Python RegEx

A Regular Expression or **RegEx** is a special sequence of characters that uses a search pattern to find a string or set of strings.

It can detect the presence or absence of a text by matching it with a particular pattern and also can split a pattern into one or more sub-patterns.

Regex Module in Python

Python has a built-in module named **"re**" that is used for regular expressions in Python. We can import this module by using the import statement.

A **RegEx,** or Regular Expression, is a sequence of characters that forms a search pattern.

RegEx can be used to check if a string contains the specified search pattern.

RegEx Module

Python has a built-in package called re, which can be used to work with Regular Expressions.

Import the re module:

```
import re
```

RegEx in Python

When you have imported the re module, you can start using regular expressions:

Example

Search the string to see if it starts with "**The**" and ends with "**Spain**":

```
import re

#Check if the string starts with "The" and ends with "Spain":

txt = "The rain in Spain"

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

if x:

  print("YES! We have a match!")
```

```
else:
  print("No match")
```

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

The findall() Function

The **findall()** function returns a list containing all matches.

Example

Print a list of all matches:

```
import re

#Return a list containing every occurrence of "ai":

txt = "The rain in Spain"

x = re.findall("ai", txt)

print(x)
```

The list contains the matches in the order they are found.

If no matches are found, an empty list is returned:

Example

Return an empty list if no match was found:

```
import re

txt = "The rain in Spain"

#Check if "Portugal" is in the string:

x = re.findall("Portugal", txt)

print(x)

if (x):

  print("Yes, there is at least one match!")

else:

  print("No match")
```

The search() Function

The **search()** function searches the string for a match, and returns a Match object if there is a match.

If there is more than one match, only the first occurrence of the match will be returned:

Example

Search for the first white-space character in the string:

```
import re

txt = "The rain in Spain"

x = re.search("\s", txt)

print("The first white-space character is located in position:", x.start())
```

If no matches are found, the value **None** is returned:

```
import re

txt = "The rain in Spain"

x = re.search("Portugal", txt)

print(x)
```

The search() Function

The **search()** function searches the string for a match, and returns a Match object if there is a match.

If there is more than one match, only the first occurrence of the match will be returned:

Example

Search for the first white-space character in the string:

```
import re

txt = "The rain in Spain"

x = re.search("\s", txt)

print("The first white-space character is located in position:", x.start())
```

If no matches are found, the value **None** is returned:

```
import re

txt = "The rain in Spain"

x = re.search("Portugal", txt)

print(x)
```

The split() Function

The **split()** function returns a list where the string has been split at each match:

Example

Split at each white-space character:

```
import re

#Split the string at every white-space character:

txt = "The rain in Spain"

x = re.split("\s", txt)

print(x)
```

You can control the number of occurrences by specifying the **maxsplit** parameter:

Example

Split the string only at the first occurrence:

```
import re

#Split the string at the first white-space character:

txt = "The rain in Spain"

x = re.split("\s", txt, 1)

print(x)
```

The sub() Function

The **sub()** function replaces the matches with the text of your choice:

Example

Replace every white-space character with the number 9:

```
import re

#Replace all white-space characters with the digit "9":

txt = "The rain in Spain"

x = re.sub("\s", "9", txt)

print(x)
```

You can control the number of replacements by specifying the count parameter:

Example

Replace the first 2 occurrences:

```
import re

#Replace the first two occurrences of a white-space character with the digit 9:

txt = "The rain in Spain"

x = re.sub("\s", "9", txt, 2)

print(x)
```

Match Object

A Match Object is an object containing information about the search and the result.

Note: If there is no match, the value None will be returned, instead of the Match Object.

Example:

Do a search that will return a Match Object:

```
import re

#The search() function returns a Match object:

txt = "The rain in Spain"

x = re.search("ai", txt)

print(x)
```

The Match object has properties and methods used to retrieve information about the search, and the result:

.span() returns a tuple containing the start-, and end positions of the match.

.**string** returns the string passed into the function

.group() returns the part of the string where there was a match

Example

Print the position (start- and end-position) of the first match occurrence.

The regular expression looks for any words that starts with an upper case "S":

```
import re

#Search for an upper case "S" character in the beginning of a word, and print its position:

txt = "The rain in Spain"

x = re.search(r"\bS\w+", txt)

print(x.span())
```

Example

Print the string passed into the function:

```
import re

#The string property returns the search string:

txt = "The rain in Spain"

x = re.search(r"\bS\w+", txt)

print(x.string)
```

Example

Print the part of the string where there was a match.

The regular expression looks for any words that starts with an upper case "S":

```
import re

#Search for an upper case "S" character in the beginning of a word, and print the word:

txt = "The rain in Spain"

x = re.search(r"\bS\w+", txt)

print(x.group())
```

Metacharacters

Character	Description	Example
[]	A set of characters	"[a-m]"
\	Signals a special sequence (can also be used to escape special characters)	"\d"
.	Any character (except newline character)	"he..o"
^	Starts with	"^hello"
$	Ends with	"planet$"
*	Zero or more occurrences	"he.*o"
+	One or more occurrences	"he.+o"
?	Zero or one occurrences	"he.?o"
{}	Exactly the specified number of occurrences	"he.{2}o"
|	Either or	"falls|stays"
()	Capture and group	

Metacharacters are characters with a special meaning:

1 [] : SET OF CHARACTERS

```
import re

txt = "The rain in Spain"

#Find all lower case characters alphabetically between "a" and "m":

x = re.findall("[a-m]", txt)

print(x)
```

2 \ :

Signals a special sequence (can also be used to escape special characters)

```
import re

txt = "That will be 59 dollars"

#Find all digit characters:

x = re.findall("\d", txt)

print(x)
```

3 . : Any character (except newline character)

```
import re

txt = "hello planet"

#Search for a sequence that starts with "he", followed by two (any) characters, and an "o":

x = re.findall("he..o", txt)

print(x)
```

4 ^ : Starts with

```
import re

txt = "hello planet"

#Check if the string starts with 'hello':

x = re.findall("^hello", txt)
```

```
if x:
  print("Yes, the string starts with 'hello'")
else:
  print("No match")
```

5 $: Ends with

```
import re
txt = "hello planet"
#Check if the string ends with 'planet':
x = re.findall("planet$", txt)
if x:
  print("Yes, the string ends with 'planet'")
else:
  print("No match")
```

6 * : Zero or more occurrences

```
import re
txt = "hello planet"
#Search for a sequence that starts with "he", followed by 0 or more  (any) characters, and an "o":
x = re.findall("he.*o", txt)
print(x)
```

7 + : One or more occurrences

```
import re
txt = "hello planet"
```

```
#Search for a sequence that starts with "he", followed by 1 or more  (any) characters, and an "o":

x = re.findall("he.+o", txt)

print(x)
```

8 {} : Exactly the specified number of occurrences

```
import re

txt = "hello planet"

#Search for a sequence that starts with "he", followed excactly 2 (any) characters, and an "o":

x = re.findall("he.{2}o", txt)

print(x)
```

9 | : Either or

```
import re

txt = "The rain in Spain falls mainly in the plain!"

#Check if the string contains either "falls" or "stays":

x = re.findall("falls|stays", txt)

print(x)

if x:

  print("Yes, there is at least one match!")

else:

  print("No match")
```

Special Sequences

A special sequence is a \ followed by one of the characters in the list below, and has a special meaning:

1. **\A : Returns a match if the specified characters are at the beginning of the string**

```
import re

txt = "The rain in Spain"

#Check if the string starts with "The":
```

```
x = re.findall("\AThe", txt)

print(x)

if x:
  print("Yes, there is a match!")
else:
  print("No match")
```

2. **\b**: **Returns a match where the specified characters are at the beginning or at the end of a word**

```
import re

txt = "The rain in Spain"

#Check if "ain" is present at the beginning of a WORD:

x = re.findall(r"\bain", txt)

print(x)

if x:
  print("Yes, there is at least one match!")
else:
  print("No match")
```

(the "r" in the beginning is making sure that the string is being treated as a "raw string")

```
import re

txt = "The rain in Spain"

#Check if "ain" is present at the end of a WORD:

x = re.findall(r"ain\b", txt)
```

```
print(x)

if x:

  print("Yes, there is at least one match!")

else:

  print("No match")
```

3.\B:Returns a match where the specified characters are present, but NOT at the beginning (or at the end) of a word

```
import re

txt = "The rain in Spain"

#Check if "ain" is present, but NOT at the beginning of a word:

x = re.findall(r"\Bain", txt)

print(x)

if x:

  print("Yes, there is at least one match!")

else:

  print("No match")
```

(the "r" in the beginning is making sure that the string is being treated as a "raw string")

```
import re

txt = "The rain in Spain"

#Check if "ain" is present, but NOT at the end of a word:

x = re.findall(r"ain\B", txt)

print(x)

if x:

  print("Yes, there is at least one match!")

else:
```

```
print("No match")
```

3. **\d :Returns a match where the string contains digits (numbers from 0-9)**

```
import re
txt = "The rain in Spain"
#Check if the string contains any digits (numbers from 0-9):
x = re.findall("\d", txt)
print(x)
if x:
  print("Yes, there is at least one match!")
else:
  print("No match")
```

4. **\D : Returns a match where the string DOES NOT contain digits**

```
import re

txt = "The rain in Spain"

#Return a match at every no-digit character:

x = re.findall("\D", txt)

print(x)

if x:
  print("Yes, there is at least one match!")
else:
  print("No match")
```