输入输出重定向

- sys.stdin　　标准输入
- sys.stdout　　标准输出
- sys.stderr　　标准错误输出

- import sys
- #从文件读入变为从键盘输入，改变输入源
- s=sys.stdin.readlines()
- print(s)

词频统计（取自pintia网站）

- 请编写程序，对一段英文文本，统计其中所有不同单词的个数，以及词频最大的10%的单词。所谓“单词”，是指由不超过80个单词字符组成的连续字符串，但长度超过15的单词将只截取保留前15个单词字符。而合法的“单词字符”为大小写字母、数字和下划线，其它字符均认为是单词分隔符。注意“单词”不区分英文大小写，例如“PAT”和“pat”被认为是同一个单词。输入给出一段非空文本，最后以符号#结尾。输入保证存在至少10个不同的单词。输出按照词频递减的顺序，按照“词频:单词”的格式输出词频最大的前10%的单词。若有并列，则按递增字典序输出。
-

程序运行结果

输入输出重定向.py - C:\Users\lenovo\Desktop\python的MOOC\第7章\程序和文件\输入...

File Edit Format Run Options Window Help

```
import sys
s=sys.stdin.read();strs=s[:s.find('#')]
for k in set([i for i in strs if i.isalnum()==False and i!='_']):
    strs=strs.replace(k, ' ')
strs=strs.rstrip(' ').lower().split()
counts=dict()
for i in strs:
    k=i[:15]
    if k not in counts:
        counts[k]=1
    else:
        counts[k]+=1
ans=sorted(counts.items(), key=lambda x:(-x[1], x[0]))
print(len(counts))
for i in range(0,int(0.1*len(counts))):
    print(str(ans[i][1])+':'+ans[i][0])
```

Python 3.6.8 Shell

File Edit Shell Debug Options Window Help

```
Python 3.6.8 (tags/v3.6.8:3c6b436a57, Dec 24 2018, 00:16:47) [MSC v.1916 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>>
===== RESTART: C:\Users\lenovo\Desktop\python的MOOC\第7章\程序和文件\输入输出重定向.py =====
This is a test.
The word "this" is the word with the highest frequency.
Longlonglonglongword should be cut off, so is considered
as the same as longlonglonglonee.  But this_8 is different
than this, and this, and this...#
this line should be ignored.
23
5:this
4:is
>>> counts
{'this': 5, 'is': 4, 'a': 1, 'test': 1, 'the': 4, 'word': 2, 'with': 1, 'highest': 1, 'frequency': 1, 'longlonglonglon': 2, 'should': 1, 'be': 1, 'cut': 1, 'off': 1, 'so': 1, 'considered': 1, 'as': 2, 'same': 1, 'but': 1, 'this_8': 1, 'different': 1, 'than': 1, 'and': 2}

>>>
```

输入处理

- “ s=sys.stdin.read()”表示重定向为键盘输入
- 输入“Ctrl-d”结束输入
- “ s[:s.find(‘#’)]”
- 表示取输入字符串，以符号“#”结尾

产生词频字典

```
for k in set([i for i in strs if i.isalnum()==False and i!='_']):
    strs=strs.replace(k, ' ') #其它字符均认为是单词分隔符
#去掉空格，全部变小写，变成列表
strs=strs.rstrip(' ').lower().split() #全部变小写
counts=dict()
for i in strs:
        k=i[:15]      #取前15个字符
        if k not in counts:
                counts[k]=1
        else:
                counts[k]+=1
```

排序及输出

- #词频递减的顺序输出，从大到小
- #若有并列，则按递增字典序，从小到大
- #用负数把从大到小变为从小到大
- ans=sorted(counts.items(), key=lambda x:(-x[1], x[0]))
- print(len(counts))
-
- for i in range(0,int(0.1*len(counts))): 词频最大的前10%
  print(str(ans[i][1])+':'+ans[i][0])