关于BeautifulSoup的总结（一）

功能

BeautifulSoup是用来从HTML或XML中提取数据的Python库。

导入

使用方法：

```
from bs4 import BeautifulSoup
soup = BeautifulSoup(html)
```

编码

soup使用Unicode编码。

对象种类

有四种类型：Tag，NavigableString，BeautifulSoup，Comment。
BeautifulSoup将文档转化为树形结构，每个节点都是上述四种类型的Python对象。

1.Tag

与XML和HTML中Tag对象相同。
如：
soup = BeautifulSoup(Extremely bold)
soup.b就是一个Tag对象。

1. Name
   tag.name　可获取，可更改

2. Attribute
   一个Tag对象可以有多个属性，操作方法和字典相同，如上述Tag对象b就有一个class属性：
   soup.b['class']
   或者使用get方法*soup.b.get('class')*

获取所有属性键值对：
soup.b.attrs

tag的属性可添加、删除（del soup.b['class']）、修改，和字典方法相同。

如果一个属性key对应多个value，则返回一个value的list，如：
css_soup = BeautifulSoup(' ')
css_soup.p['class']
输出：*["body", "strikeout"]*

这种多个值的属性是需要在HTML中有定义的，如果并没有被定义为多值属性，则返回字符串：
id_soup = BeautifulSoup(' ')
id_soup.p['id']
输出*'my id'*

如果转换的是XML文档，则不会存在多值属性，返回字符串。

可以使用list或字符串对属性赋值。

2. NavigableString

Tag中的字符串即为NavigableString对象。
tag.string

在BeautifulSoup之外使用该类型，推荐转换为Unicode：
unicode(Tag.string)

tag中包含的字符串不可编辑，只能替换：
tag.string.replace_with(new string)

tag能够包含其他tag或字符串，而NavigableString则不能包含其他对象。不支持*.content*，*.string*，*find()*，只支持部分遍历文档树和搜索文档树中的属性。

3. BeautifulSoup

表示的是一个文档的全部内容，大部分情况可当做Tag对象，支持遍历文档树和搜索文档树的大部分属性。
而在HTML或XML中并没有叫做BeautifulSoup的Tag，所以并没有name和attribute属性，但是有个特殊属性：
soup.name
输出*u'[document]'*

4. Comment

Comment类型是NavigableString类型的子类，BeautifulSoup中也有同样道理的一些其他类型。

遍历文档树

BeautifulSoup对象作为一棵树，有多个节点。对于一个节点，相对于它所在的位置，有子节点、父节点、兄弟节点。

1. 子节点

一个Tag可包含多个Tag以及字符串，这些都是这个Tag的子节点。而NavigableString不会有子节点。

如果想要获得某个Tag，上述已提到方法：
soup.tag_name
通过点取属性，只能获得当前名字的第一个tag，若要获取所有，需要使用搜索文档树中的方法:
soup.find_all('tag_name')

tag的*.contents*属性可将所有子节点以列表的方式输出。
可通过tag的*.children*生成器，对所有子节点进行遍历。

*.contents*和*.children*只对获取Tag的直接子节点，*.descendants*可用于对Tag的所有子孙节点进行遍历。

如果tag只有一个NavigableString类型子节点，则可用*.string*获取。如果包含多个，使用*.strings*遍历。若输出的字符串中包含空格或空行，使用*.stripped_strings*去除。

2. 父节点

当前节点的父节点：*.parent*
当前节点的所有父辈节点：*.parents*

3. 兄弟节点

拥有同一父节点的节点之间。
.next_sibling
.previous_sibling
同理，所有兄弟节点：
.next_siblings
.previous_siblings

指向下一个或上一个解析对象：
.next_element
.previous_element
.next_elements
.previous_elements