IT Workshop - Python

By - Aswin Barath

USN - 18BTRSE031

#1- Immutable

In [1]:

```
# Integer data type
i1 = 10
i2 = 10
i1 is i2
```

Out[1]:

```
True
```

In [2]:

```
i3 = 257
i4 = 257
i3 is i4
```

Out[2]:

```
False
```

In [3]:

```
# Float data type
f1 = 10.5
f2 = 10.5
f1 is f2
```

Out[3]:

```
False
```

In [4]:

```
# Boolean data type
b1 = True
b2 = True
b1 is b2
```

Out[4]:

```
True
```

In [5]:

```
b3 = False
b4 = False
b3 is b4
```

Out[5]:

```
True
```

In [6]:

```
# String data type
s1 = 'aswin'
s2 = 'aswin'
s1 is s2
```

Out[6]:

```
True
```

In [7]:

```
# Complex data type
c1 = 10+2j
c2 = 10+2j
c1 is c2
```

Out[7]:

```
False
```

#2 - Bytes and Bytearray

Bytes and Bytearray objects contain single bytes.

Bytes is immutable sequence.

Bytes objects can be constructed by the constructor, bytes() and also from literals by using b as prefix with normal string.

In [8]:

```
a = b"This is a byte object"
print(a)
```

```
b'This is a byte object'
```

In [9]:

```
type(a)
```

Out[9]:

```
bytes
```

In [11]:

```
c = b'''Python bytes object,
This is part of my Assignment,
Hope that it is correct'''
print(c)
```

```
b'Python bytes object,\nThis is part of my Assignment,\nHope that it
is correct'
```

In [12]:

```
d = bytes('Python, bytes object', 'utf8')
print(d)
```

```
b'Python, bytes object'
```

Bytearray is mutable sequence.

Bytearrays can be constructed by the bytearray() function.

In [14]:

```
p = bytearray(b"Python Bytes")
print(p)
```

```
bytearray(b'Python Bytes')
```

In [16]:

```
q = bytearray([10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20])
print(q)
```

```
bytearray(b'\n\x0b\x0c\r\x0e\x0f\x10\x11\x12\x13\x14')
```

In [18]:

```
type(q)
```

Out[18]:

```
bytearray
```

In [17]:

```
q.append(21)
q
```

Out[17]:

```
bytearray(b'\n\x0b\x0c\r\x0e\x0f\x10\x11\x12\x13\x14\x15')
```

#3 - List Slicing

In [19]:

```
L = []
type(L)
```

Out[19]:

```
list
```

In [21]:

```
L = [1,2,3,4,5]
print(L)
```

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

In [24]:

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

```
1
2
3
4
5
```

In [25]:

```
L.append(6)
```

In [26]:

```
L
```

Out[26]:

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

In [27]:

```
L.remove(3)
L
```

Out[27]:

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

In [28]:

```
L[1:3]
```

Out[28]:

```
[2, 4]
```

In [29]:

```
L[:3]
```

Out[29]:

```
[1, 2, 4]
```

In [30]:

```
L[2:]
```

Out[30]:

```
[4, 5, 6]
```

In [31]:

```
L[:]
```

Out[31]:

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

#4 - range() function

In [36]:

```
for i in range(10):
    print(i)
```

```
0
1
2
3
4
5
6
7
8
9
```

In [33]:

```
for i in range(10.5):
    print(i)
```

```
---------------------------------------------------------------------
-------
TypeError                                 Traceback (most recent cal
l last)
<ipython-input-33-4b73b4b4a55b> in <module>
----> 1 for i in range(10.5):
      2     print(i)

TypeError: 'float' object cannot be interpreted as an integer
```

In [37]:

```
for i in range(10, 50):
    print(i, end = '\t')
```

```
10      11      12      13      14      15      16      17      18
19      20      21      22      23      24      25      26      27
28      29      30      31      32      33      34      35      36
37      38      39      40      41      42      43      44      45
46      47      48      49
```

In [34]:

```
for i in range(10.6, 50.6):
    print(i)
```

```
--------------------------------------------------------------------
-------
TypeError                                 Traceback (most recent cal
l last)
<ipython-input-34-f34e17810117> in <module>
----> 1 for i in range(10.6, 50.6):
      2     print(i)

TypeError: 'float' object cannot be interpreted as an integer
```

In [38]:

```
for i in range(10, 50, 5):
    print(i, end = '\t')
```

```
10      15      20      25      30      35      40      45
```

In [35]:

```
for i in range(10, 50, 5.5):
    print(i)
```

```
--------------------------------------------------------------------
-------
TypeError                                 Traceback (most recent cal
l last)
<ipython-input-35-f51d54e0722f> in <module>
----> 1 for i in range(10, 50, 5.5):
      2     print(i)

TypeError: 'float' object cannot be interpreted as an integer
```

#5 Tuple data type

Tuple is a immutable datatype

In [1]:

```
List = [1,2,3,4,5]
Tuple = (1,2,3,4,5)
```

In [2]:

```
Tuple
```

Out[2]:

```
(1, 2, 3, 4, 5)
```

In [3]:

```
type(Tuple)
```

Out[3]:

```
tuple
```

In [7]:

```
for i in Tuple:
    print(i, end = '\t')
```

```
1       2       3       4       5
```

#6 Set data type

1. No duplicate items

2. Insertion order is not preserved

In [4]:

```
Set = {1,5,9,12,78}
```

In [5]:

```
Set
```

Out[5]:

```
{1, 5, 9, 12, 78}
```

In [6]:

```
type(Set)
```

Out[6]:

```
set
```

In [8]:

```
for i in Set:
    print(i, end = '\t')
```

```
1       5       9       12      78
```