



1-5 总结字典中添加数据的方法

在字典中添加数据的方法有很多种，小慕为大家总结以下几种：

第一种：通过中括号的形式添加

```
dict_1 = {
    "name": "小红"
}
print("在这里打印一下dict只有name：", dict_1)
dict_1["age"] = 20
print("在这里打印一下dict添加了age：", dict_1)
dict_1["name"] = "小李"
print("已经存在key了对应的value会被覆盖：", dict_1)
```

my_sisongwei x

```
C:\Users\user\Envs\django1.1\Scripts\python.exe C:/Users/user
在这里打印一下dict只有name： {'name': '小红'}
在这里打印一下dict添加了age： {'name': '小红', 'age': 20}
已经存在key了对应的value会被覆盖： {'name': '小李', 'age': 20}
```

第二种：字典的内置函数update，无论是添加还是修改都属于更新字典

在update括号中传入一个新的字典key和value就可以了。字典会自己去检测key是否存在，存在就会覆盖。

```
dict_1 = {
    "name": "小红"
}
dict_1.update({"key": "value"})
print("给update括号中传入一个新字典key/value：", dict_1)
dict_1.update({"name": "小青"})
print("如果字典中原本就有key则value会被覆盖：", dict_1)
```

my_sisongwei x

```
C:\Users\user\Envs\django1.1\Scripts\python.exe C:/Users/user/Desktop/
给update括号中传入一个新字典key/value： {'name': '小红', 'key': 'value'}
如果字典中原本就有key则value会被覆盖： {'name': '小青', 'key': 'value'}
```

第三种：字典的setdefault()函数

Python 字典 setdefault() 函数和 get()方法 类似，如果键不存在于字典中，将会添加键并将值设为默认值，如果字典中包含有给定键，则返回该键对应的值，否则返回为该键设置的值。

```
dict_1 = {}

dict_1.setdefault('name', 'xiao')
print("用setdefault添加对应的key/value", dict_1)
dict_1.setdefault('age')
print("如果字典中没有key也没有传入value会添加key默认值为None", dict_1)
```

my_sisongwei x

```
C:\Users\user\Envs\django1.1\Scripts\python.exe C:/Users/user/Desktop/Temp/my_h
用setdefault添加对应的key/value {'name': 'xiao'}
如果字典中没有key也没有传入value会添加key默认值为None {'name': 'xiao', 'age': None}
```

下一节