

文件对象

1

文件对象

—— 文件打开方法

open及file内建函数

- 作为打开文件之门的“钥匙”，内建函数open()以及file()提供了初始化输入/输出（I/O）操作的通用接口
- 成功打开文件后时候会返回一个文件对象，否则引发一个错误
- open()方法和file()方法可以完全相互替换
- 基本语法：
`file_object = open(file_name, access_mode='r', buffering=-1)`

文件对象访问模式

文件模式	操 作
r	以读方式打开（文件不存在则报错）
w	以写方式打开（文件存在则清空，不存在则创建）
a	以追加模式打开（必要时创建新文件）
r+	以读写模式打开（参见r）
w+	以读写模式打开（参见w）
a+	以读写模式打开（参见a）

2

文件对象

—— 文件输入

read方法

- read()方法用来直接读取字节到字符串中，最多读取给定数目个字节
- 如果没有给定size参数（默认值为-1）或者size值为负，文件将被读取直至末尾

```
>>> data = fobj.read()  
>>> print(data)
```

readline方法

- 读取打开文件的一行（读取下个行结束符之前的所有字节）
- 然后整行，包括行结束符，作为字符串返回
- 它也有一个可选的size参数，默认为-1，代表读至行结束符
- 如果提供了该参数，那么在超过size个字节后会返回不完整的行

```
>>> data = fobj.readline()
>>> print(data)
```

readlines方法

- readlines()方法读取所有（剩余的）行然后把它们作为一个字符串列表返回

```
>>> data = fobj.readlines()  
>>> print(data)
```


文件迭代

- 如果需要逐行处理文件，可以结合for循环迭代文件
- 迭代文件的方法与处理其他序列类型的数据类似

```
>>> fobj = open('star.py')  
>>> for eachLine in fobj:  
...     print(eachLine, end= " ")
```

3

文件对象

—— 文件输出

write方法

- write()内建方法功能与read()和readline()相反。它把含有文本数据或二进制数据块的字符串写入到文件中
去
- 写入文件时，不会自动添加行结束标志，需要程序员手工输入

```
>>> fobj.write('Hello World!\n')  
13
```

writelines方法

- 和readlines()一样，writelines()方法是针对列表的操作
- 它接受一个字符串列表作为参数，将它们写入文件
- 行结束符并不会被自动加入，所以如果需要的话，必须在调用writelines()前给每行结尾加上行结束符

```
>>> fobj.writelines(['Hello World!\n', 'python programing\n'])
```



更多精彩...



<http://bj.linux.tedu.cn/>
企业QQ：86198501