18/01/2022

```
                                Operators
                                ---------
```

Operator is a symbol that performs certain operations.
Python provides 6 main operators.

1.Arithmetic Operators.
2.Realational or Comparision Opeartors.
3.Logical Operators.
4.Bitwise Opearators.(Not Required)
5.Assignment Operators.
6.special Opeartors.

1.Arithmetic Operators:-

+ -->Addition

```
12+90
102
9.89+7.67
17.560000000000002
6+9j + 7+8j
(13+17j)
'K'+'S'
'KS'
True + False
1
```

- -->substraction

```
90-67
23
7.890-6.89
1.0
7+9j - 6-9j
(1+0j)
'H'-'L'
Traceback (most recent call last):
File "<pyshell#9>", line 1, in <module>
'H'-'L'
TypeError: unsupported operand type(s) for -: 'str' and 'str'
True - True
0
```

* -->Multiplication

```
5678
4368
6.897.78
```

```
53.6042
(9+8j) * (8+8j)
(8+136j)
'KS'*3
'KSKSKS'
True *556
556
```

/ -->Division

```
67/8
8.375
68.89/6.45
10.68062015503876
(67+9j) / (8+7j)
(5.300884955752212-3.5132743362831858j)
'K'/7
Traceback (most recent call last):
File "<pyshell#23>", line 1, in <module>
'K'/7
TypeError: unsupported operand type(s) for /: 'str' and 'int'
True / 7
0.14285714285714285
```

% -->modulo

```
62%6
2
```

// -->Floor Division

```
67//5
13
```

** -->Exponent

```
23
8
42
16
```

2.Realational or Comparision Opeartors.

Relational operator always return the boolean value.

```
,<,>=,<=,==(equal to),!=(Not equal)
```

Greater than:-

```
45>78
False
67>56
True
```

Less than :-

```
56<78
True
78<45
False
```

Greater that equal to :-

```
78>=90
False
67>=67
True
```

Less than equal to:-

```
67<= 78
True
78 <= 67
False
78<= 78
True
```

Equal to:-

```
89 == 89
True
78 ==90
False
```

Not equal to:-

```
67 != 89
True
56 != 56
False
```

Logical Opeartors:-

and , or , not

and :- If both the arguments are True then only result is True otherwise False

or:- If atleast One argument is True then Result is True Othersie False.

not :- Complement

and:-Always return the biggest value.

T T T
T F F
F F F

```
56 and  78
78
0 and 67
0
```

or :- If a arguements evaluates to True then result is this arguement otherwise second one.

if one arguement is True then True otherwise False.

always smallest value is return.

```
T T  T
T F  T
F F  F
```

```
10 or 20
10
78 or 4550
78
0 or 67
67
```

not:- if a argument is evalutates to False then result is True otherwise False.

```
not 6
False
not 0
True
```

Assignment Operators:-

we can use assignment operator to assign value to the variable

a = 10 (= is assignment opeartor)

we can combine assignment operator with some other operator to form compound assignment operator,

ex :-

x = x+10 ---------- x+=10

Possiable list of all compoundment operator

```
+=
-=
*=
/=
%=
//=
**=
```

Special Operator:-

1.identity Opeartors
2.membership Opeartos

1.identity Opeartors

we can use idenity operators for address comparision.

2 idenitity opeartor is available.

1. is(if variable indicate same data)
2. is not(if variable indicate different data)

x is y return True if both x and y are pointing to the same object.

```
a = 10
b = 10
a is b
True
```

```
a= 89
b = 90
a is b
False
```

x is not y return True if x and y are pointing to the differnet object.

```
a = 90
b = 89

a is not b
True

a = 10
b = 10
a is not b
False
```

2.Memebership Opeartor:-

we can use membership operators to check whether the given object present in the given collection(String,List,Srt,Tuple or Dict)

in:- Returns True if the given obejct present in the specified collection.

```
a = 'Vicky'
'k' in a
True
'h' in a
False
```

```
b = [10,20,30,40]
89 in b
False
40 in b
True
```

not in:- Returns True if the given object not present in the specified collection.

```
a = 'Vicky'
'k' not in a
False
'h' not in a
True
```