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

The following method compiles a regular expression pattern into a [regular expression object](https://docs.python.org/3/library/re.html#re-objects), which can be used for matching using its [match()](https://docs.python.org/3/library/re.html#re.Pattern.match), [search()](https://docs.python.org/3/library/re.html#re.Pattern.search) and other methods.

|  |
| --- |
| *re.compile(pattern, flags=0)* |

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

Python raw string treats backslash (\) as a literal character. Hence, we can avoid escaping of backslashes if we use raw strings in Regex objects.

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

It returns a corresponding [match object](https://docs.python.org/3/library/re.html#match-objects) if there is a match. It returns None if no position in the string matches the pattern.

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

The [group()](https://docs.python.org/3/library/re.html#re.Match.group) method of the match object returns 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?**

Group(0) covers the entire match, group(1) covers the first parenthesis, group(2) covers the second 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?**

They can be escaped with backslash.

Example: \., \(, \)

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

The findall() method returns a string list when the regular expression has no groups. The findall() method returns a list of string tuples if the regular expression has groups.

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

| character means either or.

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

? stands for 0 or 1 instance of preceding pattern.

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

+ one or more occurrences

\* zero or more occurrences

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

{4} means exactly 4 occurrences of pattern left to it.

{4,5} means at least 4, but not more than 5 occurrences of pattern left to it.

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

\d => single digit

\w => single word characters (characters from a to Z, digits from 0-9, and the underscore \_ character)

\s single white space

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

\D => single character that is not a digit

\W => single non word character

\S => single non space character

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

.\*? refers non greedy search for a match.

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

[0-9a-z]

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

re.I flag is used to make a normal expression in regular expression case insensitive.

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

* . character normally matches any character (except newline character).
* If re.DOTALL is passed as 2nd argument in re.compile(), then . character matches any character including a newline.

**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 hens'

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

This flag allows you to write regular expressions that look nicer and are more readable by allowing you to visually separate logical sections of the pattern with white space and add comments.

**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.I)