

Expressions and Variables

Functions

Conditionals

Video: Comparing Things
4 min

Reading: Comparison Operators with Equations
10 min

Reading: Comparison Operators with Strings
10 min

Reading: Logical Operators
10 min

Video: Branching with if Statements
2 min

Reading: if Statements Recap
10 min

Video: else Statements
3 min

Reading: else Statements and the Modulo Operator
10 min

Video: elif Statements
3 min

Reading: Complex Branching with elif Statements
10 min

Reading: Study Guide: Conditionals
10 min

Practice Quiz: Practice Quiz: Conditionals
5 questions

Module Review

Comparison Operators with Equations

The following examples demonstrate how to use comparison operators with the data types **int** (integers, whole numbers) and **float** (number with a decimal point or fractional value). Comparison operators return Boolean results. As you learned previously, Boolean is a data type that can hold only one of two values: **True** or **False**.

The comparison operators include:

- ==** (equality)
- !=** (not equal to)
- >** (greater than)
- <** (less than)
- >=** (greater than or equal to)
- <=** (less than or equal to)

PART 1: Equality == and Not Equal To != Operators

In Python, you can use comparison operators to compare values. When a comparison is made, Python returns a Boolean result: **True** or **False**. Note that Boolean data types are not string data types (Boolean **True** is not equal to the string "True").

- To check if two values are the same, use the **equality operator: ==**
- To check if two values are **not** the same, use the **not equal to operator: !=**

The print() function can be used to display the results of the comparisons.

Examples:

```
1
2 print(32 == 30+2) # The == operator checks if the 2 values are
3 True             # equal to each other. If they are equal,
4                 # Python returns a True result.
5
6
7 print(5+10 == 6+7) # If the two values are not equal, as in the
8 False             # expression 5+10 == 6+7 (or 15 == 13), Python
9                 # returns a False result.
10
11
12 print(10-4 != 10+4) # The != operator checks if the 2 values are
13 True              # NOT equal to each other. If true, Python
14                 # returns a True result.
15
16
17 print(9/3 != 3*1)  # In this last example, 9/3 != 3*1 (or 3 != 3)
18 False            # is False. So, Python returns a False value.
```

The equality == operator versus the equals = operator

It is important to note that the equality == comparison operator performs a different task than the equals = assignment operator. The equals = operator assigns the value on the right side of the equals = to the object (e.g., a variable) on the left side of the equals = operator.

Examples:

```
1
2 # The = equals assignment operator is used to assign a value to a
3 # variable.
4
5 my_variable = 3*5      # Assigns a value to my_variable
6 print(my_variable)     # Printing the variable returns the
7 15                    # value assigned to the variable.
8
9
10
11
12 # The == equality comparison operator checks if the values of the two
13 # expressions on either side of the == operator are equivalent to one
14 # another.
15
16 print(my_variable == 3*5) # Printing the variable returns a Boolean
17 True                    # True or False result.
```

PART 2: Greater Than > and Less Than < Operators

The comparison operators greater than > and less than < also return a **True** or **False** Boolean result after comparing two values.

- To check if one value is larger than another value, use the greater than operator: >
- To check if one value is smaller than another value, use the less than operator: <

Examples:

```
1
2 print(11 > 3*3)      # The > operator checks if the left value is
3 True               # greater than the right value. If true, it
4                 # returns a True result.
5
6
7 print(4/2 > 8-4)     # If the > operator finds that the left value
8 False            # is NOT greater than the right value, the
9                 # comparison will return a False result.
10
11
12 print(4/2 < 8-4)     # The < operator checks if the left value is
13 True             # less than the right side. If true, the
14                 # comparison returns a True result.
15
16
17 print(11 < 3*3)      # If the < operator finds that the left side is False
18 False            # a False result.
```

NOT less than the right value, Python returns

PART 3: Greater Than or Equal to >= and Less Than or Equal to <= Operators

Like the other comparison operators, the greater than or equal to >= and less than or equal to <= operators return a **True** or **False** Boolean result when a comparison is made.

- To check if one value is larger than or equal to another value, use the greater than or equal to operator: >=
- To check if one value is smaller than or equal to another value, use the less than or equal to operator: <=

Examples:

```
1
2 print(12*2 >= 24)    # The >= operator checks if the left value is
3 True                # greater than or equal to the right value.
4                 # If one of these conditions is true,
5                 # Python returns a True result. In this case
6                 # the two values are equal. So, the comparison
7                 # returns a True result.
8
9
10 print(18/2 >= 15)    # If the >= comparison determines that the left False
11 False              # value is NOT greater than or equal to the
12                 # right, it returns a False result.
13
14 print(12*2 <= 30)    # The <= operator checks if the left value is
15 True               # less than or equal to the right value. In
16                 # this case, the left value is less than the
17                 # right value. Again, if one of the two
18                 # conditions is true, Python returns a True
19                 # result.
20
21
22 print(15 <= 18/2)    # If the <= comparison determines that the left
23 False              # value is NOT less than or equal to the right
24                 # value, the comparison returns a False result.
```

PART 4: Practice

If you would like more practice using comparison operators, feel free to create your own comparisons using the code block below. Note that there is no feedback associated with this code block.

1

Run

Reset

For additional Python practice, the following links will take you to several popular online interpreters and codepads:

- [Welcome to Python](#)
- [Online Python Interpreter](#)
- [Create a new Repl](#)
- [Online Python 3 Compiler /Interpreter](#)
- [Compile Python 3 Online](#)
- [Your Python Trinket](#)

Key takeaways

Python comparison operators return Boolean results: **True** or **False**.

Symbol	Name	Expression	Description
==	Equality operator	a == b	a is equal to b
!=	Not equal to operator	a != b	a is not equal to b
>	Greater than operator	a > b	a is larger than b
>=	Greater than or equal to operator	a >= b	a is larger than or equal to b
<	Less than operator	a < b	a is smaller than b