Comparison Operators

Now, we'll learn how to perform comparisons in Python using comparison operators.

We'll cover the following

- Comparisons

Comparison operators can be used to compare values in mathematical terms.

Operator	Purpose	Notation
`>`	Greater Than	In-fix
`<`	Less Than	In-fix
`>=`	Greater Than or Equal To	In-fix
`<=`	Less Than or Equal To	In-fix
`==`	Equal To	In-fix
`!=`	Not Equal To	In-fix
`is`	Equal To (Identity)	In-fix

Operator	Purpose	Notation
`is not`	Not Equal To (Identity)	In-fix

Comparisons

The result of a comparison is always a bool.

If the comparison is correct, the value of the bool will be `True`. Otherwise, its value will be `False`.

The `==` and `!=` operators compare the **values** of both operands. However, the identity operators, `is` and `is not`, check whether the two operands are the **exact same object**. We don't need to delve into them yet, but you'll understand their utility later on.

Let's look at a few examples:

```
num1 = 5
num2 = 10
num3 = 10
print(num2 > num1)  # 10 is greater than 5
print(num1 > num2)  # 5 is not greater than 1

print(num2 == num3)  # Both have the same val
print(num3 != num1)  # Both have different va

print(3 + 10 == 5 + 5)  # Both are not equal
print(3 <= 2)  # 3 is not less than or equal

```

As we can see in **line 4**, `num2` is indeed greater than `num1`. Hence, the result is `True`. On the other hand, **line 5** contains an incorrect comparison, which results in `False`.

Next, we'll study the **assignment operators**.

← Back
Arithmetic Operators

Next →
Assignment Operators

☑ Completed

Report an

Ask a Question