complex data types

In [4]:

```
#crreating a complex number
#z=real+imaginary*1j
z=3+4j
z
```

Out[4]:

```
(3+4j)
```

In [5]:

```
#accessing the real and imaginary part
print(z.real)
print(z.imag)
```

```
3.0
4.0
```

In [11]:

```
#operation with complex numbers
z=1+3j
z1=3+2j
print('add:',z+z1)
print('dif:',z+z1)
print('mul:',z*z1)
```

```
add: (4+5j)
dif: (4+5j)
mul: (-3+11j)
```

In [14]:

```
#using builtin functions
z=2+4j
print(abs(z)) #magnitude
print(z.conjugate()) #conjugate
```

```
4.47213595499958
(2-4j)
```

In [7]:

```
#complex number in cmath model
import cmath
z=1+1j
#getting phase angle
print(cmath.phase(z))  #radians
```

```
0.7853981633974483
```

In [8]:

```
#polar form of a complex number(magnitude,angle)
print(cmath.polar(z))
```

```
(1.4142135623730951, 0.7853981633974483)
```

In [9]:

```
#sqrt of acomplex number
print(cmath.sqrt(z))
```

```
(1.09868411346781+0.45508986056222733j)
```

usage of print

In [15]:

```
# print is use for printing the output
a=10
b=20
a
b
```

Out[15]:

```
20
```

In [17]:

```
#for printing two or any values we have to use print
a=3
b=4
print(a)
print(b)
```

```
3
4
```

In [18]:

```
print(8)
print(6,20)
print('python')
print(10,20,'python')
```

```
8
6 20
python
10 20 python
```

In [19]:

```
num1=21
num2=26
add=num1+num2
print(add)
```

```
47
```

print result with string

In [20]:

```
num1=34
num2=20
add=num1+num2
print('The addition of',num1,'and',num2,'is=',add)
```

```
The addition of 34 and 20 is= 54
```

print Format method

In [22]:

```
a=10
b=60
c=a+b
print('addition of {} and {} is {}'.format(a,b,c))   # for every variable we sho
```

```
addition of 10 and 60 is 70
```

In [30]:

```
n1=2
n2=3
n3=5
avg=(n1+n2+n3)/3
```

```
avg1=round((n1+n2+n3)/3,2)
print('avg of{},{},{} is {} or {}'.format(n1,n2,n3,avg,avg1))
```

```
avg of2,3,5 is 3.3333333333333335 or 3.33
```

format using fstring

In [31]:

```
num1=37
num2=30
add=num1+num2
print(f'The addition of {num1} and {num2} is= {add}') # always prefer this
```

```
The addition of 37 and 30 is= 67
```

In [32]:

```
a=10
b=60
c=a+b
print(f'additin of{a} and{b} is {c}')
```

```
additin of10 and60 is 70
```

end statement

In [33]:

```
print('hi')
print('everyone')
```

```
hi
everyone
```

In [35]:

```
print('hi',end=' ')  # end is used for joining
print('everyone')
```

```
hi everyone
```

seprator

In [36]:

```
#here one print statement only we use insisde one print statement we have multip
print('hello','everyone','i am','sumayya',sep='----')
```

```
hello----everyone----i am----sumayya
```

In [37]:

```
print('hello','everyone','i am','sumayya',sep='$')
```

```
hello$everyone$i am$sumayya
```

In [38]:

```
print(3,'.')
```

```
3 .
```

In [40]:

```
print(3,'.',sep='')
```

```
3.
```

In [41]:

```
print(1,2,end=' ')
print(3,'.',sep='')
```

```
1 2 3.
```

string

In [1]:

```
#single line commentstart wit #
greeting='hello' # string can be in sigle quotes or double""
print(greeting)
print(len(greeting))
```

```
hello
5
```

In [3]:

```
#multiline string
multiline_string='''i'm sumayya taskeen
i'm a cse graduate currently studying
fullstack datasciense with ai'''
print(multiline_string)
```

```
i'm sumayya taskeen
i'm a cse graduate currently studying
fullstack datasciense with ai
```

In [4]:

```
multiline_string="""i'm sumayya taskeen
i'm a cse graduate currently studying
fullstack datasciense with ai"""
print(multiline_string)
```

```
i'm sumayya taskeen
i'm a cse graduate currently studying
fullstack datasciense with ai
```

In [5]:

```
#string concatanation
first='sumayya'
last='taskeen'
space=' '
full_name=first+space+last
print(full_name)
```

```
sumayya taskeen
```

In [7]:

```
print(len(full_name))
```

```
15
```

In [10]:

```
print(len(first)>len(last))
```

```
False
```

In [11]:

```
#unpacking characters
language='python'
a,b,c,d,e,f=language
print(a)
print(b)
print(c)
print(d)
print(e)
print(f)
```

```
p
y
t
h
o
n
```

In [12]:

```
#accessing charaters by index
language='python'
print(language[0])
```

```
p
```

In [13]:

```
print(language[1])
print(language[2])
print(language[3])
print(language[4])
```

```
y
t
h
o
```

In [14]:

```
print(language[-1])
print(language[-2])
print(language[-3])
print(language[-4])
```

```
n
o
h
t
```

In [16]:

```
#slicing
name='sumayya'
print(name[:3])
print(name[2:3])
print(name[3:])
first=name[:2]
last=name[2:]
print(first)
print(last)
```

```
sum
m
ayya
su
mayya
```

In [17]:

```
#skippipng characterwhile spitting  string
s=name[0:2:4]
print(s)
```

```
s
```

In [20]:

```
#escape sequence
print('python is a very intresting lanuage \n do u all agree')# \n is used for l
print('python is a very intresting lanuage\tdo u all agree') # used for some spa
print('day1 \t3\t5')
```

```
python is a very intresting lanuage
 do u all agree
python is a very intresting lanuage     do u all agree
day1    3       5
```

string methods

In [21]:

```
# capitalize()

greeting='hello everyone'
print(greeting.capitalize())
```

```
Hello everyone
```

In [25]:

```
#count()
print(greeting.count('a'))
print(greeting.count('e'))
print(greeting.count('l'))
```

```
0
4
2
```

In [28]:

```
#endswith()
print(greeting.endswith('one'))
print(greeting.endswith('tion'))
```

```
True
False
```

In [32]:

```
#expand tabs()
greeting='hello\teveryone'
print( greeting.expandtabs())
print( greeting.expandtabs(20))
```

```
hello   everyone
hello               everyone
```

In [44]:

```
#find()
challenge='study python in 30 days'
print(challenge.find(d))
print(challenge.isalnum())
```

```
9
False
```

In [43]:

```
#isalnum()  checks alphanumeric character
challenge='thirtydaysofpython'
print(challenge.isalnum())
```

```
True
```

In [39]:

```
greet='3hello'
print(greet.isalnum())
```

```
True
```

In [46]:
```
#isalpha() checks if all characters are alphabets
print(greet.isalpha())
name='hello'
print(name.isalpha())
```

```
False
True
```

In [48]:
```
challenge='thirty days of python'
print(challenge.find('y'))
```

```
5
```

In [50]:
```
#isdigit()
challenge='twenty'
print(challenge.isdigit())
```

```
False
```

In [51]:
```
challenge='30'
print(challenge.isdigit())
```

```
True
```

In [53]:
```
#is decimal()
n1='10'
n2='2.5'
print(n1.isdecimal())
print(n2.isdecimal())
```

```
True
False
```

In [55]:
```
#isidentifier()
name='taskeen'
print(name.isidentifier())
n2='3taskeen'
print(n2.isidentifier())
```

```
True
False
```

In [57]:
```
#islower() isupper()
n1='SUMAYYA'
n2='taskeen'
print(n1.islower())
print(n2.islower())
print(n1.isupper())
print(n2.isupper())
```

```
False
True
True
False
```

In [58]:
```
#isnumeric()
num='10'
print(num.isnumeric())
print('ten'.isnumeric())
```

```
True
False
```

In [60]:

```
#join
fruits=['banana','apple','orange']
add='*'.join(fruits)
print(add)
```

```
banana*apple*orange
```

In [68]:

```
#strip()
#string.strip([chars])
#If no argument is passed, it removes whitespace (spaces, tabs, newlines) from t

#If characters are passed, it removes all those characters from the beginning an
data="###welcome###"
hello=data.strip('#')
print(hello)

```

```
welcome
```

In [69]:

```
#replace() it replace substring inside
challenge='thirty days of python'
print(challenge.replace('python','coding'))
```

```
thirty days of coding
```

In [70]:

```
#split  split string from left
print(challenge.split())
```

```
['thirty', 'days', 'of', 'python']
```

In [71]:

```
#title()
print(challenge.title())
```

```
Thirty Days Of Python
```

In [73]:

```
#swapspace()
print(challenge.swapcase())
```

```
THIRTY DAYS OF PYTHON
```

In [77]:

```
#startswith()
print(challenge.startswith('thirty'))
print(challenge.startswith('python'))
```

```
True
False
```

In [ ]: