

python_字典

字典：

字典取值快，通过key去找

字典是无序的，因为其不是按照顺序进行取值的，是按照key和value进行取值的

1、定义

info={} #定义空字典

info={"name":"zhaozhao","age":23,"sex":"男"} #定义字典

2、增加KEY

增加KEY两种方式：

info["key"]=value #字典该KEY存在改变其值，不存该KEY则增加该KEY

info.setdefault(key,value) #字典该KEY存在不改变其值，不存该KEY则增加该KEY

```
info={"name":"zhaozhao"}

info["sex"]=23

info["name"]="zhaozhao1"

print(info)
```

```
D:\study\python\test\venv\Scripts\python.exe D:/study/python/test/dd.py

{'name': 'zhaozhao1', 'sex': 23}
```

```
info={"name":"zhaozhao"}

info.setdefault("sex","男")

info.setdefault("name","zhaozhao2")

print(info)

D:\study\python\test\venv\Scripts\python.exe D:/study/python/test/dd.py

{'name': 'zhaozhao', 'sex': '男'}
```

3、删除

pop: 指定key删除，key不存在会报错

del: 指定key删除，key不存在会报错

popitem: 随机返回并删除字典中的一对键和值。如果字典已经为空，却调用了此方法，就报出KeyError异常

```
info={"name":"zhaozhao","sex":"男","age":"10","班级":"1班","成绩":"90"}
```

```
info.pop("name")
```

```
print("pop:",info)
```

```
del info["sex"]
```

```
print("del:",info)
```

```
info.popitem()
```

```
print("popitem:",info)
```

```
pop: {'sex': '男', 'age': '10', '班级': '1班', '成绩': '90'}
```

```
del: {'age': '10', '班级': '1班', '成绩': '90'}
```

```
popitem: {'age': '10', '班级': '1班'}
```

4、字典取值

如果key存在，2者没有什么区别，如果key不存在，info["key"]这种方式程序报错，info.get("phone")这种方式返回none

```
info={"name":"zhaozhao","sex":"男","age":"10","班级":"1班","成绩":"90"}
```

```
print(info["name"])
```

```
print(info.get('name'))
```

```
print("=====")
```

```
print(info.get('ab'))
```

```
print(info["ab"])
```

5、获取key集合，获取value集合

```
print(info.keys()) #获取字典所有的key
```

```
print(info.values()) #获取字典所有的value
```

```
info={"name":"zhaozhao","sex":"男","age":"10","班级":"1班","成绩":"90"}

print(info.keys())

print(info.values())
```

D:\study\python\test\venv\Scripts\python.exe D:/study/python/test/dd.py

```
dict_keys(['name', 'sex', 'age', '班级', '成绩'])

dict_values(['zhaozhao', '男', '10', '1班', '90'])
```

6、判断key是否存在于字典

`info.haskey("key")` #python2

`key in info` #python3用于这种方式判断key是否存在于字典

```
info={"name":"zhaozhao","sex":"男","age":"10","班级":"1班","成绩":"90"}

cname='name'

dname='zhaozhao'

#key是否在key集合中

if cname in info.keys():

    print("sucess")

#value是否在values集合中

if dname in info.values():

    print("sucess2")

#key是否在字典中

if cname in info:

    print("sucess3")
```

7、一个字典追加另一个字典

```
info={"name":"zhaozhao","sex":"男","age":"10","班级":"1班","成绩":"90"}
```

```
info2={"score":"89"}
```

```
info.update(info2)
```

```
print(info)
```

```
D:\study\python\test\venv\Scripts\python.exe D:/study/python/test/dd.py
```

```
{'name': 'zhaozhao', 'sex': '男', 'age': '10', '班级': '1班', '成绩': '90', 'score': '89'}
```

8、循环字典值

```
info={"name":"zhaozhao","sex":"男","age":"10","班级":"1班","成绩":"90"}
```

```
for k in info:
```

```
    print("%s => %s"%(k,info[k]))
```

```
for k,v in info.items():
```

```
    print("%s => %s"%(k,v))
```

这两种第一种性能上说快一些，第二种性能弱一些