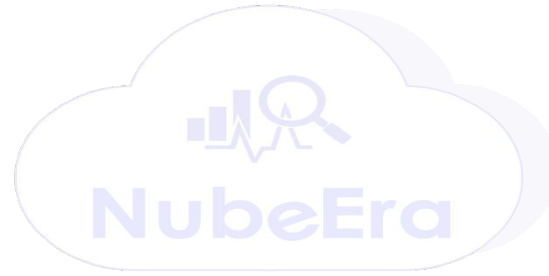


Operators





- Concept
- Types of Operators
 - Unary operators
 - Binary operators
 - Ternary operators
- Summary





- **Concept**

- Operators are used to perform operations on variables and values.
- Examples
 - `print(10+20)`
- Here “+” is operator and “10” ,”20” are operands.
- **Operators** are special symbols that represent computations like addition and multiplication. The values the operator uses are called **operand**





- **Types of Operators**

1. Unary Operators

- Minus(-)
 - To represent negative numbers.
 - Examples
 - 1] -10
 - 2]-20

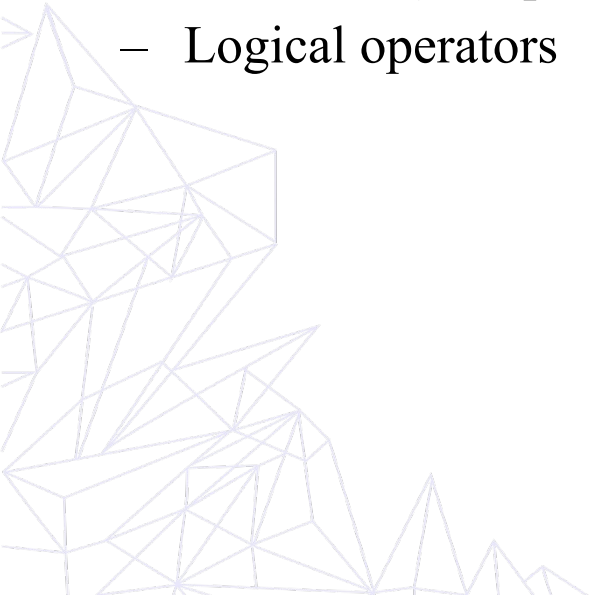
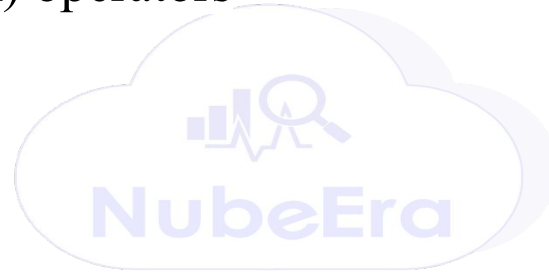
NubeEra





2.Binary Operator

- Arithmetic operators
- Relational(Comparison) operators
- Logical operators





□ Arithmetic Operator

- Addition $x + y$
- Subtraction $x - y$
- Multiplication $x * y$
- Division x / y
- Modulus $x \% y$
- Exponentiation $x ** y$
- Floor division $x // y$

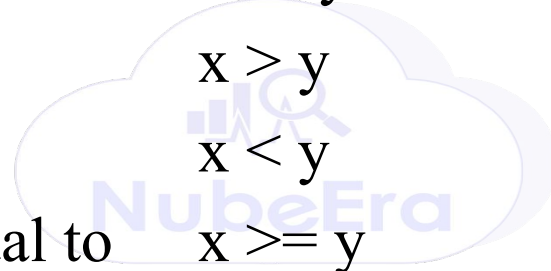
NubeEra





□ Relational(Comparison)Operator

- Equal to $x == y$
- Not equal to $x != y$
- Greater than $x > y$
- Less than $x < y$
- Greater than or equal to $x >= y$
- Less than or equal to $x <= y$





□ Logical Operator

■ and

Returns True if both statements are true $x < 5$ and $x < 10$

■ or

Returns True if one of the statements is true $x < 5$ or $x < 4$

■ not

Reverse the result, returns False if the result is true $\text{not}(x < 5$
and $x < 10)$





3.Ternary Operator(conditional expressions)

- that evaluate something based on a condition being true or false.
- It simply allows testing a condition in a **single line** replacing the multiline if-else making the code compact.

- **Syntax :**

[on_true] if [expression] else [on_false]

- **Example**

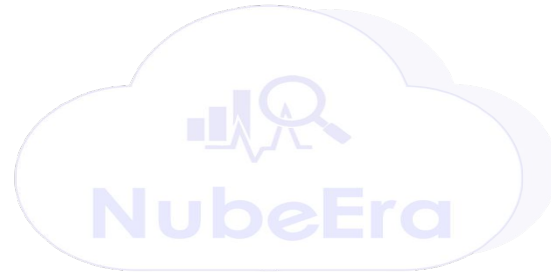
a=10 b=20

min = a if a < b else b





- Concept
- Types of Operators
 - Unary operators
 - Binary operators
 - Ternary operators



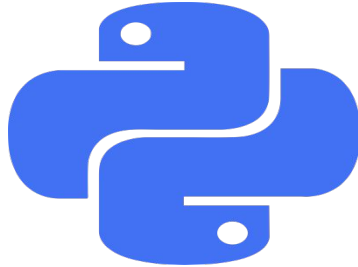
[Demo](#)



???

The Important thing is not to
stop

Questioning



RECORDED



CORPORATE



INTERACTIVE



CORPORATE



RECORDED



INTERACTIVE