LECTURE3

The difference between identifier and variable:

Identifier is used to name a variable, function, structure etc.

Variable is used to name a memory location, which holds a value.

All identifiers are not variable, but all variables are identifiers.

Eg: area//area ()

When there is a float number inside a string, we cannot use int () to transfer the type because it is against the rules.

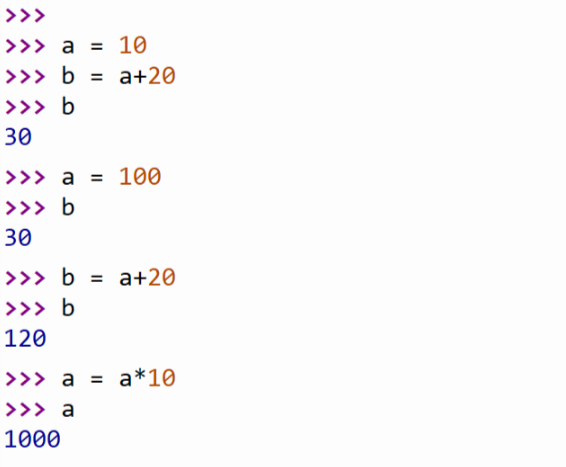
Input ("Please enter the amount you deposit:")->the type is a string “10”/”10.1”

p = int (input ("Please enter the amount you deposit:")) is illegal when the input is a float number.

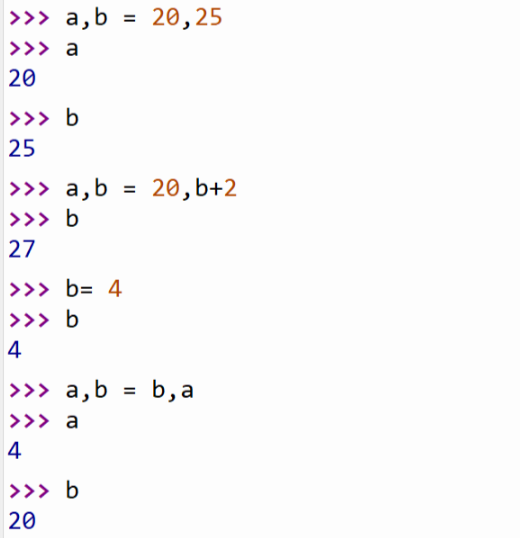
The two correct way to deal with:

1. Just input the int number like “101”
2. p = float (input ("Please enter the amount you deposit:"))

When a number 100 was assigned to a, the value of b is still 30, b = a +20 would not operate automatically. Thus, we need to do b = a + 10 manually.



Type () is used to check the type of variables, like int or float etc.

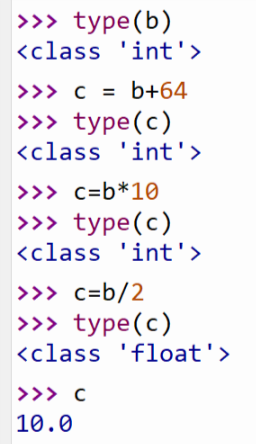


a += 3 -> a = a + 3

a -= 3 -> a = a – 3

a \*= 3 -> a = a \* 3

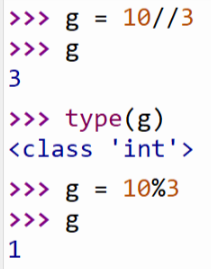
a /= 3 -> a = a / 3 (When you used int numbers or any type of numbers to divide value, the result is always the float number.)



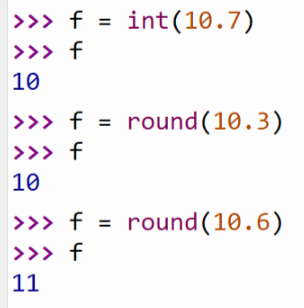
10//3 does integer division

10/3 does floating point division

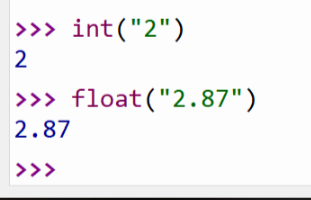
10%3 = 1 is the remainder of the integer division of 10 by 3.

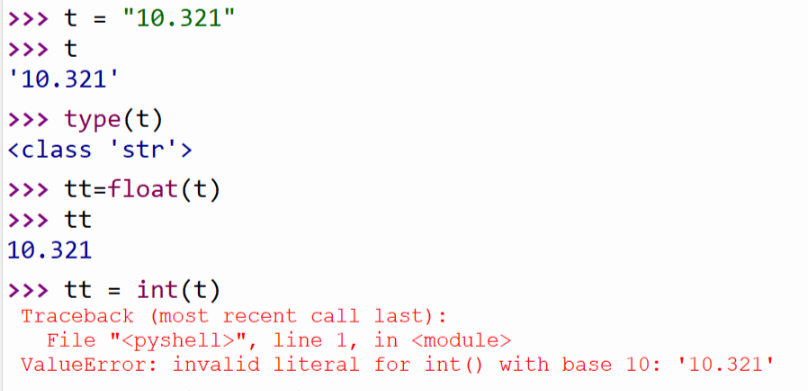


The difference between int () and round ()

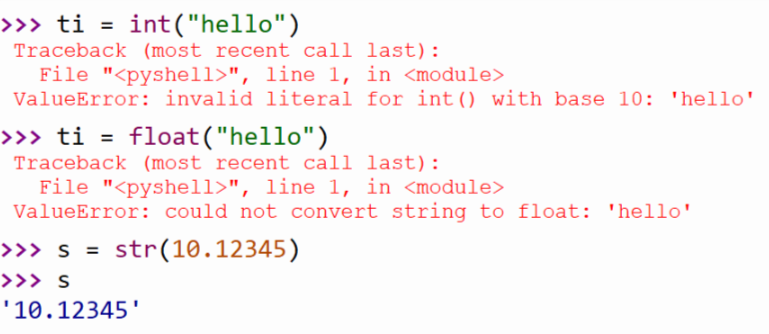


You cannot just transfer the str into int directly when the string has float number inside.





You cannot turn the str into the float or int, but conversely, you can transfer the number into the string.

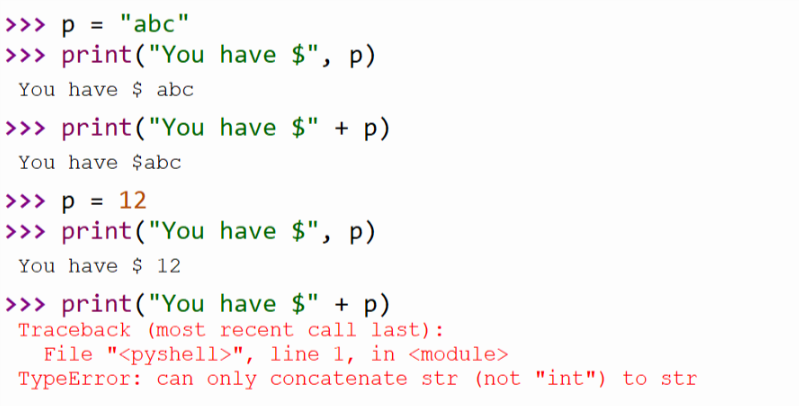


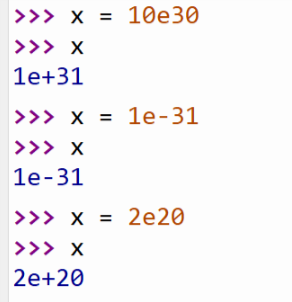
Output the string:

A = print (“the total amount is $” + str (10))

And it shows the total amount is $10.

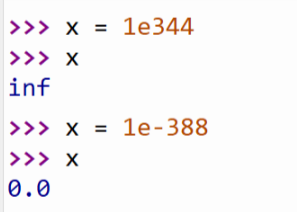
Notice: this type of print can only be used in the string (eg: “12”) , it isn’t used in the number print. However, another way can be both used in the string print and the number print.





Over-flow -> inf

Under-flow -> 0.0



>>> list (range (10))

[0, 1, 2, 3, 4, 5, 6, 7, 8, 9] <- the result

