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

*re.compile()*

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

*This is to avoid escape characters in the string.*

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

*Search method either returns None or re.MatchObject.*

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

*group() method returns the matched text group.*

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?

*group(0) – entire match.*

*group(1) – first character set within the parenthesis*

*group(2) – second character set within the 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?

*If we want to have the parenthesis and periods in the string itself, we have to escape it like \., \(, \).*

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

*When there is no group string list is returned, when there are groups list of string tuples is returned.*

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

*| is or operator.*

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

*? is used to signify zero or one of the preceding group, or be used to signify non greedy matching.*

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

*+ - matches 1 or more*

*\* - matches 0 or more*

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

*{4} – matches exactly 4 instances of preceding group*

*{4,5} – matches between 4 and 5 instances of preceding group.*

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

*\d – single digit*

*\w – single word*

*\s – single space*

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

*\D – not a digit*

*\W – not a word*

*\S – not a space*

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

*I think the question is difference between .\* and .\*?*

*.\* - match the previous elements as many times as possible.*

*.\*? – match the previous elements as fewer times as possible.*

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

*[0-9a-z] or [a-z0-9]*

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

*Mention re.IGNORECASE in re.compile as 2nd parameter.*

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

*. character matches everything except new line character. re.DOTALL in re.compile as second argument will also match newline character.*

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

*'X drummers, X pipers, five rings, X hen'*

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

*re.VERBOSE allows us to add whitespace and comments to the string passed to re.compile()*

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)

*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)

*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.'

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