

```

#字典
#字典也是一种存储的数据类型，它是以键-值对的形式出现,字典也是可变对象
#键和值永远成对出现
#字典本身是无序的
# dict1={'A':'apple','B':'book'}
# dict2={'B':'book','A':'apple'}
# print(dict1==dict2)

#新增字典中的元素,如果键，在字典中不存在，则新增到字典中，否则就变成了更新
# dict1['c']='Cake'
#如果键已经存在于字典中，则进行值的更新
# dict1['A']='ace'
# print(dict1)
#一次新增或修改多个值，可以用update()方法
# dict1.update({'D':'duck','E':'earth'})
# dict1.update({'A':'cake','B':'blue'})
# dict1.update({'A':'cake','B':'blue','D':'duck','E':'earth'})
# print(dict1)
#删除字典中的值
# del dict1['A'],dict1['B']
# print(dict1)
#清空字典
# dict1={} #重新赋值了一个空字典dict1，内存中的地址重新生成了
# dict1.clear() #清空字典，内存中的地址不变

#判断某个值是否位于字典中，此时是根据键来判断
# if 'cake' in dict1:
#     print('在字典里有')
# else:
#     print('不在字典里')

#字典中的键能存放哪些数据类型？键可以是字符串，数字，元组,不可以是列表，字典，也就是说必须是不可变对象
#值能存放哪些数据类型？值可以存放任意数据类型
# dict3={'A':(1,2)}
#字典的唯一性，定义字典时，如果输入多个相同的键，那么以最后输入的键值对为准
# dict1={'A':'apple','A':'apple2'}
# print(dict1)
# print(dict1.keys()) #显示字典中所有的键，其返回值是类列表
# for i in dict1.keys():
#     print(i) #显示字典中所有的键，其返回值是类列表
#可以将类列表通过list()函数转换为列表
# for i in range(0,len(list(dict1.keys()))):
#     print(list(dict1.keys())[i])
# print(dict1.values()) #显示字典中所有的值，其返回值是类列表
#遍历字典中的值
# for i in dict1.values():
#     print(i)
#遍历字典中所有的元素
# for key,value in dict1.items():
#     print(key,value)

import json
#json格式本身是字符串，以键-值对的形式出现
# data1=''{'

```

```
#     "aac003" : "tom",
#     "tel" : "13959687639",
#     "crm003" : "1",
#     "crm004" : "1"
# }'''
# print(type(data1))
# temp=json.loads(data1) #将json转换为字典
# print(type(temp))
# temp2=json.dumps(temp) #将字典转回为json格式
# print(temp2)

#load与loads的区别，前者是从文件中读取json文本，后者是直接在变量中读取json内容
# with open ('d:/json3.txt') as file1:
#     temp3=json.load(file1)
# print (type(temp3))

#dump与dumps的区别，前者是将数据写入到文件中，后者是直接在代码中转换
# with open ('d:/json30.txt','w+') as file1:
#     json.dump(data1,file1)
#     file1.seek(0)
#     print(file1.read())
```