

# 4

## Day 4: Operators

### Operators

Types of operators:

- Arithmetic operators (+, -, \*, /, %, \*\*, //)
- Assignment operators(=, +=, -=, \*=, /=)
- Comparison operators(==, !=, >, <, >=, <=)
- Logical operators(and, or, not)
- Identity operators
- Membership operators
- Bitwise operators

### Comparison Operators

Comparison operators are used to compare two values.

```
x = 9
y = 6

# == (Equal)
print(x==y)
""" It is different from "=", since "=" is an assignment operator
   assigns value to variable."""

# != (Not equal)
print(x!=y)
""" The output will be boolean(True/ False)."
```

```
# > (Greater than)
print(x>y)

# < (Less than)
print(x>y)

# >= (Greater than or equal to)
print(x>=y)

# <= (Less than or equal to)
print(x<=y)
```

## Logical Operators

Logical operators are used to combine conditional operators.

```
x = 9

# and (Returns True if both statements are True)
print(x > 3 and x < 10)

# or (Returns True if one of the statements is True)
print(x>3 or x<4)

# not (Reverse the result)
print(not(x<3))
```

## Identity Operators

Identity operators are used to compare the objects, not if they are equal, but if they are actually the same object, with the same memory location.

```
x = [1, 2, 3] # [] It's a list
y = [1, 2, 3]
z = x
```

```
# is
print(x is z)
# returns True because z is the same object as x

print(x is y)
# returns False because x is not the same object as y, even if t

print(x == y)
# to demonstrate the difference between "is" and "=="
# this comparison returns True because x is equal to y

# is not
print(x is not z)
# returns False because z is the same object as x

print(x is not y)
# returns True because x is not the same object as y, even if t

print(x != y)
# to demonstrate the difference between "is not" and "!="
# this comparison returns False because x is equal to y
```



### Python Operators Test (Comparison, Logical & Identity).