Python快速入门

嵩天



Python基本编程解析(中)

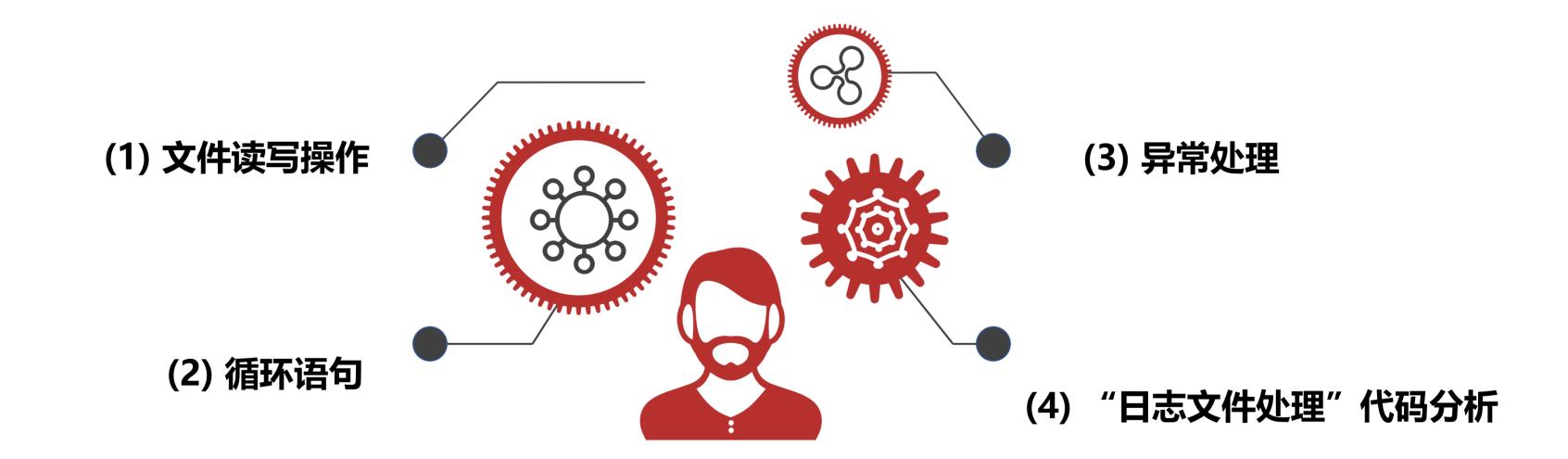
嵩天







单元开篇



Python基本编程解析(中)



单元开篇

目的: 概要了解Python编程的基本知识

学习编写10行的Python代码,开启编程之旅

- 知道 Python编程的基本知识
- 2 理解 Python日志文件分析程序
- 3 能独立编写 10行左右类似功能的Python程序

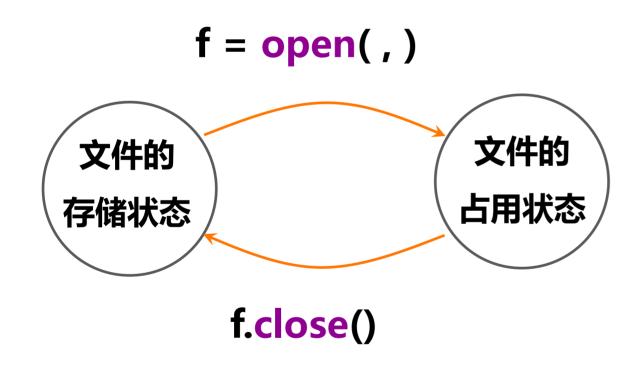
Python基本编程解析(中)





文件打开与关闭

文件处理的步骤: 打开-操作-关闭



- open()函数打开文件
- 变量f是文件句柄, 内部是文件对象
- · .close()方法关闭文件

文件打开与关闭

文件处理的步骤: 打开-操作-关闭

文件的打开模式

打开模式举例	描述
'r'	只读模式,默认值,如果文件不存在,返回FileNotFoundError
'w'	覆盖写模式,文件不存在则创建,存在则完全覆盖

- f = open('f.txt', 'r') 只读模式打开文本文件
- f = open('f.txt', 'w') 覆盖写模式打开文本文件

文件的全文读入

一次性全部读入文本

```
f = open("f.txt", "r")
txt = f.read()
# 对全文txt进行处理
f.close()
```

文件的按行读入

按行读入文本

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

for line in f:
    # 每行文本保存在line中,可逐行处理

f.close()
```



循环语句

Python包括两种循环: 遍历循环 无限循环

• 遍历循环: 由for和in组成,从遍历结构中逐一获得元素进行循环

• 无限循环: 使用while保留字, 由条件控制循环过程

循环语句

遍历循环:从遍历结构中逐一提取元素,放入循环变量

for <循环变量> in <遍历结构>:
 <语句块>

循环语句

无限循环: 反复执行语句快, 直到条件不满足时结束

```
while <条件>:

《条件>:

《语句块>
```





异常处理

如果文件不存在,怎么办?

```
f = open("sensor-data-no-exist.txt", "r")
for line in f:
    print(line)
f. close()
```

异常处理

如果文件不存在,怎么办?

```
Traceback (most recent call last):
line 1, in <module>
    f = open("sensor-data-no-exist.txt", "r")
FileNotFoundError: [Errno 2] No such file or directory: 'sensor-data-no-exist.txt'
```

异常处理

异常处理:对程序执行异常的处理机制,提高用户体验

try:

<语句块1>

except:

<语句块2>

- 使用try-except保留字
- 捕获语句块1中异常,采用语句块2进行处

理



```
#SensorReader.py
#2018-02-28 01:03:16.33393 19.3024 38.4629 45.08 2.68742
try:
    f = open("sensor-data.txt", "r")
   avg, cnt = 0, 0
    for line in f:
        Is = line.split()
        cnt += 1
       avg += eval(ls[2])
    print("平均的温度值是: {:.2f}".format(avg / cnt))
   f.close()
except:
   print("文件打开错误")
```

```
#SensorReader.py
    #2018-02-28 01:03:16.33393 19.3024 38.4629 45.08 2.68742
    try:
        f = open("sensor-data.txt", "r")
        avg, cnt = 0, 0
        for line in f:
            Is = line.split()
            cnt += 1
            avg += eval(ls[2])
        print("平均的温度值是: {:.2f}".format(avg / cnt))
        f.close()
    except:
        print("文件打开错误")
```

```
#SensorReader.py
#2018-02-28 01:03:16.33393 19.3024 38.4629 45.08 2.68742
try:
    f = open("sensor-data.txt", "r")
    avg, cnt = 0, 0
    for line in f:
        ls = line.split()
                                                           try
        cnt += 1
        avg += eval(ls[2])
                                                           except
    print("平均的温度值是: {:.2f}".format(avg / cnt))
    f.close()
except:
    print("文件打开错误")
```

```
#SensorReader.py
#2018-02-28 01:03:16.33393 19.3024 38.4629 45.08 2.68742
try:
    f = open("sensor-data.txt", "r")
                                                          文件打开
   avg, cnt = 0, 0
    for line in f:
        Is = line.split()
        cnt += 1
       avg += eval(ls[2])
    print("平均的温度值是: {:.2f}".format(avg / cnt))
   f. close()
                                                          文件关闭
except:
   print("文件打开错误")
```



```
#SensorReader.py
#2018-02-28 01:03:16.33393 19.3024 38.4629 45.08 2.68742
try:
   f = open("sensor-data.txt", "r")
   avg, cnt = 0, 0
                                                         变量初值
    for line in f:
        Is = line.split()
       cnt += 1
                                                         统计行数
       avg += eval(ls[2])
                                                         计算总和
   print("平均的温度值是: {:.2f}".format(avg / cnt))
   f.close()
except:
   print("文件打开错误")
```

```
#SensorReader.py
#2018-02-28 01:03:16.33393 19.3024 38.4629 45.08 2.68742
try:
    f = open("sensor-data.txt", "r")
   avg, cnt = 0, 0
    for line in f:
        ls = line.split()
        cnt += 1
       avg += eval(ls[2])
    print("平均的温度值是: {:.2f}".format(avg / cnt))
   f.close()
except:
   print("文件打开错误")
```

遍历每行



```
#SensorReader.py
#2018-02-28 01:03:16.33393 19.3024 38.4629 45.08 2.68742
try:
    f = open("sensor-data.txt", "r")
   avg, cnt = 0, 0
    for line in f:
        Is = line.split()
       cnt += 1
       avg += eval(ls[2])
    print("平均的温度值是: {:.2f}".format(avg / cnt))
   f.close()
except:
   print("文件打开错误")
```



识别每列

字符串的split()方法

split()方法: 能够根据空格分隔字符串, 结果保存在列表变量中

```
ls = "1 2 3 4 5 6 7".split()
print(ls)
```

['1', '2', '3', '4', '5', '6', '7']

```
#SensorReader.py
#2018-02-28 01:03:16.33393 19.3024 38.4629 45.08 2.68742
try:
    f = open("sensor-data.txt", "r")
   avg, cnt = 0, 0
    for line in f:
        Is = line.split()
        cnt += 1
       avg += eval(ls[2])
    print("平均的温度值是: {:.2f}".format(avg / cnt))
   f.close()
except:
   print("文件打开错误")
```

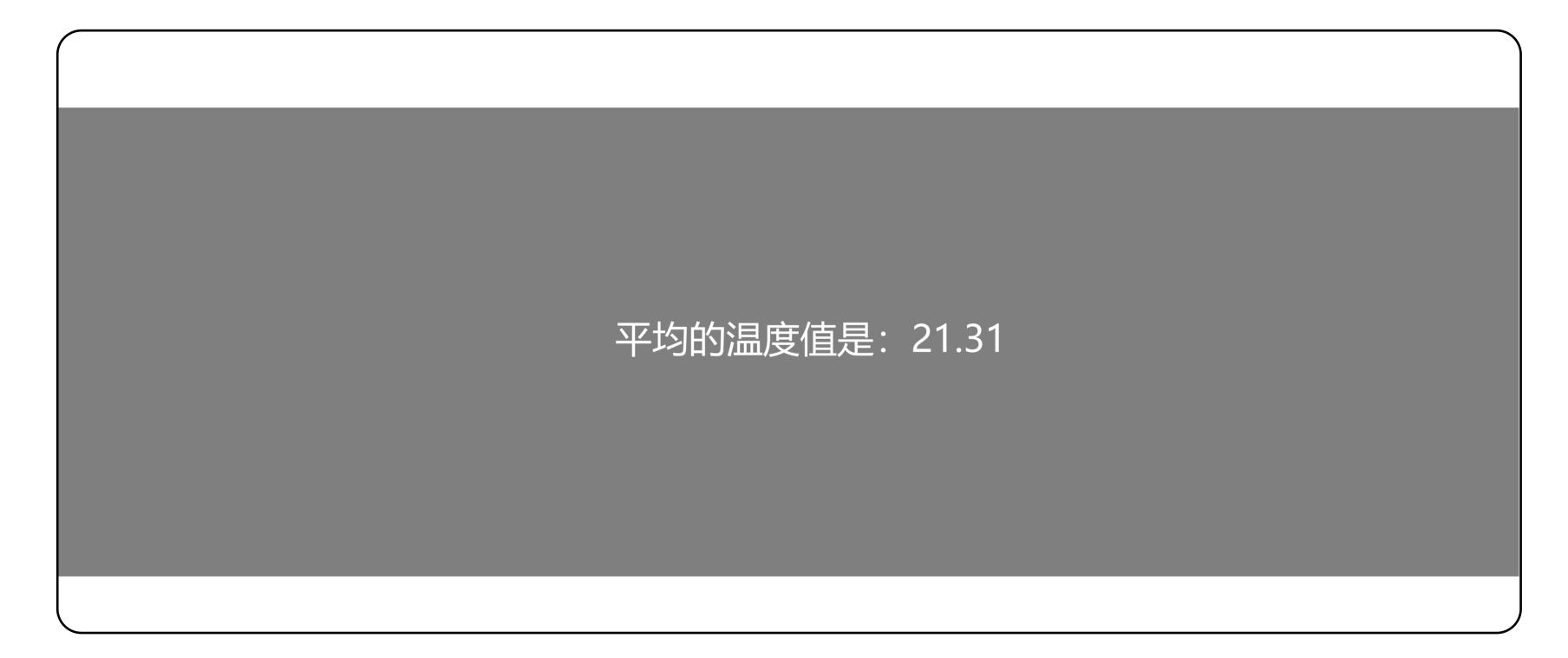
选取温度



```
#SensorReader.py
#2018-02-28 01:03:16.33393 19.3024 38.4629 45.08 2.68742
try:
    f = open("sensor-data.txt", "r")
   avg, cnt = 0, 0
    for line in f:
        Is = line.split()
       cnt += 1
       avg += eval(ls[2])
   print("平均的温度值是: {:.2f}".format(avg / cnt))
                                                         打印结果
   f.close()
except:
   print("文件打开错误")
                                                         打印结果
```



运行结果





Thank you