

# TUPLE

tuple is denoted by square bracket()

tuple can allow all the data types

tuple is immutable ¶

duplicates are allowed in tuple

we can assign more than one value to the tuple

we can perform two functions

count

index

```
In [1]:  ▶ t=()  
t
```

```
Out[1]:  ()
```

```
In [2]:  ▶ type(t)
```

```
Out[2]:  tuple
```

```
In [3]:  ▶ t=(10,20,30)  
t
```

```
Out[3]:  (10, 20, 30)
```

```
In [4]:  ▶ len(t)
```

```
Out[4]:  3
```

```
In [5]:  ▶ t1=(10,2.3,'nit',1+2j,True,10,20)  
t1
```

```
Out[5]:  (10, 2.3, 'nit', (1+2j), True, 10, 20)
```

```
In [6]:  print(t)
         print(t1)
```

```
(10, 20, 30)
(10, 2.3, 'nit', (1+2j), True, 10, 20)
```

```
In [7]:  t.count(10)
```

```
Out[7]: 1
```

```
In [8]:  t1.count(10)
```

```
Out[8]: 2
```

```
In [9]:  t1.count(2.3)
```

```
Out[9]: 1
```

```
In [10]: t1
```

```
Out[10]: (10, 2.3, 'nit', (1+2j), True, 10, 20)
```

```
In [11]: t1.index(2.3)
```

```
Out[11]: 1
```

```
In [12]: t
```

```
Out[12]: (10, 20, 30)
```

```
In [13]: t
```

```
Out[13]: (10, 20, 30)
```

```
In [14]: t[0]
```

```
Out[14]: 10
```

```
In [15]: t[0]=100
```

```
--
```

```
TypeError
```

```
Traceback (most recent call las
```

```
t)
```

```
<ipython-input-15-3382f43ca263> in <module>
```

```
----> 1 t[0]=100
```

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

**tuple object does not support item assignment means it is immutable tuple cannot be changed**

```
In [16]: t
```

```
Out[16]: (10, 20, 30)
```

```
In [17]: t2=t*3
```

```
In [18]: t2
```

```
Out[18]: (10, 20, 30, 10, 20, 30, 10, 20, 30)
```

```
In [19]: t1
```

```
Out[19]: (10, 2.3, 'nit', (1+2j), True, 10, 20)
```

```
In [20]: t1[0:6]
```

```
Out[20]: (10, 2.3, 'nit', (1+2j), True, 10)
```

```
In [21]: t1[1:4]
```

```
Out[21]: (2.3, 'nit', (1+2j))
```

```
In [22]: t2
```

```
Out[22]: (10, 20, 30, 10, 20, 30, 10, 20, 30)
```

```
In [23]: t2[1:6:2]
```

```
Out[23]: (20, 10, 30)
```

```
In [24]: t2
```

```
Out[24]: (10, 20, 30, 10, 20, 30, 10, 20, 30)
```

```
In [25]: t2[0:8:1]
```

```
Out[25]: (10, 20, 30, 10, 20, 30, 10, 20)
```

```
In [26]: t2[0:9:1]
```

```
Out[26]: (10, 20, 30, 10, 20, 30, 10, 20, 30)
```

```
In [27]: t1
```

```
Out[27]: (10, 2.3, 'nit', (1+2j), True, 10, 20)
```

In [28]: `t1[:]`

Out[28]: (10, 2.3, 'nit', (1+2j), True, 10, 20)

In [29]: `t1`

Out[29]: (10, 2.3, 'nit', (1+2j), True, 10, 20)

In [30]: `t1[-1:]`

Out[30]: (20,)

In [31]: `t1[-2:]`

Out[31]: (10, 20)

In [32]: `t1[:-1]`

Out[32]: (10, 2.3, 'nit', (1+2j), True, 10)

In [33]: `t1[:-2]`

Out[33]: (10, 2.3, 'nit', (1+2j), True)

In [34]: `t1[::-1]`

Out[34]: (20, 10, True, (1+2j), 'nit', 2.3, 10)

In [35]: `t1[::-2]`

Out[35]: (20, True, 'nit', 10)

In [36]: `t1[::-3]`

Out[36]: (20, (1+2j), 10)

In [37]: `t=(1,2,3,4,5,6,7,8)`  
`t`

Out[37]: (1, 2, 3, 4, 5, 6, 7, 8)

In [38]: `t[3:-1]`

Out[38]: (4, 5, 6, 7)

In [39]: `t`

Out[39]: (1, 2, 3, 4, 5, 6, 7, 8)

```
In [40]: t[1:4]
```

```
Out[40]: (2, 3, 4)
```

```
In [41]: t
```

```
Out[41]: (1, 2, 3, 4, 5, 6, 7, 8)
```

```
In [42]: t[1:]
```

```
Out[42]: (2, 3, 4, 5, 6, 7, 8)
```

```
In [43]: t
```

```
Out[43]: (1, 2, 3, 4, 5, 6, 7, 8)
```

```
In [44]: t[:5]
```

```
Out[44]: (1, 2, 3, 4, 5)
```

```
In [45]: t
```

```
Out[45]: (1, 2, 3, 4, 5, 6, 7, 8)
```

```
In [46]: t[1:7:3]
```

```
Out[46]: (2, 5)
```

```
In [47]: t
```

```
Out[47]: (1, 2, 3, 4, 5, 6, 7, 8)
```

```
In [48]: t[::-1]
```

```
Out[48]: (8, 7, 6, 5, 4, 3, 2, 1)
```

```
In [49]: t[::1]
```

```
Out[49]: (1, 2, 3, 4, 5, 6, 7, 8)
```

```
In [50]: t
```

```
Out[50]: (1, 2, 3, 4, 5, 6, 7, 8)
```

```
In [51]: t[::-3]
```

```
Out[51]: (8, 5, 2)
```

```
In [52]: ❏ for i in t:  
          print(i)
```

```
1  
2  
3  
4  
5  
6  
7  
8
```

```
In [53]: ❏ for i in enumerate(t1):  
          print(i)
```

```
(0, 10)  
(1, 2.3)  
(2, 'nit')  
(3, (1+2j))  
(4, True)  
(5, 10)  
(6, 20)
```

```
In [54]: ❏ range()
```

```
-----  
--  
TypeError                                Traceback (most recent call las  
t)  
<ipython-input-54-5bcbe005bf48> in <module>  
----> 1 range()  
  
TypeError: range expected 1 argument, got 0
```

**range should not be empty we have to pass up to three arguments value**

```
In [55]: ❏ range(5)
```

```
Out[55]: range(0, 5)
```

```
In [56]: ❏ list(range(5))
```

```
Out[56]: [0, 1, 2, 3, 4]
```

```
In [57]: ❏ range(10,20)
```

```
Out[57]: range(10, 20)
```

```
In [58]: ▶ list(range(10,20))
```

```
Out[58]: [10, 11, 12, 13, 14, 15, 16, 17, 18, 19]
```

```
In [59]: ▶ range(10,50,5)
```

```
Out[59]: range(10, 50, 5)
```

```
In [60]: ▶ list(range(10,50,5))
```

```
Out[60]: [10, 15, 20, 25, 30, 35, 40, 45]
```

```
In [61]: ▶ range(10,50,5,1)
```

```
-----  
--  
TypeError                                Traceback (most recent call last)  
  <ipython-input-61-64d8c3cd5c46> in <module>  
----> 1 range(10,50,5,1)  
  
TypeError: range expected at most 3 arguments, got 4
```

we cannot pass more than three arguments

## Tuple creation

```
In [62]: ▶ tup1=()
```

```
In [63]: ▶ tup2=(10,30,60)           # tuple of integers
```

```
In [64]: ▶ tup3=(10.77,30.66,60.89)  # tuple of float
```

```
In [65]: ▶ tup4=('one','two','three') #tuple of strings
```

```
In [66]: ▶ tup5=('akshitha',25,(50,100),(150,90)) #nested tuples
```

```
In [67]: ▶ tup6=(100,'akshitha',17.65) # tuple with mixed data types
```

```
In [68]: ▶ tup7=('akshitha',21,50.6,1+2j,(19,2),{'kts','sri'})
```

```
In [69]: ➤ len(tup7)
```

```
Out[69]: 6
```

## tuple indexing

```
In [70]: ➤ tup2[0]
```

```
Out[70]: 10
```

```
In [71]: ➤ tup4[0]
```

```
Out[71]: 'one'
```

```
In [72]: ➤ tup4[-1]
```

```
Out[72]: 'three'
```

```
In [73]: ➤ tup4[0][0]    # nested indexing access the first character of the first tuple
```

```
Out[73]: 'o'
```

## tuple slicing

```
In [92]: ➤ mytuple=('one','two','three','four','five','six','seven','eight')
```

```
In [75]: ➤ mytuple[0:3]
```

```
Out[75]: ('one', 'two', 'three')
```

```
In [76]: ➤ mytuple[2:5]
```

```
Out[76]: ('three', 'four', 'five')
```

```
In [77]: ➤ mytuple[:2]
```

```
Out[77]: ('one', 'two')
```

```
In [78]: ➤ mytuple[:3]
```

```
Out[78]: ('one', 'two', 'three')
```

```
In [79]: ➤ mytuple[:2]
```

```
Out[79]: ('one', 'two')
```



```
In [80]: mytuple[-3:]
```

```
Out[80]: ('six', 'seven', 'eight')
```

```
In [81]: mytuple[:-3]
```

```
Out[81]: ('one', 'two', 'three', 'four', 'five')
```

```
In [82]: mytuple[-2:]
```

```
Out[82]: ('seven', 'eight')
```

```
In [83]: mytuple
```

```
Out[83]: ('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

```
In [84]: mytuple[:-1]
```

```
Out[84]: ('one', 'two', 'three', 'four', 'five', 'six', 'seven')
```

## Remove & change items

```
In [86]: mytuple
```

```
Out[86]: ('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

```
In [87]: del mytuple[0]
```

```
-----  
--  
TypeError                                Traceback (most recent call last)  
  <ipython-input-87-667a276aa503> in <module>  
    ----> 1 del mytuple[0]
```

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

**tuple is immutable so we cannot delete the tuple items**

```
In [88]: mytuple[0]=1
```

```
-----  
--  
TypeError                                Traceback (most recent call las  
t)  
<ipython-input-88-bce2db940981> in <module>  
----> 1 mytuple[0]=1  
  
TypeError: 'tuple' object does not support item assignment
```

**tuple is immutable so we cannot change tuple items**

```
In [89]: del mytuple      #deleting entire tuple is possible
```

```
In [90]: mytuple()
```

```
-----  
--  
NameError                                Traceback (most recent call las  
t)  
<ipython-input-90-60caedb4f3f0> in <module>  
----> 1 mytuple()  
  
NameError: name 'mytuple' is not defined
```

## loop through a tuple

```
In [93]: mytuple
```

```
Out[93]: ('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

```
In [94]: for i in mytuple:  
         print(i)
```

```
one  
two  
three  
four  
five  
six  
seven  
eight
```

```
In [95]: ➤ for i in enumerate(mytuple):  
          print(i)
```

```
(0, 'one')  
(1, 'two')  
(2, 'three')  
(3, 'four')  
(4, 'five')  
(5, 'six')  
(6, 'seven')  
(7, 'eight')
```

## True membership

```
In [103]: ➤ mytuple
```

```
Out[103]: ('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

```
In [98]: ➤ 'one' in mytuple
```

```
Out[98]: True
```

```
In [99]: ➤ 'ten' in mytuple
```

```
Out[99]: False
```

```
In [100]: ➤ if 'three' in mytuple:  
            print('Three is present in the tuple')  
            else:  
                print('Three is not present in the tuple')
```

```
Three is present in the tuple
```

```
In [102]: ➤ if 'eleven' in mytuple:  
            print('eleven is present in the tuple')  
            else:  
                print('eleven is not present in the tuple')
```

```
eleven is not present in the tuple
```

## index position

```
In [104]: ➤ mytuple
```

```
Out[104]: ('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

```
In [105]: mytuple.index('one')
```

```
Out[105]: 0
```

```
In [106]: mytuple.index('five')
```

```
Out[106]: 4
```

```
In [108]: mytuple
```

```
Out[108]: ('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight')
```

```
In [ ]:
```