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

Ans: re.compile(pattern) => compile feature is responsible for generating Regex objects

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

Ans: raw string notation is used to consider all the characters within the string without escaping any character using backslash ‘\’. It is denoted as r in the pattern. Example – re.compile(r”\w+”)

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

Ans: search() method returns match object.

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

Ans: group() method can be used to get actual strings from match item.

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?

Ans: Group 0 covers both Group 1 & 2 as (group 1, group 2). Group 1 covers (\d\d\d) and Group 2 covers (\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?

Ans: It can be fit by either using raw string notation r before the string (or) by using escape character ‘\’ for each parentheses and intervals.

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

Ans: findall() method returns a string list. Pattern search on multiple occurrences makes it to return string list.

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

Ans: | refers to (either-or) case.

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

Ans: ‘?’ stands for optional pattern search.

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

Ans: ‘+’ refers to ‘Atleast 1 to more’. ‘\*’ refers to ‘zero or more’

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

Ans: The {4} matches exactly four instances of the preceding group. The {4,5} matches between four and five instances.

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

Ans: \d for digits, \w for alphanumeric including underscore, \s for whitespace characters like newline \n, tab \t, form feed \f, vertical tab \v, carriage return \r, escape \e.

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

Ans: \D opposite to \d, \W opposite to \w and \S opposite to \s.

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

Ans: .\* for greedy search and .\*? for non-greedy search.

. for any character except newline \n

\* for 0 or more characters

? for optional character

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

Ans: re.compile([a-z0-9])

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

Ans: re.compile(pattern, re.I) Passing ignore case sensitive regex flag.

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

Ans: . character normally matches any character without except newline \n. But if re.DOTALL regex flag is passed, then it matches any character along with newline \n also.

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

Ans: ‘X drummers, X pipers, X hen’ re.sub() is used for replacing pattern with ‘X’ string.

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

Ans: Allows to add comments as a part of pattern description to improve readability.

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)

Ans: 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)

Ans: 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.'

Ans: re.compile(r’(^[Alice|Bob|Carol]\s[eats|pets|throws]\s[apples|cats|baseballs].)’,re.IGNORECASE)