year (2001)
%Y-%m-%d %H:%M:%S %x%X	1999-02-22 10:44:23

examples

```
>>> datetime.strptime('20151231', '%Y%m%d')
datetime.datetime (2015, 12 31, 0, 0)
>>> datetime.now().strftime('%H:%M')
'13:37'
>>> datetime.now().strftime('%d %B %Y')
'29 October 2014'
```

import re

\d	digit
\D	non digit
\b	empty string boundary
\B	empty string non boundary
\w	alphanumeric
\W	no alphanumeric
\s	whitespace
\S	non whitespace
\A	start of string
\Z	end of string
*	0 or more
+	1 or more
?	0 or none
{x,y}	from x to y

usage

re.compile	compile regexp
re.search	search for pattern in string
re.match	checks if string starts with pattern
re.findall	return all matches of pattern
re.split	split string by the occurrences of pattern
re.sub	replace occurrences of pattern
re.group	returns one or more subgroups of the match

flags

re.I	ignore cas
re.M	multiline
re.S	dot matches all
re.X	whitespace ignored, allows comments

examples

```
>>> p = re.compile ( '\d+' )
>>> p.findall ( '12 superheroes, 2 teams ' )
[ '12' , '2' ]
>>> txt = 'super herome@stxnext.pl email'
>>> m = re.search('([\w.]+)@([\w.]+)', txt)
>>> m.group ()
'me@stnext.pl'
>>> m.group(1)
'me'
>>> m.group(2)
'stxnext.pl'
>>> txt = 'S@u!p#e$r(h!e#r*o'
>>> pattern = re.compile ('\W+' )
>>> re.sub (pattern, ' ' , txt)
'Superhero'
>>>  re.sub (pattern, ' _' , txt)
'S_u_p_e_r_h_e_r_o'
```

PYTHON CHEAT SHEET

STX NEXT

f-strings

f'Hello {name}'	f-strings are evaluated at runtime
f'13*71 is {13*71}'	valid Python expressions
f'Result is {my_func(arg)}'	calling functions
f'Hello {name.capitalize()}'	calling methods
f'{math.pi:.2f}'	3.14 - precision
f'{10:b}'	'1010' - binary
f'{200:x}'	'c8' - hexadecimal
f'{0.75:.1%}'	'75.0%' - percentage
f'{"four":10}'	'four ' - formatting width
f'{"four":@<10}'	'four@@@@@@' - fill align
f'{"four":@>10}'	'@@@@@@four'
f'{"four":@^10}'	'@@@four@@@

About STX Next

We have over 15 years of experience, 200 completed projects and 100 satisfied customers which makes us a real "Python Powerhouse". What does it mean in practice? Excellent understanding of the processes behind successful development. In STX Next the statement "the sky's the limit" has a real meaning: development and knowledge sharing works on the basis of values such as self-organization and partnership. This is our way of triggering positive energy and maintaining a good flow between people.

Deep commitment to Agile principles, non-corporate management style, employment on B2B model, flexible working hours, software craftsmanship - those are just some of the elements that emphasize our uniqueness and allow you to be free from the traditional corporate „enslavement".

Check if your are the NEXT one!

import pdb

pdb.set_trace()	enter the debugger at the calling stack frame
breakpoint()	since 3.7 enter the debugger at the calling stack frame without importing pdb module
pdb.pm()	enter post-mortem debugging of the last traceback
where	print stack trace
down	one level down
up	one level up
break *n*	set breakpoint at line number *n*
ignore *n*	ignore n breaks
step	move one step forward (goes into functions)
next	move one step forward (does not go into functions)
return	step out of function (execute until return)
continue	continue
list [n]	list source of current file
args	print arguments of current function
restart	restart program
quit	quit and abort program execution
pp	pretty print variable
until [n]	continue execution until the line with the line number greater than the current one is reached