Tutorial_Session2

September 28, 2018

1 String and List

1.1 Strings

- A string is a sequence of characters.
- A string may be specified by placing the member characters of the sequence within quotes (single, double or triple).
- Triple quotes are typically used for strings that span multiple lines.
- Strings are immutable. Ones defined their contents can not be changed

```
In [1]: message = 'Hello Gita'

In [2]: print(type(message), id(message))

<class 'str'> 2183166254960
```

1.1.1 Computing Length using len function

```
In [3]: print(len(message))

10
```

1.2 String Operators

- Concatenation: Concatenates two strings
- String Multiplication: Replicates a string given number of times

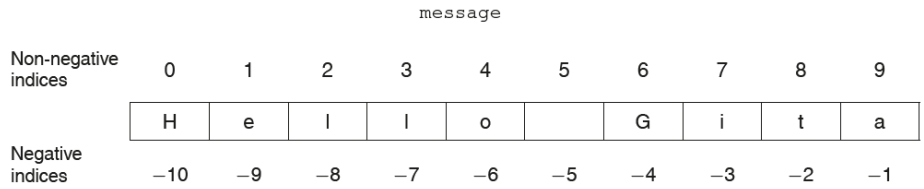
```
In [4]: s='Good'
        ss=s+ 'Boy'
        sss=s * 3
        print(ss)
        print(sss)
        print('My age is='+ 25)

GoodBoy
GoodGoodGood
```

```
---------------------------------------------------------------------------

TypeError                                 Traceback (most recent call last)

<ipython-input-4-eeeabf345569> in <module>()
  4 print(ss)
  5 print(sss)
----> 6 print('My age is='+ 25)


TypeError: must be str, not int
```

```
In [6]: print('My age is=' + str(25))

My age is=25
```

1.2.1 Indexing

- Individual characters within a string are accessed using a technique known as indexing.




```
In [7]: message='Hello Gita'
        index = len(message) - 1
        print(message[0], message[6], message[index], message[-1])

H G a a
```

```
In [8]: print(message[15])

---------------------------------------------------------------------------

IndexError                                Traceback (most recent call last)

<ipython-input-8-a801df50d8d1> in <module>()
----> 1 print(message[15])


IndexError: string index out of range
```

1.2.2 Slicing

- In order to extract the substring comprising the character sequence having indices from start to end-1, we specify the range in the form start:end.
- Python also allows us to extract a subsequence of the form *start:end:inc*.

```
In [9]: message = 'Hello Sita'
        print(message[0:5], message[-10:-5])
```

```
Hello Hello
```

```
In [10]: print(message[0:len(message):2])
         print(message[:])
```

```
HloSt
Hello Sita
```

```
In [11]: message='HelloSeeta'
         print(message[len(message)-1::-1])
         print()
         print(message[len(message)::-2])
         print()
         print(message[len(message):5:-1])
         print()
         print(message[len(message):5:-2])
```

```
ateeSolleH

aeSle

atee

ae
```

1.2.3 Membership Operator in

- Python also allows us to check for membership of the individual characters or substrings in strings using in operator.

```
In [12]: 'h' in 'hello'
```

```
Out[12]: True
```

```
In [13]: 'ell' in 'hello'
```

```
Out[13]: True
```

```
In [14]: 'h' in 'Hello'
```

```
Out[14]: False
```

1.3 Built-in Functions on Strings

1.3.1 Function: count

- For counting number of occurrences of a substring.

```
In [15]: 'Encyclopedia'.count('c')
```

```
Out[15]: 2
```

```
In [16]: vowels = 'AEIOUaeiou'
         vowelCount = 0
         for ch in vowels:
             vowelCount += 'Encyclopedia'.count(ch)
         print(vowelCount)
```

```
5
```

1.3.2 Functions find and rfind

- Function **find**: Returns the index of the first occurrence of a string.
- Function **rfind**: Returns the index of the last occurrence of a string.
- Returns -1 if the searched string is not present in the source string.

```
In [17]: colors = 'green, red, blue, red, red, green'
         print("colors.find('red'): ", colors.find('red'))
         print("colors.rfind('red'): ", colors.rfind('red'))
         print("colors.find('orange'): ", colors.find('orange'))
         colors.count('red')
```

```
colors.find('red'):  7
colors.rfind('red'):  23
colors.find('orange'):  -1
```

```
Out[17]: 3
```

1.3.3 Functions capitalize, title, lower, upper, and swapcase

- Function **capitalize**: converting the first letter of a string to uppercase character and converting the remaining letters in the string to lowercase.
- Function **title**: Capitalize the first letter of each word in a string and change the remaining letters to lowercase.
- Function **lower**: Convert all letters in a string to lowercase.
- Function **upper**: Convert all letters in a string to uppercase.

```
In [18]: 'python IS a Language'.capitalize()
```

```
Out[18]: 'Python is a language'
```

```
In [19]: 'python IS a PROGRAMMING Language'.title()
```

```
Out[19]: 'Python Is A Programming Language'
```

```
In [20]: emailId1 = 'geek@gmail.com'
         emailId2 = 'Geek@gmail.com'
         emailId1.lower() == emailId2.lower()
```

```
Out[20]: True
```

1.3.4 Function swapcase

```
In [21]: 'AnilKumar'.swapcase()
```

```
Out[21]: 'aNILkUMAR'
```

1.3.5 Functions islower, isupper, isalpha, isdigit, and isalnum

```
In [22]: 'python'.islower()
```

```
Out[22]: True
```

```
In [23]: 'Python'.isupper()
```

```
Out[23]: False
```

```
In [24]: '9953799924'.isdigit()
```

```
Out[24]: True
```

```
In [25]: 'Nikhil Kumar'.isalpha()
```

```
Out[25]: False
```

```
In [26]: password = 'Kailash107Ganga'
         password.isalnum()
```

```
Out[26]: True
```

1.3.6 Function replace

- It allows to replace part of a string by another string.
- It takes two arguments as inputs. The first argument is used to specify the substring that is to be replaced. The second argument is used to specify the string that replaces the first string.

```
In [27]: message = 'Amey my friend, Amey my guide'
```

```
In [28]: message.replace('Amey', 'Vihan')
```

```
Out[28]: 'Vihan my friend, Vihan my guide'
```

```
In [29]: message='Amey my friend, Amey my guide'
         message.replace('Ram','Seeta')
         print(message)
```

```
Amey my friend, Amey my guide
```