String Cheatsheet

Sequences of characters are referred to as strings. Strings can be any length and can include any character such as letters, numbers, symbols, and whitespace (spaces, tabs, new lines).

A string is a sequence of characters. String is ordered i.e. sequence matters, immutable.(cannot modify).

Creating a String

1) By putting the string in inverted commas ''

```
String="Data Science"
String
# Output: 'Data Science'
```

2) By using a function 'str'

```
Z=str('python ')
Z
# Output: 'python '
```

Indexing and Slicing Strings

A single character can be accessed with bracket notation ([index]), or a substring can be accessed using slicing ([start:end]). Indexing with negative numbers counts from the end of the string.

```
str = 'yellow'
str[4:6]
# Output: 'ow'
```

```
str = 'yellow'
str[:3]
# Output: 'yel'
```

```
str = 'yellow'
str[-3:]
# Output: 'low'
```

String Operations

1)Length of strings-

In Python, the built-in len() function can be used to determines the length of object .

```
length = len("Hello")
print(length)
# Output: 5
```

```
length =len("pink","blue")
print(length)
#output: 2
```

2)String concatenation-

To combine the content of two strings into a single string.

```
x = 'Ram, '
y = 'sham.'
z = x + y
print(z)
# Output: Ram, sham
```

3)Strings are immutable-

```
s='Republic'
s[0]='Z'
s
# output: We get a error
```

4)“in” operator in string-

```
s='TISSUES'
'I' in s
#output: True
```

```
s='TISSUES'
‘k’ in s
#output: False
```

5)String Method.upper()

```
A='Republic’
A=A.upper()
#output: ‘REPUBLIC'
```

6)String Method.lower()

```
A='Republic’
A=A.lower()
#output: ‘republic'
```

7) String Method.count()

```
s='TISSUES S s'
s.count('S')
#output: 4
```

8)String Method.capitalize()

```
s="python for data Science"
s.capitalize()
#output: ‘Python for data Science’
```

9) String Method.title()

```
s=" happy journey "
s.title()
#output: ‘ Happy Journey ‘
```

10)String Method.lstrip()

Removes empty spaces from left .

```
s="   happy journey   "
s.lstrip()
#output: ‘happy journey     ‘
```

11)String Method.rstrip()

Removes empty spaces from right .

```
s="   happy journey   "
s.rstrip()
#output : ‘     happy journey‘
```

12)String Method.strip() -Removes all empty spaces from left and right.

```
s="   happy journey   "
s.strip()
#output : ‘happy journey‘
```

13)String replace-

```
fruit = "Strawberry"
print(fruit.replace('r', 'R'))

# #Output: StRawbeRRy
```

14)String Method.split( )-The string method .split() splits a string into a list of items:

- If no argument is passed, the default behavior is to split on whitespace.
- If an argument is passed to the method, that value is used as the delimiter on which to split the string.

```
text = "Silicon Valley"
text.split( )
#Output:   ['Silicon', 'Valley']
```

15)String Method.join( )-

The string method .join( ) concatenates a list of strings together to create a new string joined with the desired delimiter.

```
x = "-".join(["academy", "is", "awesome"])
x
# Output:   academy-is-awesome
```

16) String Method.find( )-

The Python string method .find() returns the index of the first occurrence of the string passed as the argument. It returns -1 if no occurrence is found.

```
name = "ram sham"
print(mountain_name.find("a"))
# Prints 1 in the console.
```

17) Iterate string -

To iterate through a string in Python, "for...in" notation is used.

```
str = "hello"
for c in str:
 print(c)
# h
# e
# l
# l
# o
```

18) Multiplication of string-

```
a="1"
b=3*a
b
#output: '111'
```