

Logical Operator



Table of contents

01

AND Operator

02

OR Operator

03

NOT Operator

04

Combining Logical Operators

05

Practical Examples

01

AND Operator



AND Operator – Truth Table

Condition 1	Condition 2	Condition and Condition 2
True	True	True
True	False	False
False	True	False
False	False	False

AND Operator

The **and** operator returns True if both expressions on either side of the operator are True. If either one of the expressions is False, it returns False.

Syntax:

condition1 and condition2

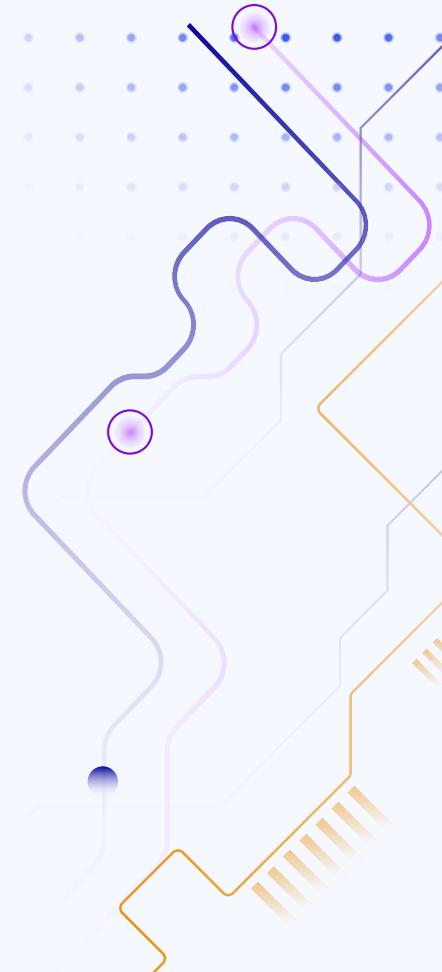
Python

```
x = 5  
y = 10  
if x > 3 and y < 15:  
    print("Both conditions are True")
```

In this example, both conditions ($x > 3$ and $y < 15$) are True, so the message "Both conditions are True" will be printed.

02

OR Operator



OR Operator - Truth Table

Condition 1	Condition 2	Condition or Condition 2
True	True	True
True	False	True
False	True	True
False	False	False

OR Operator

The **or** operator returns **True** if **at least one** of the expressions on either side of the operator is **True**. It only returns **False** if both expressions are **False**.

Syntax:

condition1 or condition2

Python

```
x = 5  
y = 10  
if x > 10 or y < 15:  
    print("At least one condition is True")
```

In this example, the second condition (**y < 15**) is **True**, so the message "At least one condition is True" will be printed even though the first condition (**x > 10**) is **False**.

03

NOT Operator



NOT Operator – Truth Table

Condition	Not Condition
True	False
False	True

NOT Operator

The **not** operator is a unary operator that **inverts** the boolean value of the expression it is applied to. If the expression is **True**, **not** makes it **False**, and if it is **False**, **not** makes it **True**.

Syntax:

not condition

Python

```
x = 5
if not x > 10:
    print("x is not greater than 10")
```

In this example, the condition **x > 10** is **False**, so **not** inverts it to **True**, and the message "x is not greater than 10" will be printed.

04

Combining Logical Operators

Combining Logical Operators

Logical operators can be combined to evaluate complex expressions. When combining **and**, **or**, and **not**. Python uses the following precedence rules:

1. '**not**' has the highest precedence.
2. '**and**' has the next highest precedence.
3. '**or**' has the lowest precedence.

Parentheses can be used to override the default precedence and make the code more readable.

Combining Logical Operators

Python

```
x = 5
y = 10
z = 15

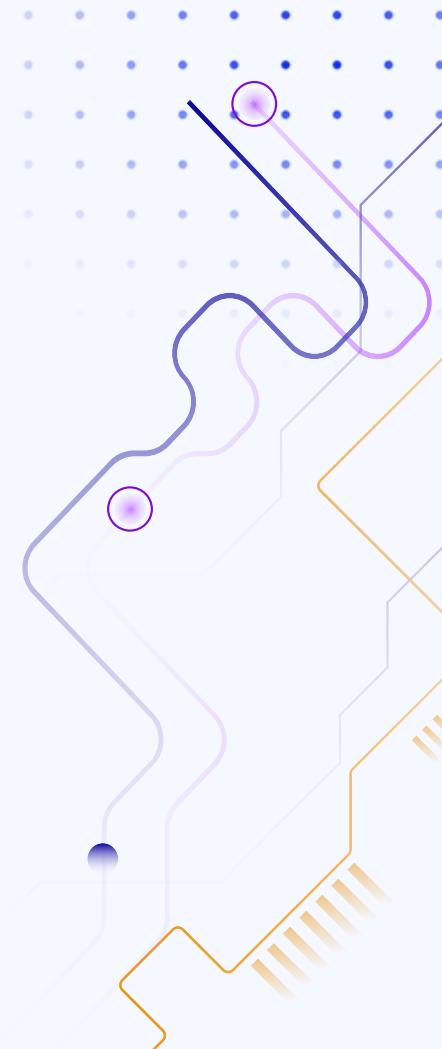
if (x < y or y > z) and not (x + y < z):
    print("Complex condition is True")
```

In this example, the expression is broken down as follows:

1. `(x < y or y > z)` evaluates to `True` because `x < y` is `True`.
2. `not (x + y < z)` evaluates to `True` because `x + y < z` is `False`, and `not` inverts it.
3. Finally, `True and True` evaluates to `True`, so the message "Complex condition is True" is printed.

05

Practical Use Cases



Filtering Data

Logical operators are commonly used in filtering data, such as retrieving specific records from a dataset.

Python

```
students = [
    {"name": "Alice", "age": 20, "grade": "A"},
    {"name": "Bob", "age": 19, "grade": "B"},
    {"name": "Charlie", "age": 22, "grade": "C"},  
]  
  
for student in students:  
    if student["age"] > 18 and student["grade"] == "A":  
        print(student["name"])
```

This code prints the names of students who are older than 18 and have an "A" grade.

Validating Input

Logical operators can validate user input in form submissions or command-line programs.

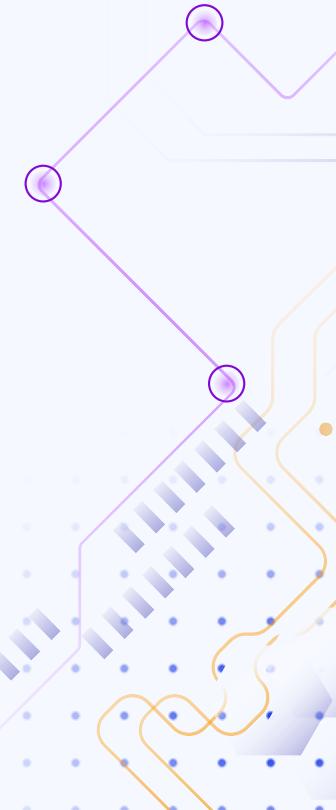
Python

```
username = input("Enter username: ")
password = input("Enter password: ")

if username == "admin" and password == "secret":
    print("Access granted")
else:
    print("Access denied")
```

Here, both the username and password must be correct for access to be granted.

Knowledge Reinforcement



1

Question

What will be the output of the following code?

Python

```
x = 10
y = 20
if x > 5 and y < 25:
    print("Condition is True")
else:
    print("Condition is False")
```

Answer

- a) Condition is True
- b) Condition is False
- c) Error
- d) None of the above

1

Question

What will be the output of the following code?

Python

```
x = 10
y = 20
if x > 5 and y < 25:
    print("Condition is True")
else:
    print("Condition is False")
```

Answer

- a) Condition is True
- b) Condition is False
- c) Error
- d) None of the above

2

Question

What does the **or** operator return if both expressions are **False**?

Answer

- a) True
- b) False
- c) None
- d) Error

2

Question

What does the **or** operator return if both expressions are **False**?

Answer

a) True

b) False

c) None

d) Error

3

Question

Which of the following operators has the **highest precedence**?

Answer

- a) and
- b) or
- c) not
- d) None of the above

3

Question

Which of the following operators has the **highest precedence**?

Answer

- a) and
- b) or
- c) **not**
- d) None of the above

4

Question

What will be the output of the following code?

Python

```
x = 15
if not x < 20:
    print("Condition is True")
else:
    print("Condition is False")
```

Answer

- a) Condition is True
- b) Condition is False
- c) Error
- d) None of the above

4

Question

What will be the output of the following code?

Python

```
x = 15
if not x < 20:
    print("Condition is True")
else:
    print("Condition is False")
```

Answer

- a) Condition is True
- b) Condition is False**
- c) Error
- d) None of the above

5

Question

Which of the following conditions will result in **True**?

Answer

- a) True and False
- b) False or False
- c) not (True)
- d) True or False

5

Question

Which of the following conditions will result in **True**?

Answer

- a) True and False
- b) False or False
- c) not (True)
- d) True or False**



6

Question

What will be the output of the following code?

Python

```
x = 7
y = 14
if (x > 5 or y < 10) and y % 2 == 0:
    print("Expression is True")
else:
    print("Expression is False")
```

Answer

- a) Expression is True
- b) Expression is False
- c) Error
- d) None of the above

6

Question

What will be the output of the following code?

Python

```
x = 7
y = 14
if (x > 5 or y < 10) and y % 2 == 0:
    print("Expression is True")
else:
    print("Expression is False")
```

Answer

- a) Expression is True
- b) Expression is False
- c) Error
- d) None of the above

7

Question

The **and** operator returns **True** only when **both conditions are True**.

Answer

- a) True
- b) False

7

Question

The **and** operator returns **True** only when **both conditions are True**.

Answer

a) True

b) False

8

Question

The **not** operator **inverts** the boolean value of the expression it is applied to.

Answer

- a) True
- b) False

8

Question

The **not** operator **inverts** the boolean value of the expression it is applied to.

Answer

a) True

b) False

9

Question

In the expression $15 > 10$ or $3 < 5$, the result of the expression will be _____.

Answer



9

Question

In the expression $15 > 10$ or $3 < 5$, the result of the expression will be _____.

Answer

True

10

Question

The logical operator with the **lowest precedence** is _____.

Answer



10

Question

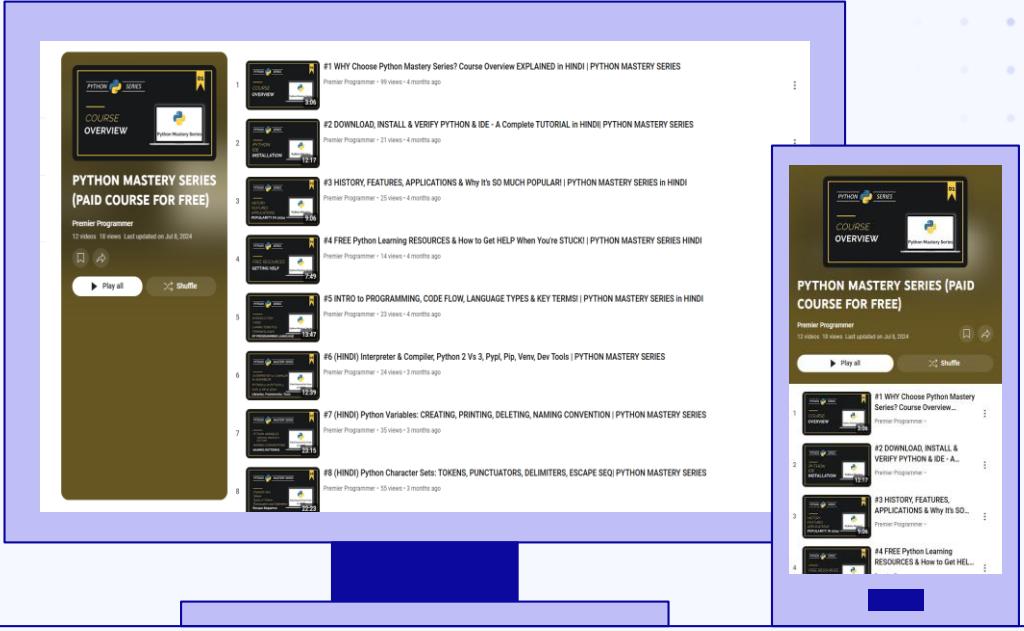
The logical operator with the **lowest precedence** is _____.

Answer

OR

WATCH

Level up your coding with each episode in this focused Python series.



Next Video!

Membership Operator-
In Depth

