

Python 3 教程

Python3 教程

Python3 简介

Python3 环境搭建

Python3 VScode

Python3 基础语法

Python3 基本数据类型

Python3 解释器

Python3 注释

Python3 运算符

Python3 数字 (Number)

Python3 字符串

Python3 列表

Python3 元组

Python3 字典

Python3 集合

Python3 编程第一步

Python3 条件控制

Python3 循环语句

Python3 迭代器与生成器

Python3 函数

Python3 数据结构

← Python3 标准库概览

Python 测验 →

分类导航

HTML / CSS

JavaScript

服务端

数据库

数据分析

移动端

XML 教程

ASP.NET

Web Service

开发工具

网站建设

Advertisement

Python 按键(key)或值(value)对字典进行排序



给定一个字典，然后按键(key)或值(value)对字典进行排序。

实例1：按键(key)排序

```
def dictionary():  
  
    # 声明字典  
    key_value = {}  
  
    # 初始化  
    key_value[2] = 56  
    key_value[1] = 2  
    key_value[5] = 12  
    key_value[4] = 24  
    key_value[6] = 18  
    key_value[3] = 323  
  
    print ("按键(key)排序:")  
  
    # sorted(key_value) 返回重新排序的列表  
    # 字典按键排序  
    for i in sorted(key_value):  
        print ((i, key_value[i]), end = " ")  
  
def main():  
    # 调用函数  
    dictionary()  
  
# 主函数  
if __name__ == "__main__":  
    main()
```

执行以上代码输出结果为：

按键(key)排序：

(1, 2) (2, 56) (3, 323) (4, 24) (5, 12) (6, 18)

实例2：按值(value)排序

```
def dictionary():  
  
    # 声明字典  
    key_value = {}
```

反馈/建议



Python3 模块

Python3 输入和输出

Python3 File

Python3 OS

Python3 错误和异常

Python3 面向对象

Python3 命名空间/作用域

Python3 标准库概览

Python3 实例

Python 测验

Python3 高级教程

Python3 正则表达式

Python3 CGI编程

Python3

MySQL(mysql-connector)

Python3

MySQL(PyMySQL)

Python3 网络编程

Python3 SMTP发送邮件

Python3 多线程

Python3 XML 解析

Python3 JSON

Python3 日期和时间

Python3 内置函数

```
# 初始化
key_value[2] = 56
key_value[1] = 2
key_value[5] = 12
key_value[4] = 24
key_value[6] = 18
key_value[3] = 323

print ("按值(value)排序:")
print(sorted(key_value.items(), key = lambda kv:(kv[1], kv[0])))

def main():
    dictionary()

if __name__=="__main__":
    main()
```

执行以上代码输出结果为:

按值(value)排序:

[(1, 2), (5, 12), (6, 18), (4, 24), (2, 56), (3, 323)]

实例 3 : 字典列表排序

```
lis = [{ "name" : "Taobao", "age" : 100},
{ "name" : "Runoob", "age" : 7 },
{ "name" : "Google", "age" : 100 },
{ "name" : "Wiki" , "age" : 200 }]

# 通过 age 升序排序
print ("列表通过 age 升序排序: ")
print (sorted(lis, key = lambda i: i['age']) )

print ("\r")

# 先按 age 排序, 再按 name 排序
print ("列表通过 age 和 name 排序: ")
print (sorted(lis, key = lambda i: (i['age'], i['name'])))

print ("\r")

# 按 age 降序排序
print ("列表通过 age 降序排序: ")
print (sorted(lis, key = lambda i: i['age'],reverse=True) )
```

执行以上代码输出结果为:

列表通过 age 升序排序:

[{'name': 'Runoob', 'age': 7}, {'name': 'Taobao', 'age': 100}, {'name': 'Google', 'age': 100}, {'name': 'Wiki', 'age': 200}]

列表通过 age 和 name 排序:

[{'name': 'Runoob', 'age': 7}, {'name': 'Google', 'age': 100}, {'name': 'Taobao', 'age': 100}, {'name': 'Wiki', 'age': 200}]

QingClo

英特尔® 云创新

全/网/底

云服务

1核2

首购

¥59

续购1折

免费DNS

立即

股票代码



反馈/建议

Python3
MongoDB

Python3 urllib

Python uWSGI 安
装配置

Python3 pip

列表通过 `age` 降序排序:

```
[{'name': 'Wiki', 'age': 200}, {'name': 'Taobao', 'age': 100}, {'name': 'Google', 'age': 100}, {'name': 'Runoob', 'age': 7}]
```



[Python3 实例](#)

← Python3 标准库概览

Python 测验 →

点我分享笔记

在线实例

- [HTML 实例](#)
- [CSS 实例](#)
- [JavaScript 实例](#)
- [Ajax 实例](#)
- [jQuery 实例](#)
- [XML 实例](#)
- [Java 实例](#)

字符集&工具

- [HTML 字符集设置](#)
- [HTML ASCII 字符集](#)
- [HTML ISO-8859-1](#)
- [HTML 实体符号](#)
- [HTML 拾色器](#)
- [JSON 格式化工具](#)

最新更新

- [CSS clip-path 属性](#)
- [CSS @charset 规则](#)
- [CSS grid-row 属性](#)
- [CSS grid-template...](#)
- [CSS grid-template...](#)
- [CSS grid-template...](#)

站点信息

- [意见反馈](#)
- [免责声明](#)
- [关于我们](#)
- [文章归档](#)

关注微信

