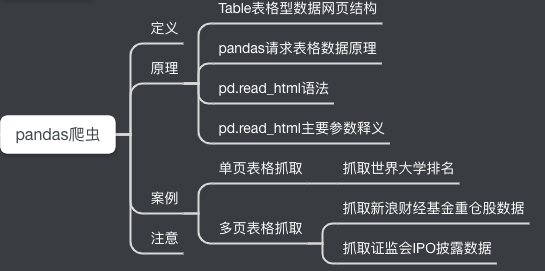
Pandas爬虫，竟能如此简单！  
众所周知，一般的爬虫套路无非是构造请求、解析网页、提取要素、存储数据等步骤。构造请求主要用到requests库，提取要素用的比较多的有xpath、bs4、css和re。一个完整的爬虫，代码量少则几十行，多则几百行，对于新手来说学习成本还是比较高的。  
那么，有没有什么方法只用几行代码就能爬下所需数据呢？答案是pandas。J哥自从知道了这个神器，尝试了多个网页数据爬取，屡战屡胜，简直不能再舒服！这家伙也太适合初学爬虫的小伙伴玩耍了吧！

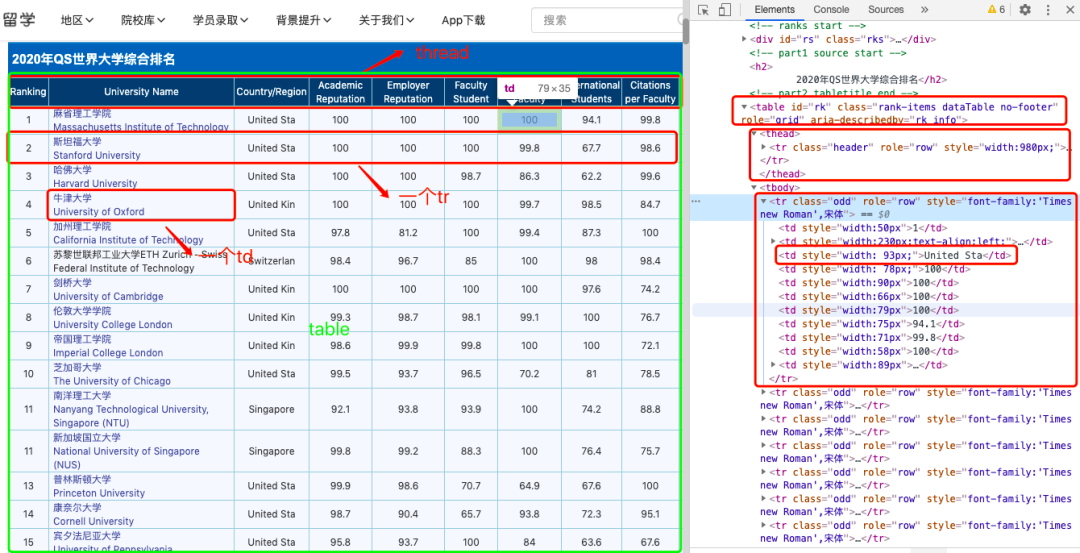


**定 义**

pandas中的pd.read\_html()这个函数，功能非常强大，可以轻松实现抓取Table表格型数据。无需掌握正则表达式或者xpath等工具，短短的几行代码就可以将网页数据抓取下来。

**原 理**

**一.Table表格型数据网页结构**pandas适合抓取Table表格型数据，那么咱们首先得知道什么样的网页具有Table表格型数据结构(有html基础的大佬可自行跳过这一part)。  
我们先来看个简单的例子。（快捷键F12可快速查看网页的HTML结构）



**指南者留学网**  
从这个世界大学排行的网站可以看出，数据存储在一个table表格中，thread为表头，tbody为表格数据，tbody中的一个tr对应表中的一行，一个td对应一个表中元素。  
  
我们再来看一个例子：



**新浪财经网**  
  
也许你已经发现了规律，以Table结构展示的表格数据，大致的网页结构如下：

<table class="..." id="...">

     <thead>

     <tr>

     <th>...</th>

     </tr>

     </thead>

     <tbody>

        <tr>

            <td>...</td>

        </tr>

        <tr>...</tr>

        <tr>...</tr>

        ...

        <tr>...</tr>

        <tr>...</tr>

    </tbody>

</table>

**Table表格一般网页结构**

只要网页具有以上结构，你就可以尝试用pandas抓取数据。  
**二.pandas请求表格数据原理**



**基本流程**

针对网页结构类似的表格类型数据，pd.read\_html可以将网页上的表格数据都抓取下来，并以DataFrame的形式装在一个list中返回。

**三.pd.read\_html语法及参数**

## 基本语法

pandas.read\_html(io, match='.+', flavor=None,

header=None,index\_col=None,skiprows=None,

attrs=None, parse\_dates=False, thousands=', ',

encoding=None, decimal='.', converters=None, na\_values=None,

keep\_default\_na=True, displayed\_only=True）

## 主要参数

**io** ：接收网址、文件、字符串；

**parse\_dates**：解析日期；

**flavor**：解析器；

**header**：标题行；

**skiprows**：跳过的行；

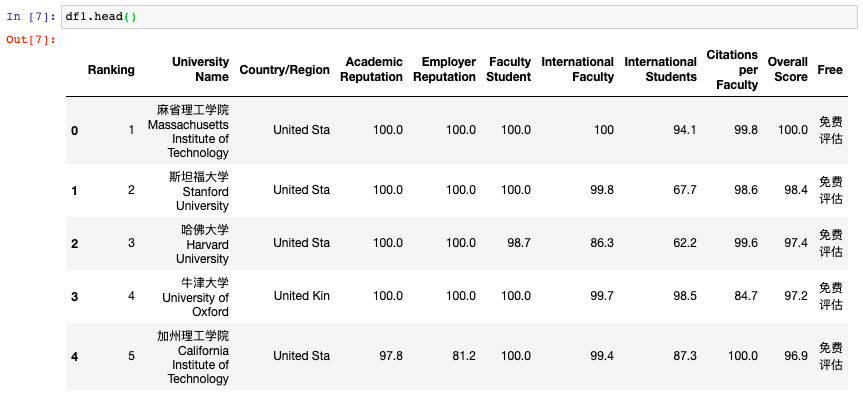
**attrs**：属性，比如 attrs = {'id': 'table'}

**实 战**

**一.案例1：抓取世界大学排名（1页数据）**

1import pandas as pd   
2import csv  
3url1 = 'http://www.compassedu.hk/qs'  
4df1 = pd.read\_html(url1)[0] #0表示网页中的第一个Table  
5df1.to\_csv('世界大学综合排名.csv',index=0)

没错，5行代码，几秒钟就搞定，我们来预览下爬取到的数据：



**世界大学排行榜**

**二.案例2：抓取新浪财经基金重仓股数据（6页数据）**

1import pandas as pd  
2import csv  
3df2 = pd.DataFrame()  
4for i in range(6):  
5    url2 = 'http://vip.stock.finance.sina.com.cn/q/go.php/vComStockHold/kind/jjzc/index.phtml?p={page}'.format(page=i+1)  
6    df2 = pd.concat([df2,pd.read\_html(url2)[0]])  
7    print('第{page}页抓取完成'.format(page = i + 1))  
8df2.to\_csv('./新浪财经数据.csv',encoding='utf-8',index=0)

没错，8行代码搞定，还是那么简单。如果对翻页爬虫不理解，可查看本公众号历史原创文章**「**[**实战|手把手教你用Python爬虫(附详细源码)**](http://mp.weixin.qq.com/s?__biz=MzAwMjYxNjcwMw==&mid=2456403634&idx=1&sn=41e3e71b16be2a0a25c1eaff063e16af&chksm=8d5aa2a2ba2d2bb4dc8e13bac62d8fef84910535d3c88c1d346de4f30f0995687cef810a3f74&scene=21#wechat_redirect)**」**，如果对DataFrame合并不理解，可查看本公众号历史原创文章**「**[**基础|Pandas常用知识点汇总(四)**](http://mp.weixin.qq.com/s?__biz=MzAwMjYxNjcwMw==&mid=2456405204&idx=1&sn=6d2d56b721976fad84bcac2356732289&chksm=8d5ad8c4ba2d51d2dd8f77601a2e7c41835bf682e2a903dbdd6a18b7110a9cd05bce6f94686b&scene=21#wechat_redirect)**」**。

我们来预览下爬取到的数据：

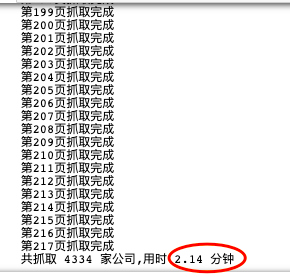


**基金重仓股数据**

**三.案例3：抓取证监会披露的IPO数据（217页数据）**

1import pandas as pd  
 2from pandas import DataFrame  
 3import csv  
 4import time  
 5start = time.time() #计时  
 6df3 = DataFrame(data=None,columns=['公司名称','披露日期','上市地和板块','披露类型','查看PDF资料']) #添加列名  
 7for i in range(1,218):    
 8    url3 ='http://eid.csrc.gov.cn/ipo/infoDisplay.action?pageNo=%s&temp=&temp1=&blockType=byTime'%str(i)  
 9    df3\_1 = pd.read\_html(url3,encoding='utf-8')[2]  #必须加utf-8，否则乱码  
10    df3\_2 = df3\_1.iloc[1:len(df3\_1)-1,0:-1]  #过滤掉最后一行和最后一列（NaN列）  
11    df3\_2.columns=['公司名称','披露日期','上市地和板块','披露类型','查看PDF资料'] #新的df添加列名  
12    df3 = pd.concat([df3,df3\_2])  #数据合并  
13    print('第{page}页抓取完成'.format(page=i))  
14df3.to\_csv('./上市公司IPO信息.csv', encoding='utf-8',index=0) #保存数据到csv文件  
15end = time.time()  
16print ('共抓取',len(df3),'家公司,' + '用时',round((end-start)/60,2),'分钟')

这里注意要对抓下来的Table数据进行过滤，主要用到iloc方法，详情可查看本公众号往期原创文章**「**[**基础|Pandas常用知识点汇总(三)**](http://mp.weixin.qq.com/s?__biz=MzAwMjYxNjcwMw==&mid=2456405086&idx=1&sn=650942fdeb7ebe117404445a5a64936f&chksm=8d5ad84eba2d5158b4a381dad6e098d7cb020f4833db52bfb89183242959b365c2049bcaa853&scene=21#wechat_redirect)**」**。另外，我还加了个程序计时，方便查看爬取速度。



2分钟爬下217页4334条数据，相当nice了。我们来预览下爬取到的数据**：**



**上市公司IPO数据**

需要注意的是，并不是所有表格都可以用pd.read\_html爬取，有的网站表面上看起来是表格，但在网页源代码中不是table格式，而是list列表格式。这种表格则不适用read\_html爬取，得用其他的方法，比如selenium。

[用Python爬取淘宝4403条大裤衩数据进行分析，终于找到可以入手的那一条](http://mp.weixin.qq.com/s?__biz=MzAwMjYxNjcwMw==&mid=2456405405&idx=1&sn=61da338357f0b4182e5f2690ae4b20c9&chksm=8d5ad98dba2d509baf7c1bba37410b4894a6c98b445be40db702f2681c4bc7797df3deb69c0e&scene=21#wechat_redirect)  
  
[睡地摊or租房？爬取某大型房产网站24685个房源信息并分析，助你选择](http://mp.weixin.qq.com/s?__biz=MzAwMjYxNjcwMw==&mid=2456405069&idx=1&sn=f74df8b3f4fc9383d3da2da7211dc612&chksm=8d5ad85dba2d514b7b128b5b361b8d496f1bac5498950ffc073cf714151b3917e948854bbe04&scene=21#wechat_redirect)  
  
[摆地摊or打工，爬取某大型招聘网站3万条招聘信息并分析，作何选择?](http://mp.weixin.qq.com/s?__biz=MzAwMjYxNjcwMw==&mid=2456404450&idx=1&sn=4cee4c349aaadfaba69a9ec3919b42e3&chksm=8d5addf2ba2d54e43b6306f68524c9c602a9a9e77041071b792d0bd4ebe96a0061d3c8d7616e&scene=21#wechat_redirect)  
[实战|利用Python获取财务金融数据的几种实用方法](http://mp.weixin.qq.com/s?__biz=MzAwMjYxNjcwMw==&mid=2456404246&idx=1&sn=1e3b647baf6b2c208033cc2b5376f9fa&chksm=8d5add06ba2d5410810b39467900035efcd9acf240998ceebbe1456b44631069d72b8bc12b1e&scene=21#wechat_redirect)  
[实战|Python爬虫并用Flask框架搭建可视化网站](http://mp.weixin.qq.com/s?__biz=MzAwMjYxNjcwMw==&mid=2456404005&idx=1&sn=381e740a8169ff5ac31a5df587383cb0&chksm=8d5adc35ba2d5523121f17d7ddb7bc06ae262548f0b8ca9e6020928fdc78e5bfdee9a40f5edc&scene=21#wechat_redirect)