

Video 28 : Regular Expressions 2

I continue teaching about Regular Expressions. We'll learn how to match any character, whitespace, numbers, one or more items, and we'll learn how to tell if an email address is legitimate or not.

Matching Any Character

We saw that `.` matches any character, but what if we want to match a period. Backslash the period. You do the same with `[,]` and others.

CODE

```
rand_str = "F.B.I. I.R.S. CIA"
print("Matches :", len(re.findall("\.\.\.", rand_str)))
```

Matching Whitespace

We can match many whitespace characters

CODE

```
rand_str = """This is a long
string that goes
on for many lines"""

print(rand_str)

# Remove newlines
regex = re.compile("\n")

rand_str = regex.sub(" ", rand_str)

print(rand_str)

# You can also match
# \b : Backspace
# \f : Form Feed
# \r : Carriage Return
# \t : Tab
# \v : Vertical Tab

# You may need to remove \r\n on Windows
```

Matching Any Single Number

CODE

```
# \d can be used instead of [0-9]
# \D is the same as [^0-9]

randStr = "12345"
```

```
print("Matches :", len(re.findall("\d", randStr)))
```

Matching Multiple Numbers

You can match multiple digits by following the \d with {numOfValues}

CODE

```
# Match 5 numbers only
if re.search("\d{5}", "12345"):
    print("It is a zip code")

# You can also match within a range
# Match values that are between 5 and 7 digits
num_str = "123 12345 123456 1234567"

print("Matches :", len(re.findall("\d{5,7}", num_str)))
```

Matching Any Single Letter or Number

CODE

```
# \w is the same as [a-zA-Z0-9_]
# \W is the same as [^a-zA-Z0-9_]

ph_num = "412-555-1212"

# Check if it is a phone number
if re.search("\w{3}-\w{3}-\w{4}", ph_num):
    print("It is a phone number")

# Check for valid first name between 2 and 20 characters
if re.search("\w{2,20}", "Ultraman"):
    print("It is a valid name")
```

Matching WhiteSpace

CODE

```
# \s is the same as [\f\n\r\t\v]
# \S is the same as [^\f\n\r\t\v]

# Check for valid first and last name with a space
if re.search("\w{2,20}\s\w{2,20}", "Toshio Muramatsu"):
    print("It is a valid full name")
```

Matching One or More

```
# + matches 1 or more characters

# Match a followed by 1 or more characters
print("Matches :", len(re.findall("a+", "a as ape bug")))
```

Python Problem for you to Solve

Create a Regex that matches email addresses from a list. Translate the following rules into a Regex.

1. 1 to 20 lowercase and uppercase letters, numbers, plus `._%+-`
2. An `@` symbol
3. 2 to 20 lowercase and uppercase letters, numbers, plus `.-`
4. A period
5. 2 to 3 lowercase and uppercase letters

Solution

```
emailList = "db@aol.com m@.com @apple.com db@.com"
```

```
print("Email Matches :", len(re.findall("[w._%+-]{1,20}@[w.-]{2,20}.[A-Za-z]{2,3}",  
                                     emailList)))
```

In the next video I'll continue down the path of turning you into a Regex Expert!