

**1. What is the name of the feature responsible for generating Regex objects?**

Answer : compile function in re module (re.compile()) is responsible for Regex objects.

**2. Why do raw strings often appear in Regex objects?**

Answer: raw strings appear in regex objects as it uses '\ ' character as a normal character and not a special character.

**3. What is the return value of the search() method?**

Answer: Match Objects are return value of the search() method.

**4. From a Match item, how do you get the actual strings that match the pattern?**

Answer: group() method is used to get the actual strings that match the pattern.

**5. In the regex which created from the r'(\d\d\d)-(\d\d\d-\d\d\d\d)', what does group zero cover? Group 2? Group 1?**

Answer : Group 0 : (\d\d\d)-(\d\d\d-\d\d\d\d)

Group 1 : (\d\d\d)

Group 2 : -(\d\d\d-\d\d\d\d)

**6. In standard expression syntax, parentheses and intervals have distinct meanings. How can you tell a regex that you want it to fit real parentheses and periods?**

Answer : Can be done using escape character ( '\ ' )

**7. The findall() method returns a string list or a list of string tuples. What causes it to return one of the two options?**

Answer : If the regex has no groups, a list of strings is returned. If the regex has groups, a list of tuples of strings is returned.

**8. In standard expressions, what does the | character mean?**

Answer : | character stands for “either or”.

re.findall('and' | 'is', 'He is a boy') ; it will search for either and or is in given string.

**9. In regular expressions, what does the character stand for?**

Answer : Characters in regular expression has some specified meaning that is used to get required output. E.g. ^ : starts with , \$ : ends with.

**10. In regular expressions, what is the difference between the + and \* characters?**

Answer : + checks for one or more occurrences and \* checks for zero and more occurrences.

**11. What is the difference between {4} and {4,5} in regular expression?**

Answer : {4} matches for exactly 4 occurrences, while {4,5} will match for between 4 and 5 occurrences.

**12. What do you mean by the \d, \w, and \s shorthand character classes signify in regular expressions?**

Answer : \d : returns output if provided string contains digit.

\w : returns output if provided string contains word character .

\s : returns output if provided string contains space character.

**13. What do means by \D, \W, and \S shorthand character classes signify in regular expressions?**

Answer : \D : returns output if provided string doesn't contains digit.

\W : returns output if provided string doesn't contains word character.

\S : returns output if provided string doesn't contains space character.

**14. What is the difference between .\*? and .\*?**

Answer : \* is a greedy mode, which tends to get the longest string that meets the conditions;  
.\*? is a non-greedy mode, which tends to get the shortest string that can meet the conditions

**15. What is the syntax for matching both numbers and lowercase letters with a character class?**

Answer : Either [0-9a-z] or [a-z0-9]

**16. What is the procedure for making a normal expression in regex case insensitive?**

Answer: Using re.IGNORECASE in flag parameters can be used to do so.

**17. What does the . character normally match? What does it match if re.DOTALL is passed as 2<sup>nd</sup> argument in re.compile()?**

Answer : . character matches any character but not newline character, re.DOTALL also matches the new line character.

**18. If numReg = re.compile(r'\d+'), what will numRegex.sub('X', '11 drummers, 10 pipers, five rings, 4 hen') return?**

Answer : ['11','10','4']

**19. What does passing re.VERBOSE as the 2<sup>nd</sup> argument to re.compile() allow to do?**

Answer : re.VERBOSE allows to add whitespace and comments to the string passed.

**20. How would you write a regex that match a number with comma for every three digits? It must match the given following:**

'42'

'1,234'

'6,368,745'

but not the following:

'12,34,567' (which has only two digits between the commas)

'1234' (which lacks commas)

Answer : re.compile(r'^\d{1,3}(\,\d{3})\*\$')

21. How would you write a regex that matches the full name of someone whose last name is Watanabe? You can assume that the first name that comes before it will always be one word that begins with a capital letter. The regex must match the following:

'Haruto Watanabe'

'Alice Watanabe'

'RoboCop Watanabe'

but not the following:

'haruto Watanabe' (where the first name is not capitalized)

'Mr. Watanabe' (where the preceding word has a nonletter character)

'Watanabe' (which has no first name)

'Haruto watanabe' (where Watanabe is not capitalized)

Answer : `re.compile(r'[A-Z][a-z]*\sWatanabe ')`

22. How would you write a regex that matches a sentence where the first word is either Alice, Bob, or Carol; the second word is either eats, pets, or throws; the third word is apples, cats, or baseballs; and the sentence ends with a period? This regex should be case-insensitive. It must match the following:

'Alice eats apples.'

'Bob pets cats.'

'Carol throws baseballs.'

'Alice throws Apples.'

'BOB EATS CATS.'

but not the following:

'RoboCop eats apples.'

'ALICE THROWS FOOTBALLS.'

'Carol eats 7 cats.'

Answer : `re.compile(r'(Alice|Bob|Carol)\s(eats|pets|throws)\s(apples|cats|baseballs)\.',  
flag=re.IGNORECASE)`