

U0LSCQQNR : <<https://docs.python.org/2/library/functions.html#round>>

U5YQR6Q1K : On documents there is a note about that: Note The behavior of round() for floats can be surprising: for example, round(2.675, 2) gives 2.67 instead of the expected 2.68. This is not a bug: it's a result of the fact that most decimal fractions can't be represented exactly as a float. See Floating Point Arithmetic: Issues and Limitations for more information.

U0LSCQQNR : yup. what's the question?

U5YQR6Q1K : look this function: qtd_lata = round(120 / (18 * 3), 2)

U5YQR6Q1K : returns: 2.0

U5YQR6Q1K : But if i don't use round function

U0LSCQQNR : are you using python 2 or 3?

U1BP42MRS : Related: Division changed in python: <<https://www.python.org/dev/peps/pep-0238/>>

U1BP42MRS : You can just make any of those numbers a float

U5YQR6Q1K : Python 2.7.9

U0LSCQQNR : ```python 2

In [5]: 120 / (18 * 3)

Out[5]: 2

3

>>> 120 / (18 * 3)

2.2222222222222223

```

U1BP42MRS : `120.0 / (18 \* 3)`

U57JJD5S8 : <@U5LNXQH3> I tried using virtualenv to install requirements.txt that way but had huge problems

U1NSCAY6R : or `from \_\_future\_\_ import division`

U1BP42MRS : Or :point\_up::skin-tone-2:

U5YQR6Q1K : >>> from \_\_future\_\_ import division>>> 140 / (18 \* 3)

2.5925925925925926

U5YQR6Q1K : in this case i need 3

U5YQR6Q1K : understood

U1NSCAY6R : or `from \_\_future\_\_ import division`

U5YQR6Q1K : ?

U5YQR6Q1K : If i use round

U5YQR6Q1K : its show 2.0

U1NSCAY6R : if you dont want to switch versions of python, use `from \_\_future\_\_ import division` or change one value to a float

U1NSCAY6R : what happens when you called `round` after using `from \_\_future\_\_ import division`?

U5YQR6Q1K : try run this:

U5YQR6Q1K : from \_\_future\_\_ import divisionround(140 / (18 \* 3), 2)

U5YQR6Q1K : the result is 2.59

U5YQR6Q1K : i need in this case rounding to 3.0

U1NSCAY6R : the second argument to `round` is how many significant digits to give, you're asking it for 2

U1NSCAY6R : hence it being 2.59

U1NSCAY6R : remove the second argument to the round function to get a whole number

U0LSCQQNR : @flasp look at code formatting with slack

U0LSCQQNR : <<https://get.slack.help/hc/en-us/articles/202288908-Format-your-messages#code-blocks>>

U5YQR6Q1K : perfect JOE

U1NSCAY6R : ```from \_\_future\_\_ import division

round(140 / (18 \* 3))

```

U1NSCAY6R : should equal 3.0

U5YQR6Q1K : very good man

U5YQR6Q1K : so easy

U5YQR6Q1K : ahauuaa

U5LNXQH3 : <@U57JJD5S8> virtualenv won't change the underlying problem, unfortunately. It just changes where the final packages get stored

U57JJD5S8 : yeah

U57JJD5S8 : i cant edit setup.py to remove #include io.h for some reason

U5LNXQHN3 : Why do you think there is that include inside setup.py?

U5LNXQHN3 : #include is a C/C++ construct, setup.py is Python