### 自带的方法

#### 需要重新赋值

1. title() 单词首字母大写
2. upper() 全部大写
3. lower() 全部小写
4. strip() 去掉两边的空格 lstrip() rstrip()
5. str() 转成字符串
6. set(数组) 集合 去掉数组里重复的值, 得到{集合} for value in set(数组)
7. split(标识) 字符串切分成数组

###### 数组

1. sorted(数组) 顺序 sorted(数组, reverse = True) 反序
2. range(1, 5, 步长) 循环输出1~4, 直接输出range(1, 5)
3. list(range(1, 5)) 输出数组[1, 2,3,4]
4. min(数组)
5. max(数组)
6. sum(数组)
7. 数组[索引1: 索引2] 切片,输出也是数组 索引1省略,默认从0开始, 索引2省略,默认到结尾 数组[-3:] 输出最后3位

#### 直接改变原来的值

###### 数组

1. append() 末尾添加元素
2. insert(索引, 新元素) 任何位置插入元素
3. del 数组[索引] / 字典[键] 删除元素
4. pop() 删除列表末尾元素, 返回被删除的值 pop(索引)
5. remove(元素值) 删除已知元素值 重复值, 只删除第一个, 全部删除要用循环
6. sort() 顺序 sort(reverse = True) 反序
7. reverse() 反转列表顺序

#### 列表(数组)

1. 索引-1, 返回最后一个元素, -2, 倒数第二个d
2. len(数组) 获取数组的长度
3. 元祖: 不能被修改的数组a = (1, 2), 但可以整个数组重新赋值
4. ‘a值’in sList 判断a是否在列表中 not in 不在列表中

#### 其他

1. If-elif-else
2. 字典遍历: for key, value in a.items():
3. a.keys() 获取字典的所有key -> 数组 for key in a
4. a.values() 获取字典的所有value -> 数组 for value in a
5. input 输入的数值是字符串型, 用int(变量) 转成数值
6. while break退出循环 continue
7. def 函数名(): 定义一个函数, 形参默认空字符串表示可选,并放置末尾,不会报错
8. 向函数传递列表的副本, 就不会更改原来的列表
9. 形参 \*变量名 能够获取所有的实参, 不管实参有多少数量 -> 元祖 \*\*变量名 -> 字典{}
10. 先匹配位置实参和关键字实参, 余下的再到任意数量实参
11. 导入模块 import ; 导入特定的函数 form 模块import 函数1, 函数2, 函数3 as 别名; from 模块 import \* 导入所有函数(不推荐)
12. 首字母大写指的是类
13. 继承: 在子类的init方法中 super().\_init\_(形参)
14. class Car():  
     def \_\_init\_\_(self, name):  
     self.name = name  
      
    class Battery():  
     def \_\_init\_\_(self, size="70"):  
     self.size = size  
     def battery\_size(self):  
     print('打印size: ' + self.size)  
      
    # 继承  
    class MyCar(Car):  
     def \_\_init\_\_(self, size):  
     super().\_\_init\_\_(size)  
     self.battery = Battery()  
      
    my\_car = MyCar('hahhaha')  
    my\_car.battery.battery\_size()

打印size: 70

1. from collections import OrderedDict

favorite\_languages = OrderedDict() OrderedDict类 -> 记录了键值对的添加顺序

1. with open(文件名) as 缓存变量名:
2. 文件读取 read()到达文件末尾时返回一个空字符串
3. readlines() 逐行读取 readline() 逐字读取
4. 文件写入: open(文件名, ‘w’) 变量名.write(写入的内容) 附加模式 ‘a’ 读取和写入’r+’
5. 捕获错误: try 代码 except 报错 (pass,什么都不做) else 正常
6. 存储数据 json.dump(变量名, 文件变量名) 读取json数据 json.load(文件变量名)