

enjoyzier

Python之CSV模块

博客园 首页 新随笔 联系 管理 订阅

随笔- 50 文章- 0 评论- 7

昵称: enjoyzier
园龄: 2年2个月
粉丝: 12
关注: 0
+加关注

< 2019年11月 >

日	一	二	三	四	五	六
27	28	29	30	31	1	2
3	4	5	6	7	8	9
10	11	12	13	14	15	16
17	18	19	20	21	22	23
24	25	26	27	28	29	30
1	2	3	4	5	6	7

Python之CSV模块

1. CSV简介

CSV(Comma Separated Values)是逗号分隔符文本格式，常用于Excel和数据库的导入和导出，Python标准库的CSV模块提供了读取和写入CSV格式文件的对象。

1.1 csv.reader对象和csv文件的读取

csv.reader(csvfile, dialect='excel', **fmtparams)，主要用于文件的读取，返回一个reader对象用于在csv文件内容上进行行迭代。

参数csvfile是文件对象或者list对象;dialect 用于指定csv的格式模式不同程序输出的csv格式有细微差别；fmtparams是一系列参数列表，主要用于设置特定的格式，以覆盖dialect中的格式。

csv.reader对象是可迭代对象，包含以下属性:

- 1. csv.reader().dialect #返回其dialect
- 2. csv.reader().line_num #返回读入的行数

示例1：使用reader对象读取csv文件scores.csv，文件内容如下：

学号,姓名,性别,班级,语文,数学,英语
100001,小雨,女,1班,72,85,87
100002,小雪,女,2班,67,87,77
100003,小宇,男,3班,88,78,78
100004,小天,男,1班,76,87,84
100005,小军,男,3班,79,86,83

+

reader对象

1.2 csv.writer对象和csv文件的写入

csv.writer(csvfile, dialect='excel', **fmtparams)，主要用于把列表数据写入到csv文件。

其中参数csvfile是任何支持write()方法的对象，通常为文件对象；dialect 和fmtparams与csv.reader对象构造函数中的参数意义相同。

csv.writer对象包含以下属性和方法:

- 1. writer.writerow(row) #方法，写入一行数据
- 2. writer.writerows #方法，写入多行数据
- 3. writer.dialect #只读属性，返回其dialect

示例2：使用writer对象写入csv文件

📄

```
import csv
def writecsv(csvfilepath):#列表方式写入
    rows=[(100006, '小江', '男', '1班', '77', '79', '80'), (100007, '小美', '女', '4班', '77', '88', '80')]
    with open(csvfilepath, 'a+', newline='') as csvfile:
        writer = csv.writer(csvfile, dialect='excel')
        writer.writerows(rows)#写入多行
        print(writer.dialect)

if __name__ == '__main__':
```

搜索

找找看

谷歌搜索

常用链接

- 我的随笔
- 我的评论
- 我的参与
- 最新评论
- 我的标签

随笔分类

- CSS(1)
- django(11)
- ffmpeg/rtmp(1)
- git和jenkins(2)
- HTML(2)
- JS(1)
- MongoDB(1)
- mysql(3)
- python常用库及扩展库(13)
- pywinauto(2)
- selenium(2)
- 接口测试(4)
- 爬虫(9)

随笔档案

- 2019年4月(4)
- 2019年3月(10)
- 2019年2月(2)
- 2019年1月(2)
- 2018年9月(6)
- 2018年8月(5)
- 2018年7月(8)
- 2018年6月(11)
- 2018年5月(2)

最新评论

```
writecsv(r'E:\2018-12-19\scores.csv')
```

1.3 csv.DictReader对象和csv文件的读取

使用csv.reader对象从csv文件读取数据，结果为列表对象row，需要通过索引row[i]访问。如果希望通过csv文件的首行标题字段名访问，则可以使用csv.DictReader对象读取。

csv.DictReader(csvfile,fieldnames=None,restkey=None,restval=None,dialect='excel',*args,**kwds)

其中，csvfile是文件对象或list对象；fieldnames用于指定字段名，如果没有指定，则第一行为字段名；restkey和restval用于指定字段名和数据个数不一致时所对应的字段名或数据值，其他参数同reader对象。

DictReader对象属性和方法:

方法:
csv.DictReader().__next__()# 称之为next(reader)
属性:
csvreader.dialect # 解析器使用的方言的只读描述。
csvreader.line_num #返回读入的行数
csvreader.fieldnames #返回标题字段名

示例3：使用DictReader对象读取csv文件

```
import csv
def readcsv2(csvfilepath):
    with open(csvfilepath,newline='') as f:
        f_csv = csv.DictReader(f)
        for row in f_csv:
            print(row['姓名'],row['班级'])
            # print('fieldnames:',f_csv.fieldnames)
            # print('dialect:',f_csv.dialect)
            # print('line_num:',f_csv.line_num)

if __name__=='__main__':
    readcsv2(r'E:\2018-12-19\scores.csv')
```

1.4 csv.DictWriter对象和csv文件的写入

csv.DictWriter(csvfile,fieldnames,restval = "",extrasaction = 'raise',dialect = 'excel',*args,**kwds)

extrasaction用于指定多余字段时的操作，其他参数同上。

DictWriter对象的属性和方法:

方法:
csvwriter.writerow(row) # 将row写入writer的文件对象,根据当前方言进行格式化。支持迭代
csvwriter.writerows(rows) # 将行中的所有元素写入编写器的文件对象,并根据当前方言进行格式化。支持迭代
DictWriter.writeheader() # 写入标题字段名
属性:
csvwriter.dialect # 使用的方言只读描述

示例4：使用DictWriter对象写入csv文件

```
import csv
def writecsv2(csvfilepath):
    headers = ['学号','姓名','性别','班级','语文','数学','英语']
    rows = [{ '学号':'100001','姓名':'小鱼','性别':'男','班级':'1班','语文':'72','数学':'82','英语':'85'},
            { '学号':'100002','姓名':'小高','性别':'女','班级':'6班','语文':'74','数学':'88','英语':'85'}
    ]
    with open(csvfilepath,'a+',newline='') as f:
        f_csv = csv.DictWriter(f,headers)
        f_csv.writerows(rows)

if __name__=='__main__':

    writecsv2(r'E:\2018-12-19\scores.csv')
```

1. Re:mysql新建用户，修改权限
好好
- 浩然正气
2. Re:win10下一分钟快速搭建rtmp推流服务器
自己笔记本按照此方法搭建成功了，但阿里云服务器下的win7系统，nginx提示Oxc000007b，.netvc++也都弄了，最终没解决
- 张亚南
3. Re:selenium抓取动态网页数据
您好！我是51testing软件测试网小编，您的这篇文章写得很好，我们想要转载让更多人看到，希望您能够同意，非常感谢！
- 51testing软件测试
4. Re:Django_学生管理系统
博主 teacher.py没有写吗？
而且 views下的 ajax也没有 能提供一下吗？
- wxc_dz
5. Re:大话软件测试(欧立奇)---软件测试入门书籍推荐
@ 柯汐新链接： 提取码： ywev 。建议先找本软件测试理论看，<<大话软件测试>>作为闲时看...
- enjoyzier

阅读排行榜

1. mysql新建用户，修改权限(34807)
2. Windows GUI程序自动化之pywinauto(8359)
3. Django的时间字段DateTimeField(4662)
4. python之openpyxl模块(3808)
5. windows系统下Jenkins环境搭建与基本使用(2853)

评论排行榜

1. 大话软件测试(欧立奇)---软件测试入门书籍推荐(2)
2. win10下一分钟快速搭建rtmp推流服务器(1)
3. selenium抓取动态网页数据(1)
4. mysql新建用户，修改权限(1)
5. Django_学生管理系统(1)

推荐排行榜

1. mysql新建用户，修改权限(2)
2. Django之组合查询(1)
3. Windows GUI程序自动化之pywinauto(1)

1.5 csv文件格式化参数和Dialect对象

1.5.1 csv 文件格式化参数

创建reader/writer对象时，可以指定csv文件格式化命名参数。

参数说明：

delimiter 用于分隔字段的分隔符。默认为","

lineterminator 用于写操作的行结束符，默认为"\n"。读操作将忽略此选项，它能认出跨平台的行结束符

quotechar 用于带有特殊字符（如分隔符）的字段的引用符号。默认为'"'

quoting 引用约定。可选值包括

csv.QUOTE_ALL (引用用所有字段)

csv.QUOTE_MINIMAL (引用如分隔符之类特殊字符的字段) 默认

csv.QUOTE_NONNUMERIC (非数字字段)

csv.QUOTE_NON (不引用)

skipinitialspace 忽略分隔符后面的空白符。默认为False

doublequote 如何处理字段内的引用符号。如果为True，字符串中的双引号使用"\"表示；如果为False，使用转义字符escapechar指定的字符

escapechar 用于对分隔符进行转义的字符串

strict 如果为True，读入错误格式的CSV行时将导致csv.Error；默认值为False

示例5： csv文件格式化参数示例

```
import csv
def writecsv3(csvfilepath):
    headers = ['学号', '姓名', '性别', '班级', '语文', '数学', '英语']
    rows = [{ '学号': '100010', '姓名': '小南', '性别': '男', '班级': '1班', '语文': '70', '数学': '89', '英语': '85'},
            { '学号': '100011', '姓名': '小风', '性别': '女', '班级': '6班', '语文': '79', '数学': '89', '英语': '85'}
    ]
    with open(csvfilepath, 'a+', newline='') as f:
        f_csv = csv.DictWriter(f, headers, delimiter = ',', quoting = csv.QUOTE_ALL)
        f_csv.writerows(rows)
if __name__ == '__main__':
    writecsv3(r'E:\2018-12-19\scores.csv')
```

1.5.2 Dialect 对象

若干格式化参数可以组成Dialect对象，Dialect对象包含对应于命名格式化参数的属性。可以创建 Dialect或其派生类的对象，然后传递给reader或writer的构造函数

可以使用下列csv模块的函数，创建Dialect对象。

csv.register_dialect(name[, dialect], **fmtparams):使用命名参数，注册一个名称。

csv.unregister_dialect(name):取消注册的名称。

csv.get_dialect(name):获取注册的名称的Dialect对象，无注册时csv.Error。

csv.list_dialects():所有注册Dialect对象的列表。

另外可以使用csv模块函数，获取和设置字段的长度限制:csv.field_size_limit([new_limit])

示例6： Dialect对象示例

```
import csv
def writecsv4(csvfilepath):
    csv.register_dialect('mydialect', delimiter = '*', quoting = csv.QUOTE_ALL)
    headers = ['学号', '姓名', '性别', '班级', '语文', '数学', '英语']
    rows = [{ '学号': '100013', '姓名': '小北', '性别': '男', '班级': '1班', '语文': '70', '数学': '80', '英语': '85'},
            { '学号': '100014', '姓名': '小琴', '性别': '女', '班级': '6班', '语文': '77', '数学': '89', '英语': '85'}
    ]
    with open(csvfilepath, 'a+', newline='') as f:
        f_csv = csv.DictWriter(f, headers, dialect='mydialect')
        f_csv.writerows(rows)
if __name__ == '__main__':
```

```
writetocsv4(r'E:\2018-12-19\scores.csv')
```



2. 使用pandas处理大型csv文件

2.1 pandas简介

Pandas 即Python Data Analysis Library，是为了解决数据分析而创建的第三方工具，它不仅提供了丰富的数据模型，而且支持多种文件格式处理，包括CSV、HDF5、HTML 等，能够提供高效的大型数据处理。其支持的两种数据结构——Series 和DataFrame——是数据处

理的基础。下面先来介绍这两种数据结构。

Series：它是一种类似数组的带索引的一维数据结构，支持的类型与NumPy兼容。如果不指定索引，默认为0到N-1。通过obj.values() 和obj.index() 可以分别获取值和索引。当给Series 传递一个字典的时候，Series 的索引将根据字典中的键排序。如果传入字典的时候同时重新指定了index 参数，当index 与字典中的键不匹配的时候，会出现时数据丢失的情况，标记为NaN。在pandas 中用函数isnull() 和 notnull() 来检测数据是否丢失。

```
import pandas

>>> obj1 = Series([1, 'a', (1,2), 3], index=['a', 'b', 'c', 'd'])
>>> obj1#value 和index 一一匹配
a 1
b a
c (1, 2)
d 3
dtype: object
>>> obj2=Series({"Book":"Python","Author":"Dan","ISBN":"011334","Price":25},index=
['book','Author','ISBN','Price'])
>>> obj2.isnull()
book True # 指定的index 与字典的键不匹配，发生数据丢失
Author False
ISBN True # 指定的index 与字典的键不匹配，发生数据丢失
Price False
dtype: bool
```

DataFrame：类似于电子表格，其数据为排好序的数据列的集合，每一列都可以是不同的数据类型，它类似于一个二维数据结构，支持行和列的索引。和Series 一样，索引会自动分配并且能根据指定的列进行排序。使用最多的方式是通过一个长度相等的列表的字典来构建。构建一个DataFrame 最常用的方式是用一个相等长度

列表的字典或NumPy 数组。DataFrame 也可以通过columns 指定序列的顺序进行排序。

```
data = {'OrderDate': ['1-6-10', '1-23-10', '2-9-10', '2-26-10', '3-15-10'],
... 'Region': ['East', 'Central', 'Central', 'West', 'East'],
... 'Rep': ['Jones', 'Kivell', 'Jardine', 'Gill', 'Sorvino']}
>>>
>>> DataFrame(data,columns=['OrderDate','Region','Rep'])# 通过字典构建，按照columns指定的顺序排序
OrderDate Region Rep
0 1-6-10 East Jones
1 1-23-10 Central Kivell
2 2-9-10 Central Jardine
3 2-26-10 West Gill
4 3-15-10 East Sorvino
```

2.2 pandas处理csv文件

Pandas中处理CSV文件的函数主要为read_csv()和to_csv()这两个，其中read_csv()读取CSV文件的内容并返回DataFrame,to_csv() 则是其逆过程。

1) 读取指定行和列的数据

具体的实现代码如下：

```
df = pd.read_csv("SampleData.csv",nrows=5,usecols=
['OrderDate','Item','Total'])
```

方法read_csv()的参数nrows 指定读取文件的行数，usecols 指定所要读取的列的列名，如果没有列

名,可直接使用索引0、1、...、n-1。上述两个参数对大文件处理非常有用,可以避免读入整个文件而只选取所需要部分进行读取。

```

import pandas as pd
def readcsv3(csvfilepath):
    df = pd.read_csv(csvfilepath, nrows=3, usecols=['学号', '姓名', '班级'], encoding='gbk')
    print(df)

if __name__ == '__main__':
    readcsv3(r'E:\2018-12-19\scores.csv')
#输出
#           学号  姓名  班级
#    0  100001  小雨  1班
#    1  100002  小雪  2班
#    2  100003  小宇  3班

```

如果出现错误提示: UnicodeDecodeError: 'ascii' codec can't decode byte 0xb8

原因: 含有中文字符, 将encoding设置为gbk

2) 设置CSV文件与excel兼容

将scores.csv文件内容修改如下:

```

学号, 姓名, 性别, 班级, 语文, 数学, 英语
100001, 小雨, 女, 1班, 72, 85, 87
100002, 小雪, 女, 2班, 67, 87, 77
100003, 小宇, 男, 3班, 88, 78, 78
100004, 小天, 男, 1班, 76, 87, 84
100005, 小军, 男, 3班, 79, 86, 83
100006, 小江, 男, 1班, 77, 79, 80, 44
100007, 小美, 女, 4班, 77, 88, 80, 44
100001, 小鱼, 男, 1班, 72, 82, 85, 33
100002, 小高, 女, 6班, 74, 88, 85, 77

```

下面的代码用于设置CSV 文件与excel 兼容, error_bad_lines=False 会直接忽略不符合要求的记录。

```

import pandas as pd
def readcsv4(csvfilepath):
    dia = csv.excel()
    df = pd.read_csv(csvfilepath, dialect=dia, error_bad_lines=False, encoding='gbk')
    print(df)

if __name__ == '__main__':
    readcsv4(r'E:\2018-12-19\scores.csv')
#输出
#           学号  姓名  性别  班级  语文  数学  英语
#    0  100001  小雨  女    1班    72    85    87
#    1  100002  小雪  女    2班    67    87    77
#    2  100004  小天  男    1班    76    87    84
#    3  100005  小军  男    3班    79    86    83

```

3) 对文件进行分块处理并返回一个可迭代的对象

分块处理可以避免将所有的文件载入内存, 仅在使用的時候读入所需内容。参数chunksize设置分块的文件行数, 2表示每一块包含2个记录。将参数iterator 设置为True时, 返回值为TextFileReader, 它是一个可迭代对象。

来看下面的例子, 当chunksize=2、iterator=True 时, 每次输出为包含2个记录的块。

```

import pandas as pd
def readcsv5(csvfilepath):
    df = pd.read_csv(csvfilepath, encoding='gbk', chunksize=2, iterator=True)
    print(df) #<pandas.io.parsers.TextFileReader object at 0x00000293FA870320>
    print(iter(df).__next__())

```

```
if __name__ == '__main__':
    readcsv5(r'E:\2018-12-19\scores.csv')
#输出
#      学号  姓名  性别  班级  语文  数学  英语
# 0  100001  小雨  女   1班   72   85   87
# 1  100002  小雪  女   2班   67   87   77
```

>>>>待续

分类: [python常用库及扩展库](#)

好文要顶

关注我

收藏该文

enjoyzier

关注 - 0

粉丝 - 12

0

0

[+加关注](#)

« 上一篇: [win10下一分钟快速搭建rtmp推流服务器](#)
» 下一篇: [Python之XML模块](#)

posted @ 2019-02-27 11:15 enjoyzier 阅读(329) 评论(0) [编辑](#) [收藏](#)

[刷新评论](#) [刷新页面](#) [返回顶部](#)

注册用户登录后才能发表评论，请 [登录](#) 或 [注册](#)，[访问](#) 网站首页。

- 【推荐】超50万行VC++源码: 大型组态工控、电力仿真CAD与GIS源码库
- 【活动】京东云服务器_云主机低于1折，低价高性能产品备战双11
- 【推荐】天翼云双十一提前开抢，1核1G云主机3个月仅需59元
- 【推荐】阿里云双11冰点钜惠，热门产品低至一折等你来抢！
- 【福利】个推四大热门移动开发SDK全部免费用一年，限时抢！

- 相关博文:
- [11.1 csv--CSV文件读写库](#)
 - [11.1 csv--CSV文件读写库](#)
 - [Python2.7-csv](#)
 - [Python csv.md](#)
 - [Python CSV模块简介](#)
- » 更多推荐...

- 最新 IT 新闻:
- Facebook又爆隐私丑闻：100多第三方开发者违规访问用户数据
 - 金山办公：发行价确定为45.86元/股
 - 美“天鹅座”货运飞船抵达空间站
 - 欧盟文件敦促发行公共数字货币 可能封杀Facebook加密货币
 - 给黑洞拍照片不算什么，下一步要拍视频了
- » 更多新闻...

历史上的今天：
2019-02-27 [Python之CSV模块](#)

