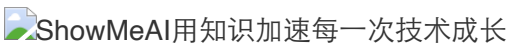


# 图解 Python 编程(16) | 字典

🕒 2021-11-09    👁 1568    💬 0    📁 工具教程 python 编程语言



作者：韩信子@ShowMeAI  
教程地址：<https://www.showmeai.tech/tutorials/56>  
本文地址：<https://www.showmeai.tech/article-detail/79>  
声明：版权所有，转载请联系平台与作者并注明出处  
收藏ShowMeAI查看更多精彩内容

## Python字典(Dictionary)

字典是另一种可变容器模型，且可存储任意类型对象。

字典的每个键值 **key=>value** 对用冒号：分割，每个键值对之间用逗号，分割，整个字典包括在花括号 {} 中 ,格式如下所示：

**d = {key1 : value1, key2 : value2 }**

### 字典 (Dictionary)

```
dict = { key1 : value1 , key2 : value2 , key3 : value3 }
```

字典

key1

key2

key3

value1

value2

value3

Search | Wechat

ShowMeAI 研究中心

<http://www.showmeai.tech/>

键一般是唯一的，如果重复最后的一个键值对会替换前面的，值不需要唯一。

1. >>> dict = {'a': 1, 'b': 2, 'b': '3'}

2. >>> dict['b']

3. '3'

4. >>> dict

5. {'a': 1, 'b': '3'}

6.

7. # dict = {'name': 'runoob', 'likes': 123, 'url': 'www.runoob.com'}

8. dict = {'name': 'ShowMeAI', 'likes': 1000000, 'url': 'www.showmeai.tech'}

### 一个字典的实例

key

value

key

value

key

value

```
dict = {'name': 'ShowMeAI', 'likes': 1000000, 'url': 'www.showmeai.tech'}
```

item1

item2

item3

Search | Wechat

ShowMeAI 研究中心

<http://www.showmeai.tech/>

值可以取任何数据类型，但键必须是不可变的，如字符串，数字或元组。

一个简单的字典实例：

1. dict1 = {'Alice': '2341', 'Beth': '9102', 'Cecil': '3258'}

2.

3. dict2 = { 'abc': 123, 98.6: 37, (1,2):345 }

### 字典部分操作 a = {}

这两种效率更高

判断key - 'name'是否存在	name in a	name in a.keys()	a.has_key('name')
增加某个key -- 'age'	a['age'] = 20		
获取某个key的值	a['name']		
获取所有的value	a.values()		
获取所有的key	a.keys()		
删除key	a.pop('name')	a.pop('name',0)	pop操作有返回值 返回key的value值

如果key不存在, 会报错    增加默认值, 不会报错

Search | Wechat

ShowMeAI 研究中心

<http://www.showmeai.tech/>

## 访问字典里的值

把相应的键放入熟悉的方括弧，下面是代码示例（代码可以在在线python3环境中运行）：

```
1. dict = {'Name': 'ShowMeAI', 'Color': 'Blue', 'Class': 'First'}
2.
3. print(dict['Name']): ", dict['Name'])
4. print(dict['Color']): ", dict['Color'])
```

以上实例执行结果：

```
1. dict['Name']: ShowMeAI
2. dict['Color']: Blue
```

如果用字典里没有的键访问数据，会输出错误如下：

```
1. dict = {'Name': 'ShowMeAI', 'Color': 'Blue', 'Class': 'First'}
2.
3. print(dict['Age']): ", dict['Age'])
```

以上实例输出结果：

```
1. Traceback (most recent call last):
2.   File "<string>", line 5, in <module>
3. KeyError: 'Age'
```

## 修改字典

向字典添加新内容的方法是增加新的键/值对，修改或删除已有键/值如下代码示例（代码可以在在线python3环境中运行）：

```
1. dict = {'Name': 'Zara', 'Age': 7, 'Class': 'First'}
2.
3. dict['Age'] = 8 # 更新
4. dict['School'] = "ShowMeAI" # 添加
5.
6.
7. print(dict['Age']): ", dict['Age'])
8. print(dict['School']): ", dict['School'])
```

以上实例输出结果：

```
1. dict['Age']: 8
2. dict['School']: ShowMeAI
```

## 删除字典元素

能删单一的元素也能清空字典，清空只需一项操作。

显示删除一个字典用del命令，如下实例：

```
1. dict = {'Name': 'Zara', 'Age': 7, 'Class': 'First'}
2.
3. del dict['Name'] # 删除键是'Name'的条目
4. dict.clear()    # 清空字典所有条目
5. del dict        # 删除字典
6.
7. print(dict['Age']): ", dict['Age'])
8. print(dict['School']): ", dict['School'])
```

但这会引发一个异常，因为用del后字典不再存在：

```
1. Traceback (most recent call last):
2.   File "<string>", line 9, in <module>
3. TypeError: 'type' object is not subscriptable
```

注：del()方法后面也会讨论。

### （1）字典键的特性

字典值可以没有限制地取任何python对象，既可以是标准的对象，也可以是用户定义的，但键不行。

两个重要的点需要记住：

- 1) 不允许同一个键出现两次。创建时如果同一个键被赋值两次，后一个值会更新前一个，如下实例：

```
1. dict = {'Name': 'Zara', 'Age': 7, 'Name': 'ShowMeAI'}
2.
3. print(dict['Name']): ", dict['Name'])
```

以上实例输出结果：

```
1. dict['Name']: ShowMeAI
```

- 2) 键必须不可变，所以可以用数字，字符串或元组充当，所以用列表就不行，如下实例：

```
1. dict = {'Name': 'Zara', 'Age': 7}
2.
3. print("dict['Name']:", dict['Name'])
```

以上实例输出结果：

```
1. Traceback (most recent call last):
2. File "<string>", line 3, in <module>
3. TypeError: unhashable type: 'list'
```

## 字典内置函数与方法

Python字典包含了以下内置函数：

函数及描述	作用
cmp(dict1, dict2)	比较两个字典元素。
len(dict)	计算字典元素个数，即键的总数。
str(dict)	输出字典可打印的字符串表示。
type(variable)	返回输入的变量类型，如果变量是字典就返回字典类型。

Python字典包含了以下内置方法：

函数及描述	作用
dict.clear()	删除字典内所有元素
dict.copy()	返回一个字典的浅复制
dict.fromkeys(seq[, val])	创建一个新字典，以序列 seq 中元素做字典的键，val 为字典所有键对应的初始值
dict.get(key, default=None)	返回指定键的值，如果值不在字典中返回default值
dict.has_key(key)	如果键在字典dict里返回true，否则返回false
dict.items()	返回可遍历的(键, 值) 元组数组的视图对象
dict.keys()	返回一个字典所有的键的视图对象
dict.setdefault(key, default=None)	和get()类似, 但如果键不存在于字典中，将会添加键并将值设为default
dict.update(dict2)	把字典dict2的键/值对更新到dict里
dict.values()	返回字典中的所有值的视图对象
pop(key[,default])	删除字典给定键 key 所对应的值，返回值为被删除的值。key值必须给出。否则，返回default值。
popitem()	返回并删除字典中的最后一对键和值。

## 视频教程

也可以点击 [这里](#) 到B站查看有【中英字幕】的版本

【双语字幕+资料下载】Python 3全系列...

去bilibili观看

分享

扫一扫 手机看

00:00 / 13:34

360P

进入bilibili,一起发弹幕吐槽!

去吐槽

## 一键运行所有代码

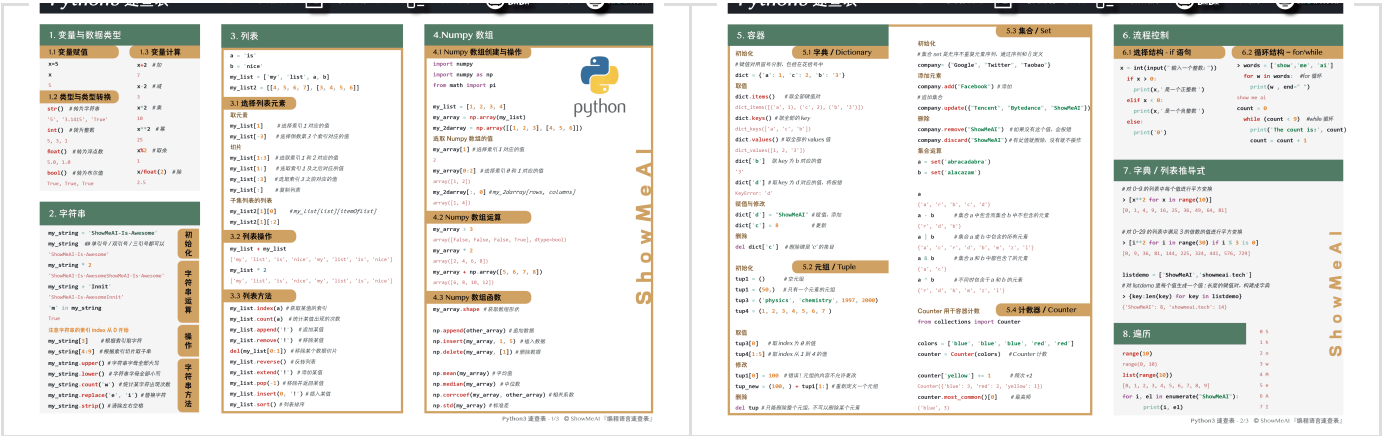
图解Python编程系列 配套的所有代码，可前往ShowMeAI 官方 [GitHub](#)，下载后即可在本地 Python 环境中运行。能访问 Google 的宝宝也可以直接借助 Google Colab一键运行与交互学习！

## 下载Python要点速查表

Awesome cheatsheets | [ShowMeAI速查表大全](#) 系列包含『编程语言』『AI技能知识』『数据科学工具库』『AI垂直领域工具库』四个板块，追平到工具库当前最新版本，并跑通了所有代码。点击 [官网](#) 或 [GitHub](#) 获取~



### Python 速查表（部分）

## 拓展参考资料

- Python教程 - Python3文档
- Python教程 - 廖雪峰的官方网站

## ShowMeAI图解Python编程系列推荐（要点速查版）

- ShowMeAI 图解 Python 编程(1) | 介绍
- ShowMeAI 图解 Python 编程(2) | 安装与环境配置
- ShowMeAI 图解 Python 编程(3) | 基础语法
- ShowMeAI 图解 Python 编程(4) | 基础数据类型
- ShowMeAI 图解 Python 编程(5) | 运算符
- ShowMeAI 图解 Python 编程(6) | 条件控制与if语句
- ShowMeAI 图解 Python 编程(7) | 循环语句
- ShowMeAI 图解 Python 编程(8) | while循环
- ShowMeAI 图解 Python 编程(9) | for循环
- ShowMeAI 图解 Python 编程(10) | break语句
- ShowMeAI 图解 Python 编程(11) | continue语句
- ShowMeAI 图解 Python 编程(12) | pass语句
- ShowMeAI 图解 Python 编程(13) | 字符串及操作
- ShowMeAI 图解 Python 编程(14) | 列表
- ShowMeAI 图解 Python 编程(15) | 元组
- ShowMeAI 图解 Python 编程(16) | 字典
- ShowMeAI 图解 Python 编程(17) | 集合
- ShowMeAI 图解 Python 编程(18) | 函数
- ShowMeAI 图解 Python 编程(19) | 迭代器与生成器
- ShowMeAI 图解 Python 编程(20) | 数据结构
- ShowMeAI 图解 Python 编程(21) | 模块
- ShowMeAI 图解 Python 编程(22) | 文件读写
- ShowMeAI 图解 Python 编程(23) | 文件与目录操作
- ShowMeAI 图解 Python 编程(24) | 错误与异常处理
- ShowMeAI 图解 Python 编程(25) | 面向对象编程
- ShowMeAI 图解 Python 编程(26) | 命名空间与作用域
- ShowMeAI 图解 Python 编程(27) | 时间和日期

## ShowMeAI系列教程精选推荐

- 大厂技术实现：推荐与广告计算解决方案
- 大厂技术实现：计算机视觉解决方案
- 大厂技术实现：自然语言处理行业解决方案
- 图解Python编程：从入门到精通系列教程
- 图解数据分析：从入门到精通系列教程
- 图解AI数学基础：从入门到精通系列教程
- 图解大数据技术：从入门到精通系列教程
- 图解机器学习算法：从入门到精通系列教程
- 机器学习实战：手把手教你玩转机器学习系列
- 深度学习教程：吴恩达专项课程·全套笔记解读
- 自然语言处理教程：斯坦福CS224n课程·课程带学与全套笔记解读
- 深度学习与计算机视觉教程：斯坦福CS231n·全套笔记解读

## 图解 Python 编程(16) | 字典

< 上一篇

图解 Python 编程(15) | 元组

下一篇 >

图解 Python 编程(17) | 集合