Python程序设计

第九讲 Web应用与网络爬虫

开发网络爬虫程序

张 华

WHU

网络爬虫概述

网络爬虫是什么

- 网络爬虫（网络蜘蛛、网络机器人、网页追逐者），可以按照指定的规则（网络爬虫算法）自动浏览或抓取网络中的信息。
- 通过Python可以很轻松的编写爬虫程序。

网络爬虫概述

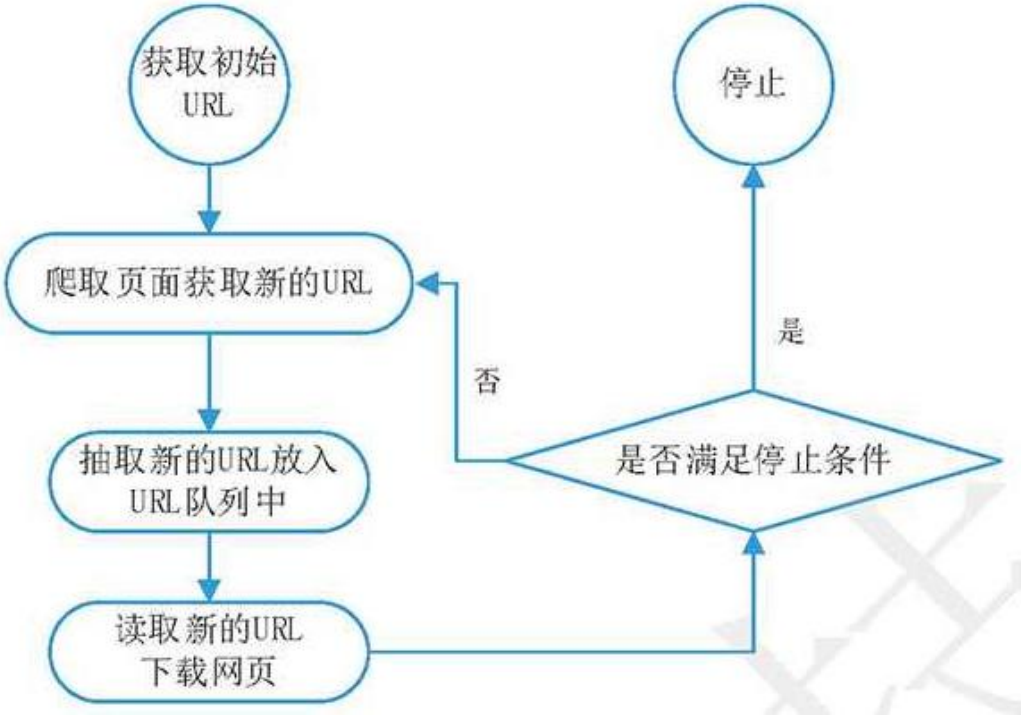
网络爬虫的分类

- 按照实现的技术和结构可以分为以下几种类型：
  - 通用网络爬虫（全网爬虫）
  - 聚焦网络爬虫（主题爬虫）
  - 增量式网络爬虫（抓取有更新的网页）
  - 深层网络爬虫（除了表层网页，也爬取深层网页）
  - 几类爬虫的组合体

网络爬虫概述

网络爬虫的基本原理

通用网络爬虫的工作流程：



网络爬虫的常用技术

网络请求

- 通过HTTP协议请求URL地址对应的网页，并下载网页。
- 可以使用以下方式：
  - **urllib模块**
    - Python自带，包括urlopen()函数。
  - **urllib3模块**
    - 一个功能强大，条理清晰，用于HTTP客户端的Python库。
    - Anaconda已集成
  - **requests模块**
    - 基于urllib编写的，比urllib更加方便，操作更人性化。
    - Anaconda已集成

网络爬虫的常用技术

网络请求示例

通过requests模块请求指定网页

```
>>> import requests
>>> response = requests.get('http://www.baidu.com')
>>> print(response.status_code)
200
>>> print(response.text)
```

```
<!DOCTYPE html>
<!--STATUS OK--><html> <head><meta http-equiv=content-type content=text/html;charset=utf-8><meta http-equiv=X-UA-Compati
ble content=IE=Edge><meta content=always name=referrer><link rel=stylesheet type=text/css href=http://s1.bdstatic.com/r/
www/cache/bdorz/baidu.min.css><title>ç□¾åº¦ä,□ä,□ï¼□ä½  å° ±ç□¥é□□</title></head> <body link=#0000cc> <div id=wr
apper> <div id=head> <div class=head_wrapper> <div class=s_form> <div class=s_form_wrapper> <div id=lg> <img hidefocus=t
rue src=//www.baidu.com/img/bd_logo1.png width=270 height=129> </div> <form id=form name=f action=//www.baidu.com/s clas
s=fm> <input type=hidden name=bdorz_come value=1> <input type=hidden name=ie value=utf-8> <input type=hidden name=f valu
e=8> <input type=hidden name=rsv_bp value=1> <input type=hidden name=rsv_idx value=1> <input type=hidden name=tn value=b
aidu><span class="bg s_ipt_wr"><input id=kw name=wd class=s_ipt value maxlength=255 autocomplete=off autofocus></span><s
pan class="bg s_btn_wr"><input type=submit id=su value=ç□¾åº¦ä,□ä,□ class="bg s_btn"></span> </form> </div> </div> <d
iv id=u1> <a href=http://news.baidu.com name=tj_trnews class=mnav>æ□° é□»</a> <a href=http://www.hao123.com name=tj_tr
hao123 class=mnav>hao123</a> <a href=http://map.baidu.com name=tj_trmap class=mnav>å□° å□¾</a> <a href=http://v.baidu.
com name=tj_trvideo class=mnav>è§ □é¢□</a> <a href=http://tieba.baidu.com name=tj_trtieba class=mnav>è´´å□ §</a> <nos
cript> <a href=http://www.baidu.com/bdorz/login.gif?login&amp;tpl=mn&amp;u=http%3A%2F%2Fwww.baidu.com%2f%3fbdorz_come%3d
1 name=tj_login class=lb>ç□»å½□</a> </noscript> <script>document.write('<a href="http://www.baidu.com/bdorz/login.gif?
login&tpl=mn&u='+ encodeURIComponent(window.location.href+ (window.location.search === "" ? "?" : "&")+ "bdorz_come=1")+
'" name="tj_login" class="lb">ç□»å½□</a>');</script> <a href=//www.baidu.com/more/ name=tj_briicon class=bri style="d
isplay: block;">æ□´å□□äº §å□□</a> </div> </div> </div> <div id=ftCon> <div id=ftConw> <p id=lh> <a href=http://home
.baidu.com>å□³äº □ç□¾åº¦</a> <a href=http://ir.baidu.com>About Baidu</a> </p> <p id=cp>&copy;2017 Baidu <a h
ref=http://www.baidu.com/duty/>ä½¿ç□¨ç□¾åº¦å□□å¿□è¯»</a>  <a href=http://jianyi.baidu.com/ class=cp-feedback>
æ□□è§ □å□□é¦ □</a> äº¬ICPè¯□030173å□ •  <img src=//www.baidu.com/img/gs.gif> </p> </div> </div> </div>
</body> </html>
```

网络爬虫的常用技术

解析网页，提取信息

- BeautifulSoup是一个可以从HTML或XML文件中提取数据的Python库。
  - 它能够通过你喜欢的转换器实现惯用的文档导航、查找、修改文档的方式，能够帮你节省数小时甚至数天的工作时间。
  - Anaconda已集成该库。
- BeautifulSoup不仅支持HTML解析器，还支持一些第三方的解析器，如 lxml，html5lib，但是需要安装相应的库。如果不安装，则会使用Python默认的解析器。
  - Anaconda已集成lxml和html5lib。

网络爬虫的常用技术

解析网页，提取信息

几种HTML解析器的用法和优缺点

解 析 器	用 法	优 点	缺 点
Python 标准库	BeautifulSoup(markup, "html.parser")	Python 标准库 执行速度适中	（在 Python 2.7.3 或 3.2.2 之前的版本中） 文档容错能力差
lxml 的 HTML 解析器	BeautifulSoup(markup, "lxml")	速度快 文档容错能力强	需要安装 C 语言库
lxml 的 XML 解析器	BeautifulSoup(markup, "lxml-xml") BeautifulSoup(markup, "xml")	速度快 唯一支持 XML 的解析器	需要安装 C 语言库
html5lib	BeautifulSoup(markup, "html5lib")	最好的容错性 以浏览器的方式解析文档 生成 HTML5 格式的文档	速度慢 不依赖外部扩展

网络爬虫的常用技术

解析网页示例

- 用BeautifulSoup解析爬到的百度主页。

```
>>> html_doc = response.text
>>> from bs4 import BeautifulSoup
>>> soup = BeautifulSoup(html_doc, 'lxml')
>>> print(soup.prettify())   # prettify()是对解析后的代码进行格式化
```

```
<!DOCTYPE html>
<!--STATUS OK-->
<html>
 <head>
  <meta content="text/html;charset=utf-8" http-equiv="content-type"/>
  <meta content="IE=Edge" http-equiv="X-UA-Compatible"/>
  <meta content="always" name="referrer"/>
  <link href="http://s1.bdstatic.com/r/www/cache/bdorz/baidu.min.css" rel="stylesheet" type="text/css"/>
  <title>
   ç□¾å°¦ä,□ä,□ï¼□ä½  å° ±ç□¥é□□
  </title>
 </head>
 <body link="#0000cc">
  <div id="wrapper">
   <div id="head">
    <div class="head_wrapper">
     <div class="s_form">
      <div class="s_form_wrapper">
       <div id="lg">
        <img height="129" hidefocus="true" src="//www.baidu.com/img/bd_logo1.png" width="270"/>
       </div>
       <form action="//www.baidu.com/s" class="fm" id="form" name="f">
        <input name="bdorz_come" type="hidden" value="1"/>
        <input name="ie" type="hidden" value="utf-8"/>
```

网络爬虫的常用技术

解析网页：HTML的基本概念

文本
声音
图像
视频
超文本

HTML是WWW(World Wide Web)的信息组织方式

- 标记后的信息可形成信息组织结构，增加了信息维度
- 标记后的信息可用于通信、存储或展示
- 标记后的信息更利于程序理解和运用

网络爬虫的常用技术

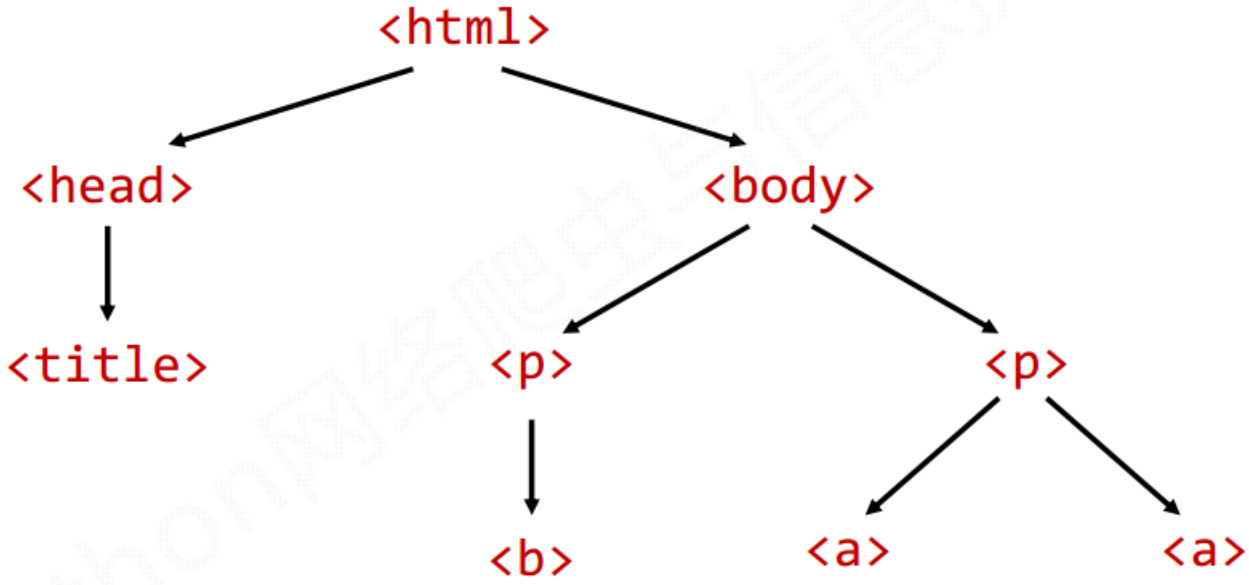
解析网页：HTML的基本结构

```
<html>
    <head>
        <title>This is a python demo page</title>
    </head>
    <body>
        <p class="title">
            <b>The demo python introduces several python courses.</b>
        </p>
        <p class="course">
            Python is a wonderful general-purpose programming language. You can learn Python from novice to
            professional by tracking the following courses:
            <a href="http://www.icourse163.org/course/BIT-268001" class="py1" id="link1">Basic Python</a>
            and
            <a href="http://www.icourse163.org/course/BIT-1001870001" class="py2" id="link2">Advanced Python</a>
            .
        </p>
    </body>
</html>
```

HTML通过预定义的<>…</>标签形式组织不同类型的信息

网络爬虫的常用技术

解析网页：HTML的基本结构

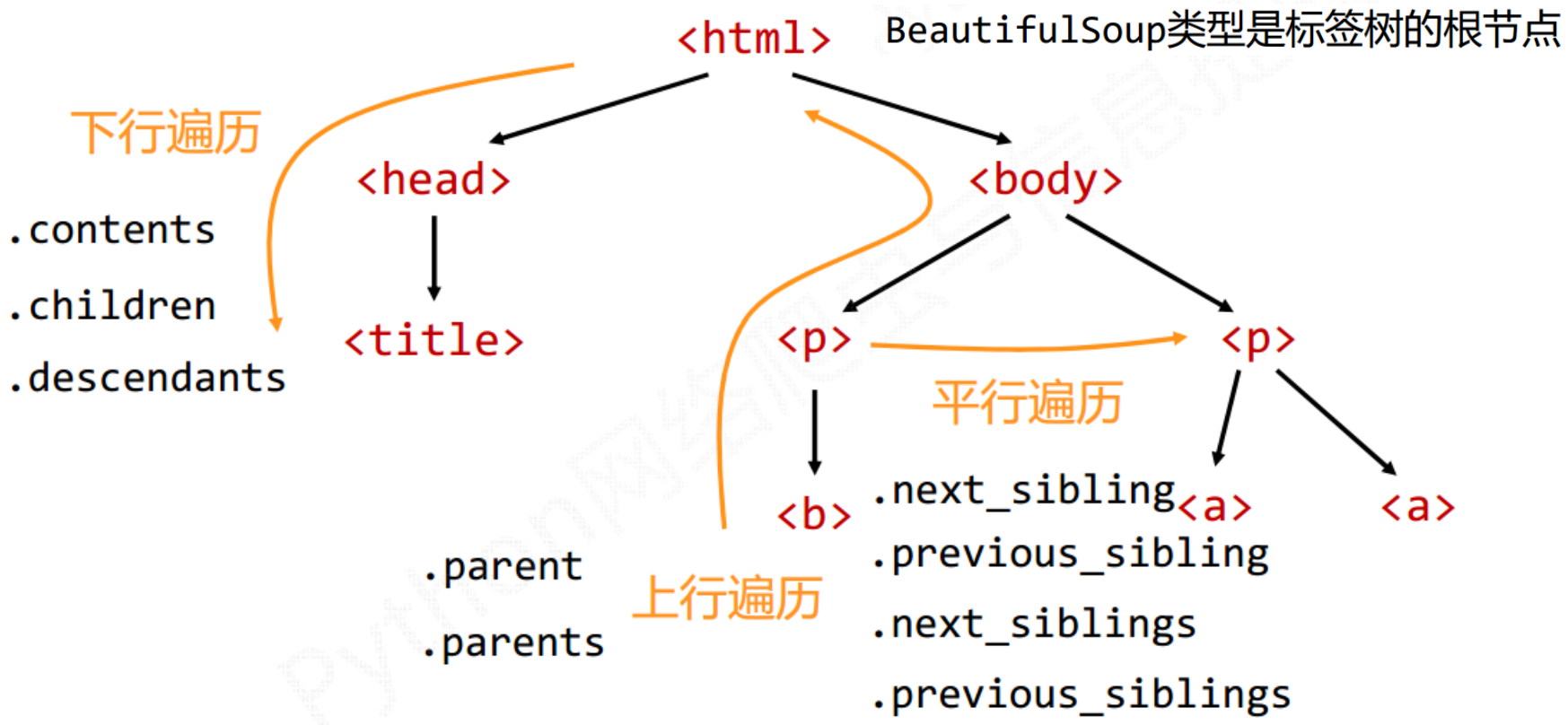


<>…</>构成了所属关系，形成了标签的树形结构

网络爬虫的常用技术

解析网页：HTML的基本结构

用BeautifulSoup遍历标签树



网络爬虫开发常用框架

开发网络爬虫常用的框架

- 爬虫框架就是一些爬虫项目的半成品。
  - 可以将一些爬虫常用的功能写好，然后留下一些接口，在不同的爬虫项目当中调用适合自己项目的接口，再编写少量的代码实现自己需要的功能。
  - 框架可以为开发人员节省很多精力和时间。
- 常用的框架
  - **Scrapy**
  - **Crawley**
  - **PySpider**

网络爬虫开发技术路线选择

requests vs. Scrapy

页面级爬虫	网站级爬虫
功能库	框架
并发性考虑不足，性能较差	并发性好，性能较高
重点在于页面下载	重点在于爬虫结构
定制灵活	一般定制灵活，深度定制困难
上手十分简单	入门稍难

网络爬虫开发技术路线选择

- **选择用哪种技术路线？**
  - 非常小的需求，用**requests**库
  - 不太小的需求，用**Scrapy**框架
  - 对于定制程度很高的需求（不考虑规模），打算自搭框架，**requests > Scrapy**

案例1：两岸四地最好大学排名

目标

- http://www.zuihaodaxue.cn/Greater_China_Ranking2018_0.html
- 提取大学排名

最好大学网
ZUIHAODAXUE.COM

网站首页 | 中国大学排名 | 世界大学排名 | 原创分析 | 要闻资讯 | 院校信息 | 会议

首页 / 中国大学排名 / 软科中国两岸四地大学排名 2018

软科中国两岸四地大学排名 2018

2018 2017 2016 2015 2014 2013 2012 2011

两岸四地百强排名 | 排名方法 | 相关资源 | 相关文章

排名	学校	地区	总分	指标得分						
				人才培养						
				研究生比例(5%)	留学生比例(5%)	师生比(5%)	博士学位授予数(10%)		校友获奖(10%)	
							总量	师均	总量	生均
1	清华大学（北京）	大陆	100.0	89.1	9.0	69.3	74.6	71.3	44.4	21.6
2	北京大学	大陆	80.6	94.3	8.1	57.3	100.0	100.0	55.6	23.4
3	清华大学（新竹）	台湾	65.0	72.0	10.3	40.9	13.3	66.4	66.7	100.0

案例1：两岸四地最好大学排名

功能描述

- 输入：大学排名URL链接
- 输出：大学排名信息的屏幕输出
  - 排名，大学名称，地区，总分
- 技术路线：**requests + bs4**
- 定向爬虫：仅对输入URL进行爬取，不扩展爬取

案例1：两岸四地最好大学排名

可行性

用浏览器查看网页源码


```
<tbody>
<tr><td>1</td><td class="align-left"><a target="_blank"href="World-University-Rankings/Tsinghua-University.html"><div align="left">清华大学（北京）</div></a></td><td>大陆</td><td>100.0</td><td class="hidden-xs">89.1</td><td class="hidden-xs">9.0</td><td class="hidden-xs">69.3</td><td class="hidden-xs">74.6</td><td class="hidden-xs">71.3</td><td class="hidden-xs">44.4</td><td class="hidden-xs">21.6</td></tr><tr><td>2</td><td class="align-left"><a target="_blank"href="World-University-Rankings/National-Tsing-Hua-University.html"><div align="left">清华大学（新竹）</div></a></td><td>台湾</td><td>65.0</td><td class="hidden-xs">72.0</td><td class="hidden-xs">10.3</td><td class="hidden-xs">40.9</td><td class="hidden-xs">13.3</td><td class="hidden-xs">66.4</td><td class="hidden-xs">66.7</td><td class="hidden-xs">100.0</td></tr><tr><td>4</td><td class="align-left"><a target="_blank"href="World-University-Rankings/The-Chinese-University-of-Hong-Kong.html"><div align="left">香港中文大学</div></a></td><td>香港</td><td>62.8</td><td class="hidden-xs">24.6</td><td class="hidden-xs">30.1</td><td class="hidden-xs">68.7</td><td class="hidden-xs">33.7</td><td class="hidden-xs">67.4</td><td class="hidden-xs">0.0</td><td class="hidden-xs">0.0</td></tr><tr><td>5</td><td class="align-left"><a target="_blank"href="World-University-Rankings/Zhejiang-University.html"><div align="left">浙江大学</div></a></td><td>大陆</td><td>60.6</td><td class="hidden-xs">76.3</td><td class="hidden-xs">7.5</td><td class="hidden-xs">51.6</td><td class="hidden-xs">83.7</td><td class="hidden-xs">77.8</td><td class="hidden-xs">0.0</td><td class="hidden-xs">0.0</td></tr><tr><td>6</td><td class="align-left"><a target="_blank"href="World-University-Rankings/University-of-Science-and-Technology-of-China.html"><div align="left">中国科学技术大学</div></a></td><td>大陆</td><td>57.9</td><td class="hidden-xs">100.0</td><td class="hidden-xs">3.1</td><td class="hidden-xs">52.3</td><td class="hidden-xs">45.2</td><td class="hidden-xs">95.4</td><td class="hidden-xs">0.0</td><td class="hidden-xs">0.0</td></tr><tr><td>7</td><td class="align-left"><a target="_blank"href="World-University-Rankings/Shanghai-Jiao-Tong-University.html"><div align="left">上海交通大学</div></a></td><td>大陆</td><td>56.2</td><td class="hidden-xs">85.3</td><td class="hidden-xs">8.0</td><td class="hidden-xs">54.6</td><td class="hidden-xs">57.4</td><td class="hidden-xs">65.8</td><td class="hidden-xs">0.0</td><td class="hidden-xs">0.0</td></tr><tr><td>8</td><td class="align-left"><a target="_blank"href="World-University-Rankings/The-University-of-Hong-Kong.html"><div align="left">香港大学</div></a></td><td>香港</td><td>54.9</td><td class="hidden-xs">65.3</td><td class="hidden-xs">46.3</td><td class="hidden-xs">30.6</td><td class="hidden-xs">/</td><td class="hidden-xs">/</td><td class="hidden-xs">0.0</td><td class="hidden-xs">0.0</td></tr><tr><td>9</td><td class="align-left"><a target="_blank"href="World-University-Rankings/National-Taiwan-University.html"><div align="left">台湾大学</div></a></td><td>台湾</td><td>54.6</td><td class="hidden-xs">70.3</td><td class="hidden-xs">10.0</td><td class="hidden-xs">50.9</td><td class="hidden-xs">29.2</td><td class="hidden-xs">46.1</td><td class="hidden-xs">55.6</td><td class="hidden-xs">32.9</td></tr><tr><td>10</td><td class="align-left"><a
```

案例1：两岸四地最好大学排名

可行性

试验爬取网页

```
>>> import requests
>>> response =
requests.get('http://www.zuihaodaxue.cn/Greater_China_Ranking2018_0.html')
>>> response.status_code
200
>>> response.text
```

'ï»¿<!DOCTYPE html>\r\n<html lang="zh">\r\n<head>\r\n<meta charset="utf-8">\r\n<meta name="viewport" content="width=device-width, initial-scale=1">\r\n<meta name="keywords"\r\n\tcontent="2018,ä,ÛåÂ²,å\x9b\x9bå\x9c°åÛ§å\xad¦æ\x8e\x92å\x90\x8d,ä,\xadå\x9b½åÛ§é\x99\x86é«\x98æ\xa0¡æ\x8e\x92å\x90\x8d,ä,\xadå\x9b½,é¦\x99æ,¯,å\x8f°æ'¼,æ¼³é\x97¨,åÛ§é\x99\x86,åÛ§å\xad¦,æ\x8e\x92å\x90\x8d">\r\n<meta name="description"\r\n\tcontent="2018ä,\xadå\x9b½ä,ÛåÂ²,å\x9b\x9bå\x9c°åÛ§å\xad¦æ\x8e\x92å\x90\x8dä°\x8e9æ\x9c\x8822æ\x97¥ç\x94±æ\x9c\x80å¥½åÛ§å\xad¦ç½\x91æ\xad£å¼\x8få\x8f\x91å,\x83ï¼\x8cæ,\x85å\x8d\x8eåÛ§å\xad¦ï¼\x88å\x8c\x97äº¬ï¼\x89è¿\x9eç»\xadä°\x94æ¬¡è\x9d\x89è\x81\x94ä,ÛåÂ²,å\x9b\x9bå\x9c°åÛ§å\xad¦æ\x8e\x92å\x90\x8dæ¦\x9cé¦\x96ï¼\x8cæ,\x85å\x8d\x8eåÛ§å\xad¦ï¼\x88æ\x96°ç«'ï¼\x89å\x92\x8cå\x8f°æ'¼åÛ§å\xad¦å\x88\x86è\x8e•ç¬¬äº\x8cå\x92\x8cç¬¬ä,\x89å\x90\x8dä\x80\x82å\x8c\x97äº¬åÛ§å\xad¦ä»\x8eå\x8e»å'´ç\x9a\x84ç¬¬äº\x94å\x90\x8dä,\x8aå\x8d\x87å\x88°ä»\x8aå'´ç\x9a\x84ç¬¬å\x9b\x9bå\x90\x8dä\x80\x82">\r\n<meta name="author" content="æ\x9c\x80å¥½åÛ§å\xad¦ç½\x91">\r\n<link rel="shortcut icon" href="houtai/templates/images/favicon.png" />\r\n<title>2018ä,\xadå\x9b½

案例1：两岸四地最好大学排名

程序结构

- 步骤1：从网络上获取大学排名网页内容
  - 定义函数：**getHTMLText()**
- 步骤2：提取网页内容中信息到合适的数据结构
  - 定义函数： **fillUnivList()**
- 步骤3：利用数据结构展示并输出结果
  - 定义函数： **printUnivRanking()**

案例1：两岸四地最好大学排名

程序实现

```
def main():
    univ_list = []
    url = 'http://www.zuihaodaxue.cn/Greater_China_Ranking2018_0.html'
    html_doc = getHTMLText(url)
    fillUnivList(univ_list, html_doc)
    printUnivRanking(univ_list, 25)


if __name__ == '__main__':
    main()
```

案例1：两岸四地最好大学排名

程序实现

```
import requests
import bs4

def getHTMLText(url):
    try:
        r = requests.get(url, timeout=30)
        r.raise_for_status()
        r.encoding = r.apparent_encoding
        return r.text
    except:
        return ""
```

案例1：两岸四地最好大学排名

程序实现

```
def fillUnivList(univ_list, html_doc):
    soup = bs4.BeautifulSoup(html_doc, 'html.parser')
    for tr in soup.find('tbody').children:
        if isinstance(tr, bs4.element.Tag):
            tds = tr('td')
            univ_list.append([tds[0].string, tds[1].string,
                              tds[2].string, tds[3].string])
```

案例1：两岸四地最好大学排名

程序实现

```
def printUnivRanking(univ_list, num):
    print('{:3}{:20}{:6}{:6}'.format('排名','学校','地区','总分'))
    for i in range(num):
        univ = univ_list[i]
        name = ('{:11}'.format(univ[1])).replace(' ', '  ')
        loc = ('{:4}'.format(univ[2])).replace(' ', '  ')
        print('{:5}{:11}{:4}{:8}'.format(univ[0], name, loc, univ[3]))
```

- 中英文混排对齐问题
  - 当中文字符宽度不够时，采用英文字符填充；中英文字符占用宽度不同。
  - 采用中文字符的空格填充 chr(12288)，此办法失效。
  - 于是，把1个西文空格换成2个西文空格。

案例1：两岸四地最好大学排名

程序实现

排名	学校	地区	总分
1	清华大学（北京）	大陆	100.0
2	北京大学	大陆	80.6
3	清华大学（新竹）	台湾	65.0
4	香港中文大学	香港	62.8
5	浙江大学	大陆	60.6
6	中国科学技术大学	大陆	57.9
7	上海交通大学	大陆	56.2
8	香港大学	香港	54.9
9	台湾大学	台湾	54.6
10	复旦大学	大陆	54.2
11	北京师范大学	大陆	52.8
12	香港科技大学	香港	50.1
13	香港城市大学	香港	46.2
14	交通大学（新竹）	台湾	45.8
15	南京大学	大陆	44.7
16	中山大学（广州）	大陆	42.8
17	华中科技大学	大陆	42.0
18	中国医药大学	台湾	41.6
19	阳明大学	台湾	39.8
20	香港理工大学	香港	39.2
21	澳门科技大学	澳门	39.0
22	武汉大学	大陆	38.9
23	西安交通大学	大陆	38.7
24	哈尔滨工业大学	大陆	38.1
25	成功大学	台湾	37.7

网络爬虫注意事项

网络爬虫引发的问题

- “性能骚扰”
  - Web服务器默认接收人类访问。
  - 受限于编写水平和目的，网络爬虫将会为Web服务器带来巨大的资源开销。
- 法律风险
  - 服务器上的数据有产权归属。
  - 网络爬虫获取数据后牟利将带来法律风险。
- 隐私泄露
  - 网络爬虫可能具备突破简单访问控制的能力，获得被保护数据，从而泄露个人隐私。

网络爬虫注意事项

- 服务器对网络爬虫的限制
  - 来源审查：判断User-Agent进行限制
    - 检查来访HTTP协议头的User-Agent域，只响应浏览器或友好爬虫的访问。
  - 发布公告：Robots协议
    - 告知所有爬虫网站的爬取策略，要求爬虫遵守。
  - 服务器可能还会自定义一些参数验证访问是否合法。
  - 服务器还会限制IP，限制IP的访问速度。