EXPRESSIONS

Sequence of characters that helps you find if a pattern exists (or not) using a specialized syntax. Used for searching or checking the validity (aka sanitizing) of a field like an email address or phone number. Example: The wildcard * is like an expression which means return all of a search set.

The following are basic patterns that match single character.

\n new line (windows \r\n in Java)

\t tab

\s space

^ start of a string

. period means match any character except \n

^ Caret means match a pattern at start of string

$ Dollar sign means match a pattern at end of string

CHEATSHEET

https://www.debuggex.com/cheatsheet/regex/python

EXPRESSION MODULE

Import re

phoneNumRegex = re.compile(r'\d\d\d-\d\d\d-\d\d\d\d') #create mask format

mo = phoneNumRegex.search('My number is 415-555-4242.')

print('Phone number found: ' + mo.group())

SEARCH FUNCTION

First argument is sting pattern and second argument is string to be searched.

Using \s

import re

re.search(r'Eat\scake', 'Eat cake').group()

Using ^ Carat

re.search(r'^Eat', 'Eat cake').group()

print(re.search(r'Co.k.e', 'Cookie').group())

We use group(num) or groups() of match to get the expression from match function. Group will get 1 out of the result whereas groups gets everything.

MATCH FUNCTION

re.match(pattern, string to be searched, compilation flag)

The compilation flag modifies how your expression works. Flags include I (case-insensitivty), S (match any character), A (ASCII characters only), M (matching on multi-lines)

import re

pattern = r"Cookies"

sequence = "Cookie"

if re.match(pattern, sequence):

  print("Match!")

else: print("Not a match!")

The r in the pattern means raw to prevent python from confusing special characters such as \\. This example r was not necessary.

There is a difference between match versus search function. Match checks for a match at beginning of a string (by default) whereas search checks for a match anywhere in the string.