Video 30 : Regular Expressions 4

We continue on our path towards becoming Regex Experts in this video. We'll cover String Boundaries, Subexpressions, and solve another problem.

String Boundaries

```
# ^ : Matches the beginning of a string if outside of
#     a [ ]
# $ : Matches the end of a string

# Grab everything from the start of the string to @
rand_str = "Match everything up to @"

regex = re.compile(r"^.*[^@]")

matches = re.findall(regex, rand_str)

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

for i in matches:
    print(i)

# Grab everything from @ to the end of the line
rand_str = "@ Get this string"

regex = re.compile(r"[^@\s].*$")

matches = re.findall(regex, rand_str)

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

for i in matches:
    print(i)

# Grab the 1st word of each line using the the multiline
# code which allows for the targeting of each line after
# a line break with ^
rand_str = '''Ape is big
Turtle is slow
Cheetah is fast'''

regex = re.compile(r"(?m)^.*?\s")

matches = re.findall(regex, rand_str)

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

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

Subexpressions

Subexpressions are parts of a larger expression. If you want to match for a large block, but only want to return part of it. To do that surround what you want with ( )

CODE

```
# Get just the number minus the area code
rand_str = "My number is 412-555-1212"

regex = re.compile(r"412-(.*)")

matches = re.findall(regex, rand_str)

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

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

Python Problem for you to Solve

Get just the numbers minus the area codes from this string to solve this problem.

```
rand_str = "412-555-1212 412-555-1213 412-555-1214"
```

Solution

```
rand_str = "412-555-1212 412-555-1213 412-555-1214”
regex = re.compile(r"412-(.{8})")

matches = re.findall(regex, rand_str)

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

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

Multiple Subexpressions

```
# You can have multiple subexpressions as well
# Get both numbers that follow 412 separately
rand_str = "My number is 412-555-1212"

regex = re.compile(r"412-(.*)-(.*)")

matches = re.findall(regex, rand_str)

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

print(matches[0][0])
print(matches[0][1])
```

You are getting very good at using Regular Expressions at this point. Now we take it to the next level. Next time I’ll cover Back References, Back Reference Substitutions, Look Ahead, Look Behind, and Negative Look Ahead & Behind.