Operators

Arithmetic Operator (+,-,,/,//,%,*,+)

```
In [1]: x,y = 10,5
```

```
In [2]: x+5
```

```
Out[2]: 15
```

```
In [3]: x-y
```

```
Out[3]: 5
```

```
In [4]: x*y
```

```
Out[4]: 50
```

```
In [5]: x/y
```

```
Out[5]: 2.0
```

```
In [6]: x//y
```

```
Out[6]: 2
```

```
In [7]: x%y
```

```
Out[7]: 0
```

```
In [8]: x**y
```

```
Out[8]: 100000
```

Assignment operator (=,+=,-=,*=,/=,//=)

```
In [9]: x=2
```

```
In [10]: x=x+2
         x
```

```
Out[10]: 4
```

```
In [11]: x+=2
         x
```

```
Out[11]: 6
```

In [13]:
```
x-=2
x
```

Out[13]:
```
2
```

In [14]:
```
x*=2
x
```

Out[14]:
```
4
```

In [16]:
```
x/=2
x
```

Out[16]:
```
1.0
```

In [17]:
```
x//=2
x
```

Out[17]:
```
0.0
```

Unary Operator

--> Unary means - 1 | Binary means 2 --> Here we are applying minus operator(-) on the operand n ; the value of m becomes -7, which indicates it as negative value

In [18]:
```
n = 7
n
```

Out[18]:
```
7
```

In [19]:
```
m = -(n)
m
```

Out[19]:
```
-7
```

In [20]:
```
-n
```

Out[20]:
```
-7
```

Relational Operators

We are using this operator for comparing (<,>,<=,>=,==)

In [22]:
```
a = 10
b = 16
```

In [23]:
```
a>b
```

Out[23]:
```
False
```

In [24]:
```
a<b
```

```
Out[24]: True
```

```
In [25]: a == b
```

```
Out[25]: False
```

```
In [26]: a>=b
```

```
Out[26]: False
```

```
In [27]: a<=b
```

```
Out[27]: True
```

Logical Operator

AND, OR, NOT

```
In [28]: a = 10
         b = 5
```

```
In [33]: a<b and b<a
```

```
Out[33]: False
```

```
In [32]: a<12 and b>4
```

```
Out[32]: True
```

```
In [34]: a>6 or b>4
```

```
Out[34]: True
```

```
In [35]: a>15 or b<2
```

```
Out[35]: False
```

```
In [36]: x = False
         x
```

```
Out[36]: False
```

```
In [37]: not x
```

```
Out[37]: True
```

```
In [38]: x = not x
         x
```

```
Out[38]: True
```

```
In [40]: not x
```

```
Out[40]: False
```

Number System

Binary Octal Decimal HexaDecimal

```
In [ ]:  Decimal ---> Binary
         Octal ----> Hexa
```

```
In [41]:  25

Out[41]:  25
```

```
In [42]:  bin(25)

Out[42]:  '0b11001'
```

```
In [43]:  bin(30)

Out[43]:  '0b11110'
```

```
In [44]:  int(0b11001)

Out[44]:  25
```

```
In [45]:  int(0b11110)

Out[45]:  30
```

```
In [46]:  oct(25)

Out[46]:  '0o31'
```

```
In [47]:  int(0o31)

Out[47]:  25
```

```
In [48]:  hex(25)

Out[48]:  '0x19'
```

```
In [49]:  hex(16)

Out[49]:  '0x10'
```

```
In [50]:  hex(16)

Out[50]:  '0x10'
```

```
In [51]:  hex(1)

Out[51]:  '0x1'
```

```
In [52]:  0x10

Out[52]:  16
```

In [53]:
```
0x1
```

Out[53]:
```
1
```

Swapping two Variables in Python

In [54]:
```
a = 5
b = 6
```

In [55]:
```
a = b
b = a
print(a)
print(b)
```

```
6
6
```

In [59]:
```
# in the aboce senario we lost the value of 5
a1 = 7
b1 = 8
```

In [60]:
```
temp = a1
a1 = b1
b1 = temp
```

In [61]:
```
print(a1)
print(b1)
```

```
8
7
```

In [62]:
```
a2 = 5
b2 = 6
```

In [63]:
```
# swap variables in another way
a2 = a2 + b2
b2 = a2 - b2
a2 = a2 - b2
```

In [64]:
```
print(a2)
print(b2)
```

```
6
5
```

Betwise Operator

- WE HAVE 6 OPERATORS COMPLEMENT (~) || AND (&) || OR (|) || XOR (^) || LEFT SHIFT (<<) || RIGHT SHIFT (>>)

In [65]:
```
print(bin(12))
print(bin(13))
```

```
0b1100
0b1101
```

```
In [66]:  # Compliment
          ~12

Out[66]:  -13

In [67]:  ~14

Out[67]:  -15

In [68]:  # AND
          12 & 13

Out[68]:  12

In [69]:  # OR
          12 | 13

Out[69]:  13

In [70]:  1 & 0

Out[70]:  0

In [71]:  # OR
          1 | 0

Out[71]:  1

In [73]:  print(bin(35))
          print(bin(40))

         0b100011
         0b101000

In [74]:  35 | 40

Out[74]:  43

In [75]:  # XOR
          35 ^ 40

Out[75]:  11

In [76]:  10<<1

Out[76]:  20

In [77]:  12<<2

Out[77]:  48

In [78]:  10>>1

Out[78]:  5

In [79]:  12>>2
```

```
Out[79]: 3
```

```
In [ ]:
```