Q1. What is the benefit of regular expressions?

ANS:

A Regular Expression is used for identifying a search pattern in a text string. It also helps in finding out the correctness of the data and even operations such as finding, replacing and formatting the data is possible using Regular Expressions.

Q2. Describe the difference between the effects of "(ab)c+" and "a(bc)+." Which of these, if any, is the unqualified pattern "abc+"?

ANS:

f = A’ + B + C

details

starting with f = A’ + AB + AB’C + AB’CD

the third and fourth terms reduce to

AB’C + AB’CD = AB’C( 1 + D) = AB’C(1) = AB’C

f = A’ + AB + AB’C

we can write the A’ term A’ = A’( 1 + B + B’C)

f = A’( 1 + B + B’C) + AB + AB’C

regroup

f = A’ + (A’B + AB) + (A’B’C + AB’C)

f = A’ + B(A’ + A) + B’C( A’ +A) = A’ + B(1) + B’C(1)

f = A’ + B + B’C

rewrite B as B(1 + C)

f = A’ + B + BC + B’C = A’ + B + C(B + B’) = A’ + B + C(1)

f = A’ + B + C

Q3. How much do you need to use the following sentence while using regular expressions?

ANS:

A Regular Expressions (RegEx) is a special sequence of characters that uses a search pattern to find a string or set of strings. It can detect the presence or absence of a text by matching it with a particular pattern, and also can split a pattern into one or more sub-patterns. Python provides a re module that supports the use of regex in Python. Its primary function is to offer a search, where it takes a regular expression and a string. Here, it either returns the first match or else none.

Example:

* Python3

|  |
| --- |
| import re    s = 'GeeksforGeeks: A computer science portal for geeks'    match = re.search(r'portal', s)    print('Start Index:', match.start())  print('End Index:', match.end()) |

Output

Start Index: 34

End Index: 40

Q4. Which characters have special significance in square brackets when expressing a range, and under what circumstances?

ANS:

The <period>, <left-square-bracket>, and <backslash> shall be special except when used in a bracket expression (see RE Bracket Expression). An expression containing a '[' that is unescaped and is not part of a bracket expression produces undefined results.

Q5. How does compiling a regular-expression object benefit you?

ANS:

We can combine a regular expression pattern into pattern objects, which can be used for pattern matching. It also helps to search a pattern again without rewriting it.

Q6. What are some examples of how to use the match object returned by re.match and re.search?

ANS:

The**re.search()** and **re.match()** both are functions of re module in python. These functions are very efficient and fast for searching in strings. The function searches for some substring in a string and returns a match object if found, else it returns none.

There is a difference between the use of both functions. Both return the first match of a substring found in the string, but **re.match()** searches only from the beginning of the string and return match object if found. But if a match of substring is found somewhere in the middle of the string, it returns none.   
While **re.search()** searches for the whole string even if the string contains multi-lines and tries to find a match of the substring in all the lines of string.

Q7. What is the difference between using a vertical bar (|) as an alteration and using square brackets as a character set?

ANS:

This allows you to apply a quantifier to the entire group or to restrict alternation to part of the regex. Only parentheses can be used for grouping. Square brackets define a character class, and curly braces are used by a quantifier with specific limits.

Q8. In regular-expression search patterns, why is it necessary to use the raw-string indicator (r)? In   replacement strings?

ANS:

Raw strings help you get the "source code" of a RegEx safely to the RegEx parser, which will then assign meaning to character sequences like \d , \w , \n , etc...