In [ ]:

```
string:
    --> A string is a group of characters or sequence of characters....
    --> Strings are immutable
        * once we defined,they can't be changed.

    --> strings can be created by enclosing characters inside a single quote
                      (or) double quote
```

In [1]:

```
s='python'
print(s)
```

```
python
```

In [2]:

```
s="python"
print(s)
```

```
python
```

In [3]:

```
s="""python"""
s
```

Out[3]:

```
'python'
```

In [4]:

```
a="7347mnfjdfh@@%"
print(a)
```

```
7347mnfjdfh@@%
```

In [5]:

```
a="387467"
print(a)
```

```
387467
```

In [6]:

```
a
```

Out[6]:

```
'387467'
```

In [7]:

```
s
```

Out[7]:

```
'python'
```

In [ ]:

```
--> Indexing

  1.Positive index : 0,1,2,3,.....................
  2.Negative index : -1,-2,-3,-4.....................

    syntax:   string_variable[position]
```

In [ ]:

```
--> string slicing:
    * cutting into pieces
syntax: string_Variable[start:end:step]
```

In [8]:

```
# indexing
s="python"
print(s)
```

```
python
```

In [9]:

```
type(s)
```

Out[9]:

```
str
```

In [10]:

```
d=input()
print(d)
```

```
workshop
workshop
```

In [ ]:

```
# 5
# odd
```

In [11]:

```
d
```

Out[11]:

```
'workshop'
```

In [12]:

```
d[0]
```

Out[12]:

```
'w'
```

In [13]:

```
d[4]
```

Out[13]:

```
's'
```

In [14]:

```
d[7]
```

Out[14]:

```
'p'
```

In [15]:

```
# positive index
print(d[0])
print(d[1])
print(d[2])
print(d[3])
print(d[4])
print(d[5])
print(d[6])
print(d[7])
```

```
w
o
r
k
s
h
o
p
```

In [ ]:

```
# negative index
# workshop--- string
```

In [16]:

```
d[-1]
```

Out[16]:

```
'p'
```

In [17]:

```
d[-5]
```

Out[17]:

```
'k'
```

In [18]:

```
d[-6]
```

Out[18]:

```
'r'
```

In [19]:

```
f="desktop"
for i in f:   # desktop   ---> d  e  s   k.............
    print(i)  #           ---> d  e  s   k...........
```

```
d
e
s
k
t
o
p
```

In [20]:

```
f="desktop"
for i in f:
    print(i,end='')
```

```
desktop
```

In [21]:

```
# min,max,sum,sorted,len---> predefined functions
c="python"
print(min(c))
print(max(c))
print(sorted(c))
print(len(c))
print(sum(c))
```

```
h
y
['h', 'n', 'o', 'p', 't', 'y']
6

---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-21-64f2e3f936e7> in <module>
      5 print(sorted(c))
      6 print(len(c))
----> 7 print(sum(c))

TypeError: unsupported operand type(s) for +: 'int' and 'str'
```

In [ ]:

```
# slicing
# python ---  pyt
starting index--- inclued
ending index--- excluded
```

In [22]:

```
x="python"
x[0:3]
```

Out[22]:

```
'pyt'
```

In [23]:

```
# pytho
x[0:5]
```

Out[23]:

```
'pytho'
```

In [24]:

```
# ytho
x[1:5]
```

Out[24]:

```
'ytho'
```

In [25]:

```
x[:]
```

Out[25]:

```
'python'
```

In [26]:

```
x[::]
```

Out[26]:

```
'python'
```

In [27]:

```
x[:3]
```

Out[27]:

```
'pyt'
```

In [28]:

```
x[3:]
```

Out[28]:

```
'hon'
```

In [29]:

```
x[1:]
```

Out[29]:

```
'ython'
```

In [30]:

```
# python
x[0:6:2]

# 0+2=2 2+2=4  4+2=6
```

Out[30]:

```
'pto'
```

In [31]:

```
# ph
# python-   0+3=3  3+3=6
x[0:6:3]
```

Out[31]:

```
'ph'
```

In [32]:

```
x='python workshop'
x[0:15:3]
# 0+3=3  3+3=6 6+3=9 9+3=12 12+3=15
```

Out[32]:

```
'ph rh'
```

In [33]:

```
# string concatenation
s1="book"
s2="pen"
print(s1+" "+s2)
```

```
book pen
```

In [34]:

```
x="python"
x[::-1]
```

Out[34]:

```
'nohtyp'
```

In [35]:

```
# pythonworkshop-- ytho
# x[1:-1]
c='pythonworkshop'
c[1:-1]
```

Out[35]:

```
'ythonworksho'
```

In [36]:

```
#string methods....
print(dir(str),end=' ')
```

```
['__add__', '__class__', '__contains__', '__delattr__', '__dir__', '__doc_
_', '__eq__', '__format__', '__ge__', '__getattribute__', '__getitem__', '__
getnewargs__', '__gt__', '__hash__', '__init__', '__init_subclass__', '__ite
r__', '__le__', '__len__', '__lt__', '__mod__', '__mul__', '__ne__', '__new_
_', '__reduce__', '__reduce_ex__', '__repr__', '__rmod__', '__rmul__', '__se
tattr__', '__sizeof__', '__str__', '__subclasshook__', 'capitalize', 'casefo
ld', 'center', 'count', 'encode', 'endswith', 'expandtabs', 'find', 'forma
t', 'format_map', 'index', 'isalnum', 'isalpha', 'isascii', 'isdecimal', 'is
digit', 'isidentifier', 'islower', 'isnumeric', 'isprintable', 'isspace', 'i
stitle', 'isupper', 'join', 'ljust', 'lower', 'lstrip', 'maketrans', 'partit
ion', 'replace', 'rfind', 'rindex', 'rjust', 'rpartition', 'rsplit', 'rstri
p', 'split', 'splitlines', 'startswith', 'strip', 'swapcase', 'title', 'tran
slate', 'upper', 'zfill']
```

In [37]:

```
w="python training"
print(w.capitalize())   # It converts first letter of string into capital
```

```
Python training
```

In [38]:

```
w="python training"
print(w.casefold())
```

```
python training
```

In [39]:

```
w="PYTHON WORKSHOP"
w.casefold()  # It converts uppercase letters into lowercase letters
```

Out[39]:

```
'python workshop'
```

In [40]:

```
d="python"
d.center(23,'@')
```

Out[40]:

```
'@@@@@@@@@python@@@@@@@@'
```

In [41]:

```
w="python training"
w.title()
```

Out[41]:

```
'Python Training'
```

In [42]:

```
w="hai 4232rtg"
w.title()
```

Out[42]:

```
'Hai 4232Rtg'
```

In [43]:

```
w="python training"
w.count('i')
```

Out[43]:

```
2
```

In [44]:

```
w.count('n')
```

Out[44]:

```
3
```

In [45]:

```
s="python"
print(s.upper())
```

```
PYTHON
```

In [46]:

```
s="PYTHON"
print(s.lower())
```

```
python
```

In [49]:

```
s="ASFGHJLANCKNINCJS"
s.swapcase()
```

Out[49]:

```
'asfghjlancknincjs'
```

In [50]:

```
s='Python'
s.casefold()
```

Out[50]:

```
'python'
```

In [51]:

```
c="python program"
print(c.startswith('p'))
print(c.endswith('m'))
print(c.endswith('u'))
```

```
True
True
False
```

In [52]:

```
a="python workshop"
print(a.find("o"))
print(a.rfind("o"))
print(a.find("v"))
print(a.index("p"))
print(a.rindex("p"))
print(a.index("m"))
```

```
4
13
-1
0
14

---------------------------------------------------------------------------
ValueError                                Traceback (most recent call last)
<ipython-input-52-058bf30239fb> in <module>
      5 print(a.index("p"))
      6 print(a.rindex("p"))
----> 7 print(a.index("m"))

ValueError: substring not found
```

In [53]:

```
s="34work"
print(s.isalpha())  # It returns true when our entire string is only alphabets
print(s.isalnum())
```

```
False
True
```

In [54]:

```
d="342ojdhfd"
d.isalnum()  # It returns true when our entire string is combination of numbers and alphabe
```

Out[54]:

```
True
```

In [55]:

```
k="jhdrh@%"
k.isascii()  # It gives always TRUE
```

Out[55]:

```
True
```

In [56]:

```
s="3478374"
s.isdigit()   # It return true if all characters in a string are digits
```

Out[56]:

```
True
```

In [57]:

```
s1="workshop"
print(s1.islower())
print(s1.isupper())
```

```
True
False
```

In [58]:

```
s2="EIRIOEH"
print(s2.isupper())
print(s2.islower())
```

```
True
False
```

In [59]:

```
d=" "
d.isspace()
```

Out[59]:

```
True
```

In [60]:

```
d=""
d.isspace()
```

Out[60]:

```
False
```

In [61]:

```
z="           work          "
print(z.strip())
print(z.lstrip())
print(z.rstrip())
```

```
work
work
           work
```

In [62]:

```
d="py@th%on onl@ine work%sh%op"
print(d.split())
print(d.split("%"))
print(d.split("@"))
```

```
['py@th%on', 'onl@ine', 'work%sh%op']
['py@th', 'on onl@ine work', 'sh', 'op']
['py', 'th%on onl', 'ine work%sh%op']
```

In [63]:

```
d="work"
d.replace('w','t')
```

Out[63]:

```
'tork'
```

In [64]:

```
s="python online\ntraining"
s.splitlines()
```

Out[64]:

```
['python online', 'training']
```

In [65]:

```
g="a","p","s","s","d","c"
print(" ".join(g))
print("@".join(g))
print("$".join(g))
print("".join(g))
```

```
a p s s d c
a@p@s@s@d@c
a$p$s$s$d$c
apssdc
```

In [66]:

```
s="my name is {name}".format(name="ashok")
print(s)
```

```
my name is ashok
```

In [ ]:

```
Tasks:
1.    # input:  "A P S S D C"
        # output:  Apssdc
```

In [ ]:

```
2.# input:    python workshop
     # output:   P.Workshop

  # input:  Puvvala Ambika Pavani
  # output: P.Ambika Pavani
```

In [ ]:

```
3.print the words which are not ends with vowels

 # input: Apssdc Ambika Anu Pavani
 # output: Apssdc
```

In [ ]:

```
4. input: "Python Workshop 848353"
   output : Capital letters count  : 2
            smaller letters count  : 12
            digits count: 6
```

In [ ]:

```

```