从链表到字符串

>>> silly = ['We', 'called', 'him', 'Tortoise', 'because', 'he', 'taught', 'us', '.']

>>> ' '.join(silly)

'We called him Tortoise because he taught us .'>>

';'.join(silly)

'We;called;him;Tortoise;because;he;taught;us;.'

>>> ''.join(silly)

'WecalledhimTortoisebecausehetaughtus.'

>>>

‘;’.join()的意思是，取出silly中的所有项目，将他们连接成一个大的字符串，使用；作为分隔符。

我们常常需要为了更好地维护和阅读文本，需要对文本进行字符串格式表达。

>>> '%s->%d;' % ('cat',3)

'cat->3;'

>>> '%s->' % 'cat'

'cat->'

>>> 'I want a %s rught now' % 'coffee'

'I want a coffee rught now'

>>>

>>> '%s wants a %s %s' % ('Lee', 'sandwich', 'for lunch')

'Lee wants a sandwich for lunch'

>>>

%s和d%被称为转换说明符。他们以%字符开始，以一个转换字符如s或d结束。其中包含转换说明符的字符串被称为格式化字符串。

>>> menu = ['sandwich','span fritter','pancake']

>>> template = 'Lee wants a %s right now'

>>> for snack in menu:

print template % snack

Lee wants a sandwich right now

Lee wants a span fritter right now

Lee wants a pancake right now

>>>

文本对齐

>>> '%6s' % 'dog' #右对齐

' dog'

>>> '%-6s' % 'dog' #负号表示左对齐

'dog '

>>> line='%-\*s' #如果事先不知道文本的宽度，可以用\*代替。

>>> width = 6

>>> line % (width,'dog')

'dog '

>>>

>>> from \_\_future\_\_ import division

>>> count, total = 3205,9375

>>> "accuracy for %d words: %2.4f%%" % (total, 100 \* count/total)

'accuracy for 9375 words: 34.1867%'

>>> "accuracy for %d words: %2.4f%" % (total, 100 \* count/total)

Traceback (most recent call last):

File "<pyshell#29>", line 1, in <module>

"accuracy for %d words: %2.4f%" % (total, 100 \* count/total)

ValueError: incomplete format

>>> "accuracy for %d words: %2.4f" % (total, 100 \* count/total)

'accuracy for 9375 words: 34.1867'

>>>

%2.4f%% 2表示小数点前有两位，4表示小数点后有4位，因为34.1867后面需要跟一个百分比符号，为了不引起歧义，规定再加一个%，表示直接输出前一个%。

>>> "accuracy for %d words: %.4f" % (total, 100 \* count/total)

'accuracy for 9375 words: 34.1867'

>>> "accuracy for %d words: %.f" % (total, 100 \* count/total)

'accuracy for 9375 words: 34'

>>>

.4f表示不知道小数点前有几位

.f表示取整

def tabulate(cfdist, words, categories):

print '%-16s' % 'Category', # column headings

for word in words:

print '%6s' % word,

print

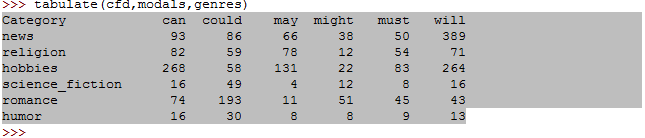
for category in categories:

print '%-16s' % category, # row heading

for word in words: # for each word

print '%6d' % cfdist[category][word], # print table cell

print # end the row

>>> genres = ['news', 'religion', 'hobbies', 'science\_fiction', 'romance', 'humor']  
>>> modals = ['can', 'could', 'may', 'might', 'must', 'will']  


制表是对齐方法最需要的地方之一。

将结果写入文件。

>>> import pprint

>>> output\_file = open('D:\KuaiPan\Python\NLP\Chapter 3\output.txt','w')

>>> output\_file

<open file 'D:\\KuaiPan\\Python\\NLP\\Chapter 3\\output.txt', mode 'w' at 0x05600180>

>>> words = set(nltk.corpus.genesis.words('english-kjv.txt'))

>>> for word in sorted(words):

output\_file.write(word + "\n")

>>>

>>> output\_file.write(str(len(words)) + "\n")

>>> output\_file.close()

