STRING 2

In [ ]:
```
It is the time now every one should prepare a document on strings concept, write your understanding on Strings
You need to cover following topics

1) Reading strings
2) Type, len,max,min
3)Concatenation
4) Mutability conceot
5) String index
6) String slice
7) String methods
Capitalize/Casefold/upper/lower
Index/find
Strip-rstrip-lstrip
Startswith/endswith
Count
Replace

You need to write at least one example
```

In [ ]:
```
```

In [2]:
```
name='python'
print(type(name))
```

```
<class 'str'>
```

In [3]:
```
len(name)
```

Out[3]:
```
6
```

In [4]:
```
max(name)
```

Out[4]:
```
'y'
```

In [5]:
```
min(name)
```

Out[5]:
```
'h'
```

In [6]:
```
type(name)
```

Out[6]:
```
str
```

In [2]:
```
name='python'
print(name)
```

```
python
```

In [4]:
```
# doc string ---triple cotes
# multiple lines
# in jupyter--markdown
# notepad
# vs code
# pycharm

string1=""" hi how are you, i am good i am learning python """
string1
```

Out[4]:
```
' hi how are you, i am good i am learning python '
```

In [6]:
```
- triple cotes means doc string
- it is information about the code
```

In [14]:
```
print("hello 'python'")
print('hello "python"')

#Entire string will double quotes, the highlighted string is single quotes
#Entire string will single quotes, the highlighted string is double quotes
```

In [15]:
```
string1='python' #python
string2=string1  #python
string3='hello'  #hello
string4=string2  #python
string5=string4
string5
```

Out[15]:
```
'python'
```

In [17]:
```
string1='python'
len(string1)
```

Out[17]:
```
6
```

In [18]:
```
type(string1)
```

Out[18]:
```
str
```

In [19]:
```
max(string1)  #ascii value
```

Out[19]:
```
'y'
```

In [20]:
```
min(string1)
```

Out[20]:
```
'h'
```

In [24]:
```
'p'in 'python'
'y'in 'python'
't'in 'python'
'h'in 'python'
'o'in 'python'
'n'in 'python'

#for i in 'python':
    #print(i,ord(i))
```

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

In [ ]:
```
string1='hai'
string2='hellow'
string1+string2 # possible i.e concatination
string1-string2  # not possible
string1*string2  # not possible
string1/string2  #not possible
```

In [9]:
```
string1='hello'
string2='python'
string3='how are you'
print(string1+string2+string3)    ##concatination = series of string
```

```
hellopythonhow are you
```

In [3]:
```
#this program about infinite while loop
#the original password='python'
#function:
#arguments: None
#return: None
def pwd():
    password='python'
    user_password=input('Enter the password ')
    if user_password==password:
        print('you won')
    else:
        print('wrong password')

#def pwd():



#if user_password==password:
   # print('you won')
   # else:
       # print('wrong password')
```

In [5]:
```
pwd()
```

```
Enter the password python
you won
```

index

In [45]:
```
string1='python'
len(string1)

# the python index start with zeros.
```

Out[45]:
```
6
```

In [16]:
```
string1='python'
i=0
for i in range(len(string1)):
    print('the index of {} is {}'.format(string1[i],i,end=' '))
```

```
the index of p is 0
the index of y is 1
the index of t is 2
the index of h is 3
the index of o is 4
the index of n is 5
```

In [2]:
```
string1='python'
i=0
while i<len(string1):
    print('the index of {} is'.format((string1[i],len(string1)-i)))
    i=i+1
```

```
the index of ('p', 6) is
the index of ('y', 5) is
the index of ('t', 4) is
the index of ('h', 3) is
the index of ('o', 2) is
the index of ('n', 1) is
```

In [ ]:
```
# wap ask the user print the letters from a given word
# using for loop
# 'the negative index of p is:-6'
# 'the negative index of y is:-5'
# i want letter as well as attached index number
```

In [7]:
```
name4='python'
i=0
for i in range(len(name4)):

    print('The negative index of {} is: {}'.format(name4[i],i-len(name4)))
    i+=1
```

```
The negative index of p is: -6
The negative index of y is: -5
The negative index of t is: -4
The negative index of h is: -3
The negative index of o is: -2
The negative index of n is: -1
```

In [8]:
```
# how to print -6 -5 -4 -3 -2 -1 using while loop
name3='python'
i=0
while i<len(name3):
    print('The negative index of {} is: {}'.format(name3[i],i-len(name3)))
    i+=1
```

```
The negative index of p is: -6
The negative index of y is: -5
The negative index of t is: -4
The negative index of h is: -3
The negative index of o is: -2
The negative index of n is: -1
```

In [9]:
```
name='python'
for i in range(len(name)):
    print('The positive index is {}, the negative index is {}, for {}'.format(i,i-len(name),name[i]))
```

```
The positive index is 0, the negative index is -6, for p
The positive index is 1, the negative index is -5, for y
The positive index is 2, the negative index is -4, for t
The positive index is 3, the negative index is -3, for h
The positive index is 4, the negative index is -2, for o
The positive index is 5, the negative index is -1, for n
```

In [10]:
```
name=input('Enter the word')
for i in range(len(name)):
    print('The positive index {}, the negative index is {}, for {}'.format(i,i-len(name),name[i]))
```

```
Enter the wordpython
The positive index 0, the negative index is -6, for p
The positive index 1, the negative index is -5, for y
The positive index 2, the negative index is -4, for t
The positive index 3, the negative index is -3, for h
The positive index 4, the negative index is -2, for o
The positive index 5, the negative index is -1, for n
```

In [3]:
```
sent='hai hai how are you'
count=0
for i in range(len(sent)):
    if sent[i]=='a':
        count=count+1
        print(i)
print('no. of repeted a are',count)
```

```
1
5
12
no. of repeted a are 3
```

slice

slice

In [4]:
```
string1='hello how are you'

# this concept is same as range() in for loop concept

# slice(start,stop,step)   range(start,stop,step)

string1[2:10]
#start=2
#stop=10-1=9

# h  e  l  l  o      h  o  w     a    r   e    y  o  u
  0  1  2  3  4   5  6  7  8  9  10  11  12 13 14 15 16
```

Out[4]:
```
'llo how '
```

In [13]:
```
string1[::]  # [start:last char:step]

# start value=first letter   if nothing is mention
# stop value=last letter     if nothing is mention
# step value=+ve direction   if nothing is mention

print(string1[:0])
print(string1[::len(string1)])
print(string1[:])
print(string1[::])
```

```
h
hello how are you
hello how are you
```

In [14]:
```
string1[::-1]

# in negative
#start = u
#stop = h
```

Out[14]:
```
'uoy era woh olleh'
```

String methods Capitalize/Casefold/upper/lower

In [16]:
```
name='snehal raikwar'
name.capitalize()

# capitalize makes first character was upper case and rest is lower case
```

Out[16]:
```
'Snehal raikwar'
```

UPPER AND LOWER

In [17]:
```
name='hello'
help(name.upper)
```

```
Help on built-in function upper:

upper() method of builtins.str instance
    Return a copy of the string converted to uppercase.
```

In [18]:
```
name.upper()

# in upper keyword makes complete sentence is upper case
```

Out[18]:
```
'HELLO'
```

In [19]:
```
help(name.lower)
```

```
Help on built-in function lower:

lower() method of builtins.str instance
    Return a copy of the string converted to lowercase.
```

In [20]:
```
name='PYTHON'
name.lower()
```

Out[20]:
```
'python'
```

In [22]:
```
name='hello how ARE YOU'

print(name.capitalize())
print(name.upper())
print(name.lower())
```

```
Hello how are you
HELLO HOW ARE YOU
hello how are you
```

casefold

In [25]:
```
name='hello how are you'
help(name.casefold)
print(name.casefold())
```

```
Help on built-in function casefold:

casefold() method of builtins.str instance
    Return a version of the string suitable for caseless comparisons.

hello how are you
```

In [51]:
```
string1='python'
string2='P'
string4='Y'
string3=string1[2:]
print(string2+string4+string3)
```

```
PYthon
```

count

In [53]:
```
string1='hai hai hai ola'
print(string1.count('hai'))
print(string1.count('h'))
print(string1.count('o'))
```

```
3
3
1
```

In [49]:
```
string2='hello hello hello hello'
string2.count('o',4)
# 4 means we are counting o from 4th index.
```

Out[49]:
```
4
```

In [ ]:

In [ ]:

REPLACE

In [ ]:
```
string1='hello welcome'


#i want to replace 'o' with 0
```

In [33]:
```
string1='hello welcome'
string1.replace('o','O')
```

Out[33]:
```
'hellO welcOme'
```

In [45]:
```
string1='welcome'
string1[:2]+'L'+string1[3:]    # slice + concetanation
```

Out[45]:
```
'weLcome'
```

In [46]:
```
string1.replace('l','L')
```

Out[46]:
```
'weLcome'
```

In [47]:
```
string2='naresh it'
string2.replace('a','$$$$')
```

Out[47]:
```
'n$$$$resh it'
```

Index/find Strip-rstrip-lstrip

In [1]:
```
string1='Hospital'
string1.index('p')

# index of p is 3
# starts with zero
```

Out[1]:
```
3
```

In [26]:
```
string1='tiktoktiktok'
```

```
print(string1.count('t'))
print(string1.index('t'))
f1=string1.index('t')
s1=string1.index('t',f1+1)
t1=string1.index('t',s1+1)
r1=string1.index('t',t1+1)
print(string1.index('t',f1))
print(string1.index('t',s1))
print(string1.index('t',t1))
print(string1.index('t',r1))
```

```
4
0
0
3
6
9
```

find

In [30]:
```
string1='tiktok'
string1.find('t')
f_o=string1.find('t')
s_o=string1.find('t',f_o+1)
print(f_o,s_o)
```

```
0 3
```

split

In [49]:
```
string1='tik tok tik tok'
string1.split()
print(string1.split())
```

```
['tik', 'tok', 'tik', 'tok']
```

In [50]:
```
string1.split(',')
```

Out[50]:
```
['tik tok tik tok']
```

In [40]:
```
string2='hai how are you'
string2.split()
```

Out[40]:
```
['hai', 'how', 'are', 'you']
```

In [53]:
```
var1='we are'
var2='going to home'
print(var1,var2,sep=' - ')
```

```
we are - going to home
```

In [55]:
```
#sent2='mobin.sourav@uma.com'
sent2='mobin.souravsahoo@narehit.ac.in'
#between '@' and '.' (Second)
index1=sent2.index('@')
first_dot_index=sent2.index('.')
sec_dot_index=sent2.index('.',first_dot_index+1)
sent2[index1+1:sec_dot_index]
```

Out[55]:
```
'narehit'
```

In [56]:
```
sent1='snehal.raikwar@gmail.com'
f1=sent1.index('.')
s1=sent1.index('@')
sec_dot_index=sent1.index('@',f1+1)
print(sent1[f1+1:sec_dot_index])
```

```
raikwar
```

In [57]:
```
sent1.index('.')
```

Out[57]:
```
6
```

In [58]:
```
sent1.index('@')
```

Out[58]:
```
14
```

strip-lstrip-rstrip:

In [62]:
```
#strip = it will remove spaces both sides
#lstrip = it will remove spaces left side
#rstrip = it will remove spaces right side

string1=' hai how are you '
string1.strip()
```

Out[62]:
```
'hai how are you'
```

In [67]:
```
string1.lstrip()
```

Out[67]:
```
'hai how are you '
```

In [66]:
```
string1.rstrip()
```

Out[66]:
```
' hai how are you'
```

In [73]:
```
string1= '*******hospital*******'
print(string1.strip('*'))
print(string1.lstrip('*'))
print(string1.rstrip('*'))
```

```
hospital
hospital*******
*******hospital
```

In [72]:
```
string1.lstrip('*')
```

Out[72]:
```
'hospital*******'
```

startswith-endswith

In [76]:
```
string1='hello how are you..i am fine'
string1.startswith('h')
```

Out[76]:
```
True
```

In [77]:
```
string1.endswith('fine')
```

Out[77]:
```
True
```

In [78]:
```
string1.startswith('are')
```

Out[78]:
```
False
```

In [79]:
```
string1.endswith(string1)
```

Out[79]:
```
True
```

In [80]:
```
string1.startswith(string1)
```

Out[80]:
```
True
```

In [ ]:
```

```