Assignment 7 Python Basic Snehal Jadhav.

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

re module is used for generation of Regex objects {re.compitle()}

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

to prevent backslash from excaping.

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

if it finds the pattern then it returns match object else none.

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

group() method returns matched 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? Group1?

group 0 covers whole part

group 1 covers first part

group 2 covers second part

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?

\., \(, and \)

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 we enclose our pattern inside "()" it will return tuple

when we enclose our pattern without using "()" it will return tuple

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

in standard ecpression " | " means Alternation / OR operand.

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

it stands for match zero or one of the preceding group

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

+ means one or more

\* means zero or more

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

{4} Matches the expression to its left 4 times, and not less.

{4,5) Matches the expression to its left 4 to 5 times, and not less.

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

\d Matches digits, which means 0-9.

\w Matches alphanumeric characters, which means a-z, A-Z, and 0-9. It also matches the underscore, \_.

\s Matches whitespace characters, which include the \t, \n, \r, and space characters.

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

\D Matches any non-digits.

\W Matches any non-word character.

\S Matches non-whitespace characters.

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

.\*? adds ?, a quantifier that matches the preceding element (the ".\*" here) 0 or 1 times.

.\* adds \*, a quantifier that matches the preceding element (the "." here) 0 or more times.

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

import re

def match(text):

patterns = '^[a-z0-9]\*$'

if re.search(patterns,text):

return "matched"

else:

return"not matched"

print(match("rushikehs123"))

matched

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

use re.IGNORECASE to search, match, or sub

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

dot character matches everything in input except newline character

By passing re. DOTALL as the second argument to re. compile() , you can make the dot character match all characters, including the 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 hens'

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

it will allow to add whitespace and comments to 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})\*$')

re.compile(r'^\d{1,3}(,\d{3})\*$', re.UNICODE)

### 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]\*\sNakamoto')

re.compile(r'[A-Z][a-z]\*\sNakamoto', re.UNICODE)

### 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)

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

re.IGNORECASE|re.UNICODE)