爬中真的还是很重要的

以后是数据为王的时代

一: HtmlAgilityPack来解析html

使用HtmlAgilityPack类库,可以不用正则表达式来解析html而直接用xpath来解析对应的节点

HtmlAgilityPack API简明介绍

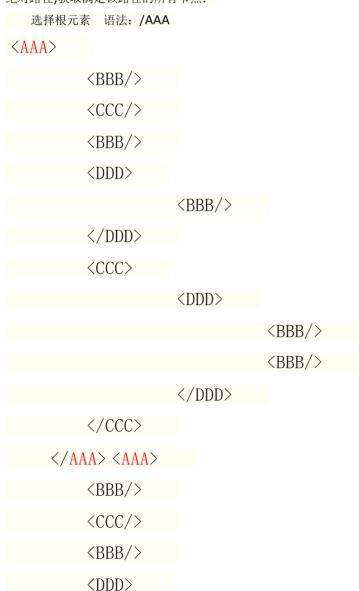
在HtmlAgilityPack中常用到的类有HtmlDocument、HtmlNodeCollection、HtmlNode和HtmlWeb等。

其流程一般是先获取HTML,这个可以通过HtmlDocument的Load()或LoadHtml()来加载静态内容,或者也可以HtmlWeb的Get()或Load()方法来加载网络上的URL对应的HTML。

得到了HtmlDocument的实例之后,就可以用HtmlDocument的DocumentNode属性,这是整个HTML文档的根节点,它本身也是一个HtmlNode,然后就可以利用HtmlNode的SelectNodes()方法返回多个HtmlNode的集合对象HtmlNodeCollection,也可以利用HtmlNode的SelectSingleNode()方法返回单个HtmlNode

二: xPath来查找对应的节点

1、斜线/从根节点选择,基本的XPath语法类似于在一个文件系统中定位文件,如果路径以斜线 / 开始,那么该路径就表示到一个元素的绝对路径,获取满足该路径的所有节点:



```
<BBB/>
</DDD>
<CCC>
   <DDD>
       <BBB/>
     <BBB/>
 </DDD>
</CCC>
</AAA>
选择AAA的所有CCC子元素 语法: /AAA/CCC
<AAA>
<BBB/>
 <CCC/>
 <BBB/>
 <DDD>
 <BBB/>
</DDD>
 <CCC>
   <DDD>
        <BBB/>
      <BBB/>
 </DDD>
</CCC>
</AAA>
2、如果路径以双斜线 // 开头,则表示选择文档中所有满足双斜线//之后规则的元素(无论层级关系),//会做全文档扫描
<AAA>
<BBB/>
 <CCC/>
 <BBB/>
 <DDD>
 <BBB/>
</DDD>
<CCC>
```

```
<DDD>
            <BBB/>
                <BBB/>
  </DDD>
</CCC>
</AAA>
选择所有父元素是DDD的BBB元素 语法: //DDD/BBB
<AAA>
<BBB/>
<CCC/>
 <BBB/>
 <DDD>
   <<u>BBB</u>/>
 </DDD>
 <CCC>
      <DDD>
           <BBB/>
            <BBB/>
  </DDD>
</CCC>
</AAA>
3、点"."选取当前节点
4、".."选取当前节点的父节点
5、"@"选取属性
选择所有的id属性 语法: //@id 注意: 这里选取的是属性而不是方法。
<AAA>
 <BBB id = "b1"/>
 <BBB id = "b2"/>
<BBB name = "bbb"/>
<BBB/>
</AAA>
选择有id属性的BBB元素 语法: //BBB[@id] 区别于上面的属性选择
<AAA>
<<u>BBB</u> id = "b1"/>
```

```
\langle BBB | id = "b2"/\rangle
<BBB name = "bbb"/>
<BBB/>
</AAA>
选择id属性为"id1"的BBB元素 语法: //BBB[@id="b1"]
<AAA>
<<u>BBB</u> id = "b1"/>
<BBB id = "b2"/>
<BBB name = "bbb"/>
<BBB/>
</AAA>
选择有任意属性的BBB元素 语法: //BBB[@*]
<AAA>
\langle BBB | id = "b1"/\rangle
\langle BBB \mid id = "b2"/\rangle
<BBB name = "bbb"/>
<BBB/>
</AAA>
选择不具有任何属性的BBB元素 语法: //BBB[not(@*)]
<AAA>
\langle BBB \mid id = "b1"/\rangle
<BBB id = "b2"/>
<BBB name = "bbb"/>
<BBB/>
</AAA>
```

本篇主要学习XPath 使用路径表达式在 XML 文档中选取节点。斜线"/"是从根节点选择,双斜线"//"从匹配选择的当前节点选择文档中的节点,而不考虑它们的位置,点"."选取当前节点,两点".."选取当前节点的父节点,"@"为选取属性。