Author：Gaoyang

运算

abs(x)：求x的绝对值

In [70]:

```
abs(-10)
```

Out[70]:

```
10
```

x**y: 求x的y次方

In [71]:

```
2**8
```

Out[71]:

```
256
```

round(x)：四舍五入，不保留小数

round(x, d)：四舍五入，保留d位小数

In [72]:

```
round(3.5)
```

Out[72]:

```
4
```

In [73]:

```
round(3.5555, 2)
```

Out[73]:

```
3.56
```

int(x)：将x取整，舍去小数部分

In [74]:

```
int(3.5555)
```

Out[74]:

```
3
```

float(x)：将x变成浮点数，增加小数部分

In [75]:

```
float(3)
```

Out[75]:

```
3.0
```

bin(x): 将x转换为2进制

In [76]:

```
bin(17)
```

Out[76]:

```
'0b10001'
```

字符串

string.title()：让每个单词的首字母都改成大写

In [77]:

```
"harrow school".title()
```

Out[77]:

```
'Harrow School'
```

string.lower()：字符串全字符小写

In [78]:

```
"ALL CAPS".lower()
```

Out[78]:

```
'all caps'
```

string.upper()：字符串全字符大写

In [79]:

```
"all lower".upper()
```

Out[79]:

```
'ALL LOWER'
```

string.replace(old, new)：所有old的子串被new子串替换

In [80]:

```
"abababab a".replace("ab","xy")
```

Out[80]:

```
'xyxyxyxya'
```

string.strip()：同时删去字符串两端空白

In [81]:

```
"    123     ".strip()
```

Out[81]:

```
'123'
```

string.strip(chars)：删去字符串两边出现的字符chars

In [82]:

```
"aaaaaaa123aaaaa".strip("a")
```

Out[82]:

```
'123'
```

string.count('xxx')：获取xxx在string中出现的次数

In [83]:

```
"thisisanapple".count("a")
```

Out[83]:

```
2
```

string.split()：以空格为分隔符将字符串拆分成多个部分，并储存在一个列表里

In [84]:

```
"harrow school is a good school".split()
```

Out[84]:

```
['harrow', 'school', 'is', 'a', 'good', 'school']
```

string.split(sep)：返回列表，根据sep将字符串分隔

In [85]:

```
"131-1000-9999-19199-13219".split("-")
```

Out[85]:

```
['131', '1000', '9999', '19199', '13219']
```

len(string)：获取字符串的长度

In [86]:

```
len("howlongisthisstring")
```

Out[86]:

```
19
```

ord(x)：获取x的Unicode编码

In [87]:

```
ord("A")
```

Out[87]:

```
65
```

str(num)：将数字num转换成字符串

In [88]:

```
str(100)
```

Out[88]:

```
'100'
```

string[::-1]：倒转字符串

In [89]:

```
"hellothere"[::-1]
```

Out[89]:

```
'erehtolleh'
```

"{} {}".format("hello", "world")：不设置指定位置，按默认顺序输出 hello world

In [90]:

```
"{} is from {}".format("CM", "China")
```

Out[90]:

```
'CM is from China'
```

列表

list= []：创建空列表

In [91]:

```
a = []
print(a)
```

```
[]
```

range(firstNum, lastNum)：生成一系列数字，包含firstNum，不包含lastNum。通常在遍历时使用，如for num in range(1, 5)

range(firstNum, lastNum, step)：指定步长生成一系列数字，包含firstNum，不包含lastNum，步长为step

In [92]:

```
for i in range(1,10):
    print(i,i**2)
```

```
1 1
2 4
3 9
4 16
5 25
6 36
7 49
8 64
9 81
```

In [93]:

```
for i in range(1,10,2):
    print(i,i**2)
```

```
1 1
3 9
5 25
7 49
9 81
```

list = [num**2 for num in range(1s, 11)]：使用列表解析生成数字列表

In [94]:

```
a = [x**2 for x in range(1,10)]
print(a)
```

```
[1, 4, 9, 16, 25, 36, 49, 64, 81]
```

sum(list)：获取数字列表中的总和

In [95]:

```
a = [1,2,3,4,50]
print(sum(a))
```

```
60
```

list.append(object)：将object元素添加到列表末尾

In [96]:

```
a = [1,2,3]
a.append(4)
print(a)
```

```
[1, 2, 3, 4]
```

list.insert(index, object)：将object元素添加到列表任意位置，需要指定索引index

In [97]:

```
a = [1,2,3]
a.insert(1,"here")
print(a)
```

```
[1, 'here', 2, 3]
```

list.remove(object)：将列表中元素object删除，只删除第一个指定的值

In [98]:

```
a = [1,2,"xy",3]
a.remove("xy")
print(a)
```

```
[1, 2, 3]
```

list.sort()：根据首字母对列表进行排序，永久性修改

In [99]:

```
a = [256,45,45,15,52]
a.sort()
print(a)
```

```
[15, 45, 45, 52, 256]
```

list.reverse()：反转列表排列顺序

In [100]:

```
a = [1,2,3,4]
a.reverse()
print(a)
```

```
[4, 3, 2, 1]
```

len(list)：获取list列表的长度

In [101]:

```
a = [1,"a","b"]
len(a)
```

Out[101]:

```
3
```

list[: lastNum]：切片，获取从头开始到lastNum-1的元素组成新列表

In [102]:

```
a = [0,1,2,3,4,5,6,7]
print(a[:3])
```

```
[0, 1, 2]
```

字典

In [103]:

```
alien0 = {'color': 'green', 'points': 5}
print(alien0)
```

```
{'color': 'green', 'points': 5}
```

In [104]:

```
favorite_languages = {
    'jen': 'python',
    'sarah': 'ruby',
    'edward': 'ruby',
    'phil': 'python',
    }
print(favorite_languages)
```

```
{'jen': 'python', 'sarah': 'ruby', 'edward': 'ruby', 'phil': 'python'}
```

map[key]：根据key获取value，如果key为字符串需要用引号标识

In [105]:

```
a = {"name":"GY","level":10}
a["name"]
```

Out[105]:

```
'GY'
```

map.items()：返回字典中所有键值对的信息。可用于用于map字典的遍历，如：for key, value in map.items()

In [106]:

```
a = {"name":"GY","level":10}
for key, value in a.items():
    print(key,value)
```

```
name GY
level 10
```

map[key] = value：给map字典新增一对键值对 / 或修改某键的值

In [107]:

```
a = {"name":"GY","level":10}
a["level"] = 10+3
a
```

Out[107]:

```
{'name': 'GY', 'level': 13}
```

key in map：判断键key是否在字典map中

In [108]:

```
a = {"name":"GY","level":10}
print("name" in a)
```

```
True
```

输入输出

content = input("mesage")：获取用户输入内容

In [109]:

```
m = input("message:")
print(m)
```

```
message:hello!
hello!
```

print(content)：将content打印输出到控制台

In [110]:

```
print("hello world...!")
```

```
hello world...!
```

print(content,end=end)：将content打印输出到控制台，用end作为结束

In [111]:

```
print("hello",end="**")
print("hello",end="&&")
print("hello",end=">>")
```

```
hello**hello&&hello>>
```

随机数

记得导入random模块

In [112]:

```
import random
```

randint(a, b)：生成一个[a , b]之间的整数

In [113]:

```
random.randint(1,5)
```

Out[113]:

```
1
```

choice(seq)：从序列seq中堆积选择一个元素

In [114]:

```
a = ["apple","orange","grape","peach"]
random.choice(a)
```

Out[114]:

```
'peach'
```

uniform(a, b)：生成一个[a, b]之间的随机小数

In [115]:

```
random.uniform(9,10)
```

Out[115]:

```
9.078288058707354
```

shuffle(seq)：将序列seq中的元素随机排列，返回打乱后的序列

In [116]:

```
a = [1,2,3,4,5,6]
random.shuffle(a)
a
```

Out[116]:

```
[3, 5, 4, 2, 1, 6]
```

逻辑 ¶

True: 真

In [119]:

```
10>2
```

Out[119]:

```
True
```

False: 假

In [120]:

```
10<2
```

Out[120]:

```
False
```

in: 包含

In [117]:

```
"a" in "apple"
```

Out[117]:

```
True
```

and: 交集

In [122]:

```
x = 1
x<2 and x>0
```

Out[122]:

```
True
```

not: 反面

In [123]:

```
not 2<10
```

Out[123]:

```
False
```