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

**Ans**: The re.compile() function is responsible for generating Regex objects.

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

**Ans**: Raw strings are used in Regex objects to prevent the backslashes from being treated as escape characters. This is useful when defining patterns that include backslashes.

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

**Ans**: The search() method returns a Match object if a match is found, or None if no match is found.

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

**Ans**: You can use the group() method on the Match object to get the actual strings that match the pattern.

1. 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**: In created regex,

Group 0 covers the entire match.

Group 1 covers the first set of parentheses (\d\d\d).

Group 2 covers the second set of parentheses (\d\d\d-\d\d\d\d).

1. 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**: You can use a backslash \ to escape parentheses and periods, like \( \) and \..

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

**Ans**: If the regex pattern contains groups, findall() returns a list of string tuples. Otherwise, it returns a list of strings.

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

**Ans**: The | character means "or" in regular expressions, allowing you to match either the pattern on its left or the pattern on its right.

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

**Ans**: In regular expressions, the ^ character asserts the start of a line.

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

**Ans**: The + character matches one or more occurrences of the preceding pattern.

The \* character matches zero or more occurrences of the preceding pattern.

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

**Ans**: {4} specifies exactly four occurrences of the preceding pattern.

{4,5} specifies a range, matching between four and five occurrences of the preceding pattern.

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

**Ans**: \d matches any digit (equivalent to [0-9]).

\w matches any word character (alphanumeric + underscore).

\s matches any whitespace character (spaces, tabs, and newlines).

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

**Ans**: \D matches any non-digit.

\W matches any non-word character.

\S matches any non-whitespace character.

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

**Ans**: .\*? performs a non-greedy match, matching as little as possible.

.\* performs a greedy match, matching as much as possible.

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

**Ans**: [0-9a-z] or [a-z0-9] matches both numbers and lowercase letters.

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

**Ans**: Use the re.IGNORECASE flag.

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

**Ans**: The . character normally matches any character except a newline.

If re.DOTALL is passed, it matches any character including a newline.

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

**Ans**: It will replace all occurrences of one or more digits with 'X', so it will return 'X drummers, X pipers, five rings, X hens'.

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

**Ans**: It allows you to add whitespace and comments to the regex for better readability.

1. 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**: pattern r'^\d{1,3}(,\d{3})\*$'.

1. 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**: '^[A-Z][a-zA-Z]\*\sWatanabe$'

1. 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**: '^(Alice|Bob|Carol)\s(eats|pets|throws)\s(apples|cats|baseballs)\.$'