Strip

In [1]:
```
txt = "   abc  def  ghi   "
txt.lstrip()
```

Out[1]:
```
'abc  def  ghi   '
```

In [2]:
```
txt.rstrip()
```

Out[2]:
```
'   abc  def  ghi'
```

In [3]:
```
txt.strip()
```

Out[3]:
```
'abc  def  ghi'
```

Escape Character

In [4]:
```
gvk = "my favourite color is "Gray""
```

```
  Cell In[4], line 1
    gvk = "my favourite color is "Gray""
                                  ^
SyntaxError: invalid syntax
```

In [6]:
```
gvk = "my favourite color is \"Gray\""
```

In [7]:
```
gvk
```

Out[7]:
```
'my favourite color is "Gray"'
```

List

In [9]:
```
l1=[]
```

In [10]:
```
l1
```

Out[10]:
```
[]
```

In [11]:
```
print(type(l1))
```

```
<class 'list'>
```

In [12]:
```
type(l1)
```

Out[12]:
```
list
```

In [13]:
```
l1=[10,20,30,40,50,'Hi',True,1+2j,1234.454]
```

In [14]:
```
l1
```

Out[14]:
```
[10, 20, 30, 40, 50, 'Hi', True, (1+2j), 1234.454]
```

In [16]:
```
l1[0]
```

Out[16]:
```
10
```

In [17]:
```
l1[-1]
```

Out[17]:
```
1234.454
```

In [18]:
```
l1[1:6]
```

Out[18]:
```
[20, 30, 40, 50, 'Hi']
```

In [19]:
```
l1[:-1]
```

Out[19]:
```
[10, 20, 30, 40, 50, 'Hi', True, (1+2j)]
```

In [20]:
```
l1[2:-1]
```

Out[20]:
```
[30, 40, 50, 'Hi', True, (1+2j)]
```

In [ ]:
```

```

In [21]:
```
l1
```

Out[21]:
```
[10, 20, 30, 40, 50, 'Hi', True, (1+2j), 1234.454]
```

In [22]:
```
l1[-1:]
```

Out[22]:
```
[1234.454]
```

In [23]:
```
l1[::-1]
```

Out[23]:
```
[1234.454, (1+2j), True, 'Hi', 50, 40, 30, 20, 10]
```

In [24]:
```
l1[1::]
```

Out[24]:
```
[20, 30, 40, 50, 'Hi', True, (1+2j), 1234.454]
```

In [25]:
```
l2=['one', 'two', 'three']
```

In [26]:
```
l2
```

Out[26]:
```
['one', 'two', 'three']
```

In [27]:
```
l2.append("nine")
```

```
In [28]: l2
Out[28]: ['one', 'two', 'three', 'nine']
In [29]: l2.insert(3, 'four')
In [30]: l2
Out[30]: ['one', 'two', 'three', 'four', 'nine']
In [31]: l2.insert(1,'one')
In [32]: l2
Out[32]: ['one', 'one', 'two', 'three', 'four', 'nine']
In [33]: l2.remove('one')
In [34]: l2
Out[34]: ['one', 'two', 'three', 'four', 'nine']
In [35]: l2.pop()
Out[35]: 'nine'
In [36]: l2
Out[36]: ['one', 'two', 'three', 'four']
In [38]: l2.pop(3)
Out[38]: 'four'
In [39]: l2
Out[39]: ['one', 'two', 'three']
In [40]: del l1[2]
In [41]: l2
Out[41]: ['one', 'two', 'three']
In [42]: l1
Out[42]: [10, 20, 40, 50, 'Hi', True, (1+2j), 1234.454]
In [43]: del l1[2]
In [44]: l2
```

Out[44]: ['one', 'two', 'three']

In [45]:
```
l1
```

Out[45]: [10, 20, 50, 'Hi', True, (1+2j), 1234.454]

In [46]:
```
l1[2]=30
l1
```

Out[46]: [10, 20, 30, 'Hi', True, (1+2j), 1234.454]

In [47]:
```
l2
```

Out[47]: ['one', 'two', 'three']

In [48]:
```
l2.clear()
```

In [49]:
```
l2
```

Out[49]: []

In [50]:
```
del l2
```

In [51]:
```
l2
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[51], line 1
----> 1 l2

NameError: name 'l2' is not defined
```

In [54]:
```
l1
```

Out[54]: [10, 20, 30, 'Hi', True, (1+2j), 1234.454]

In [57]:
```
l2=l1
```

In [58]:
```
l2
```

Out[58]: [10, 20, 30, 'Hi', True, (1+2j), 1234.454]

In [59]:
```
id(l1) , id(l2)
```

Out[59]: (1770956042624, 1770956042624)

In [60]:
```
l3= l1.copy()
```

In [61]:
```
l3
```

Out[61]: [10, 20, 30, 'Hi', True, (1+2j), 1234.454]

In [62]:
```
id(l3)
```

Out[62]:
```
1770974451776
```

In [63]:
```
l1[0]=0
```

In [64]:
```
l1
```

Out[64]:
```
[0, 20, 30, 'Hi', True, (1+2j), 1234.454]
```

In [65]:
```
l2
```

Out[65]:
```
[0, 20, 30, 'Hi', True, (1+2j), 1234.454]
```

In [66]:
```
l3
```

Out[66]:
```
[10, 20, 30, 'Hi', True, (1+2j), 1234.454]
```

In [67]:
```
l1=l2+l3
```

In [68]:
```
l1
```

Out[68]:
```
[0,
 20,
 30,
 'Hi',
 True,
 (1+2j),
 1234.454,
 10,
 20,
 30,
 'Hi',
 True,
 (1+2j),
 1234.454]
```

In [70]:
```
l1.extend(l2)
```

In [71]:
```
l1
```

Out[71]:
```
[0,
 20,
 30,
 'Hi',
 True,
 (1+2j),
 1234.454,
 10,
 20,
 30,
 'Hi',
 True,
 (1+2j),
 1234.454,
 0,
 20,
 30,
 'Hi',
 True,
 (1+2j),
 1234.454]
```

In [72]:
```
l2
```

Out[72]:
```
[0, 20, 30, 'Hi', True, (1+2j), 1234.454]
```

In [76]:
```
1121 in l2
```

Out[76]:
```
False
```

In [77]:
```
0 in l1
```

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

In [78]:
```
if True in l1:
    print('True is present in L1')
else:
    print('True is not present in L1')
```

```
True is present in L1
```

In [81]:
```
if 'Australia'in l1:
    print('Australia is present in L1')
else:
    print('Australia is not present in L1')
```

```
Australia is not present in L1
```

In [82]:
```
l3
```

Out[82]:
```
[10, 20, 30, 'Hi', True, (1+2j), 1234.454]
```

In [83]:
```
l3.reverse()
```

In [84]:
```
l3
```

Out[84]: [1234.454, (1+2j), True, 'Hi', 30, 20, 10]

In [86]:
```
l3.reverse()
```

In [87]:
```
l3
```

Out[87]: [10, 20, 30, 'Hi', True, (1+2j), 1234.454]

In [88]:
```
l3 = l3[::-1]
```

In [89]:
```
l3
```

Out[89]: [1234.454, (1+2j), True, 'Hi', 30, 20, 10]

In [90]:
```
l3 = l3[1::]
```

In [91]:
```
l3
```

Out[91]: [(1+2j), True, 'Hi', 30, 20, 10]

In [92]:
```
l3 = l3[1:]
```

In [93]:
```
l3
```

Out[93]: [True, 'Hi', 30, 20, 10]

In [94]:
```
l3 = l3[1::]
```

In [95]:
```
l3
```

Out[95]: ['Hi', 30, 20, 10]

In [96]:
```
l3.sort()
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[96], line 1
----> 1 l3.sort()

TypeError: '<' not supported between instances of 'int' and 'str'
```

In [97]:
```
l3 = l3[1::]
```

In [98]:
```
l3
```

Out[98]: [30, 20, 10]

In [99]:
```
l3.sort()
```

In [100…
```
l3
```

Out[100… [10, 20, 30]

In [101…
```
l3.sort(reverse=True)
```

In [102…
```
l3
```

Out[102… [30, 20, 10]

In [103…
```
l3.sort()
```

In [104…
```
l3
```

Out[104… [10, 20, 30]

In [106…
```
l4=['o','q','s','a','p']
sorted(l4)
```

Out[106… ['a', 'o', 'p', 'q', 's']

In [107…
```
l4
```

Out[107… ['o', 'q', 's', 'a', 'p']

In [108…
```
l2
```

Out[108… [0, 20, 30, 'Hi', True, (1+2j), 1234.454]

In [109…
```
for i in l2:
    print(i)
```

```
0
20
30
Hi
True
(1+2j)
1234.454
```

In [111…
```
for i in enumerate(l2):
    print(i)
```

```
(0, 0)
(1, 20)
(2, 30)
(3, 'Hi')
(4, True)
(5, (1+2j))
(6, 1234.454)
```

In [112…
```
l1
```

Out[112…
```
[0,
 20,
 30,
 'Hi',
 True,
 (1+2j),
 1234.454,
 10,
 20,
 30,
 'Hi',
 True,
 (1+2j),
 1234.454,
 0,
 20,
 30,
 'Hi',
 True,
 (1+2j),
 1234.454]
```

In [114…
```
l1.count(20)
```

Out[114… 3

In [115…
```
l1.count(True)
```

Out[115… 3

In [116…
```
all(l1)
```

Out[116… False

In [117…
```
any(l1)
```

Out[117… True

In [118…
```
l4
```

Out[118… ['o', 'q', 's', 'a', 'p']

In [120…
```
l5=[0,0]
```

In [121…
```
all(l5)
```

Out[121… False

In [122…
```
l6=[1,3,5]
```

In [123…
```
l6
```

Out[123… [1, 3, 5]

In [124…
```
all(l6)
```
Out[124…
```
True
```
In [125…
```
any(l6)
```
Out[125…
```
True
```

Tuple

In [126…
```
t1=(1,2,3,4,5)
```
In [127…
```
t1
```
Out[127…
```
(1, 2, 3, 4, 5)
```
In [128…
```
type(t1)
```
Out[128…
```
tuple
```
In [130…
```
t2=(1,2,'hi',True,1+2j,[1,2],[3,4],(1,5),('s','d'))
t2
```
Out[130…
```
(1, 2, 'hi', True, (1+2j), [1, 2], [3, 4], (1, 5), ('s', 'd'))
```
In [131…
```
len(t2)
```
Out[131…
```
9
```
In [132…
```
len(t1)
```
Out[132…
```
5
```
In [134…
```
t2[0]
```
Out[134…
```
1
```
In [135…
```
t2
```
Out[135…
```
(1, 2, 'hi', True, (1+2j), [1, 2], [3, 4], (1, 5), ('s', 'd'))
```
In [140…
```
t2[2][0]
```
Out[140…
```
'h'
```
In [141…
```
t2[-1]
```
Out[141…
```
('s', 'd')
```

In [142…
```
t2[:3]
```

Out[142…
```
(1, 2, 'hi')
```

In [143…
```
t2[:-2]
```

Out[143…
```
(1, 2, 'hi', True, (1+2j), [1, 2], [3, 4])
```

In [144…
```
t2[-3:]
```

Out[144…
```
([3, 4], (1, 5), ('s', 'd'))
```

In [145…
```
t2[:]
```

Out[145…
```
(1, 2, 'hi', True, (1+2j), [1, 2], [3, 4], (1, 5), ('s', 'd'))
```

In [146…
```
l1
```

Out[146…
```
[0,
 20,
 30,
 'Hi',
 True,
 (1+2j),
 1234.454,
 10,
 20,
 30,
 'Hi',
 True,
 (1+2j),
 1234.454,
 0,
 20,
 30,
 'Hi',
 True,
 (1+2j),
 1234.454]
```

In [147…
```
l2
```

Out[147…
```
[0, 20, 30, 'Hi', True, (1+2j), 1234.454]
```

In [148…
```
t1
```

Out[148…
```
(1, 2, 3, 4, 5)
```

In [149…
```
t2
```

Out[149…
```
(1, 2, 'hi', True, (1+2j), [1, 2], [3, 4], (1, 5), ('s', 'd'))
```

In [151…
```
t3=t1
```

In [152…
```
t3
```

Out[152…
```
(1, 2, 3, 4, 5)
```

In [153…
```
t3[0]=7
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[153], line 1
----> 1 t3[0]=7

TypeError: 'tuple' object does not support item assignment
```

In [154…
```
del t3[1]
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
Cell In[154], line 1
----> 1 del t3[1]

TypeError: 'tuple' object doesn't support item deletion
```

In [155…
```
del t3 # deleting entire tuple is possible
```

In [156…
```
t3
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Cell In[156], line 1
----> 1 t3

NameError: name 't3' is not defined
```

In [157…
```
t2
```

Out[157…
```
(1, 2, 'hi', True, (1+2j), [1, 2], [3, 4], (1, 5), ('s', 'd'))
```

In [158…
```
for i in t2:
    print(i)
```

```
1
2
hi
True
(1+2j)
[1, 2]
[3, 4]
(1, 5)
('s', 'd')
```

In [159…
```
for i in enumerate(t2):
    print(i)
```

```
(0, 1)
(1, 2)
(2, 'hi')
(3, True)
(4, (1+2j))
(5, [1, 2])
(6, [3, 4])
(7, (1, 5))
(8, ('s', 'd'))
```

In [160…
```
l2
```

Out[160…
```
[0, 20, 30, 'Hi', True, (1+2j), 1234.454]
```

In [161…
```
'Hi' in t2
```

Out[161…
```
False
```

In [162…
```
t2
```

Out[162…
```
(1, 2, 'hi', True, (1+2j), [1, 2], [3, 4], (1, 5), ('s', 'd'))
```

In [163…
```
type(t2)
```

Out[163…
```
tuple
```

In [164…
```
type(l2)
```

Out[164…
```
list
```

In [165…
```
'hi' in t2
```

Out[165…
```
True
```

In [166…
```
t2
```

Out[166…
```
(1, 2, 'hi', True, (1+2j), [1, 2], [3, 4], (1, 5), ('s', 'd'))
```

In [169…
```
if [1, 2] in t2:
    print('it is present')
else:
    print('it is not present')
```

```
it is present
```

In [178…
```
t2.index(True)
```

Out[178…
```
0
```

In [179…
```
t2
```

Out[179…
```
(1, 2, 'hi', True, (1+2j), [1, 2], [3, 4], (1, 5), ('s', 'd'))
```

In [182…
```
t2.index([1, 2])
```

Out[182…
```
5
```

In [181…
```
t2.index(True)
```

Out[181…
```
0
```

In [184…
```
if False in t2:
    print('it is present')
else:
    print('it is not')
```

```
it is not
```

In [185…
```
t1
```

Out[185…
```
(1, 2, 3, 4, 5)
```

In [187…
```
t3=t1
```

In [188…
```
t3
```

Out[188…
```
(1, 2, 3, 4, 5)
```

In [190…
```
t4=(2,5,65,6,33)
sorted(t4)
```

Out[190…
```
[2, 5, 6, 33, 65]
```

In [197…
```
sorted(t4, reverse=True)
```

Out[197…
```
[65, 33, 6, 5, 2]
```

In [ ]: