

文件操作步骤

- 打开文件-open
- 读写文件
 - 读---把文件的内容读到变量里-read
 - 写---把变量的值写到文件内容里-write
- 关闭文件-close

读取第一个文件内容

- 打开文件
 - file = open(要打开文件的路径和文件名, "r")
 - file是一个变量名,代表文件操作对象
 - open的第一个参数一定要文件存在, 如果文件不存在,会报错
 - "r"意思是用只读方式打开文件
- 读取文件内容
 - txt = file.read()
 - txt是定义一个变量,代表存放读取到的文件内容
 - file是上一个open返回的文件操作对象
 - read是file对象的一个方法,用来读取文件内容
- 关闭文件
 - file.close()
 - 文件打开后,一定要记得关闭

```
1
2 # 第一步: 打开文件
3 file = open(r"C:\file\temp\a.txt", "r")
4 # 第二步: 读取文件内容
5 txt = file.read()
6 print(txt) # 为了显示文件内容
7 # 第三步: 关闭文件
8 file.close()
```

写第一个文件内容

- 打开文件
 - file = open("要打开的文件路径和文件名", "w")
 - 第二个参数w代表用写的方式打开文件
 - 如果用w方法打开文件,文件名不存在,会创建一个新文件,存在的话会覆盖已有文件
- 写文件
 - file.write(要写入的内容)
- 关闭文件
 - file.close()

```
1 # 第一步:打开文件
2 file = open(r"c:\file\temp\b.txt", "w")
3 # 第二步:写内容
4 file.write("hello python")
5 # 第三步:关闭文件
6 file.close()
7 # 验证程序是否执行成功?看c盘file目录的temp目录下是否生成了文件b.txt
8 # b.txt文件内容是否为hello python
```

注意项:如果open第二个参数是"r",打开文件后只能用read读,不能用write写

如果open第二个参数是"w",打开文件后只能用write写,不能用read读

追加写文件内容

- 打开文件
 - file = open("文件路径和文件名", "a")
 - "a"当文件不存在的时候创建新文件
 - 当文件存在的时候,在文件后面追加内容
- 写文件
 - file.write("要写入文件的内容")
- 关闭文件
 - file.close()

```
1
2 file = open(r"c:\file\temp\c.txt", "a")
3 file.write("hello\n")
4 file.close()
```

总结打开文件的方式

- open函数第二个参数的意思是打开文件的方式
 - r只读
 - w只写
 - a追加写
 - rb用二进制方法打开,只读
 - wb用二进制方法打开,只写
 - ab用二进制方法打开,追加

打开文件的时候,指定字符集

- 如果文件格式是utf8的,open打开文件的时候需要指定字符集
 - file = open("文件名", "r", encoding="utf8")
- 如果出现以下错误提示,基本就是由于字符集导致的错误

```
1 UnicodeDecodeError:
```

```
1 file = open(r"C:\file\teach\05python\day09\3-code\01-open函数.py", "r",  
encoding="utf8")  
2 txt = file.read()  
3 print(txt)  
4 file.close()
```

课堂练习-修改文件内容

```
1 # 有文件a.txt  
2 # 内容如下:  
3 # 我爱你python, 我爱你  
4 #  
5 # 写代码,把a.txt内容中的python替换为world  
6 #  
7 # 思路,先把a.txt文件内用r方式读出来  
8 # 用字符串的replace方法把python替换为world  
9 # 在用w方式把替换后的内容写回到a.txt里面  
10 file = open(r"C:\file\temp\a.txt", "r")  
11 txt = file.read()  
12 a = txt.replace("python", "world")  
13 file.close() # 这一步必须关闭  
14 file = open(r"C:\file\temp\a.txt", "w")  
15 file.write(a)  
16 file.close()
```

复制文件

```
1 # 把文件a.txt复制为b.txt  
2 # 思路:把a.txt打开,读取全部内容  
3 # 把b.txt用w方法打开,把内容写入  
4 file = open(r"C:\file\temp\a.txt", "r")  
5 txt = file.read()  
6 file.close()  
7 file = open(r"C:\file\temp\b.txt", "w")  
8 file.write(txt)  
9 file.close()
```

课堂练习-合并文件

```
1 # 有文件a.txt  
2 # 内容如下:  
3 #  
4 # 我爱你python, 我爱你  
5 #  
6 # 有文件b.txt  
7 # 内容如下:  
8 #  
9 # hello  
10 #  
11 # 把文件a.txt和b.txt合并为c.txt  
12 # 思路:把文件a.txt内如读出来放到变量a
```

```

13 # 把文件b.txt内如读出来放到变量b
14 # 把a + b 结果写入的c.txt中
15 file = open(r"c:\file\temp\a.txt", "r")
16 a = file.read()
17 file.close()
18 file = open(r"c:\file\temp\b.txt", "r")
19 b = file.read()
20 file.close()
21 file = open(r"c:\file\temp\c.txt", "w")
22 file.write(a + b)
23 file.close()

```

按行读取文件内容

- readline()按行读取文件内容
- 由于read一次把文件所有内容都读取到内存中,如果文件特别大,会消耗内存
- 每调用一次readline,内部的文件指针就会向下一行移动,这样的结果是下次再次调用readline会自动读取下一行
- readline读取到文件最后,返回""
- 如果要通过readline来读取文件所有内容
 - 写一些死循环
 - 在循环内部调用readline,读取文件每一行
 - 如果readline返回为"", 终止这个死循环

```

1 file = open(r"c:\file\temp\a.txt", "r")
2 # 在循环开始的时候,不知道文件有多少行,所以也不能决定循环多少次
3 while True:
4     txt = file.readline()
5     # 如果readline读取到文件最后末尾,返回""
6     if txt == "":
7         break
8     print(txt, end="")
9
10 file.close()

```

课堂练习-读取文件偶数行

```

1 # 有文件a.txt,有很多行,只显示偶数行
2 # 思路:只读方式打开a.txt,读取所有行,但只显示偶数行
3 # 做一个循环读取所有行,在循环的时候做一个计数器,每次循环计数器+1
4 # 可以判断计数器是偶数还是奇数,如果是偶数那么就print
5
6 file = open(r"c:\file\temp\a.txt", "r")
7 index = 1
8 while True:
9     txt = file.readline()
10    if txt == "":
11        break
12    if index % 2 == 0: # 如果条件成立,代表index为偶数

```

```
13     print(txt, end="")
14     index += 1
15
16 file.close()
17
```

readlines

- 一下读取文件所有行,返回一个列表,列表中的一个成员就是文件中的一行,文件有多少行,列表就有多少成员

```
1 file = open(r"c:\file\temp\a.txt", "r")
2 list1 = file.readlines()
3 for n in list1:
4     print(n, end="")
5 file.close()
```

课堂练习-计算文件中最大数字与最小数字的差

```
1 # 有个文件a.txt,内容每一行是一个整数,最大值和最小值的差
2 # 思路:
3 # 把文件每一行 做为list的每个成员(文件的每一行是字符串)
4 # 只有max(list) - min (list)
5 #
6 # 先做一个空的list
7 # 循环读每行,把每行用int转化为整数,append到空的list里面
8 # 循环读完后,就可以max(list) - min(list)
9 file = open(r"c:\file\temp\a.txt", "r")
10 list1 = []
11 while True:
12     txt = file.readline()
13     if txt == "":
14         break
15     list1.append(int(txt))
16 file.close()
17 print(max(list1) - min(list1))
```

- 字符串比较大小的原理

```
1 list1 = [2, 12, 5]
2 list2 = ['2', '12', '5']
3 print(max(list1), min(list1))
4 print(max(list2), min(list2))
5 # 如果长得像数字的字符串比较大小,比较原理是先比较字符串的一个字符
```

with open语法

- 这个语法传统的file = open的一种简化语法, 不需要明确的调用close关闭文件

```
1 # file = open(r"c:\file\temp\a.txt", "r")
2 # txt = file.read()
3 # print(txt)
4 # file.close()
5 # 对等代码
6 with open(r"c:\file\temp\a.txt", "r") as file:
7     txt = file.read()
8     print(txt)
```

json文件

- 一般是定义数据使用的
- 在互联网传递数据的时候, 很常见的一种文件格式
- 所有数据用一对大括号括起来
- 大括号内部是键值对, 键和值用冒号分隔
- 多个键值对用逗号分隔
- 字符串用双引号
- 数字不需要引号
- 列表用中括号
- 对象用大括号

```
1 {
2     "name": "tom",
3     "age": 20,
4     "addr": [1, 2, 3]
5 }
```

读取json文件的方法

- 第一步要导入json模块
 - import json
- 第二步打开json文件
 - json中有中文, 一般字符集都是utf8, 打开文件的时候要指定字符集为utf8
 - open(json文件名, "r", encoding="utf8")
- 第三步: 调用json模块的load方法, 读取文件内容
 - data = json.load(file)
- 第四步: 关闭open打开的文件
 - file.close()

```
1 import json
2 file = open("a.json", "r", encoding="utf8")
3 data = json.load(file) # 把json文件的内容转化为python的字典
4 file.close()
5 # print(data)
6 for n in data:
7     print(n, data[n])
```

操作json文件常见错误

错误1:json格式错误

```
1 json.decoder.JSONDecodeError
```

错误2:扩展名不是json,是其他的

```
1 json文件的扩展名必须是.json
```

错误3:自己创建了一个文件名叫json.py

```
1 AttributeError: module 'json' has no attribute 'load'
```

写json文件

- 导入json模块
 - import json
- 用只写方式打开json文件
 - open(json文件,"w", encoding="utf8")
- 用dump方法把字典内容写入到json文件中
 - ensure_ascii = False代表中文不转义
- 关闭文件

```
1 import json
2 dict1 = {"name":"刘备", "age":20, "sex":"男"}
3 file = open("data.json", "w", encoding="utf8")
4 json.dump(dict1, file, ensure_ascii=False)
5 file.close()
```

课堂练习-修改json文件内容

2. 课堂练习

1.文件 test.json 内容如下:

```
{  
  "name": "诸葛亮",  
  "sex": "男",  
  "age": 24  
}
```

```
1  # 思路:  
2  # 先把内容从test.json文件中读出来  
3  # 读出来的结果是一个字典  
4  # 把字典中键age对应 的值修改为30  
5  # 再把字典写回到test.json文件中  
6  import json  
7  file = open("test.json", "r", encoding="utf8")  
8  dict1 = json.load(file)  
9  file.close()  
10 dict1["age"] = 30  
11 file = open("test.json", "w", encoding="utf8")  
12 json.dump(dict1, file, ensure_ascii=False)  
13 file.close()
```

在pycharm中安装课程需要的包-演示



