Tuple

- Is immutable
  - tuple does not support item assignment
- is faster
- no write operations are supported, only read operations are allowed (tuple's are readonly data type)

Creating tuples

In [1]:
```
# homogenous tuple
t1 = (1,2,3,4)

# heterogenous tuple
t2 = (1,'two',3,'four')
```

In [2]:
```
t1
```

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

In [3]:
```
t2
```

Out[3]:
```
(1, 'two', 3, 'four')
```

tuple with only 1 element

In [2]:
```
# this is a wring syntex
t3 = (1)
type(t3)
```

Out[2]:
```
int
```

In [4]:
```
t4 = ('int')
type(t4)
```

Out[4]:
```
str
```

In [ ]:
```

```

In [3]:
```
t3 = (1,)
type(t3)
```

Out[3]:
```
tuple
```

In [5]:
```
t4 = ('int',)
type(t4)
```

Out[5]:
```
tuple
```

In [ ]:
```

```

In [6]:
```
t4 = (1,2,3,(4,5))
t4
```

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

In [ ]:
```

```

Using type conversion for the creation of the tuples

In [7]:
```
t5 = tuple('go')
type(t5)
```

Out[7]:
```
tuple
```

In [8]:
```
t5
```

Out[8]:
```
('g', 'o')
```

In [ ]:
```

```

Creating tuple with multiple items

In [9]:
```
t1 = tuple([1,2,3,4])
t1
```

Out[9]:
```
(1, 2, 3, 4)
```

In [12]:
```
t2 = tuple((1,2,3,4))
t2
```

Out[12]:
```
(1, 2, 3, 4)
```

Creating list with multiple items

In [10]:
```
l1 = list([1,2,3,4])
l1
```

Out[10]:
```
[1, 2, 3, 4]
```

In [11]:
```
l2 = list((1,2,3))
l2
```

Out[11]:
```
[1, 2, 3]
```

In [ ]:
```

```

Deleting tuples

In [13]:
```
t1 = (12,3,4)
t1
```

Out[13]:
```
(12, 3, 4)
```

In [14]:
```
del t1
```

In [15]:
```
t1
```

```
---------------------------------------------------------------------------
NameError                                 Traceback (most recent call last)
Input In [15], in <module>
----> 1 t1

NameError: name 't1' is not defined
```

In [ ]:
```

```

Operations on tuple

- concatenate
- multiply
- loop & iterate
- use membership operators

all the opern are same as list

even the fucntn are all same as list

In [16]:
```
t1 = (12,3,4)
t2 = (4,5,6)
t1+t2
```

Out[16]:
```
(12, 3, 4, 4, 5, 6)
```

In [17]:
```
t1*2
```

Out[17]:
```
(12, 3, 4, 12, 3, 4)
```

In [ ]:
```

```