Jamia Millia Islamia

Dept. of Computer Science

Subject: Pattern Matching Using Python
Assignment Topic: 20 String Methods in Python

Submitted By: Wasit Shafi **Submitted To:** Prof.Usman Malik
Roll No: 18MCA054
Date: 29-SEP-2020

Q. Write down 20 string function in python with their running code as an exapmle

Sol.

1.upper() : The upper() method return a string where all characters are in upper case.
Example:

```
s = 'Hello World'
s2 = s.upper()
print(s2)
```

Output : HELLO WORLD

2.lower() : The lower() method return a string where all characters are in lower case.
Example:

```
s = 'Hello World'
s2 = s.lower()
print(s2)
```

Output : hello world

3.swapcase() : The swapcase() method return a string where all upper case lettters are in lower case and vice versa.
Example:

```
s = 'Hello World'
s2 = s.swapcase()
print(s2)
```

Output : hELLO wORLD

4.replace() : The replace() method replaces a specified phrase with another specified phrase. All the occurrences of the specified phrase will be replaced.

```
s = 'Hello World, Hello World'
s2 = s.replace('Hello', 'New')
print(s2)
```

Output : s = 'New World, New World

```
s = 'Hello World, Hello World'
s2 = s.replace('Hello', 'New')
print(s2)
```

5.capitalize() : The capitalize() method returns a string where the first character is upper case.
Example:

```
s = 'hello world'
s2 = s.capitalize()
print(s2)
```

Output : Hello world

6.index() : The index() method return the first occurence of the specified value if not found it raises an exception.
Example:

```
s = 'hello world'
print(s.index('wor'))
```

Output : 6

7.find() : The find() method return the first occurence of the specified value if not found it returns -1.
Example:

```
s = 'hello world'
print(s.find('wor'))
print(s.find('python'))
```

Output : 6
-1

8.strip() : The strip() method removes any leading and trailing specific characters. By default it removes spaces.
Example:

```
s = '        hello world    '
print(s.strip())
s = 'xxxxxxhello worldxxxxxxxxx'
print(s.strip(''))
print(s.strip('x'))
```

Output : hello world

```
xxxxxxhello worldxxxxxxxxx
 hello world
```

9.split() : The split() method splits a string into a list. It takes one argument as separator, if no argument then whitespace will be the separator.
Example:

```
s = 'hello world'
print(s.split())
print(s.split('l'))
```

Output : ['hello', 'world
['he', '', 'o wor', 'd']

10.count() : The count() method return the number of times a specified string appears in the string.
Example:

```
s = 'hello world, hello world, hello'
print(s.count('hello'))
print(s.count('world'))
```

Output : 3
2

11.startwith() : The startswith() method returns True if the string starts with the specified value, otherwise False.
Example:

```
s = 'hello world, hello world’
print(s.startswith('hello'))
print(s.startswith('world'))
```

Output : True
False

12.rfind() : The rfind() method finds the last occurrence of the specified value. The rfind() method returns -1 if the value is not found. The rfind() method is almost the same as the rindex() method.
Example:

```
s = 'hello world, hello world'
print(s.rfind('wor'))
print(s.rfind('python'))
```

Output : 19

-1

13.title() : The title() method returns a string where the first character in every word is upper case.
Example:

```
s = 'hello world, hello world, hello'
print(s.title())
```

Output : Hello World, Hello World, Hello

14.partition() : The partiton() method searches for a specified string, and splits the string into a tuple containing three elements.

1. The first element contains the part before the specified string.

2. The second element contains the specified string.

3. The third element contains the part after the string.

It partitions the string on first occurences of the specified, incase specified is not present in string then second and third partition will be empty

Example:

```
s = 'the quick brown fox jumps over to the lazy dog. Tit for tat'
print(s.partition('the'))
print(s.partition('fox'))
print(s.partition('hello'))
```

Output :

```
('', 'the', ' quick brown fox jumps over to the lazy dog. Tit for tat')
('the quick brown ', 'fox', ' jumps over to the lazy dog. Tit for tat')
('the quick brown fox jumps over to the lazy dog. Tit for tat', '', '')
```

15.isspace() : The isspaces() method returns True if all the characters in a string are whitespaces, otherwise False.

Example:

```
s = 'hello world'
print(s.isspace())
s = '   '
print(s.isspace())
s = ''
print(s.isspace())
```

Output :

```
False
True
False
```

16.islower() : The islower() method returns True if all the characters are in lower case, otherwise False.

Example:

```
s = 'Hello World'
print(s.islower())
s = 'hello world'
print(s.islower())
s = 'helloworld'
print(s.islower())
```

Output :

```
False
True
True
```

17.isupper() : The isupper() method returns True if all the characters are in upper case, otherwise False.

Example:

```
s = 'Hello World'
print(s.isupper())
s = 'HELLO WORLD'
print(s.isupper())
s = 'HELLOWORLD'
print(s.isupper())
```

Output : False

```
True

True
```

18.isalnum() : The isalpha() method returns True if all the characters are alphabet letters (a-z).
Example:

```
s = 'HelloWorld'
print(s.isalpha())
s = 'Hello World'
print(s.isalpha())
s = 'Hello World 123'
print(s.isalpha())
s = 'HelloWorld@gmail.com'
print(s.isalpha())
```

Output : True

```
False

False

False
```

19.isdigits() : The isdigit() method returns True if all the characters are digits, otherwise False.
Example:

```
s = 'Hello World 123'
print(s.isdigit())
s = '123 123'
print(s.isdigit())
s = '123123'
print(s.isdigit())
```

Output : False

```
True

True
```

20.isidentifier(): The isidentifier() method returns True if the string is a valid identifier, otherwise False.

A string is considered a valid identifier if it only contains alphanumeric letters (a-z) and (0-9), or underscores (_). A valid identifier cannot start with a number, or contain any spaces.
Example:

```
s = 'hello world'
print(s.isidentifier())
s = 'hello@world'
print(s.isidentifier())
s = 'helloworld'
print(s.isidentifier())
s = 'hello1world'
print(s.isidentifier())
s = 'hello_world'
print(s.isidentifier())
```

```
Output : False
         False
         True
         True
         True
```