Python Operators

Operators are used to perform operations on variables and values

- Arithmetic operators # + - * / //
- Assignment operators # += -= *= /= =
- Comparison operators # == != > < >= <=
- Logical operators # **and** **or** **not**
- Membership operators # **in** **not in**
- Bitwise operators # **AND** **OR** **XOR** **NOT**

Arithmetic operator

```
x = int(input("Enter 1 Digits :- "))        #   10
y = int(input("Enter 2 Digits :- "))        #   2
print('Addition :- ', x+y)                  #   12
print('Subtraction :- ', x-y)               #   8
print('Multiplication :- ', x*y)            #   20
print('Division :- ', x/y)                  #   5.0
print('Floor division :- ', x//y)           #   5
print('Exponentiation :- ', x**y)           #   100
```

Assignment operator

```
x = 3       =   x = 3
x += 3      =   x = x + 3
x -= 3      =   x = x – 3
x *= 3      =   x = x * 3
x /= 4      =   x = x / 3
```

Comparison operator

```
x = int(input("Enter 1 Digits :- "))          #    10
y = int(input("Enter 2 Digits :- "))          #    2
print('Equal :- ', x==y)                      #    False
print('Not equal :- ', x!=y)                  #    True
print('Less Than :- ', x<y)                   #    False
print('Greater Than  :- ', x>y)               #    True
print('Less than equal :- ', x<=y)            #    False
print('Greater than equal :- ', x>=y)         #    True
```

Logical operators

and -- Returns True if both Statements are True

```
x = int(input("Enter Digits :- "))            #    10
print( x>5 and 12>x )                         #    True
```

or -- Returns True if one of the Statements is True

```
x = int(input("Enter Digits :- "))            #    10
print( x>5 or x<6 )                           #    True
```

not -- Reverse the result, Returns True if the result is False

```
x = int(input("Enter Digits :- "))            #    10
print(not( x>5 and x<12 ))                    #    False
```

Membership operators

in -- Returns True if a sequence with the specified value is present in the object

```
x = ["Apple", "Mango"]

print("Apple", in x)                    #       True
```

in not -- Returns True if a sequence with the specified value is not present in the object

```
x = ["Apple", "Mango"]

print("Pineapple", not in x)            #       True
```

Bitwise operators

&	AND	Sets each bit to 1 if both bits are 1
|	OR	Sets each bit to 1 if one of the two bits is 1
^	XOR	Sets each bit to 1 if only one of two bits is 1
~	NOT	Inverts all the bits