Regular Expression Assignment

In [1]:

```
# importing regex library
import regex as re
```

Question 1-

Write a Python program to check that a string contains only a certain set of characters (in this case a-z, A-Z and 0-9).

In [2]:

```
# defing a function that will check only a certain set of charecters
def only_certain_set_of_char(string):
    pattern= r"[^a-zA-Z0-9]"
    matches=re.search(pattern,string)
    return not bool(matches)
```

In [3]:

```
# Checking whether it works or not
# 1. string contains all the specified char
print("(1)", only_certain_set_of_char("ABCSpqrs1267"))

# 2. String does not have all specified char
print("(2)", only_certain_set_of_char("!@#$%&~#@"))

# 3. String contains all specified char along with that string contains some special
print("(3)", only_certain_set_of_char("AGuLPrt4580fFjy!@#$,%_"))
```

```
(1) True
(2) False
(3) False
```

Question 2-

Create a function in python that matches a string that has an a followed by zero or more b's

In [4]:

```
#defining the function as per question 2
def question_2(string):
    pattern=r"^a(b*)$"
    matches=re.search(pattern,string)
    return bool(matches)

#cheaking conditions
print("Case 1 :",question_2("abb"))
print("Case 2 :",question_2("ac"))
print("Case 3 :",question_2("ab"))
print("Case 4 :",question_2("abbx"))
print("Case 5 :",question_2("a"))
```

```
Case 1 : True
Case 2 : False
Case 3 : True
Case 4 : False
Case 5 : True
```

Question 3-

Create a function in python that matches a string that has an a followed by one or more b's

In [5]:

```
#defining the function as per question 3
def question_3(string):
    pattern=r"^a(b+)$"
    matches=re.search(pattern,string)
    return bool(matches)

#cheaking conditions
print("Case 1 :",question_3("abb"))
print("Case 2 :",question_3("ac"))
print("Case 3 :",question_3("ab"))
print("Case 4 :",question_3("abbx"))
print("Case 5 :",question_3("a"))
```

```
Case 1 : True
Case 2 : False
Case 3 : True
Case 4 : False
Case 5 : False
```

Question 4-

Create a function in Python and use RegEx that matches a string that has an a followed by zero or one 'b'.

In [6]:

```
#defining the function as per question 4
def question_4(string):
    pattern=r"^a(b?)$"
    matches=re.search(pattern,string)
    return bool(matches)

#cheaking conditions
print("Case 1 :",question_4("abb"))
print("Case 2 :",question_4("ac"))
print("Case 3 :",question_4("ab"))
print("Case 4 :",question_4("abbx"))
print("Case 5 :",question_4("a"))
```

```
Case 1 : False
Case 2 : False
Case 3 : True
Case 4 : False
Case 5 : True
```

Question 5-

Write a Python program that matches a string that has an a followed by three 'b'.

In [7]:

```
#defining the function as per question 5
def question_5(string):
    pattern=r"^a(b{3})$"
    matches=re.search(pattern,string)
    return bool(matches)

#cheaking conditions
print("Case 1 :",question_5("abbb"))
print("Case 2 :",question_5("ac"))
print("Case 3 :",question_5("ab"))
print("Case 4 :",question_5("abbbb"))
print("Case 5 :",question_5("a"))
```

```
Case 1 : True
Case 2 : False
Case 3 : False
Case 4 : False
Case 5 : False
```

Question 6-

Write a regular expression in Python to split a string into uppercase letters. Sample text: “ImportanceOfRegularExpressionsInPython” Output: [‘Importance’, ‘Of’, ‘Regular’, ‘Expression’, ‘In’, ‘Python’]

In [8]:

```
text="ImportanceOfRegularExpressionsInPython"
pattern=r'[A-Z][a-z]+'
matches=re.findall(pattern,text)
print(matches)
```

```
['Importance', 'Of', 'Regular', 'Expressions', 'In', 'Python']
```

Question 7-

Write a Python program that matches a string that has an a followed by two to three 'b'.

In [9]:

```
#defining the function as per question 7
def question_7(string):
    pattern=r"^a(b{2,3})$"
    matches=re.search(pattern,string)
    return bool(matches)

#cheaking conditions
print("Case 1 :",question_7("abbb"))
print("Case 2 :",question_7("ac"))
print("Case 3 :",question_7("abb"))
print("Case 4 :",question_7("abbbb"))
print("Case 5 :",question_7("ab"))
```

```
Case 1 : True
Case 2 : False
Case 3 : True
Case 4 : False
Case 5 : False
```

Question 8-

Write a Python program to find sequences of lowercase letters joined with a underscore.

In [10]:

```
#defining the function as per question 8
def lowercase_underscore(string):
    pattern=r"[a-z]+_[a-z]+"
    matches=re.findall(pattern,string)
    return matches

text="We are doing Assignment of regular_expression. this assignment contains severa
print(lowercase_underscore(text))

```

```
['regular_expression', 'several_methods', 'search_method', 'findall_metho
d', 'match_mrthod', 'substitute_method']
```

Question 9-

Write a Python program that matches a string that has an 'a' followed by anything, ending in 'b'.

In [11]:

```
#defining the function as per question 9
def question_9(string):
    pattern="^a.*b$"
    matches=re.search(pattern,string)
    return bool(matches)

#cheaking conditions
print("Case 1 :",question_9("abbb"))
print("Case 2 :",question_9("acdb"))
print("Case 3 :",question_9("abbcfb"))
print("Case 4 :",question_9("abbbbg"))
print("Case 5 :",question_9("abc"))
```

```
Case 1 : True
Case 2 : True
Case 3 : True
Case 4 : False
Case 5 : False
```

Question 10-

Write a Python program that matches a word at the beginning of a string.

In [12]:

```
#defining the function as per question 10
def text_match(string):
        pattern = '^\w+'
        if re.search(pattern,string):
                return ("matched.")
        else:
                return ("Not matched.")

#cheaking conditions
print("case 1 :",text_match("We are hare to learn Regular Expression."))
print("case 2 :",text_match("# symbol is used to tag in social media"))
```

```
case 1 : matched.
case 2 : Not matched.
```

Question 11-

Write a Python program to match a string that contains only upper and lowercase letters, numbers, and underscores.

In [13]:

```
#defining the function as per question 11
def question_11(string):
    pattern=r"^[A-Za-z0-9_]*$"
    if re.search(pattern,string):
        return ("Pattern matched.")
    else:
        return ("Pattern not matched.")

#cheaking conditions
print("case 1 :",question_11("we_are_hare_to_learn_regular_expression_and_work_is_sc
print("Case 2 :",question_11("we are hare to learn regular expression and work is sc
```

```
case 1 : Pattern matched.
Case 2 : Pattern not matched.
```

Question 12-

Write a Python program where a string will start with a specific number.

In [14]:

```
#defining the function as per question 12
def question_12(string):
    pattern="^[0-9]+.*"
    if re.search(pattern,string):
        return ("Pattern Matched")
    else:
        return ("Pattern not Matched")

#cheaking conditions
print("Case 1 :",question_12("150 is the score of west indies in the first innings."
print("Case 2 :",question_12("West indied  has scored 150 in the first innings."))
```

```
Case 1 : Pattern Matched
Case 2 : Pattern not Matched
```

Question 13-

Write a Python program to remove leading zeros from an IP address

In [15]:

```
ip = "255.067.022.028"
string = re.sub('\.0*', '.', ip)
print(string)
```

```
255.67.22.28
```

Question 14-

Write a regular expression in python to match a date string in the form of Month name followed by day number and year stored in a text file. Sample text : ' On August 15th 1947 that India was declared

In [16]:

```
string=" On August 15th 1947 that India was declared independent from British coloni
pattern="[A-Za-z]+\s*\d{2}\w{2}\s*\d{4}"
matches=re.search(pattern,string)
print(matches.group())
```

```
August 15th 1947
```

Question 15-

Write a Python program to search some literals strings in a string. Go to the editor Sample text : 'The quick brown fox jumps over the lazy dog.' Searched words : 'fox', 'dog', 'horse'

In [17]:

```
ptrns = [ 'fox', 'dog', 'horse' ]
text = 'The quick brown fox jumps over the lazy dog.'
for pattern in ptrns:
    print('looking for {} in {}'.format(pattern, text))
    if re.search(pattern,text):
        print('Matched')
    else:
        print('Not Matched')
```

```
looking for fox in The quick brown fox jumps over the lazy dog.
Matched
looking for dog in The quick brown fox jumps over the lazy dog.
Matched
looking for horse in The quick brown fox jumps over the lazy dog.
Not Matched
```

Question 16-

Write a Python program to search a literals string in a string and also find the location within the original string where the pattern occurs Sample text : 'The quick brown fox jumps over the lazy dog.' Searched words : 'fox'

In [18]:

```
pattern='fox'
text = 'The quick brown fox jumps over the lazy dog.'
match=re.search(pattern,text)
print("text literal is '{}'and location of the searched text is {}.".format(match.gr
```

```
text literal is 'fox'and location of the searched text is (16, 19).
```

Question 17-

In [19]:

```
pattern='exercises'
text='Python exercises, PHP exercises, C# exercises'
for match in re.findall(pattern,text):
    print("Match found :",match)
```

```
Match found : exercises
Match found : exercises
Match found : exercises
```

Question 18-

Write a Python program to find the occurrence and position of the substrings within a string.

In [20]:

```
text = 'Python exercises, PHP exercises, C# exercises'
pattern = 'exercises'
matches=re.finditer(pattern, text)
for match in matches:
    print(match.group(), " is at position ",match.span())
```

```
exercises  is at position  (7, 16)
exercises  is at position  (22, 31)
exercises  is at position  (36, 45)
```

Question 19-

Write a Python program to convert a date of yyyy-mm-dd format to dd-mm-yyyy format.

In [21]:

```
string= "India vs West Indies match fixtures are on date 2023-07-12, 2023-07-20, 202
pattern=r"(\d{4})-(\d{2})-(\d{2})"
replace=r"\3-\2-\1"
new_string=re.sub(pattern,replace,string)
print(new_string)
```

```
India vs West Indies match fixtures are on date 12-07-2023, 20-07-2023, 27
-07-2023, 29-07-2023, 01-08-2023, 03-08-2023, 06-08-2023, 08-08-2023, 12-0
8-2023 and 13-08-2023.
```

Question 20-

Write a Python program to find all words starting with 'a' or 'e' in a given string.

In [22]:

```
#defining the function as per question 20
def a_or_e(string):
    pattern="\w*a\w*|\w*e\w*"
    matches=re.findall(pattern,string)
    return matches

#cheaking conditions
print(a_or_e("India vs West Indies match fixtures"))
print(a_or_e("Ramesh got first prise in 100 m running competition."))
```

```
['India', 'West', 'Indies', 'match', 'fixtures']
['Ramesh', 'prise', 'competition']
```

Question 21-

Write a Python program to separate and print the numbers and their position of a given string.

In [23]:

```
string= "Rohit Sharma made his 44th century in the 1st test match of India Tour West
pattern=r"\d+"
matches=re.finditer(pattern,string)
for i in matches:
    print("number = {} and its position = {}".format(i.group(),i.span()))

```

```
number = 44 and its position = (22, 24)
number = 1 and its position = (42, 43)
number = 171 and its position = (134, 137)
number = 387 and its position = (146, 149)
```

Question 22-

Write a regular expression in python program to extract maximum numeric value from a string

In [24]:

```
#defining the function as per question 22
def extract_Maximum_num(string):
     numbers = re.findall('\d+',string)
     num_s = map(int,numbers)
     print(max(num_s))

extract_Maximum_num("score of each indivisual india in 1st innings Yashasvi Jaiswal

```

```
171
```

Question 23-

In [25]:

```
#defining the function as per question 23
def put_space(string):
    words = re.findall(r"[A-Z][a-z\.]*", string)

    print(' '.join(words))


put_space("SachinTendulkarIsKnownAsGodOfIndianCricket.")
```

```
Sachin Tendulkar Is Known As God Of Indian Cricket.
```

Question 24-

Python regex to find sequences of one upper case letter followed by lower case letters

In [26]:

```
#defining the function as per question 24
def upper_lower(string):
    words = re.findall(r"[A-Z][a-z\.]*", string)
    return words
upper_lower("SachinTendulkarIsKnownAsGodOfIndianCricket.")
```

Out[26]:

```
['Sachin', 'Tendulkar', 'Is', 'Known', 'As', 'God', 'Of', 'Indian', 'Crick
et.']
```

Question 25-

Write a Python program to remove duplicate words from Sentence using Regular Expression

In [27]:

```
string='Hello hello hi hi bye bye tata tata'
pattern=r"\b(\w+)(?:\W+\1\b)+"
matches=re.sub(pattern,r"\1", string, flags=re.IGNORECASE)
print(matches)
```

```
Hello hi bye tata
```

Question 26-

Write a python program using RegEx to accept string ending with alphanumeric character.

In [28]:

```
#defining the function as per question 26
def end_alphanumeric(string):
    pattern=r"[A-Za-z0-9]$"
    if re.search(pattern,string):
        print("Pattern Accepted")
    else:
        print("Pattern Denied")

end_alphanumeric("abgd45*")
end_alphanumeric("abgd45")
end_alphanumeric("abd4")
end_alphanumeric("abgd45*sv6")

```

```
Pattern Denied
Pattern Accepted
Pattern Accepted
Pattern Accepted
```

Question 27-

Write a python program using RegEx to extract the hashtags. Sample Text: text = """RT @kapil_kausik: #Doltiwal I mean #xyzabc is "hurt" by #Demonetization as the same has rendered USELESS <U+00A0> <U+00BD><U+00B1><U+0089> "acquired funds" No wo""" Output: ['#Doltiwal', '#xyzabc', '#Demonetization']

In [29]:

```
text="""RT @kapil_kausik: #Doltiwal I mean #xyzabc is "hurt" by #Demonetization as t
pattern="#\w+"
matches=re.findall(pattern,text)
print(matches)
```

```
['#Doltiwal', '#xyzabc', '#Demonetization']
```

Question 28-

Write a python program using RegEx to remove <U+..> like symbols Check the below sample text, there are strange symbols something of the sort <U+..> all over the place. You need to come up with a general Regex expression that will cover all such symbols. Sample Text: "@Jags123456 Bharat band on 28??<U+00A0> <U+00BD><U+00B8><U+0082>Those who are protesting #demonetization are all different party leaders" Output: @Jags123456 Bharat band on 28??Those who are protesting #demonetization are all different party leaders

In [30]:

```
text="@Jags123456 Bharat band on 28??<U+00A0><U+00BD><U+00B8><U+0082>Those who are p
pattern="<U\+\w{4}>"
replaced_text=re.sub(pattern,"",text)
print(replaced_text)
```

```
@Jags123456 Bharat band on 28??Those who are protesting #demonetization ar
e all different party leaders
```

Question 29-

Write a python program to extract dates from the text stored in the text file. Sample Text: Ron was born on 12-09-1992 and he was admitted to school 15-12-1999. Store this sample text in the file and then extract dates.

In [31]:

```
text="Ron was born on 12-09-1992 and he was admitted to school 15-12-1999."
pattern="\d{2}-\d{2}-\d{4}"
re.findall(pattern,text)
```

Out[31]:

```
['12-09-1992', '15-12-1999']
```

Question 30-

Write a Python program to replace all occurrences of a space, comma, or dot with a colon. Sample Text- 'Python Exercises, PHP exercises.' Output: Python:Exercises::PHP:exercises:

In [32]:

```
text="Python Exercises, PHP exercises."
pattern="[,\.\s]"
re.sub(pattern,":",text)
```

Out[32]:

```
'Python:Exercises::PHP:exercises:'
```