30th January

```
t=()
t
```

```
()
```

```
type(t)
```

```
tuple
```

```
a10=5
type(a10)
```

```
int
```

```
t1=3,4,5,6,7
type(t1)
```

```
tuple
```

```
t2=(10,20,30)
```

```
Start coding or generate with AI.
```

```
t2
```

```
(10, 20, 30)
```

```
t1
```

```
(3, 4, 5, 6, 7)
```

```
t1.count(5)
```

```
1
```

```
t.appended(10)
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-7-a9c4020d4b24> in <cell line: 0>()
----> 1 t.appended(10)

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

Archana Kapu
Feb 1, 2025

append is not applicable for tuple

```
t1
```

```
(3, 4, 5, 6, 7)
```

```
t1[0]=10
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-9-c7bad9d4135c> in <cell line: 0>()
----> 1 t1[0]=10

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

Archana Kapu
Feb 1, 2025

tuple is immutable and nothasshable

```
hdfc=(1234,'archana','DHT2A')
```

```
hdfc[1]='Anitha'
```

Archana Kapu
Feb 1, 2025

tuple is immutable

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-11-a2a2b43f30a4> in <cell line: 0>()
----> 1 hdfc[1]='Anitha'

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

```
t1
```

```
(3, 4, 5, 6, 7)
```

```
t1*3
```

```
(3, 4, 5, 6, 7, 3, 4, 5, 6, 7, 3, 4, 5, 6, 7)
```

Archana Kapu
Feb 1, 2025

represents index value of mentioned element value

```
t1.index(5)
```

```
2
```

```
t1[:]
```

```
(3, 4, 5, 6, 7)
```

```
t1[2:]
```

```
(5, 6, 7)
```

```
t1[:10]
```

```
(3, 4, 5, 6, 7)
```

Archana Kapu
Feb 1, 2025

Index error

```
t1[10]
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-18-4e90c3f6fed8> in <cell line: 0>()
----> 1 t1[10]

IndexError: tuple index out of range
```

```
for i in t1:
  print(i)
```

```
3
4
5
6
7
```

```
for i in enumerate (t1):
  print(i)
```

```
(0, 3)
(1, 4)
(2, 5)
(3, 6)
(4, 7)
```

Tuple is immutable/unhashable

indexing and slicing is allowed

multiple data types are allowed.

tuple supports only 2 functions i.e;count,index

Archana Kapu
Feb 1, 2025

multiple data types are allowed

```
t6=('Archana',2,1+2j)
```

```
type(t6)
```

```
tuple
```

```
t6[0]
```

```
'Archana'
```

```
t6[0][0]
```

```
'r'
```

```
print(t6[0][0])
print(t6[0][1])
print(t6[0][2])
print(t6[0][3])
print(t6[0][4])
print(t6[0][5])
print(t6[0][6])
```

```
A
r
c
h
a
n
a
```

```
print(t6[0][0])
print(t6[0][1])
print(t6[0][2])
print(t6[0][3])
print(t6[0][4])
print(t6[0][5])
print(t6[0][6])
print(t6[0][9])
```

```
A
r
c
h
a
n
a
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-37-890bc243fbe8> in <cell line: 0>()
      6 print(t6[0][5])
      7 print(t6[0][6])
----> 8 print(t6[0][9])

IndexError: string index out of range
```

```
print(t6[0][9])
print(t6[0][0])
print(t6[0][1])
print(t6[0][2])
print(t6[0][3])
print(t6[0][4])
print(t6[0][5])
print(t6[0][6])
```

```
---------------------------------------------------------------------------
IndexError                                Traceback (most recent call last)
<ipython-input-38-c4a8c4a14ba7> in <cell line: 0>()
----> 1 print(t6[0][9])
      2 print(t6[0][0])
      3 print(t6[0][1])
      4 print(t6[0][2])
      5 print(t6[0][3])

IndexError: string index out of range
```

to encounter this prblm, need to learn Exceptinal handling--try except finally

tuple is completed.

bitwise operators

```
6 Operators

compliment(~)

And(&)

or(|)

Xor(^)

left shift(<<)

right shift(>>)
```

```
12 & 13
```

```
12
```

```
12 | 13
```

```
13
```

```
~0
```

```
-1
```

```
35 & 40
```

```
32
```

```
~36
```

```
-37
```

```
~1
```

```
-2
```

```
12 ^ 13
```

```
1
```

```
print(bin(12))
print(bin(13))
```

```
0b1100
0b1101
```

```
print(type(str(123)))
```

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

```
 print(10 + '5')
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-48-2a425b2115a6> in <cell line: 0>()
----> 1 print(10 + '5')

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

```
print("Hello" + str(123))
```

```
Hello123
```

```
print("Hello", end="")
```

```
Hello
```

```
print(type(1/2))
```

```
<class 'float'>
```

```
len([1, 2, 3, 4])
```

```
4
```

```
[1, 2, 3] + [4, 5, 6]
```

```
[1, 2, 3, 4, 5, 6]
```

```
(1, 2) + (3, 4)
```

```
(1, 2, 3, 4)
```

```
t6
```

```
('Archana', 2, (1+2j))
```

```
t7=([1,2])
```

```
t7
```

```
[1, 2]
```

```
t8=([1,2],4,'a')
```

```
t8
```

```
([1, 2], 4, 'a')
```

```
print(list((1,2,3)))
```

```
[1, 2, 3]
```

```
tr=('a'=10)
tr
```

```
  File "<ipython-input-66-e88420cead9b>", line 1
    tr=('a'=10)
        ^
SyntaxError: cannot assign to literal here. Maybe you meant '==' instead of '='?
```

```
Start coding or generate with AI.
```

```
my_tuple = (1)
```

```
my_tuple
```

```
1
```

```
(1, 2) + (3, 4)
```

```
(1, 2, 3, 4)
```

```
my_list = [1, 2, 2, 3];
```

```
my_list.remove(2);
```

```
print(my_list)
```

```
[1, 2, 3]
```

Tuple task assignment

Tuples

1. Tuple is similar to List except that the objects in tuple are immutable which means we cannot change the elements of a tuple once assigned.
2. When we do not want to change the data over time, tuple is a preferred data type.
3. Iterating over the elements of a tuple is faster compared to iterating over a list.

Tuple Creation

```
t=()
```

```
type(t)
```

```
tuple
```

```
tup2 = (10,30,60) # tuple of integers numbers
```

```
tup2
```

```
(10, 30, 60)
```

```
tup3 = (10.77,30.66,60.89) # tuple of float numbers
```

```
tup3
```

```
(10.77, 30.66, 60.89)
```

Archana Kapu
4:13 PM Today
nested tuple

```
tup4 = ('one','two' , "three") # tuple of strings
```

```
tup4
```

```
('one', 'two', 'three')
```

```
tup5 = ('Archana', 25 ,(50, 100),(150, 90)) # Nested tuples
```

```
tup5
```

```
('Archana', 25, (50, 100), (150, 90))
```

Archana Kapu
4:15 PM Today
tuple with combination of datatypes & datatructures(list,dict,tuple)

```
tup6 = (100, 'Archana', 17.765) # Tuple of mixed data types
```

```
tup6
```

```
(100, 'Archana', 17.765)
```

```
tup7 = ('Archana', 25 ,[50, 100],[150, 90] , {'Harish' , 'Sony'} , (99,22,33))
```

```
tup7
```

```
('Archana', 25, [50, 100], [150, 90], {'Harish', 'Sony'}, (99, 22, 33))
```

```
len(tup7) #Length of list
```

```
6
```

Tuple Indexing

```
tup2[0] # Retreive first element of the tuple
```

```
10
```

```
tup4[0] # Retreive first element of the tuple
```

```
'one'
```

```
tup4
```

```
('one', 'two', 'three')
```

```
tup4[0][0] # Nested indexing - Access the first character of the first tuple element
```

```
'o'
```

```
tup4[-1] # Last item of the tuple
```

```
'three'
```

```
tup5[-1] # Last item of the tuple
```

```
(150, 90)
```

Tuple Slicing

```
mytuple = ('one' , 'two' , 'three' , 'four' , 'five' , 'six' , 'seven' , 'eight','nine','ten')
```

```
mytuple[0:3]# Return all items from 0th to 3-1=2nd index
```

```
('one', 'two', 'three')
```

```
mytuple[2:5] # List all items from 2nd to 5th index location excluding the item
```

```
('three', 'four', 'five')
```

```
mytuple[:3] # Return first three items(3-1=2nd index)
```

```
('one', 'two', 'three')
```

```
mytuple[:2] # Return first two items
```

```
('one', 'two')
```

```
mytuple[-3:] # Return last three items
```

```
('eight', 'nine', 'ten')
```

```
mytuple[-2:] # Return last two items
```

```
('nine', 'ten')
```

```
mytuple[-1] # Return last item of the tuple
```

```
'ten'
```

```
mytuple[:] # Return whole tuple
```

```
('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight', 'nine', 'ten')
```

Archana Kapu
4:26 PM Today

tuple are immutable that means we can not delete tuple items

Remove & Change Items

```
mytuple
```

```
('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight', 'nine', 'ten')
```

Archana Kapu
4:27 PM Today

Tuples are immutable which means we can't CHANGE tuple items

```
del mytuple[0]#tuple are immutable that means we can not delete tuple items
```

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-34-667a276aa503> in <cell line: 0>()
----> 1 del mytuple[0]

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

```
mytuple[0] = 1 # Tuples are immutable which means we can't CHANGE tuple items
```

Archana Kapu
4:28 PM Today

name error , as we deleted tuple and trying to call

```
---------------------------------------------------------------------------
TypeError                                 Traceback (most recent call last)
<ipython-input-35-4c2ed09725a9> in <cell line: 0>()
----> 1 mytuple[0] = 1 # Tuples are immutable which means we can't CHANGE tuple items

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

```
del mytuple # Deleting entire tuple object is possible
```

```
mytuple
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
<ipython-input-37-c6c21778968d> in <cell line: 0>()
----> 1 mytuple

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

Loop through a tuple

```
mytuple=('one' , 'two' , 'three' , 'four' , 'five' , 'six' , 'seven' , 'eight','nine','ten')
```

```
mytuple
```

```
('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight', 'nine', 'ten')
```

```
for i in mytuple:
    print(i)
```

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

```
for i in enumerate (mytuple):
  print(i)
```

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

Count

```
mytuple1 =('one', 'two', 'three', 'four', 'one', 'one', 'two', 'three')
```

```
mytuple1.count('one')#retun no of occurences of an elemnet
```

```
3
```

```
mytuple1.count('two')#retun no of occurences of an elemnet
```

```
2
```

```
mytuple1.count('four')#retun no of occurences of an elemnet
```

```
1
```

Tuple Membership

```
mytuple
```

```
('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight', 'nine', 'ten')
```

```
'two' in mytuple# Check if 'two' exist in the list
```

```
True
```

```
'eleven' in mytuple# Check if 'eleven' exist in the list
```

```
False
```

```
if 'three' in mytuple:# Check if 'two' exist in the list
  print('theree present in mytuple')
else:
  print('theree not present in mytuple')
```

```
theree present in mytuple
```

```
if 'eleven' in mytuple: # Check if 'eleven' exist in the list
    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

```
mytuple
```

```
('one', 'two', 'three', 'four', 'five', 'six', 'seven', 'eight', 'nine', 'ten')
```

```
mytuple.index('one')
```

```
0
```

```
mytuple.index('five') # Index of first element equal to 'five'
```

```
4
```

```
mytuple1
```

```
('one', 'two', 'three', 'four', 'one', 'one', 'two', 'three')
```

```
mytuple1.index('one')
```

```
0
```

Archana Kapu
4:45 PM Today

Sort in descending order

Sorting

```
mytuple2 = (43,67,99,12,6,90,67)
```

```
sorted(mytuple2)# Returns a new sorted list and doesn't change original tuple
```

```
[6, 12, 43, 67, 67, 90, 99]
```

```
sorted(mytuple2,reverse=True)# Returns a new sorted list in descending order and doesn't chang
```

```
[99, 90, 67, 67, 43, 12, 6]
```