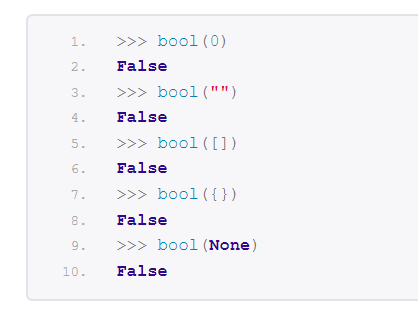
Int

Float : 1.1 1.587

Str : string = text

Value=str(‘hello’)

Bool : logical values , can only be true or false



Boolean=(3>4)

Print(Boolean)

Result: False

Type(a): we can detect the type of the data

print (7 \* 3.) 保留一位小数 ---21.0

print (2\*\*10) 2的十次方 --- 1024

print (8 % 3) 整除 保留整数 --- 2

print(3/2) ---1.5

print(3/2.) ---1.5

print(3.//2.) ---1.0 保留整数

a\_list = [2.,'aa', 0.2]

a\_list[1] ---- 1是第二个数字

another\_list = ['first', 'second', 'third', 'fourth', 'fifth']

print(another\_list[3:])

print(another\_list[:2])

print(another\_list[::2])

['fourth', 'fifth']

['first', 'second']

['first', 'third', 'fifth']

a\_third\_list.append('pink') ---在结尾加上一个element

a\_third\_list.pop()----去掉最后一个element并返回list

如果不写东西默认是最后一个，加上数字就可以选择去掉哪个

a\_third\_list.reverse() 颠倒顺序

a\_third\_list.remove('white') 去掉某一个具体的element，不能使用数字代替

a\_third\_list = a\_third\_list[:-2]

a\_third\_list

去掉最后两个element

a 1,2,3

b 4,5,6

a.append(B)

([1,2,3][4,5,6])

a\_concatenated\_list = a\_third\_list + a\_list 可以两个list直接相加

number\_list = [1, 2, 3, 4]

[i\*\*2 for i in number\_list]

直接处理i里面的数字

[i\*\*2 for i in number\_list if i>1]

downstairs=areas[0:6]

不包括最后一个数字，所以如果要前六个的话要写0：6

# Create list areas

areas = [11.25, 18.0, 20.0, 10.75, 9.50]

areas\_copy = areas[0:]

areas\_copy[0] = 5.0

print(areas)

如果改areas\_copy的内容也会连着areas一起改，所以可以通过define areas来避免改变areas内容, for example list(areas) or areas[0 :]

% = modulo (remainder)

Pour avoir le reste

8%3=2 取余

In python we use try and catch for exception handling

NO



All the python arithmetic operators

Iterate through the list ‘numbers’

For number in numbers