Q1. What is the benefit of regular expressions?

Answer =

Regular expression let us search the piece of text that matches the given pattern. The main benefit of using them is, it saves lot of time and are very eeficient to use.

Q2. Describe the difference between the effects of "(ab)c+" and "a(bc)+." Which of these, if any, is the unqualified pattern "abc+"?

Answer =

The regular expression "(ab)c+" matches strings that start with the substring "ab" and continue with one or more occurrences of the character "c". For example, the string "abccc" matches this regular expression because it starts with "ab" and is followed by three occurrences of "c".

On the other hand, the regular expression "a(bc)+" matches strings that start with the character "a" and are followed by one or more occurrences of the substring "bc". For example, the string "abcbc" matches this regular expression because it starts with "a" and is followed by two occurrences of "bc".

Q3. How much do you need to use the following sentence while using regular expressions?

import re

Answer =

We have to use it everytime we use functions of the module re

Q4. Which characters have special significance in square brackets when expressing a range, and under what circumstances?

Answer =

\ (backslash) : general escape character, is usd when you use character class metacharacters as literals inside a character class only.

^ (circumflex anchor) : negate the class, if this is the first character in the brackets(If ^ is not the first, it is not a metacharacter.)

Q5. How does compiling a regular-expression object benefit you?

Answer =

It is faster compared to using pattern ad other re functions.

Q6. What are some examples of how to use the match object returned by re.match and re.search?

Answer =

re.search() searches for the whole string even if the string contains multi-lines and tries to find a match of the substring in all the lines of string

re.match() searches only from the beginning of the string and return match object if found. But if a match of substring is found somewhere in the middle of the string, it returns none.

import re  
pattern = 'work'  
search = re.search(pattern,"It is the best way of doing good work!", re.IGNORECASE)  
print(search)  
match = re.match(pattern,"The best way for Work at coaching class!", re.IGNORECASE)  
print(match)

<\_sre.SRE\_Match object; span=(33, 37), match='work'>

None

Q7. What is the difference between using a vertical bar (|) as an alteration and using square brackets as a character set?

Answer =

| : is regex or. It will check either of any string is present in search string.

[] : This will search for any character is present in string

import re  
print(re.findall("She|he", "She is a great human!"))  
print(re.findall(r'[^\d]+',"she is great human!"))

['She']

['she is great human!']

Q8. In regular-expression search patterns, why is it necessary to use the raw-string indicator (r)? In   replacement strings?

Answer =

r indicates raw string means that the characters ahead are individual charachers not escape characters. It changes how the string literal is interpreted. Such literals are stored as they appear.