Student Name: ___________________________ **Roll No:** ________________ **Section:** ______________

Lab No. 12

Lab 12 – Introduction to Regular Expression

Objectives:

- Introduction to Regular Expression
- Metacharacters
- Functions in regular expressions

1. Introduction to Regular Expression

A RegEx, or Regular Expression, is a sequence of characters that forms a search pattern. RegEx can be used to check if a string contains the specified search pattern. We use **RegEx Module** Python has a built-in package called **re**, which can be used to work with Regular Expressions. So you need to import library **re** before you can use regular expressions in Python.

Regular expressions use two types of characters:

a) Meta characters: As the name suggests, these characters have a special meaning, similar to * in wild card.
b) Literals (like a,b,1,2…)

The most common uses of regular expressions are:

- Search a string (search and match)
- Finding a string (findall)
- Break string into a sub strings (split)
- Replace part of a string (sub)

2. What are various methods of Regular Expressions?

The 're' package provides multiple methods to perform queries on an input string. Here are the most commonly used methods:

1. re.match()
2. re.search()
3. re.findall()
4. re.split()
5. re.sub()
6. re.compile

Student Name: ______________________ **Roll No:** ______________ **Section:** ____________

3. Metacharacters

Metacharacters are characters that are interpreted in a special way by a RegEx engine. Here's a list of metacharacters:

[] . ^ $ * + ? {} () \ |

Character	Description	Example
1. []	A set of characters	"[a-m]"
2. \	Signals a special sequence (can also be used to escape special characters)	"\d"
3. .	Any character (except newline character)	"he..o"
4. ^	Starts with	"^hello"
5. $	Ends with	"world$"
6. *	Zero or more occurrences	"aix*"
7. +	One or more occurrences	"aix+"
8. {}	Exactly the specified number of occurrences	"al{2}"
9. |	Either or	"falls|stays"
10. ()	Capture and group	

You can also specify a range of characters using - inside square brackets.

[a-e] is the same as [abcde].
[1-4] is the same as [1234].
[0-39] is the same as [01239].

Regular Expression(RE) Syntax

```
import re
```

Example of w+ and ^ Expression

"^": This expression matches the start of a string
"w+": This expression matches the alphanumeric character in the string

Student Name: ____________________________ Roll No: _________________ Section: _______________

1. re.search()

Function will search the regular expression pattern and return the first occurrence. the pattern is found and “null” if the pattern is not found

In order to use search() function, you need to import re first and then execute the code. The search() function takes the "pattern" and "text" to scan from our main string

"^": This expression matches the start of a string
"w+": This expression matches the alphanumeric character in the string

Exercise 1: Write a program search the

```
import re

string = "Python is fun"

# check if 'Python' is at the beginning
match = re.search('^Python', string)

if match:
  print("pattern found inside the string")
else:
  print("pattern not found")
```

Output:

2. The findall() function:

The findall() function returns a list containing all matches.

Student Name: ______________________________ **Roll No:** __________________ **Section:** ________________

Exercise 2: Write a code that will able to extract ou from all the words in the string.

```
import  re
str = " The rain in Spain"
x  =  re.findall("Sp",  str)
print(x)
```

Output:

The list contains the matches in the order they are found. If no matches are found, an empty list is returned.

Exercise 3:

\d - Matches any decimal digit. Equivalent to [0-9]

```
# Program to extract numbers from a string

import re

string = 'hello 12 hi 89. Howdy 34'
pattern = '\d+'

result = re.findall(pattern, string)
print(result)
```

Output:

Student Name: ____________________________ Roll No: _________________ Section: _______________

4. re.split()

The re.split method splits the string where there is a match and returns a list of strings where the splits have occurred.
If there is more than one match, only the first occurrence of the match will be returned:

Exercise 4:

```
import re

string = 'Twelve:12 Eighty nine:89.'
pattern = '\d+'

result = re.split(pattern, string)
print(result)
```

Output:

5. The sub() Function:

The sub() function replaces the matches with the text of your choice:

The syntax of re.sub() is:

```
re.sub(pattern, replace, string)
```

The method returns a string where matched occurrences are replaced with the content of replace variable

Exercise 6:

```
# Program to remove all whitespaces
import re

# multiline string
string = 'abc 12\
```

Student Name:____________________________ Roll No:_________________ Section: _______________

```
de 23 \n f45 6'

# matches all whitespace characters
pattern = '\s+'

# empty string
replace = ''

new_string = re.sub(pattern, replace, string)
print(new_string)
.
```

Output:

Student Name: ____________________________ Roll No: _________________ Section: _______________

Programming Exercise (Python)

Task1:
You are working in Data Science project in which you need to find the mobile networks in certain area.

Wireless Provides	**Access Codes**	
Mobilink	030	0300, 0301, 0302, 0303, 0304, 0305, 0306, 0307, 0308, 0309, 03000
Zong	031	0310, 0311, 0312, 0313, 0314, 0315
Warid	032	0320, 0321, 0322, 0323, 0324, 0325
Ufone	033	033, 0331, 0332, 0333, 0334, 0335, 0336, 0337
Telenor	034	0340, 0341, 0342, 0343, 0344, 0345, 0346, 0347
SCOM	035	0355 (AJK & Gilgit- Baltistan)
Instaphone	036	0364

For this your task is to enter at least 10 mobile numbers in a file of different companies based on list provided to you. Extract the mobile numers such as 0332 is for Ufone, then at the end of the list calculate each mobile network in total. The required output will be:

```
Mobile Number              Network              Count
O3323328094                Ufone                 1
.
.
.

After 10 entries
The total will be
_____________
Ufone Users    : 4
Jazz Users       3
Telenor Users    2
Warid Users      1

Total Users    : 10
```