

# python标准库 configparser读取config或ini配置文件



木头人

Python + C 法力无边

10 人赞同了该文章

## 1、config 或ini文件格式

config 配置文件由两部分组成sections与items。

sections 用来区分不同的配置块

items 是sections下面的键值

格式如下：应用有多种语言环境，不同的语言采用不同的配置

```
# zh_cn.config(UTF-8)
[lang]
name=中文简体

[message]
applyLangTip          = 重启程序来应用更改。
runCommands           = 执行命令

[menu]
id                    = 96
service              = 服务
help                 = 帮助
officialSite         = 官网
officialHelp         = 帮助文档

[UI]
title                = 集成运行环境
stop                 = 停止
startZentao          = 启动
account              = 账号
password             = 密码
```

中括号([])中的为sections，每个sections可以有多个items。

## 2、configparser基础

python 对于配置文件的读取已经标准化，标准库为configparser。

引入包

```
from configparser import ConfigParser
```

实例化

```
config = ConfigParser()
```

### 3、读取config文件数据

常用方法：

**config.read(filename,encoding)** 直接读取ini文件内容,finlename 文件地址，encoding 文件编码格式

**config.sections()** 得到所有的section，并以列表的形式返回

**config.options(section)** 得到该section的所有option

**config.items(section)** 得到该section的所有键值对

**config[section][option]** 读取section中的option的值

**config.get(section,option)** 得到section中option的值，返回为string类型

**config.getint(section,option)** 得到section中option的值，返回为int类型

**config.getboolean(section,option)** 得到section中option的值，返回为bool类型

**config.getfloat(section,option)** 得到section中option的值，返回为float类型

示例：读取zh\_cn.config 文件

```
from configparser import ConfigParser

config = ConfigParser()
# 传入读取文件的地址，encoding文件编码格式，中文必须
```

```
config.read('zh_cn.config', encoding='UTF-8')

# 获取Lang 下name的值, 不存在抛出KeyError异常
print('lang>name:', config['lang']['name'])

menu = config['menu']
print('menu>officialSite:', menu['officialSite']) # 获取officialSite值

# 判断配置文件中是否存在menu块
print('menu' in config)

# 得到所有的section
print('sections:', config.sections())

# 得到该menu的所有option选项, 不存在NoSectionError异常
print('options:', config.options('menu'))

# 得到该menu的所有键值对, 不存在NoSectionError异常
print('items:', config.items('menu'))

# 得到menu中service的值, 返回为string类型, 不存在NoSectionError或NoOptionError异常
print('menu>service(str):', config.get('menu', 'service'))

# 得到 menu 中 id 的值, 返回为 int 类型
print('menu>id(int):', config.getint('menu', 'id'))

# 用法比较灵活
print('menu>id(int):', config['menu'].getint('id'))
```

## 4、新增或修改config文件数据

**config.add\_section(section)** 添加一个新的section

**config.set( section, option, value)** 对section中的option进行设置

**config.write(open(path, "w"))** 将修改的内容写回配置文件

**config[section][option]=value** 修改或在新增值

示例一：通过set方法添加值，重新创建配置文件

```
from configparser import ConfigParser

config = ConfigParser()
config.add_section('table') # 添加table section
config.set('table', 'order_th', '订单号,申请人,状态') # 对config添加值
config.set('table', 'user_th', '用户名,权限,状态') # 对config添加值
with open('config.ini', 'w', encoding='utf-8') as file:
    config.write(file) # 值写入配置文件
```

## 示例二：通过字典添加配置，重新创建配置文件

```
from configparser import ConfigParser

config = ConfigParser()
config['table'] = {'order_th': '订单号,申请人,状态', 'user_th': '用户名,权限,状态'}
with open('config.ini', 'w', encoding='utf-8') as file:
    config.write(file) # 数据写入配置文件
```

## 示例三：修改配置文件数据

```
from configparser import ConfigParser

path = 'zh_cn.config'
config = ConfigParser()
# 对文件修改必须先将文件读取到config
config.read(path, encoding='UTF-8')
config['table'] = {'order_th': '订单号,申请人,状态', 'user_th': '用户名,权限,状态'}
config['menu']['help'] = '帮助(F1)' # 对于存在的值则是修改
config['menu']['tool'] = '工具' # 对于不存在的值时增加
config.set('menu', 'help', '帮助(F1)') # 方法新增必须有sections
fo = open(path, 'w', encoding='UTF-8') # 重新创建配置文件
config.write(fo) # 数据写入配置文件
fo.close()
```

注意：配置文件修改，先要读取配置文件，然后修改其中数据。

发布于 2018-01-23