

CONFIDENTIAL & RESTRICTED

Gaurav Gupta

Regular Expressions

- Regular Expression Functions
- Special Characters and Functions
- Creating character sets and repetitions
- Capture and non-Capture groups
- Backreferences

CONFIDENTIAL & RESTRICTED

Regular Expression

- Stronger form of pattern matching
- **re** module available in python or regular expression
- *re.match(pattern, string)* : matches at start of string, None if no match
- *re.search(pattern, string)* : matches first occurrence anywhere in string
- *re.findall(pattern, string)/ finditer()* : returns all matches

CONFIDENTIAL & RESTRICTED

Examples

- Write regex to search word **Python** in a string.
- String starts with Python
- String ends with Python

CONFIDENTIAL & RESTRICTED

Regular Expression Special Characters – 1

- **^** : matches start of string
 - WAR to check whether string starts with word 'Python' or not.
- **\$** : matches end of string
 - WAR to check that string should end with '!' symbol
- **.** (dot) : matches any single character except newline

CONFIDENTIAL & RESTRICTED

Special Sequences

- `\d` : match any digit
- `\D` : match any non-digit
- `\b` : matches a word boundary
- `\s` : match any white space character
- `\S` : match any non-whitespace character
- `\w` : match any alphanumeric including _
- `\W` : match any non-alphanumeric

CONFIDENTIAL & RESTRICTED

Questions

- Find all 3 digit numbers in a string
- Match ipv4 ip Address: ex: 192.168.254.001
- WAR to check valid phone no: Valid Phone no is of the form "xxxx xxx xxx"

CONFIDENTIAL & RESTRICTED

Regular Expression Special Characters – 1

- `*` : matches preceding RE 0 or more times
- `+` : matches preceding RE 1 or more times
- `?` : matches preceding RE 0 or 1 times(non greedy)
- `\` : is used as an escape sequence

CONFIDENTIAL & RESTRICTED

Including and excluding specific characters

- `[]` : specify the characters to be included inside the `[]`
- `[x-y]` : specify ranges. ex: `[a-z]`, `[0-9]`
- `[^]` : specify inverse set ex: `[^a-b]`, `[^,.]`

CONFIDENTIAL & RESTRICTED

Repetitions

- `{n}` : matches exactly n repetitions of preceding re.
- `{m, n}` : match m up to n repetitions.

CONFIDENTIAL & RESTRICTED

Questions

- WAR to match a string that starts and ends with the word 'the', make it case insensitive.
- WAR a regex to validate PAN numbers:
AAAAANNNA
First five are alphabets, next 4 are numbers and last one is alphabet

CONFIDENTIAL & RESTRICTED

Questions

- WAR that returns all the vowels
- Create a pattern, for strings with 4 characters. First should be alphabet, second can be alphanumeric, third should be digit, last should be ,(comma) or ,(dot)
- Pattern for 5 character string with first and last character should not be digit. Second character should not be a vowel and white space. Third should not be lower case character and fourth should not be Uppercase character.

CONFIDENTIAL & RESTRICTED

Questions

- Rewrite the IpV4 regular expression.
- Rewrite the Phone Number Regular Expression.
- Write RE to match a string of length 10, with it should have first five characters as only alphabets, next 4 as alphabets or digits, and the last one is dot ' ' .

CONFIDENTIAL & RESTRICTED

Groups and non-capture groups and alternative

- `()` : used to group as a single repetition unit
- `(?:)` : makes the group as non-capturing group
- `|` : works as or operator

CONFIDENTIAL & RESTRICTED

Questions

- Rewrite the IPv4 regular expression to capture all the subnets.
- Write a RE that parses these kind of Phone numbers and returns the country code:

+91 1111 222 123

+cc dddd ddd ddd

The RE should extract CC section from the Phone Number. The phone no should be a valid no.

CONFIDENTIAL & RESTRICTED

Backreferences

- Backreference means to check for some pattern that had occurred previously in the expression
- Uses `\n` syntax, i.e. `\1` matches the first capture group and so on.
- `([a-z]) \1`

CONFIDENTIAL & RESTRICTED

Questions

- Write RE to match a string of length 5, such that it should be a palindrome, also capture all the matched characters
- Write RE to check for XML syntax that the closing tag is correct or not.