

# 文件

## Python Reference

## 读写文本文件

文本文件 (text file) 把文本保存在磁盘上。用 `open(name, mode)` 打开它, 其中模式 (mode) 说明是读还是写。始终使用 `with`, 它会帮你关闭文件。

### 写入

模式 "w" 写一个新文件, 并抹掉任何旧内容。

```
with open("notes.txt", "w") as f:
    f.write("first line\n")
    f.write("second line\n")
print("saved")           # saved
```

### 读取

模式 "r"(默认) 用来读。`.read()` 把整个文件作为一个字符串返回。

```
with open("notes.txt", "w") as f:
    f.write("hello\nworld\n")
with open("notes.txt") as f:
    print(f.read().strip())  # hello / world
```

### 逐行读取

遍历文件, 一次得到一行。`.strip()` 会去除 (remove) 末尾的换行符 (newline)。

```
with open("data.txt", "w") as f:
    f.write("Mei,88\nSam,71\n")
with open("data.txt") as f:
    for line in f:
        name, score = line.strip().split(",")
        print(name, "scored", score)
# Mei scored 88
# Sam scored 71
```

### 追加

模式 "a" 追加 (append)——它在末尾添加, 而不抹掉原有内容。

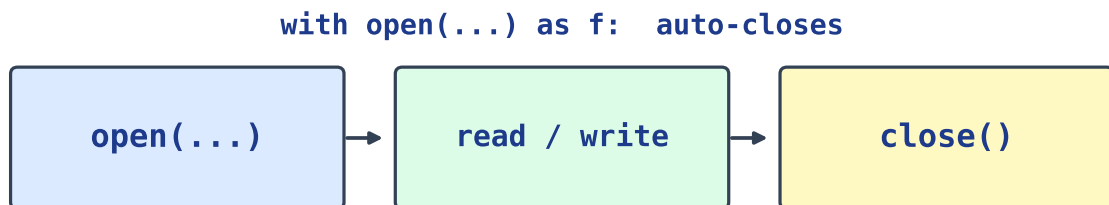
```
with open("log.txt", "w") as f:
    f.write("line 1\n")
with open("log.txt", "a") as f:
```

```
f.write("line 2\n")
with open("log.txt") as f:
    print(f.read().strip())    # line 1 / line 2
```

| 模式  | 含义         |
|-----|------------|
| "r" | 读 (默认)     |
| "w" | 写 (先抹掉)    |
| "a" | 追加 (添加到末尾) |

### 常见错误

- 一定要关闭文件, 或用 `with open(...) as f:`, 它会替你关闭。
- `read()` 把整个文件当成一个字符串返回, 每行末尾仍带着 `\n`。
- 用 `"w"` 打开会先清空文件; 要追加到末尾用 `"a"`。



Exam tip: mode 'r' reads, 'w' writes (overwrites); prefer with so the file always closes

*open → read/write → close (with auto-closes)*