

什么是csv格式

逗号分隔值（Comma-Separated Values，CSV，有时也称为字符分隔值，因为分隔字符也可以不是逗号），其文件以纯文本形式存储表格数据（数字和文本）。

CSV文件由任意数目的记录组成，记录间以某种换行符分隔；

每条记录由字段组成，字段间的分隔符是其它字符或字符串，最常见的是逗号或制表符。

所有记录都有完全相同的字段序列，通常都是纯文本文件。

建议用nodepad++、sublime等编辑器进行编辑。

csv格式规则

1. 开头是不留空，以行为单位。
2. 可含或不含列名，含列名则居文件第一行。
3. 一行数据不跨行，无空行。
4. 以半角逗号（即,）作分隔符，列为空也要表达其存在。
5. 列内容如存在半角引号（即"），替换成半角双引号（""）转义，即用半角引号（即""）将该字段值包含起来。
6. 文件读写时引号，逗号操作规则互逆。
7. 内码格式不限，可为 ASCII、Unicode 或者其他。
8. 不支持特殊字符

Python csv模块

csv模式是python内置的标准模块，用于读写csv格式的文件。

在csv模块中提供了reader、writer来读写csv格式的文件，下面我们一起看下这两个函数使用示例。

示例功能：

1. 先使用writer函数写一个csv文件
2. 使用reader函数读取上述步骤写的csv文件内容，并在console中输出

```
#-*- coding:utf-8 -*-

__author__ = "苦叶子"

import csv

if __name__ == "__main__":
    print("python csv文件读写操作示例")

    # 写csv文件
    print("写入一些简单数据到csv_data.csv文件中")
    with open('csv_data.csv', 'w', newline='') as csvfile:
```

```

    spamwriter = csv.writer(csvfile, # csvfile为打开要写的文件对象
        delimiter=',', # 分隔符
    )
    spamwriter.writerow(['csv_demo'] * 5 + ['DeepTest'])
    spamwriter.writerow(['hello', 'Study Python3', 'Auto Testing'])

    csvfile.close()

print("读取csv_data.csv内容")
with open('csv_data.csv', 'r') as f:
    reader = csv.reader(f, delimiter=',')
    for row in reader:
        print("row的类型: ", type(row))
        print(row)

        # 遍历每行中每个数据项
        for data in row:
            print(data, " ")

f.close()

```

在Python csv模块中还提供了另外一种方式来读写csv文件，就是通过字典方式来读写，其提供的主要方法为：DictReader、DictWriter，下面我们一起一个基本的示例，大家可以根据这个示例与上面的示例进行对比，看看有什么不同。

```

#-*- coding:utf-8 -*-

__author__ = "苦叶子"

import csv

if __name__ == "__main__":
    print("python csv文件字典写读操作示例")

    # 写csv文件
    print("写入一些简单数据到csv_dict_data.csv文件中")
    with open('csv_dict_data.csv', 'w') as csvfile:
        # 写csv头
        fieldnames = ['first_name', 'last_name']
        writer = csv.DictWriter(csvfile, fieldnames=fieldnames)
        writer.writeheader()

        # 写csv内容
        writer.writerow({'first_name': 'Baked', 'last_name': 'Beans'})
        writer.writerow({'first_name': 'Lovely', 'last_name': 'Spam'})
        writer.writerow({'first_name': 'Wonderful', 'last_name': 'Spam'})

    print("读取csv_dict_data.csv内容")
    with open('csv_dict_data.csv') as csvfile:

```

```
reader = csv.DictReader(csvfile)
for row in reader:
    # 遍历每行中的数据
    print(row['first_name'], row['last_name'])
```

小结

本文介绍了csv读写的两种方式，大家可以仔细比对二者的区别，掌握其应用场景。

扫一扫关注微信公众号：

