

监控日志统计IP次数

```
1 #监控服务器日志，找出每分钟访问超过100次的ip地址。
2
3 #1、每分钟读取一次文件，获取文件内容
4 #2、从文件内容里面找到ip地址
5 #3、判断每个ip出现的次数，如果超过100次，就找出来
6 import time
7 point=0
8 while True:
9     ips={} #定义字典，用于存放IP和次数
10    f=open("access.log",encoding="utf-8")
11    f.seek(point) #指针移动到上次统计的位置
12    for i in f: #通过for来逐行读取文件的内容
13        if i.strip()!='': #判断是否是空行
14            ip=i.split()[0]
15            if ip not in ips.keys():
16                ips.setdefault(ip,1)
17            else:
18                ips[ip]+=1
19    point=f.tell() #获取当前文件读的位置
20    for k,v in ips.items(): #遍历字典中所有的IP，并输出次数大于100次的IP
21        if v>100:
22            print("%s超过100次"%k)
23    f.close() #关闭文件
24    time.sleep(60)
```

备注：

1、为什么用字典不用list存放IP进行统计

答：如果用list每个IP存入数组，当IP较多，数组就会特别大，占内容，且遍历性能较弱，用字典每个IP对应一个次数，节省内存，遍历性能较高

2、除了监控日志，还可以通过该段代码进行做其他的统计工作，比如一个文件中某个人物出现次数等

3、文件读取用for i in f的方式，这样每次读取一行，占内存较小，如果用readlines，是每次把文件所有内容取出，这样较占内存

其他应用变异，比如统计一遍文章人物名称出现的次数，代码如下：

```
1 dic={"董卓":0,"貂蝉":0,"吕布":0,"孙坚":0} #人物初始值

2 with open("sanguo.txt",encoding="utf-8") as f: #with这种方式不需关闭文件

3     for i in f: #逐行遍历文件

4         for k in dic.keys(): #遍历字典人物

5             num=i.count(k) #统计当前行人物个数

6             dic[k]+=num #更新人物个数值

7     print(dic) #输出字典
```