杂记

1. “\*”和“\*\*”的拆封功能

“\*”可以拆封：元组、列表和字符串

“\*\*”可以拆封字典

*------元组打散-------*

(1, 2, 3, 4)

1 2 3 4

*------列表打散-------*

['a', 'b', 'c']

a b c

*------字符串打散-------*

hello

h e l l o

*------字典打散-------*

张三 20

tup = (1,2,3,4)

print(tup)

print(\*tup)

lst = ['a', 'b', 'c']

print(lst)

print(\*lst)

s = 'hello'

print(s)

print(\*s)

def func\_dic(name, age): #字典的拆包

print(name, age) # \*\*将字典打散

dic = {'name': '张三', 'age':20}

func\_dic(\*\*dic) # 打散的只是值，且本句与 func\_dic(name='张三'， age=20)等价

print(\*\*dic) # 报错

1. Python 中有关下标的集合都满足左闭右开原则

3、列表切片操作还可以设置步长，如：a[1: :2]表示从切片索引 1 到列表末尾，每隔两个元素取值。

4、检查列表是否为空：

if list\_name: # 列表不为空

pass

else: # 列表为空

pass

5、想要创建空集合，必须使用set()而不是{}，后者用于创建空字典。

6、匿名函数：lambda（） （λ） map（）… …

7、集合a和集合表的对称差集（symmetric difference）：存在于a或b但不同时存在。

8、集合是一个无序不重复元素的集。可以用来进行关系测试（'orange' in basket）和消除重复元素。

9、从集合中添加（.add()）或弹出（.pop()）元素，都是随机的。

10、可以使用 enumerate()遍历列表（或任何序列类型），同时获得元素索引值（for i, j in enumerate(['a', 'b', 'c'])）

11、zip() 函数用于将多个可迭代的对象（列表、元组、集合等）作为参数，将对象中对应的元素打包成一个个元组，然后返回由这些元组组成的列表。如果各个迭代器的元素个数不一致，则返回列表长度与最短的对象相同。利用“\*”号操作符，可以将元组解压为列表。

* 12、rjust() 返回一个原字符串右对齐,并使用空格填充至长度 width 的新字符串。如果指定的长度小于字符串的长度则返回原字符串。（str.rjust(width[, fillchar]) width -- 指定填充指定字符后中字符串的总长度；fillchar -- 填充的字符，默认为空格)