

Python 文件操作



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌



目录

Contents

- ◆ 文件的编码
- ◆ 文件的读取
- ◆ 文件的写入
- ◆ 文件的追加
- ◆ 文件操作综合案例

学习目标

Learning Objectives

1. 掌握文件编码的概念和常见编码

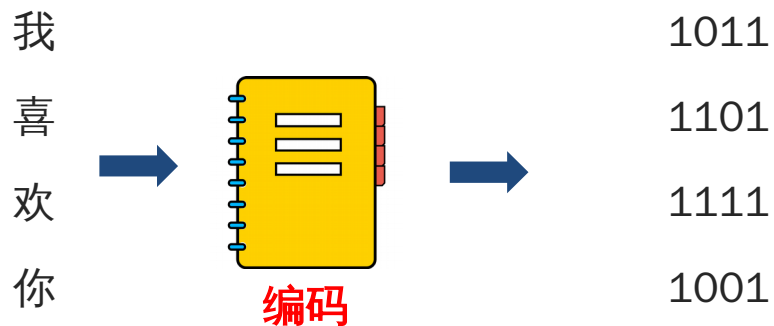
文件编码

思考：计算机只能识别：0 和 1，那么我们丰富的文本文件是如何被计算机识别，并存储在硬盘中呢？

答案：使用编码技术（密码本）将内容翻译成 0 和 1 存入。

文件编码

编码技术即：翻译的规则，记录了如何将内容翻译成二进制，以及如何将二进制翻译回可识别内容。



计算机中有许多可用编码：

- UTF-8
- GBK
- Big5
- 等

不同的编码，将内容翻译成二进制也是不同的。

文件编码

编码有许多，所以要使用正确的编码，才能对文件进行正确的读写操作呢。



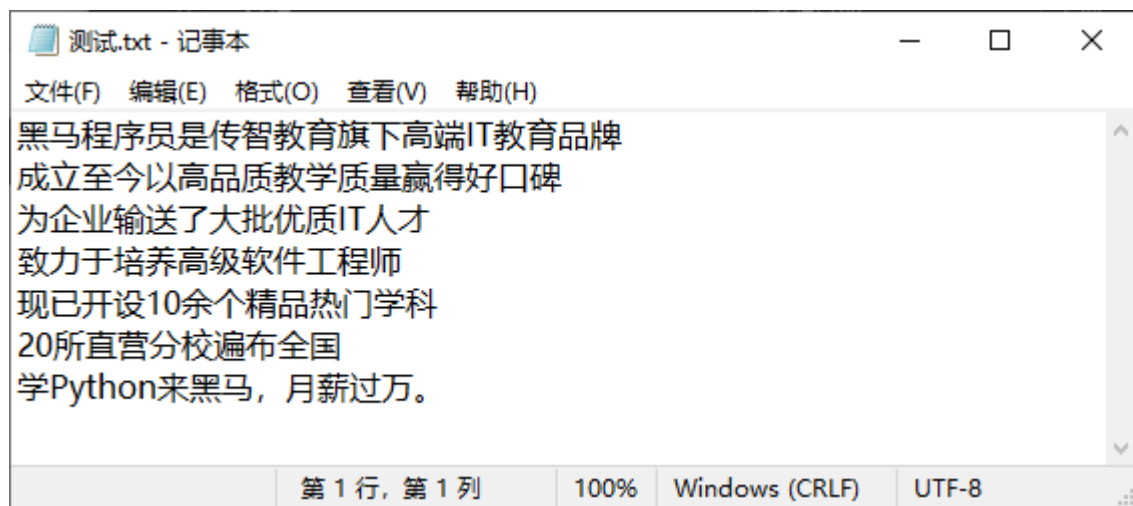
如上，如果你给喜欢的女孩发送文件，使用编码 A 进行编码（内容转二进制）。

女孩使用编码 B 打开文件进行解码（二进制反转回内容）

自求多福吧。

查看文件编码

我们可以使用 Windows 系统自带的记事本，打开文件后，即可看出文件的编码是什么：



UTF-8 是目前全球通用的编码格式

除非有特殊需求，否则，一律以 UTF-8 格式进行文件编码即可。



总结

1. 什么是编码？

编码就是一种规则集合，记录了内容和二进制间进行相互转换的逻辑。
编码有许多中，我们最常用的是 UTF-8 编码

2. 为什么需要使用编码？

计算机只认识 0 和 1，所以需要将内容翻译成 0 和 1 才能保存在计算机中。
同时也需要编码，将计算机保存的 0 和 1，反向翻译回可以识别的内容。



目录

Contents

- ◆ 文件的编码
- ◆ 文件的读取
- ◆ 文件的写入
- ◆ 文件的追加
- ◆ 文件操作综合案例

学习目标

Learning Objectives

1. 了解文件操作的作用
2. 掌握文件的打开、读取、关闭操作

什么是文件

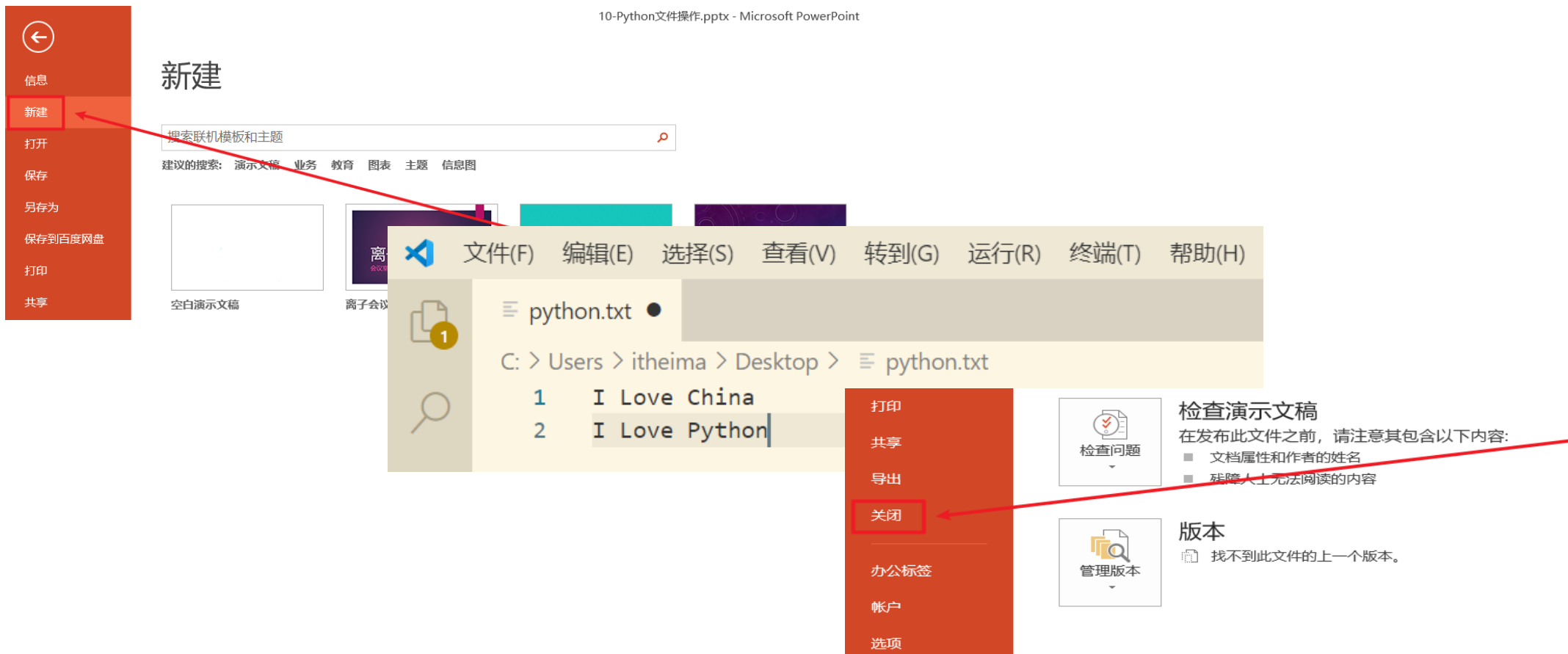
内存中存放的数据在计算机关机后就会消失。要长久保存数据，就要使用硬盘、光盘、U 盘等设备。为了便于数据的管理和检索，引入了“文件”的概念。

一篇文章、一段视频、一个可执行程序，都可以被保存为一个文件，并赋予一个文件名。操作系统以文件为单位管理磁盘中的数据。一般来说，文件可分为文本文件、视频文件、音频文件、图像文件、可执行文件等多种类别。



文件操作包含哪些内容呢？

在日常生活中，文件操作主要包括**打开**、**关闭**、**读**、**写**等操作。



10-Python文件操作.pptx - Microsoft PowerPoint

新建

搜索联机模板和主题

建议的搜索: 演示文稿 业务 教育 图表 主题 信息图

空白演示文稿

离子会议

文件(F) 编辑(E) 选择(S) 查看(V) 转到(G) 运行(R) 终端(T) 帮助(H)

python.txt

C: > Users > itheima > Desktop > python.txt

```
1 I Love China
2 I Love Python
```

打印
共享
导出
关闭
办公标签
帐户
选项

检查问题

检查演示文稿

在发布此文件之前，请注意其包含以下内容：

- 文档属性和作者的姓名
- 残障人士无法阅读的内容

版本

找不到此文件的上一个版本。

文件的操作步骤

想想我们平常对文件的基本操作，大概可以分为三个步骤（简称文件操作三步走）：

- ① 打开文件
- ② 读写文件
- ③ 关闭文件

注意事项

注意：可以只打开和关闭文件，不进行任何读写

open() 打开函数

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，语法如下

```
open(name, mode, encoding)
```

mode ：设置打开文件的模式（访问模式）：只读、写入、追加等。

encoding: 编码格式（推荐使用 UTF-8 ）

示例代码：

```
f = open('python.txt', 'r', encoding="UTF-8")  
# encoding 的顺序不是第三位，所以不能用位置参数，用关键字参数直接指定
```

注意事项

注意：此时的 `f` 是 `open` 函数的文件对象，对象是 Python 中一种特殊的数据类型，拥有属性和方法，可以使用对象 . 属性或对象 . 方法对其进行访问，后续面向对象课程会给大家进行详细的介绍。

mode 常用的三种基础访问模式

模式	描述
r	以只读方式打开文件。文件的指针将会放在文件的开头。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打开文件，并从开头开始编辑， 原有内容会被删除 。 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， 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。

读操作相关方法

read() 方法：

文件对象 .read(num)

readlines() 方法：

readlines 可以按照行的方式把整个文件中的内容进行一次性读取，并且返回的是一个列表，其中每一行的数据为一个元素。

```
f = open('python.txt')
content = f.readlines()

# ['hello world\n', 'abcdefg\n', 'aaa\n', 'bbb\n', 'ccc']
print(content)

# 关闭文件
f.close()
```


读操作相关方法

readline() 方法：一次读取一行内容

```
f = open('python.txt')

content = f.readline()
print(f' 第一行： {content}')

content = f.readline()
print(f' 第二行： {content}')

# 关闭文件
f.close()
```

读操作相关方法

for 循环读取文件行

```
for line in open("python.txt", "r"):
    print(line)
```

每一个 line 临时变量，就记录了文件的一行数据

读操作相关方法

close() 关闭文件对象

```
f = open("python.txt", "r")
```

```
f.close()
```

最后通过 close，关闭文件对象，也就是关闭对文件的占用

如果不调用 close, 同时程序没有停止运行，那么这个文件将一直被 Python 程序占用。

读操作相关方法

with open 语法

```
with open("python.txt", "r") as f:  
    f.readlines()
```

```
# 通过在 with open 的语句块中对文件进行操作  
# 可以在操作完成后自动关闭 close 文件，避免遗忘掉 close 方法
```

操作汇总

操作	功能
文件对象 = open(file, mode, encoding)	打开文件获得文件对象
文件对象 .read(num)	读取指定长度字节 不指定 num 读取文件全部
文件对象 .readline()	读取一行
文件对象 .readlines()	读取全部行，得到列表
for line in 文件对象	for 循环文件行，一次循环得到一行数据
文件对象 .close()	关闭文件对象
with open() as f	通过 with open 语法打开文件，可以自动关闭



总结

1. 操作文件需要通过 open 函数打开文件得到文件对象
2. 文件对象有如下读取方法：
 - read()
 - readline()
 - readlines()
 - for line in 文件对象
3. 文件读取完成后，要使用文件对象 .close() 方法关闭文件对象，否则文件会被一直占用



课后练习：单词计数

通过 Windows 的文本编辑器软件，将如下内容，复制并保存到：word.txt，文件可以存储在任意位置

itheima itcast python

itheima python itcast

beijing shanghai itheima

shenzhen guangzhou itheima

wuhan hangzhou itheima

zhengzhou bigdata itheima

通过文件读取操作，读取此文件，统计 itheima 单词出现的次数



目录

Contents

- ◆ 文件的编码
- ◆ 文件的读取
- ◆ 文件的写入
- ◆ 文件的追加
- ◆ 文件操作综合案例

写操作快速入门

案例演示：

```
# 1. 打开文件
f = open('python.txt', 'w')

# 2. 文件写入
f.write('hello world')

# 3. 内容刷新
f.flush()
```

- ❑ 当调用 flush 的时候，内容会真正写入文件
- ❑ 这样做是避免频繁的操作硬盘，导致效率下降（攒一堆，一次性写磁盘）

写操作注意

- 文件如果不存在，使用“w”模式，会创建新文件
- 文件如果存在，使用“w”模式，**会将原有内容清空**



总结

1. 写入文件使用 open 函数的“w”模式进行写入
2. 写入的方法有：
 - write()，写入内容
 - flush()，刷新内容到硬盘中
3. 注意事项：
 - w 模式，文件不存在，会创建新文件
 - w 模式，文件存在，会清空原有内容
 - close() 方法，带有 flush() 方法的功能



目录

Contents

- ◆ 文件的编码
- ◆ 文件的读取
- ◆ 文件的写入
- ◆ 文件的追加
- ◆ 文件操作综合案例

追加写入操作快速入门

案例演示：

```
# 1. 打开文件，通过 a 模式打开即可  
f = open('python.txt', 'a')
```

```
# 2. 文件写入  
f.write('hello world')
```

```
# 3. 内容刷新  
f.flush()
```

□ a 模式，文件存在会在最后，追加写入文件



总结

1. 追加写入文件使用 open 函数的“a”模式进行写入
2. 追加写入的方法有（和 w 模式一致）：
 - write()，写入内容
 - flush()，刷新内容到硬盘中
3. 注意事项：
 - a 模式，文件不存在，会创建新文件
 - a 模式，文件存在，会在原有内容后面继续写入
 - 可以使用“\n”来写出换行符



目录

Contents

- ◆ 文件的编码
- ◆ 文件的读取
- ◆ 文件的写入
- ◆ 文件的追加
- ◆ 文件操作综合案例

学习目标

Learning Objectives

1. 完成文件备份案例

需求分析

需求：有一份账单文件，记录了消费收入的具体记录，内容如下：

name,date,money,type,remarks

周杰轮,2022-01-01,100000,消费,正式
周杰轮,2022-01-02,300000,收入,正式
周杰轮,2022-01-03,100000,消费,测试
林俊节,2022-01-01,300000,收入,正式
林俊节,2022-01-02,100000,消费,测试
林俊节,2022-01-03,100000,消费,正式
林俊节,2022-01-04,100000,消费,测试
林俊节,2022-01-05,500000,收入,正式
张学油,2022-01-01,100000,消费,正式
张学油,2022-01-02,500000,收入,正式
张学油,2022-01-03,900000,收入,测试
王力鸿,2022-01-01,500000,消费,正式
王力鸿,2022-01-02,300000,消费,测试
王力鸿,2022-01-03,950000,收入,正式
刘德滑,2022-01-01,300000,消费,测试
刘德滑,2022-01-02,100000,消费,正式
刘德滑,2022-01-03,300000,消费,正式

同学们可以将内容复制并保存为 bill.txt 文件

需求分析

我们现在要做的就是：

- 读取文件
- 将文件写出到 bill.txt.bak 文件作为备份
- 同时，将文件内标记为测试的数据行丢弃

实现思路：

- open 和 r 模式打开一个文件对象，并读取文件
- open 和 w 模式打开另一个文件对象，用于文件写出
- for 循环内容，判断是否是测试不是测试就 write 写出，是测试就 continue 跳过
- 将 2 个文件对象均 close()

快去实现吧。



传智教育旗下高端IT教育品牌