



[Curso](#) > [Week 1...](#) > [1. Intro...](#) > [Exercis...](#)

Audit Access Expires 5 de ago de 2020

You lose all access to this course, including your progress, on 5 de ago de 2020.

Exercise 8

Finger Exercises due Aug 5, 2020 20:30 -03 *Completo*

Exercise 8

12/12 points (graded)

ESTIMATED TIME TO COMPLETE: 6 minutes

Note that you will have to answer all questions before you can click the Check button.

For each of the following expressions, indicate the value returned, or if the evaluation would lead to an error, write the word 'error' (note this is a word, not a string, no quotes). While you could simply type these expressions into your IDE, we encourage you to answer them directly since this will help reinforce your understanding of basic Python expressions.

Hint: Python boolean types

Remember that in Python words are case-sensitive. The word `True` is a Python keyword (it is the value of the Boolean type) and is not the same as the word `true`. Refer to the [Python documentation on Boolean values](#).

Hint: Priority order of Boolean operations

For these problems, it's important to understand the priority of Boolean operations. The order of operations is as follows:

1. Parentheses. Before operating on anything else, Python must evaluate all parentheticals starting at the innermost level.
2. `not` statements.
3. `and` statements.



4. `or` statements.

What this means is that an expression like

```
not True and False
```

evaluates to `False`, because the `not` is evaluated first (`not True` is `False`), then the `and` is evaluated, yielding `False and False` which is `False`.

However the expression

```
not (True and False)
```

evaluates to `True`, because the expression inside the parentheses must be evaluated first - `True and False` is `False`. Next the `not` can be evaluated, yielding `not False` which is `True`.

Overall, you should always use parenthesis when writing expressions to make it clear what order you wish to have Python evaluate your expression. As we've seen here, `not (True and False)` is different from `(not True) and False` - but it's easy to see how Python will evaluate it when you use parentheses. A statement like `not True and False` can bring confusion!

- `3 > 4`

```
False
```

✓ Answer: False

- `4.0 > 3.999`

```
True
```

✓ Answer: True

- `4 > 4`

```
False
```

✓ Answer: False

- `4 > + 4`

```
False
```

✓ Answer: False

- `2 + 2 == 4`



✓ Answer: True

- `True or False`

✓ Answer: True

- `False or False`

✓ Answer: False

- `not False`

✓ Answer: True

- `3.0 - 1.0 != 5.0 - 3.0`

✓ Answer: False

- `3 > 4 or (2 < 3 and 9 > 10)`

✓ Answer: False

- `4 > 5 or 3 < 4 and 9 > 8`

✓ Answer: True

- `not(4 > 3 and 100 > 6)`

✓ Answer: False



i Answers are displayed within the problem














Exercise 8

Ocultar discussão

Topic: Lecture 1 / Exercise 8

Show all posts ▼

por atividade recente ▼

- | | | |
|---|--|----|
|  | <u>I answered all of these in lowercase, and got them wrong. Apparently the first letter has to be capital.</u>
<u>I think it should say at the top that answers must be capitalised.</u> | 2 |
|  | <u>PEDMAS for Booleans</u>
<u>I never realized that "not" "or" "and" statements all had a specific order!! Learned something cool...</u> | 3 |
|  | <u>Why True or False is True?</u>
<u>I am confused that why True or False is True?</u> | 15 |
|  | <u>last question</u>
<u>?can anyone explain the last question</u> | 2 |
|  | <u>4 > + 4 not an error?</u>
<u>Helo, isnt 4 > + 4 supposed to be an error or is the + kind of the same as negative or positive num...</u> | 2 |
|  | <u>laptop</u>
<u>How do I do the programming interactive learning on my laptop on windows until now I'm only u...</u> | 3 |
|  | <u>good resources</u>
<u>check out JetBrains for good python resources and w3school</u> | 1 |
|  | <u>did anybody else get "True" on this problem...?</u>
<u>3 > 4 or (2 < 3 and 9 > 10) hello everybody, I ran this problem on my Python IDE and I got a true st...</u> | 2 |
|  | <u>can someone explain why (False or False) is true?</u>
<u>i went into a mental maze thinking about this can anyone try to explain it in a simple way for me ...</u> | 7 |
|  | <u>It is real that I had zero only because of a cae letter</u>
<u>even if you are recommending, we are not programming - this is discouraging</u> | 1 |
|  | <u>Here i share the simpleTruth Table for programming you must know:</u> | 1 |
|  | <u>Exercise 8 Post-Reflection + Sandbox (for course coordinators)</u>
<u>Hi there, I seem to be unable to access the exercise 8 post-reflection and the sandbox at the end ...</u> | 16 |
|  | <u>Compound Propositions in Programming</u>
<u>In programming, these not, and, or operators are called compound propositions. A proposition is</u> | 1 |

