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

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

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

Answer: Raw string (r"text") keeps regular expressions meaningful and confusion-free. And It often appears in Regex objects so that every backslash (‘\’) in a regular expression would have to be prefixed with another one escape it.

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

Answer: search() method will return the match object if there is match anywhere in string

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

Answer: using group() method

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 covers the entire match

Group 2 covers second set of parenthesis

Group1 covers first set of parenthesis

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: It can be escape with a backslash \,open parenthesis\( and close parenthesis \)

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: re.findall() method returns a list of tuples when regex has groups whereas it returns a list of string where regex has no groups.

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

Answer: | is a vertical bar which indicates alteration it gives a choice of alternatives.

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

Answer: The regex set(?:Value) matched set or SetValue, If we don’t want a group to capture its match we can use this regular expression.

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

Answer: In regular expressions + matches one or more whereas \* character matches zero or more.

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

Answer: {4} matches exactly 4 instances of the preceding group and {4,5} matches between 4 and 5 instance.

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

Answer:

\d match a character that is digit

\w match any single word character

\s match any single whitespace character

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

Answer: The \D, \W, and \S shorthand character classes match a single character that is not a digit, word, or space character, respectively.

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

Answer:

<.\*> performs Greedy repetition

<.\*?> perform Non Greedy match

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 regax case insensitive?

Answer: re.IGNORECASE as second argument in re.compile() this procedure allows for making a normal expression in regax case insensitive.

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

Answer: The .character normally matches any character except the newline character. If re.DOTALL is passed as second argument in re.compile() then the .character will also matches the new line characters.

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

Answer: numRegex.sub('X', '11 drummers, 10 pipers, five rings, 4 hen') returns X drummers, X pipers, five rings, X hens'

19. What does passing re.VERBOSE as the 2nd argument to re.compile() allow to do?

Answer: If we pass re.VERBOSE as the 2nd argument to re.compile() it allows us to add whitespace and comments to the string.

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})\*$') will create this regex, but other regex strings can produce a similar regular expression.

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]\*\s Watanabe ')

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)\.', re.IGNORECASE)