

**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.'

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

Q2 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...']}

```
Ans - import pandas as pd
data = {'SUMMARY': ['hello, world!', 'XXXXXX 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      test
2  four five six
```

Q3. 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.

```
Ans- import re
def find_words(string):
    pattern = re.compile(r'\b\w{3,5}\b')
    matches = pattern.findall(string)
    return matches
import re
def find_words(string):
    pattern = re.compile(r'\b\w{3,5}\b')
    matches = pattern.findall(string)
    return matches
```

Q4. 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.

```
Ans- import re
def find_words(string):
    pattern = re.compile(r'\b\w{3,5}\b')
    matches = pattern.findall(string)
    return matches
import re
```

```
def find_words(string):
    pattern = re.compile(r'\b\w{3,5}\b')
    matches = pattern.findall(string)
    return matches
```

Q5 .Create a function in Python to remove the parenthesis in a list of strings. The use of the re.compile() method is mandatory.

```
Ans 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

```

Data Scientist

Q6 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)"]

Ans - import re

```

# Read the text file and store its content in a variable
with open('filename.txt', 'r') as file:
    text = file.read()

```

```

# Use regular expressions to remove the parenthesis area
modified_text = re.sub(r"\([^\)]*\)", "", text)

```

```

# Save the modified text back to the text file
with open('filename.txt', 'w') as file:
    file.write(modified_text)

```

Q7 Write a regular expression in Python to split a string into uppercase letters.

Ans - import re

```

text = "ImportanceOfRegularExpressionsInPython"
result = re.findall('[A-Z][^A-Z]*', text)
print(result)

```

['Importance', 'Of', 'Regular', 'Expression', 'In', 'Python']

Q8 Create a function in python to insert spaces between words starting with numbers.

Sample Text: "RegularExpression1IsAn2ImportantTopic3InPython"

Expected Output: RegularExpression 1IsAn 2ImportantTopic 3InPython

Ans- import re

```

def insert_spaces(text):
    # Use regular expression to find words starting with numbers
    pattern = r'(\d+)([A-Za-z]+)'
    result = re.sub(pattern, r'\1 \2', text)
    return result

```

Q9. 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

Ans - import re

```
def insert_spaces(text):
```

```
    # Use regular expression to find words starting with capital letters or numbers
```

```
    pattern = r'([A-Z][a-z0-9]+|\d+)
```

```
    # Replace the matched words with a space followed by the word
```

```
    result = re.sub(pattern, r'\1', text)
```

```
    # Remove any leading or trailing spaces
```

```
    result = result.strip()
```

```
    return result
```

**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.

Ans. import re

```
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."
```

```
pattern = r"\b([A-Z][a-z]+) \d{1,2} (?:(st|nd|rd|th)? \d{4})\b"
```

```
matches = re.findall(pattern, text)
```

```
print(matches)
```

**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'

Ans- import re

```
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!')

else:

    print('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'

Ans-

```

import re
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))

```

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

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

**Pattern :** 'exercises'.

```

ANS import re
text = 'Python exercises, PHP exercises, C# exercises'
pattern = 'exercises'
for match in re.findall(pattern, text):
    print('Found "%s"' % match)

```

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

```

Ans- import re
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))

```

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

```

Ans- import re
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 YYYY-MM-DD Format: ",dt1)
print("New date in DD-MM-YYYY Format: ",change_date_format(dt1))

```

**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']

Ans-

```
import re
```

```
def find_decimal_numbers(string):
```

```
    pattern = re.compile(r'\d+\.\d{1,2}')
```

```
    decimal_numbers = re.findall(pattern, string)
```

```
    return decimal_numbers
```

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

Ans- import re

```
# Input.
```

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

```
for m in re.finditer("\d+", text):
```

```
    print(m.group(0))
```

```
    print("Index position:", m.start())
```

**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

Ans- import re

```
input_string = 'My marks in each semester are: 947, 896, 926, 524, 734, 950, 642'
```

```
numeric_values = re.findall(r'\d+', input_string)
```

```
numeric_values = [int(value) for value in numeric_values]
```

```
max_value = max(numeric_values)
```

```
print(max_value)
```

**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

Ans- import re

```
def insert_spaces(text):
    # Use regular expression to find words starting with capital letters
    pattern = r'([A-Z][a-z]+)'
    # Replace the found words with the same word followed by a space
    result = re.sub(pattern, r'\1 ', text)
    # Remove any leading or trailing spaces
    result = result.strip()
    return result
```

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

Ans- import re

```
pattern = r'[A-Z][a-z]+'
text = "This is a Sample Text with Multiple Matches"
```

```
matches = re.findall(pattern, text)
print(matches)
```

**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

```
Ans- import re
def remove_duplicates(sentence):
    pattern = r'\b(\w+)(\s+\1\b)+'
    result = re.sub(pattern, r'\1', sentence)
    return result
```

```
# Example usage
sentence = "Hello hello world world"
output = remove_duplicates(sentence)
print(output)
```

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

Ans- import re

```

def check_string(string):

    pattern = r"\w$"

    match = re.search(pattern, string)

    if match:

        return True

    else:

        return False


# Example usage

input_string = input("Enter a string: ")

if check_string(input_string):

    print("String ends with an alphanumeric character")

else:

    print("String does not end 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`

`<ed><U+00A0><U+00BD><ed><U+00B1><U+0089> "acquired funds" No wo"""`

**Expected Output:** `['#Doltiwal', '#xyzabc', '#Demonetization']`

Ans- import re

```

def extract_hashtags(text):

    hashtags = re.findall(r'#\w+', text)

    return hashtags

```

# Sample text

```

text = 'RT @kapil_kausik: #Doltiwal I mean #xyzabc is "hurt" by #Demonetization as the
same has rendered USELESS <ed><U+00A0><U+00BD><ed><U+00B1><U+0089>
"acquired funds" No wo'

```

```
# Extract hashtags

hashtags = extract_hashtags(text)

# Print the extracted hashtags

print(hashtags)
```

**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??<ed><U+00A0><U+00BD><ed><U+00B8><U+0082>Those who are protesting #demonetization are all different party leaders"  
**Expected Output:** @Jags123456 Bharat band on 28??<ed><ed>Those who are protesting #demonetization are all different party leaders

Ans- import re

```
input_text = "@Jags123456 Bharat band on
28??<ed><U+00A0><U+00BD><ed><U+00B8><U+0082>Those who are protesting
#demonetization are all different party leaders"

pattern = r"<U\+\w{4}>"

output_text = re.sub(pattern, "", input_text)

print(output_text)

@Jags123456 Bharat band on 28??<ed><ed>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.

Ans- import re

```
# Open the text file

with open('filename.txt', 'r') as file:

    text = file.read()

# Define the regular expression pattern for dates

pattern = r"\d{2}-\d{2}-\d{4}'
```



```
# Find all matches of the pattern in the text
dates = re.findall(pattern, text)

# Print the extracted dates
for date in dates:
    print(date)
```

**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.

Ans- import re

```
def remove_words(string):
```

```
    pattern = re.compile(r'\b\w{2,4}\b')
```

```
    modified_string = re.sub(pattern, "", string)
```

```
    return modified_string
```

ample\_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."

```
result = remove_words(sample_text)
```

```
print(result == expected_output) # True
```

