

Course	
Term	
Week	
Date	
Chapter. Topic	3. Decision Structures and Boolean Logic

Operators and Conditions

Siva R Jasthi

Computer Science and Cybersecurity

Metropolitan State University

Outline

1. Operators
2. Conditions (if.. elif.. else)
3. Python Code Blocks: Indentation
4. String Operations

Comparison Operators

Comparison Operators are used to compare two values.
And the result is always a Boolean value (True or False).

Operator	Name	Example
<code>==</code>	Equal	<code>x == y</code>
<code>!=</code>	Not equal	<code>x != y</code>
<code>></code>	Greater than	<code>x > y</code>
<code><</code>	Less than	<code>x < y</code>
<code>>=</code>	Greater than or equal to	<code>x >= y</code>
<code><=</code>	Less than or equal to	<code>x <= y</code>

Conditions

Conditions help us to control the program execution.

We use:

if

elif

else

key words to write the conditions

```
1 random_state = "Delaware"
2
3 message = "What is the capital of ", random_state
4 user_answer = input(message)
5
6 state_capital = "Dover"
7
8 if user_answer == state_capital:
9     print("You are correct!")
10 else:
11     print("The capital of ", random_state, "is", state_capital)
```

Conditions

Let us try some of these
“Try it yourself” exercises at
this link.

← → ↻ w3schools.com/python/python_conditions.asp

Home HTML CSS JAVASCRIPT SQL PYTHON PHP

Python Get Started
Python Syntax
Python Comments
Python Variables
Python Data Types
Python Numbers
Python Casting
Python Strings
Python Booleans
Python Operators
Python Lists
Python Tuples
Python Sets
Python Dictionaries
Python If...Else
Python While Loops
Python For Loops
Python Functions

- Greater than or equal to: `a >= b`

These conditions can be used in several ways.

An "if statement" is written by using the `if` keyword.

Example

If statement:

```
a = 33
b = 200
if b > a:
    print("b is greater than a")
```

[Try it Yourself >>](#)

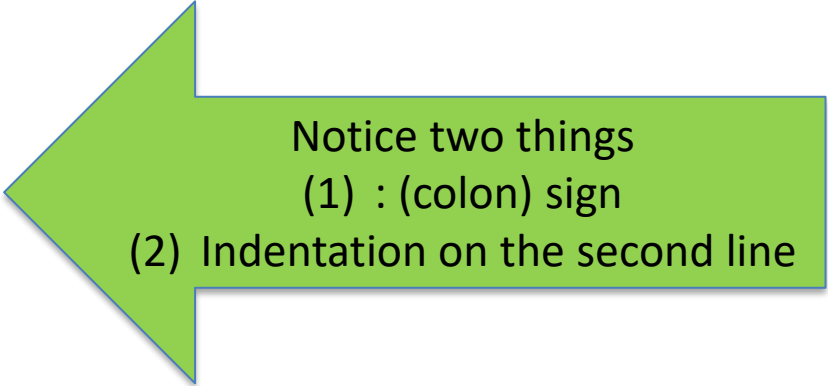
In this example we use two variables:

https://www.w3schools.com/python/python_conditions.asp

Indentation (code blocks in python)

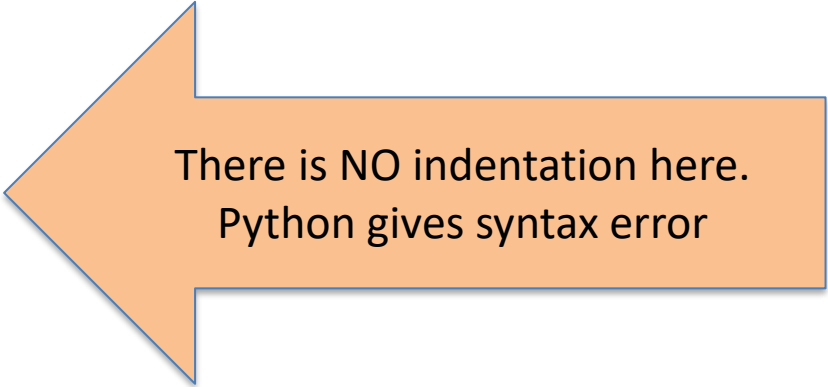
Python relies on indentation (whitespace at the beginning of a line) to define scope in the code.

```
a = 33
b = 200
if b > a:
    print("b is greater than a")
```



Notice two things
(1) : (colon) sign
(2) Indentation on the second line

```
a = 33
b = 200
if b > a:
print("b is greater than a")
```




There is NO indentation here.
Python gives syntax error

if condition

Python relies on indentation (whitespace at the beginning of a line) to define scope in the code.

```
a = 33
b = 200
if b > a:
    print("b is greater than a")
```



- Equals: `a == b`
- Not Equals: `a != b`
- Less than: `a < b`
- Less than or equal to: `a <= b`
- Greater than: `a > b`
- Greater than or equal to: `a >= b`

if .. else

The `else` keyword catches anything which isn't caught by the preceding conditions.

```
a = 200
b = 33
if b > a:
    print("b is greater than a")
else:
    print("b is not greater than a")
```


if ..elif

The **elif** keyword is python's way of saying "if the previous conditions were not true, then try this condition".

```
a = 33
b = 33
if b > a:
    print("b is greater than a")
elif a == b:
    print("a and b are equal")
```

if ..elif...else

The `else` keyword catches anything which isn't caught by the preceding conditions.

```
a = 200
b = 33
if b > a:
    print("b is greater than a")
elif a == b:
    print("a and b are equal")
else:
    print("a is greater than b")
```

nested if..elif..else conditions

You can go any number of levels to write your conditions.

[1] Remember the indentation. If you are not getting the expected result, check your indentation.

[2] Make sure to use “comparison” operators in if and elif conditions. The final result should be “Boolean”

Coding Tip (4 lines to 1 line)

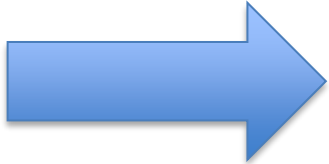
```
age = 18
```

```
If (age >=18):  
    print ("can vote")  
else:  
    print ("can not vote")
```

Can be written as follows (imagine speaking plain English)

```
print ("can vote") if (age >= 18) else print ("can not vote")
```

```
If (age >=18):  
    print ("can vote")  
else:  
    print ("can not vote")
```



```
print ("can vote") if (age >= 18) else print ("can not vote")
```

Some questions

[1] Can you have "elif" without "if"?



[2] Can you have "else" without "if"?

No

[3] Can you have "if" without "elif"?

Yes

[4] Can you have "if" without "else"?

Yes

[5] Can you have "if" inside another "if"? Yes

[6] Can you have two "elif" statements at the same level? Yes

[7] Can you have two "else" statements at the same level? No

[8] Can you have "if" and "elif" without an "else" at the same level? Yes

Summary: What did we learn today?

1. Operators (Arithmetic, Comparison, Logical)
2. Conditions (if...elif..else)
3. Python Code Blocks: Indentation