

## 3.文件操作

读取文件

文件指针

写入内容

with 结构/文件备份

编码：将str ->bytes

```
Python |  
1 print("hello world".encode())#输出b'hello world'
```

反编码：bytes ->str

```
Python |  
1 print(b'hello world'.decode())#输出hello world
```

看函数里有哪些方法：dir()

打开文件：open("绝对路径"，"打开方式",encoding='utf-8')

访问方式	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，这是默认模式。如果文件不存在，抛出异常
w	以只写方式打开文件。如果文件存在会被覆盖。如果文件不存在，创建新文件
a	以追加方式打开文件。如果该文件已存在，文件指针将会放在文件的结尾。如果文件不存在，创建新文件进行写入
r+	以读写方式打开文件。文件的指针将会放在文件的开头。如果文件不存在，抛出异常
w+	以读写方式打开文件。如果文件存在会被覆盖。如果文件不存在，创建新文件

a+

以读写方式打开文件。如果该文件已存在，文件指针将会放在文件的结尾。如果文件不存在，创建新文件进行写入

## 读取文件

Python

```
1 print(文件名.read(size))#读取指定大小的内容，如果没有指定，读取全部内容。
2 print(文件名.readline())#读取一行。
3 print(文件名.readlines())#print读取所有行并返回一个列表。
```

## 文件指针

操作指针：seek()

Python

```
1 file.seek(offset, whence)
2 '''
3 offset：要移动的字节数。
4 whence（可选）：指定偏移量的基准位置。
5 可以取以下值：
6 0：从文件开头开始计算（默认值）。
7 1：从当前位置开始计算。
8 2：从文件末尾开始计算
9 '''
10
11
```

## 写入内容

写入：file.write()

关闭文件：file.close()

```
1 # 以追加模式打开日记文件
2 file = open('diary.txt', 'a', encoding='utf-8')
3
4 # 获取用户输入的日记内容
5 content = input("请输入今天的日记: ")
6
7 # 将日记内容写入文件
8 file.write(content + "\n")
9 print("日记已保存! ")
10
11 # 关闭文件
12 file.close()
```

## with 结构/文件备份

自动管理文件的打开和关闭，避免忘记关闭文件的情况。

```
1
2 source = 'a.txt' #a是源文件,
3 destination = 'b.txt' #b是写入的文件
4
5 with open(source, 'r', encoding='utf-8') as src:
6     content = src.read() #读a
7
8 with open(destination, 'w', encoding='utf-8') as dest:
9     dest.write(content) #写b
10
11 print(f"备份成功! '{source}' 的内容已复制到 '{destination}'")
```

去除换行符: `.strip()`

分割: `.split()`返回值是一个列表