The following pages have a list of expressions. For each expression, first compute the expression in your head, without Python. Write down what you think the value is in the second column of the table. If you have no idea, write ?.

Next, use Python Interpreter to compute the same expression. Type each of these expressions one by one. You may find it easier to copy-and-paste them from this pdf file. Write down Python's result in the third column. You should always fill in the second and third column of a row before moving on to the next row. You want to learn from earlier examples before moving on to the next one.

If the two values are different, you should try to figure out why Python gave the answer that it did. Come up with a reasonable explanation and put it in the final column. You are not being graded on correctness in these labs so make your best guess at what is happening; your answer will help TA understand how to better aid you.

This lab is just supposed to be practice; do not waste too much time trying to figure things out yourself. If you do not understand something, ask TA immediately. It is very important that you understand how each expression is evaluated, so if an answer doesn't make sense, ask someone.

Do not forget to type `import math` before calling functions from math module

*Addopted from lab exercises designed by D. Gries, L. Lee, S. Marschner, W. White for Cornell University

Table 1: int and float Expressions.

Expression	*Expected Value*	*Python Displayed Value*	*Reason for Displayed Value*
2 * 3			
2 * 3.0			
2 ** 3			
2 * 3-2/2			
16/(4/(1/2))			
16/4/(1/2)			
16/(1+2)+2)			
2 * 3**2			
(2 * 3)**2			
2 ** 2 ** 0			
(2 ** 2) ** 0			
-4 - -4 - -4			
6/2			
6/4			
6//4			
6 % 3			
7 % 2			
6.2 % 4			
5<3			

Table 2: function calls

Expression	*Expected Value*	*Python Displayed Value*	*Reason for Displayed Value*
min(25, 4)			
max(25, 4)			
min(25, max(27, 4))			
abs(25)			
abs(-25)			
round(25.6)			
round(-25.6)			
round(25.64, 0)			
round(25.64, 1)			
round(25.64, 2)			
math.ceil(3.7)			
math.floor(3.7)			
type(4)			
type(7//4)			
type(7/4)			

Variables and Assignment Statements

The part involves assignment statements. You need to know the difference between expressions, which you've been working with so far, and assignment statements. Here is an example of an assignment statement:

$$b = 3 < 5$$

In particular, the above assignment statement is executed by Python as follows:

1. Python evaluates the expression on the right-hand side of the $=$ (in this case, $3 < 5$), and
2. then it makes the variable on left-hand side of the $=$, in this case, b, refer to this value.

Because it is not an expression but rather an assignment statement, Python Interpreter will not actually output a result when you type it in; it will just perform the command silently. In the table below, the first column contains either an expression or an assignment statement. If it is an expression, write the value. If it is an assignment statement, you should just write Nothing (we have done the first one for you). Because some of the entries are assignment statements, it is important that you enter the expressions or assignment statement, in exactly the order they are given.

Table 3: Variables and Assignment Statements.

Statement or Expression	*Expected Value*	*Python Displayed Value*	*Reason for Displayed Value*
i = 2	Nothing		
i			
j			
j = 1			
j			
j = j + i			
j			
i			
w = 'Hello'			
i + w			
a=1==2			
a			
a= min(pow(2,3,2), 2)			
a			