

文件的打开与关闭

想一想：

如果想用word编写一份简历，应该有哪些流程呢？

1. 打开word软件，新建一个word文件
2. 写入个人简历信息
3. 保存文件
4. 关闭word软件

同样，在操作文件的整体过程与使用word编写一份简历的过程是很相似的

1. 打开文件，或者新建立一个文件
2. 读/写数据
3. 关闭文件

<1>打开文件

在python，使用open函数，可以打开一个已经存在的文件，或者创建一个新文件

open(文件路径，访问模式)

示例如下：

```
f = open('test.txt', 'w')
```

说明：

文件路径

文件的路径分为相对路径和绝对路径两种。

- 绝对路径：指的是绝对位置，完整地描述了目标的所在地，所有目录层级关系是一目了然的。
 - 例
如：`C:/Users/chris/AppData/Local/Programs/Python/Python37/python.exe`，从电脑的盘符开始，表示的就是一个绝对路径。
- 相对路径：是从当前文件所在的文件夹开始的路径。
 - `test.txt`，是在当前文件夹查找 `test.txt` 文件
 - `./test.txt`，也是在当前文件夹里查找 `test.txt` 文件，`./` 表示的是当前文件夹。
 - `../test.txt`，从当前文件夹的上一级文件夹里查找 `test.txt` 文件。`../` 表示的是上

一级文件夹

- `demo/test.txt`，在当前文件夹里查找 `demo` 这个文件夹，并在这个文件夹里查找 `test.txt` 文件。

访问模式：

访问模式	说明
r	以只读方式打开文件。文件的指针将会放在文件的开头。如果文件不存在，则报错。 这是默认模式。
w	打开一个文件只用于写入。如果该文件已存在则将其覆盖。如果该文件不存在，创建新文件。
a	打开一个文件用于追加。如果该文件已存在，文件指针将会放在文件的结尾。也就是说，新的内容将会被写入到已有内容之后。如果该文件不存在，创建新文件进行写入。
r+	打开一个文件用于读写。文件指针将会放在文件的开头。
w+	打开一个文件用于读写。如果该文件已存在则将其覆盖。如果该文件不存在，创建新文件。
a+	打开一个文件用于读写。如果该文件已存在，文件指针将会放在文件的结尾。文件打开时会是追加模式。如果该文件不存在，创建新文件用于读写。
rb	以二进制格式打开一个文件用于只读。文件指针将会放在文件的开头。
wb	以二进制格式打开一个文件只用于写入。如果该文件已存在则将其覆盖。如果该文件不存在，创建新文件。
ab	以二进制格式打开一个文件用于追加。如果该文件已存在，文件指针将会放在文件的结尾。也就是说，新的内容将会被写入到已有内容之后。如果该文件不存在，创建新文件进行写入。
rb+	以二进制格式打开一个文件用于读写。文件指针将会放在文件的开头。
wb+	以二进制格式打开一个文件用于读写。如果该文件已存在则将其覆盖。如果该文件不存在，创建新文件。
ab+	以二进制格式打开一个文件用于读写。如果该文件已存在，文件指针将会放在文件的结尾。如果该文件不存在，创建新文件用于读写。

<2>关闭文件

`close()`

示例如下：

```
# 新建一个文件，文件名为:test.txt  
f = open('test.txt', 'w')  
  
# 关闭这个文件  
f.close()
```

千锋Python人工智能学院