**Assignment 7**

Q1. What is the name of the feature responsible for generating Regex objects?

Ans1. The feature responsible for generating Regex objects in many programming languages is often referred to as a regular expression (regex) engine or regex compiler. It is a component or module within the programming language that processes and compiles regular expressions into pattern objects that can be used for pattern matching and manipulation of text.

The regex engine analyses the regular expression pattern provided by the user and generates an internal representation of the pattern that can efficiently match and manipulate strings based on the specified pattern rules. This internal representation is often stored in a Regex object or similar data structure, which can then be used to perform various regex operations like searching, matching, substitution, and more.

Q2. Why do raw strings often appear in Regex objects?

Ans2. Raw strings are often used in Regex objects to handle special characters and escape sequences in regular expressions. Regular expressions often contain backslashes **\** that have special meaning both in regular expressions and in string literals.

To simplify the writing and reading of regular expressions, raw strings are commonly used. In a raw string, backslashes are treated as literal characters and not as escape characters. This means that a regular expression pattern can be written directly as it appears without the need to escape backslashes.

Q3. What is the return value of the search() method?

Ans3. The search() method is a function provided by regular expression (regex) libraries in many programming languages. It is used to search for a pattern within a given string and returns a match object if a match is found.

The return value of the search() method depends on whether a match is found or not. Here are the possible return values:

Match object: If a match is found, the search() method returns a match object that contains information about the match. The match object provides various methods and attributes to access details about the match, such as the matched substring, the position of the match, and more.

None: If no match is found, the search() method returns **None**. This indicates that the pattern was not found in the given string.

Q4. From a Match item, how do you get the actual strings that match the pattern?

Ans4. To get the actual strings that match the pattern from a Match object, we can use the group() method. The group() method is used to retrieve the matched substring or substrings based on the pattern.

Here's how we can use the group() method:

Without any arguments: If you call group() without any arguments, it returns the entire matched substring.

**Code:**

import re

pattern = r'\d+'

text = 'I have 42 apples and 7 oranges.'

match = re.search(pattern, text)

if match:

matched\_string = match.group()

print(matched\_string)

**Output:**

42