Regular Expression

Question 1- Write a Python program to replace all occurrences of a space, comma, or dot with a colon. Sample Text- 'Python Exercises, PHP exercises.' Expected Output: Python:Exercises::PHP:exercises:

In [2]:
```
import re
text = 'Python Exercises, PHP exercises.'
print(re.sub("[ ,.]", ":", text))
```
```
Python:Exercises::PHP:exercises:
```

Question 2- Create a dataframe using the dictionary below and remove everything (commas (,), !, XXXX, ;, etc.) from the columns except words. Dictionary- {'SUMMARY' : ['hello, world!', 'XXXXX test', '123four, five:; six...']} Expected output- 0 hello world 1 test 2 four five six

In [19]:
```
import pandas as pd
data = {'SUMMARY': ['hello, world!', 'XXXXX test', '123four, five:; six...']}
df = pd.DataFrame(data)
df['SUMMARY'] = df['SUMMARY'].str.replace('[^a-zA-Z\s]', '', regex=True)
print(df)
```
```
         SUMMARY
0    hello world
1     XXXXX test
2  four five six
```

Question 3- Create a function in python to find all words that are at least 4 characters long in a string. The use of the re.compile() method is mandatory.

In [4]:
```
import re

def find_words(string):
  pattern = re.compile(r'\b\w{3,5}\b')
  matches = pattern.findall(string)
  return matches
string = "This is a sample string with words of different lengths."
result = find_words(string)
print(result)
```
```
['This', 'with', 'words']
```

Question 4- Create a function in python to find all three, four, and five character words in a string. The use of the re.compile() method is mandatory.

In [5]:
```
import re

def find_words(string):
  pattern = re.compile(r'\b\w{3,5}\b')
  matches = pattern.findall(string)
  return matches
string = "This is a sample string with words of different lengths."
result = find_words(string)
print(result)
```
```
['This', 'with', 'words']
```

Question 5- Create a function in Python to remove the parenthesis in a list of strings. The use of the re.compile() method is mandatory. Sample Text: ["example (.com)", "hr@fliprobo (.com)", "github (.com)", "Hello (Data Science World)", "Data (Scientist)"] Expected Output: example.com hr@fliprobo.com github.com Hello Data Science World Data Scientist

In [22]:
```
import re

def remove_parentheses(strings):
  pattern = re.compile(r"\(|\)")
  modified_strings = []
  for string in strings:
   modified_string = re.sub(pattern, "", string)
  modified_strings.append(modified_string)
  return modified_strings
sample_text = ["example (.com)", "hr@fliprobo (.com)", "github (.com)", "Hello (Data Science World)", "Data (Scientist)"]
result = remove_parentheses(sample_text)
print(result)
```
```
['Data Scientist']
```

Question 6- Write a python program to remove the parenthesis area from the text stored in the text file using Regular Expression. Sample Text: ["example (.com)", "hr@fliprobo (.com)", "github (.com)", "Hello (Data Science World)", "Data (Scientist)"] Expected Output: ["example", "hr@fliprobo", "github", "Hello", "Data"] Note- Store given sample text in the text file and then to remove the parenthesis area from the text.

In [20]:
```
import re

with open("C:\\Users\\Sakshi\\Documents\\RegEx\\parenthesis_.txt") as file:
    for item in file:
        print(re.sub(r" ?\([^)]+\)", "", item), end= " ")
```
```
["example", "hr@fliprobo", "github", "Hello", "Data"]
```

Question 7- Write a regular expression in Python to split a string into uppercase letters. Sample text: “ImportanceOfRegularExpressionsInPython” Expected Output: [‘Importance’, ‘Of’, ‘Regular’, ‘Expression’, ‘In’, ‘Python’]

In [3]:
```
text= "ImportanceOfRegularExpressionsInPython"
upper_case = [s for s in re.split("([A-Z][^A-Z]*)", text) if s]

print(upper_case)
```
```
['Importance', 'Of', 'Regular', 'Expressions', 'In', 'Python']
```

Question 8- Create a function in python to insert spaces between words starting with numbers. Sample Text: “RegularExpression1IsAn2ImportantTopic3InPython" Expected Output: RegularExpression 1IsAn 2ImportantTopic 3InPython

In [5]:
```
def numbers_spaces(str1):
    return re.sub(r"(\w)([0-9])", r"\1 \2", str1)
print(numbers_spaces("RegularExpression1IsAn2ImportantTopic3InPython"))
```
```
RegularExpression 1IsAn 2ImportantTopic 3InPython
```

Question 9- Create a function in python to insert spaces between words starting with capital letters or with numbers. Sample Text: “RegularExpression1IsAn2ImportantTopic3InPython" Expected Output: RegularExpression 1 IsAn 2 ImportantTopic 3 InPython

In [13]:
```
def Insert_space(string):
    words = re.findall(r"[0-9]|[A-Z][a-z]*", string)
    print(' '.join(words))

Insert_space("RegularExpression1IsAn2ImportantTopic3InPython")
```
```
Regular Expression 1 Is An 2 Important Topic 3 In Python
```

Question 10- Use the github link below to read the data and create a dataframe. After creating the dataframe extract the first 6 letters of each country and store in the dataframe under a new column called first_five_letters. Github Link- https://raw.githubusercontent.com/dsrscientist/DSData/master/happiness_score_dataset.csv

In [ ]:

Question 11- Write a Python program to match a string that contains only upper and lowercase letters, numbers, and underscores.

In [34]:
```
def text_match(text):
    patterns = '[a-zA-Z]+[0-9_]+'
    x=re.findall(patterns,  text)
    if x:
        print(x)
        return 'Found a match!\n'
    else:
        return('Not matched!')

print(text_match("The quick brown fox jumps over the lazy dog named xYz_123."))
print(text_match("Python_Exercises_1"))
```
```
['xYz_123']
Found a match!

['Python_', 'Exercises_1']
Found a match!
```

Question 12- Write a Python program where a string will start with a specific number.

In [39]:
```
def match_num(string):
    text = re.compile(r"^6")
    if text.match(string):
        return True
    else:
        return False
print(match_num('6-1234681'))
print(match_num('7-1234681'))
```
```
True
False
```

Question 13- Write a Python program to remove leading zeros from an IP address

In [37]:
```
ip = "616.07.075.258"
string = re.sub('\.[0]*', '.', ip)
print(string)
```
```
616.7.75.258
```

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 independent from British colonialism, and the reins of control were handed over to the leaders of the Country’. Expected Output- August 15th 1947 Note- Store given sample text in the text file and then extract the date string asked format.

In [14]:
```
with open("C:\\Users\\Sakshi\\Documents\\RegEx\\date_String_in_form_of_month.txt") as file:
    for line in file:
        pattern = "([a-zA-Z]+) (\d+[a-z]+) (\d+)"
        matched = re.search(pattern, line)
        print ("Output: %s" % (matched.group()))
```
```
Output: August 15th 1947
```

Question 15- Write a Python program to search some literals strings in a string. Sample text : 'The quick brown fox jumps over the lazy dog.' Searched words : 'fox', 'dog', 'horse'

In [41]:
```
patterns = [ 'fox', 'dog', 'horse' ]
text = 'The quick brown fox jumps over the lazy dog.'
for pattern in patterns:
    print('Searching for "%s" in "%s" ->' % (pattern, text))
    if re.search(pattern,  text):
        print('Matched!\n')
    else:
            print('Not Matched!')
```
```
Searching for "fox" in "The quick brown fox jumps over the lazy dog." ->
Matched!

Searching for "dog" in "The quick brown fox jumps over the lazy dog." ->
Matched!

Searching 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 [42]:
```
pattern = 'fox'
text = 'The quick brown fox jumps over the lazy dog.'
match = re.search(pattern, text)
s = match.start()
e = match.end()
print('Found "%s" in "%s" from %d to %d ' % (match.re.pattern, match.string, s, e))
```
```
Found "fox" in "The quick brown fox jumps over the lazy dog." from 16 to 19
```

Question 17- Write a Python program to find the substrings within a string. Sample text : 'Python exercises, PHP exercises, C# exercises' Pattern : 'exercises'.

In [43]:
```
text = 'Python exercises, PHP exercises, C# exercises'
pattern = 'exercises'
for match in re.findall(pattern, text):
    print('Found "%s"' % match)
```
```
Found "exercises"
Found "exercises"
Found "exercises"
```

Question 18- Write a Python program to find the occurrence and position of the substrings within a string.

In [44]:
```
text = 'Python exercises, PHP exercises, C# exercises'
pattern = 'exercises'
for match in re.finditer(pattern, text):
    s = match.start()
    e = match.end()
    print('Found "%s" at %d:%d' % (text[s:e], s, e))
```
```
Found "exercises" at 7:16
Found "exercises" at 22:31
Found "exercises" at 36:45
```

Question 19- Write a Python program to convert a date of yyyy-mm-dd format to dd-mm-yyyy format.

In [45]:
```
def change_date_format(dt):
    return re.sub(r'(\d{4})-(\d{1,2})-(\d{1,2})', '\\3-\\2-\\1', dt)
dt1 = "2026-01-02"
print("Original date in YYY-MM-DD Format: ",dt1)
print("New date in DD-MM-YYYY Format: ",change_date_format(dt1))
```
```
Original date in YYY-MM-DD Format:  2026-01-02
New date in DD-MM-YYYY Format:  02-01-2026
```

Question 20- Create a function in python to find all decimal numbers with a precision of 1 or 2 in a string. The use of the re.compile() method is mandatory. Sample Text: "01.12 0132.123 2.31875 145.8 3.01 27.25 0.25" Expected Output: ['01.12', '145.8', '3.01', '27.25', '0.25']

In [46]:
```
def decimal_with_precision(string):

    deci_num = re.compile(r"\b([0-9]+\.[0-9]{1,2})\b")
    result = deci_num.findall(text)
    print(result)

text= "01.12 0132.123 2.31875 145.8 3.01 27.25 0.25"
decimal_with_precision(text)
```
```
['01.12', '145.8', '3.01', '27.25', '0.25']
```

Question 21- Write a Python program to separate and print the numbers and their position of a given string.

In [48]:
```
text = "The following example creates an ArrayList with a capacity of 50 elements. 4 elem"

for m in re.finditer("\d+", text):
    print(m.group(0))
    print("Index position:", m.start())
```
```
50
Index position: 62
4
Index position: 75
```

Question 22- Write a regular expression in python program to extract maximum/largest numeric value from a string. Sample Text: 'My marks in each semester are: 947, 896, 926, 524, 734, 950, 642' Expected Output: 950

In [49]:
```
def extractMax(input):

    numbers = re.findall('\d+',input)
    print(numbers)

    numbers = map(int,numbers)
    print ("Maximum Numeric value is ",max(numbers))

input = 'My marks in each semester are: 947, 896, 926, 524, 734, 950, 642'
extractMax(input)
```
```
['947', '896', '926', '524', '734', '950', '642']
Maximum Numeric value is  950
```

Question 23- Create a function in python to insert spaces between words starting with capital letters. Sample Text: “RegularExpressionIsAnImportantTopicInPython" Expected Output: Regular Expression Is An Important Topic In Python

In [50]:
```
def putSpace(input):
    words = re.findall('[A-Z][a-z]*', input)
    print(' '.join(words))

input = 'BruceWayneIsBatman'
putSpace(input)
```
```
Bruce Wayne Is Batman
```

Question 24- Python regex to find sequences of one upper case letter followed by lower case letters

In [51]:
```
def match(text):

    pattern = '[A-Z][a-z]*$'
    if re.search(pattern, text):
        return('Yes')

    else:
        return('No')

print(match("Welcome"))
print(match("WelcomeHomewelcome Hello Welcome"))
print(match("Welcomes you"))
```
```
Yes
Yes
No
```

Question 25- Write a Python program to remove continuous duplicate words from Sentence using Regular Expression. Sample Text: "Hello hello world world" Expected Output: Hello hello world

In [52]:
```
def removeDuplicateWords(input):
    regex = r'\b(\w+)(?:\W+\1\b)+'
    return re.sub(regex, r'\1', input)

str1 = "Good bye bye world world"
print(removeDuplicateWords(str1))

str2 = "Ram went went to to his home"
print(removeDuplicateWords(str2))

str3 = "Hello hello world world"
print(removeDuplicateWords(str3))

str4 = "Hello Hello world World"
print(removeDuplicateWords(str4))
```
```
Good bye world
Ram went to his home
Hello hello world
Hello world World
```

Question 26- Write a python program using RegEx to accept string ending with alphanumeric character.

In [54]:
```
regex = '[a-zA-z0-9]$'

def check_alpha_numeric(string):
    if(re.search(regex, string)):
        print("The string is ending with an alphanumeric character. \n")

    else:
         print("The string is not ending with an alphanumeric character. \n")

check_alpha_numeric("pitchumca@")
check_alpha_numeric("pitchumca123")
check_alpha_numeric("pitchum.")
check_alpha_numeric("staysafeindistancestay")
```
```
The string is not ending with an alphanumeric character.

The string is ending with an alphanumeric character.

The string is not ending with an alphanumeric character.

The string is ending with an alphanumeric character.
```

Question 27-Write a python program using RegEx to extract the hashtags. Sample 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""" Expected Output: ['#Doltiwal', '#xyzabc', '#Demonetization']

In [58]:
```
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 f
hashtags = re.findall(r"#\w+", text)

print("Tweet:\n", text)
print("\n Hashtag:\n", hashtags)
```
```
Tweet:
 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

 Hashtag:
 ['#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" Expected Output: @Jags123456 Bharat band on 28??Those who are protesting #demonetization are all different party leaders

In [59]:
```
text= "@Jags123456 Bharat band on 28??<U+00A0><U+00BD><U+00B8><U+0082>Those who are protesting #demonetization are all different party leaders"
clean_text = re.sub(r"<U\+[A-Z0-9]+>", "", text)

print("Text before:\n", text)
print("\n Text after:\n", clean_text)
```
```
Text before:
 @Jags123456 Bharat band on 28??<U+00A0><U+00BD><U+00B8><U+0082>Those who are protesting #demonetization are all different party leaders

 Text after:
 @Jags123456 Bharat band on 28??Those who are protesting #demonetization are 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. Note- Store this sample text in the file and then extract dates.

In [13]:
```
import re
with open("C:\\Users\\Sakshi\\Documents\\RegEx\\sample_text1.txt") as file:
    for line in file:
        emails = re.findall(r"\d{2}-\d{2}-\d{4}", line)
        print(emails)
```
```
['12-09-1992', '15-12-1999']
```

Question 30- Create a function in python to remove all words from a string of length between 2 and 4. The use of the re.compile() method is mandatory. Sample Text: "The following example creates an ArrayList with a capacity of 50 elements. 4 elements are then added to the ArrayList and the ArrayList is trimmed accordingly." Expected Output: following example creates ArrayList a capacity elements. 4 elements added ArrayList ArrayList trimmed accordingly.

In [60]:
```
def remove_words(str1):
    shortword = re.compile(r'\b\w{2,4}\b')
    print(shortword.sub('', text))

text = "The following example creates an ArrayList with a capacity of 50 elements. 4 elements are then added to the ArrayList and the ArrayList is trimmed acco
remove_words(text)
```
```
 following example creates  ArrayList  a capacity   elements. 4 elements   added   ArrayList   ArrayList  trimmed accordingly.
```

In [ ]: