Video 29 : Regular Expressions 3

In this video we will continue towards our path to become Regex Experts! Here is everything we have learned so far :

Regex Tricks We Have Learned

```
Did you find a match
if re.search("REGEX", my_string)

Get list of matches
print("Matches :", len(re.findall("REGEX", my_string)))

Get a pattern object
regex = re.compile("REGEX")

Substitute the match
my_string = regex.sub("substitution", my_string)

[ ]   : Match what is in the brackets
[^ ]  : Match anything not in the brackets
.     : Match any 1 character or space
+     : Match 1 or more of what proceeds
\n    : Newline
\d    : Any 1 number
\D    : Anything but a number
\w    : Same as [a-zA-Z0-9_]
\W    : Same as [^a-zA-Z0-9_]
\s    : Same as [\f\n\r\t\v]
\S    : Same as [^\f\n\r\t\v]
{5}   : Match 5 of what proceeds the curly brackets
{5,7} : Match values that are between 5 and 7 in length
```

Matching Zero or One

```
rand_str = "cat cats"

regex = re.compile("[cat]+s?")

matches = re.findall(regex, rand_str)

# Match cat or cats
print("Matches :", len(matches))

for i in matches:
    print(i)
```

Matching Zero or More

```
rand_str = "doctor doctors doctor's"

# Match doctor doctors or doctor's
regex = re.compile("[doctor]+['s]*")
```

```
matches = re.findall(regex, rand_str)

print("Matches :", len(matches))

# You can do the same by setting an interval match
regex = re.compile("[doctor]+['s]{0,2}")

matches = re.findall(regex, rand_str)

print("Matches :", len(matches))

for i in matches:
    print(i)
```

Python Problem for you to Solve

On Windows newlines are some times \n and other times \r\n

Create a regex that will grab each of the lines in this string, print out the number of matches and each line.

Solution

```
long_str = '''Just some words
and some more\r
and more
'''

print("Matches :", len(re.findall(r"[\w\s]+[\r]?\n", long_str)))

matches = re.findall("[\w\s]+[\r]?\n", long_str)

for i in matches:
    print(i)
```

Greedy & Lazy Matching

```
rand_str = "<name>Life On Mars</name><name>Freaks and Geeks</name>"

# Let's try to grab everything between <name> tags
# Because * is greedy (It grabs the biggest match possible)
# we can't get what we want, which is each individual tag
# match
regex = re.compile(r"<name>.*</name>")

matches = re.findall(regex, rand_str)

print("Matches :", len(matches))

for i in matches:
    print(i)
```

```
# We want to grab the smallest match we use *?, +?, or
# {n,}? instead

regex = re.compile(r"<name>.*?</name>")

matches = re.findall(regex, rand_str)

print("Matches :", len(matches))

for i in matches:
    print(i)
```

Word Boundaries

We use word boundaries to define where our matches start and end

\b matches the start or end of a word

```
# If we want ape it will match ape and the beginning of apex
rand_str = "ape at the apex"

regex = re.compile(r"ape")

# If we use the word boundary
regex = re.compile(r"\bape\b")

matches = re.findall(regex, rand_str)

print("Matches :", len(matches))

for i in matches:
    print(i)
```

Feeling like a Regex Expert? More is coming in the next video!