

TUPLES

- Similar to a **list** in that it contains a collection of elements in ().
- Elements in a **tuple** are immutable.
- Only has two functions.

In [3]:

```
type(())
```

Out[3]:

tuple

In [4]:

```
tup1 = (40, 50, 60, 70)
tup1
type(tup1)
```

Out[4]:

tuple

In [5]:

```
mylist = [1,2,3,4]
mylist
type(tuple(mylist))
```

Out[5]:

tuple

In [6]:

```
list(tup1)
```

Out[6]:

[40, 50, 60, 70]

In [7]:

```
tup2 = ((10, 10, 10, 10, 2, 2, 2, 3),
        ("cat", "dog", "bird"), tup1)
tup2
```

Out[7]:

((10, 10, 10, 10, 2, 2, 2, 3), ('cat', 'dog', 'bird'), (40, 50, 60, 70))

In [8]:

```
tup2.count(2)
```

```
tup2[0].count(2)
```

Out[8]:

3

In [9]:

```
tup2[1].index("bird")
tup2[1][2]
```

Out[9]:

'bird'

In [10]:

```
tup3 = tuple(range(0,21, 2))
tup3
```

Out[10]:

(0, 2, 4, 6, 8, 10, 12, 14, 16, 18, 20)

In [16]:

```
mylist = [10, 20, 30, 40]
mylist[3] = "hello"
mylist
```

Out[16]:

[10, 20, 30, 'hello']

In [20]:

```
# TUPLES ARE IMMUTABLE
tup2[2][3] = "hello"
```

TypeError Traceback (most recent call last)

<ipython-input-20-c33b03500e68> in <module>()

----> 1 tup2[2][3] = "hello"

TypeError: 'tuple' object does not support item assignment

In [1]:

```
tup4 = 1,2,3,4, False, True, True
tup4
```

Out[1]:

(1, 2, 3, 4, False, True, True)

In [2]:

```
tup5 = tup4, True, False, 100, 9.1245, "string"
tup5
```

Out[2]:

((1, 2, 3, 4, False, True, True), True, False, 100, 9.1245, 'string')

In [3]:

```
sum(tup4)
```

Out[3]:

12

In [6]:

```
tup4.index(0)  
tup4.count(1)
```

Out[6]:

3