Python
TUTORIAL

Python Comparison Operators

Summary: in this tutorial, you'll learn about Python comparison operators and how to use them to compare two values.

Introduction to Python comparison operators

In programming, you often want to compare a value with another value. To do that, you use comparison operators.

Python has six comparison operators, which are as follows:

- Less than (`<`)
- Less than or equal to (`<=`)
- Greater than (`>`)
- Greater than or equal to (`>=`)
- Equal to (`==`)
- Not equal to (`!=`)

These comparison operators compare two values and return a boolean (https://www.pythontutorial.net/python-basics/python-boolean/) value, either `True` or `False` .

And you can use these comparison operators to compare both numbers (https://www.pythontutorial.net/python-basics/python-numbers/) and strings (https://www.pythontutorial.net/python-basics/python-string/) .

Less than operator (<)

The Less Than operator (<) compares two values and returns `True` if the value on the left is less than the value on the right. Otherwise, it returns `False` :

```
left_value  < right_value
```

The following example uses the Less Than (`<`) operator to compare two numbers:

```
>>> 10 < 20
True
>>> 30 < 20
False
```

It's quite obvious when you use the less than operator with the numbers.

The following example uses the less than operator (`<`) to compare two strings:

```
>>> 'apple' < 'orange'
True
>>> 'banana' < 'apple'
False
```

The expression `'apple' < 'orange'` returns `True` because the letter `a` in `apple` is before the letter `o` in `orange` .

Similarly, the `'banana' < 'apple'` returns `False` because the letter `'b'` is after the letter `'a'` .

The following example shows how to use the less than operator with variables (https://www.pythontutorial.net/python-basics/python-variables/) :

```
>>> x = 10
>>> y = 20
>>> x < y
True
>>> y < x
False
```

Less than or equal to operator (<=)

The less than or equal to operator compares two values and returns `True` if the left value is less than or equal to the right value. Otherwise, it returns `False` :

```
left_value <= right_value
```

The following example shows how to use the less than or equal to operator to compare two numbers:

```
>>> 20 <= 20
True
>>> 10 <= 20
True
>>> 30 <= 30
True
```

And this example shows how to use the less than or equal to operator to compare the values of two variables:

```
>>> x = 10
>>> y = 20
>>> x <= y
True
```

```
>>> y <= x
False
```

Greater than operator (>)

The greater than operator (`>`) compares two values and returns `True` if the left value is greater than the right value. Otherwise, it returns `False` :

```
left_value > right_value
```

This example uses the greater than operator (`>`) to compare two numbers:

```
>>> 20 > 10
True
>>> 20 > 20
False
>>> 10 > 20
False
```

And the following example uses the greater than operator (`>`) to compare two strings:

```
>>> 'apple' > 'orange'
False
>>> 'orange' > 'apple'
True
```

Greater Than or Equal To operator (>=)

The greater than or equal to operator (`>=`) compares two values and returns `True` if the left value is greater than or equal to the right value. Otherwise, it returns `False` :

```
left_value >= right_value
```

The following example uses the greater than or equal to operator to compare two numbers:

```
>>> 20 >= 10
True
>>> 20 >= 20
True
>>> 10 >= 20
False
```

And the following example uses the greater than or equal to operator to compare two strings:

```
>>> 'apple' >= 'apple'
True
>>> 'apple' >= 'orange'
False
>>> 'orange' >= 'apple'
True
```

Equal To operator (==)

The equal to operator (`==`) compares two values and returns `True` if the left value is equal to the right value. Otherwise, it returns `False` :

```
left_value = right_value
```

The following example uses the equal to operator (`==`) to compares two numbers:

```
>>> 20 == 10
False
>>> 20 == 20
True
```

And the following example uses the equal to operator (`==`) to compare two strings:

```
>>> 'apple' == 'apple'
True
>>> 'apple' == 'orange'
False
```

Not Equal To operator (!=)

The not equal to operator (`!=`) compares two values and returns `True` if the left value isn't equal to the right value. Otherwise, it returns `False` .

```
left_value != right_value
```

For example, the following uses the not equal to operator to compare two numbers:

```
>>> 20 != 20
False
>>> 20 != 10
True
```

And the following example uses the not equal to operator to compare two strings:

```
>>> 'apple' != 'apple'
False
>>> 'apple' != 'orange'
True
```

Summary

- A comparison operator compares two values and returns a boolean value, either `True` or `False` .

- Python has six comparison operators: less than (`<`), less than or equal to (`<=`), greater than (`>`), greater than or equal to (`>=`), equal to (`==`), and not equal to (`!=`).