

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 [1]: import pandas as pd
import regex as re
```

```
In [21]: samp_text = 'Python Exercises, PHP exercises.'

output = re.sub('.$|,|\s',':',samp_text)

print(output)
```

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!', 'XXXXXX test', '123four, five;; six...']}

```
In [86]: Dictionary={'SUMMARY' : ['hello, world!', 'XXXXXX test', '123four, five;; six...']}

df = pd.DataFrame(Dictionary)
df

df['SUMMARY'][0] = re.sub(',|!','',df.iloc[0,0])
df['SUMMARY'][1] = re.sub('^X+','',df.iloc[1,0])
df['SUMMARY'][2] = re.sub('\d+|,|:|;|...$','',df.iloc[2,0])

df
```

Out[86]:

	SUMMARY
0	hello world
1	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 [6]: str1 = 'Chandrayaan-3 is a mission to demonstrate the ability to safely land and explore
str2 ='Developing and demonstrating new technologies for interplanetary missions.

pattern = '\w{4}+'
reg_pat = re.compile(pattern)

def _4char(i):
    output = reg_pat.findall(i)
    print(output)
```

```
_4char(str1)
_4char(str2)

['Chan', 'dray', 'miss', 'demo', 'nstr', 'abil', 'safe', 'land', 'expl', 'luna', 'sur
f']
['Deve', 'lopi', 'demo', 'nstr', 'atin', 'tech', 'nolo', 'gies', 'inte', 'rpla', 'net
a', 'miss', 'ions']
```

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 [10]: str1 = 'On an experimental basis, the Prop Module (PM) of Chandrayaan-3 was moved out
pattern = '\w{3,5}'
re_pat = re.compile(pattern)

def _35char(char):
    output = re_pat.findall(char)
    print(output)

_35char(str1)

['exper', 'iment', 'basis', 'the', 'Prop', 'Modul', 'Chand', 'rayaa', 'was', 'moved',
'out', 'lunar', 'orbit', 'orbit', 'aroun', 'Earth']
```

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 [57]: samp_text = ["example (.com)", "hr@fliprobo (.com)", "github (.com)", "Hello (Data Sci
samp_text

pattern = '[()]'
reg_pat = re.compile(pattern)

def data_clean():
    for i in samp_text:
        res = re.sub(reg_pat, '', i)
        print(res)

data_clean()

example .com
hr@fliprobo .com
github .com
Hello Data Science World
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.

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 [121]: samp_text = 'ImportanceOfRegularExpressionsInPython'

output = re.split('(?<=.)(?=[A-Z])', samp_text)

print(output)

['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: RegularExpression1 IsAn2 ImportantTopic3 InPython

```
In [152]: Sample_Text= 'RegularExpression1IsAn2ImportantTopic3InPython'

def insert(Sample_Text):
    pattern = r'(\d+)([A-Za-z]+)'
    result = re.sub(pattern, r'\1 \2', Sample_Text)
    return result

print(insert(Sample_Text))

RegularExpression1 IsAn2 ImportantTopic3 InPython
```

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 [37]: Sample_Text= 'RegularExpression1IsAn2ImportantTopic3InPython'

def insert1(Sample_Text):
    return re.sub(r'(\w)([A-Z])', r'\1 \2', Sample_Text)

print(insert1(Sample_Text))
```

Regular Expression1 Is An2 Important Topic3 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 [67]: url = 'https://raw.githubusercontent.com/dsrscientist/DSData/master/happiness_score_da
df = pd.read_csv(url)
df
```

Out[67]:

	Country	Region	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom
0	Switzerland	Western Europe	1	7.587	0.03411	1.39651	1.34951	0.94143	0.66557
1	Iceland	Western Europe	2	7.561	0.04884	1.30232	1.40223	0.94784	0.62877
2	Denmark	Western Europe	3	7.527	0.03328	1.32548	1.36058	0.87464	0.64938
3	Norway	Western Europe	4	7.522	0.03880	1.45900	1.33095	0.88521	0.66973
4	Canada	North America	5	7.427	0.03553	1.32629	1.32261	0.90563	0.63297
...
153	Rwanda	Sub-Saharan Africa	154	3.465	0.03464	0.22208	0.77370	0.42864	0.59201
154	Benin	Sub-Saharan Africa	155	3.340	0.03656	0.28665	0.35386	0.31910	0.48450
155	Syria	Middle East and Northern Africa	156	3.006	0.05015	0.66320	0.47489	0.72193	0.15684
156	Burundi	Sub-Saharan Africa	157	2.905	0.08658	0.01530	0.41587	0.22396	0.11850
157	Togo	Sub-Saharan Africa	158	2.839	0.06727	0.20868	0.13995	0.28443	0.36453

158 rows × 12 columns

```
In [80]: df['first_five_letters'] = df['Country'].apply (lambda x:x[0:6])
df
```

Out[80]:

	Country	Region	Happiness Rank	Happiness Score	Standard Error	Economy (GDP per Capita)	Family	Health (Life Expectancy)	Freedom
0	Switzerland	Western Europe	1	7.587	0.03411	1.39651	1.34951	0.94143	0.66557
1	Iceland	Western Europe	2	7.561	0.04884	1.30232	1.40223	0.94784	0.62877
2	Denmark	Western Europe	3	7.527	0.03328	1.32548	1.36058	0.87464	0.64938
3	Norway	Western Europe	4	7.522	0.03880	1.45900	1.33095	0.88521	0.66973
4	Canada	North America	5	7.427	0.03553	1.32629	1.32261	0.90563	0.63297
...
153	Rwanda	Sub-Saharan Africa	154	3.465	0.03464	0.22208	0.77370	0.42864	0.59201
154	Benin	Sub-Saharan Africa	155	3.340	0.03656	0.28665	0.35386	0.31910	0.48450
155	Syria	Middle East and Northern Africa	156	3.006	0.05015	0.66320	0.47489	0.72193	0.15684
156	Burundi	Sub-Saharan Africa	157	2.905	0.08658	0.01530	0.41587	0.22396	0.11850
157	Togo	Sub-Saharan Africa	158	2.839	0.06727	0.20868	0.13995	0.28443	0.36453

158 rows × 13 columns

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

In [149...]

```
str1 = 'Aero_space56'
res = re.match('^[a-zA-Z0-9_]+$', str1)
print(res)
```

```
<regex.Match object; span=(0, 12), match='Aero_space56'>
```

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

In [187...]

```
def matchst(str1):
    pattern = '^9'
    if re.match(pattern,str1):
        print('String start with number')
```

```

else:
    print('None')
matchst('9str')
matchst('abs3')

```

String start with number
None

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

In [191...]

```

def rem_zero(add):
    pattern = '\.[0]*'
    res = re.sub(pattern, '.', add)
    print(res)

rem_zero('255.09.06.01')

```

255.9.6.1

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 [199...]

```

sample_text = "On August 15th 1947 that India was declared independent from British co
file = open('samptext.txt', 'r')
r = file.read()
pattern = r"\b([A-Z][a-z]+) (\d{1,2}(?:st|nd|rd|th)?) (\d{4})\b"

result = re.search(pattern, r)
print(result.group())

```

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 [239...]

```

Sample_text = 'The quick brown fox jumps over the lazy dog.'

def findstr(str1):
    pattern = ['fox', 'dog', 'horse']

    for p in pattern:
        res = re.findall(p, str1)
        print(res)

findstr(Sample_text)

```

```
['fox']
['dog']
[]
```

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 [241...]

```
Sample_text = 'The quick brown fox jumps over the lazy dog.'

def findloc(str1):
    pattern = 'fox'
    res = re.search(pattern, str1)
    print(res)

findloc(Sample_text)

<regex.Match object; span=(16, 19), match='fox'>
```

Question 17- Write a Python program to find the substrings within a string.

Sample text : 'Python exercises, PHP exercises, C# exercises'

Pattern : 'exercises'.

In [243...]

```
Sample_text = 'Python exercises, PHP exercises, C# exercises'

def findsu(str1):
    pattern = 'exercises'

    output = re.findall(pattern, str1)
    print(output)

findsu(Sample_text)

['exercises', 'exercises', 'exercises']
```

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

In [276...]

```
Sample_text = 'Python exercises, PHP exercises, C# exercises'

def findocc(str1):
    pattern = 'exercises'

    for m in re.finditer(pattern, str1):
        a = m.start()
        b = m.end()
        print(Sample_text[a:b], a, b)

findocc(Sample_text)

exercises 7 16
exercises 22 31
exercises 36 45
```

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

In [291...]

```
sample_date = '2023-12-16'
pattern = '(\d{4})-(\d{1,2})-(\d{1,2})'

def dateconv(date):
    output = re.sub(pattern, '\2-\1-\3', date)
    print(output)

dateconv(sample_date)
```

16-12-2023

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 [305...]

```
Sample_Text= "01.12 0132.123 2.31875 145.8 3.01 27.25 0.25"
pattern = re.compile('\d+\.\d{1,2}')

def findec(dec):
    output = re.findall(pattern, dec)
    print(output)

findec(Sample_Text)
```

['01.12', '0132.12', '2.31', '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 [325...]

```
sample_text = "The chandrayan 3 was ambitious mission."
pattern = '\d+'

def findp(str1):
    for i in re.finditer(pattern, str1):
        print(i)

findp(sample_text)
```

<regex.Match object; span=(15, 16), match='3'>

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 [328...]

```
Sample_Text = 'My marks in each semester are: 947, 896, 926, 524, 734, 950, 642'
pattern = '\d+'
```

```

marks = re.findall(pattern, Sample_Text)
print(marks)
max_marks = max(marks)
print(max_marks)

['947', '896', '926', '524', '734', '950', '642']
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 [4]: Sample_Text ='RegularExpressionIsAnImportantTopicInPython'
pattern ='(?<=.)(?=[A-Z])'

def insertspace(str1):
    output = re.sub(pattern, ' ',str1)
    print(output)

insertspace(Sample_Text)

```

Regular Expression Is An Important Topic In Python

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

```

In [50]: sample_text = 'Regular Expression is an Important Topic In Python'
pattern = '[A-Z]+[a-z]*'

output = re.findall(pattern, sample_text)
print(output)

['Regular', 'Expression', 'Important', 'Topic', 'In', 'Python']

```

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 [61]: Sample_Text = "Hello hello world world"
pattern = r'\b(\w+)(?:\w+\1\b)+'

def removedup(dup):
    output = re.sub(pattern, r'\1', dup)
    print(output)

removedup(Sample_Text)

```

Hello hello world

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

```
In [91]: pattern = r'\w+$'

def alphan(str1):
    output = re.search(pattern, str1)
    print(output)

alphan('Hello alpha45_')

<regex.Match object; span=(6, 14), match='alpha45_'>
```

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 [95]: Sample_Text = """RT @kapil_kausik: #Doltiwal I mean #xyzabc is "hurt" by #Demonetizati
pattern = r'#\w+'

def exhash(h1):
    output = re.findall(pattern, h1)
    print(output)

exhash(Sample_Text)

['#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 [118...]: Sample_Text = "@Jags123456 Bharat band on 28??<U+00A0><U+00BD><U+00B8><U+0082>Those wh
pattern = r'<U\+\d+\w+>'

def remsym(str1):
    output = re.sub(pattern, '', str1)
    print(output)

remsym(Sample_Text)
```

@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 [130...]
file = open('date.txt', 'r')
r = file.read()
pattern = r'(\d{2}[-]\d{2}[-]\d{4})'

def extdate():
    output = re.findall(pattern, r)
    print(output)

extdate()
['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 [139...]
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."
pattern = re.compile(r'\b\w{2,4}\b')

def remstr(str1):
    output = re.sub(pattern, '', str1)
    print(output)

remstr(Sample_Text)
```

following example creates ArrayList a capacity elements. 4 elements added ArrayList ArrayList trimmed accordingly.

In []:

In []: