

# 爬虫爬股票哪些信息 - Python , Node.js 哪个比较适合写爬虫-股识吧

## 一、能爬取股票帐号信息吗

能爬取股票帐号信息股票账户都是证券公司信息不会对外开放查询

## 二、Python , Node.js 哪个比较适合写爬虫

作者：梁川链接：<https://www.zhihu.com/question/23643061/answer/26322652>来源：知乎著作权归作者所有。

商业转载请联系作者获得授权，非商业转载请注明出处。

主要看你定义的“爬虫”干什么用。

1、如果是定向爬取几个页面，做一些简单的页面解析，爬取效率不是核心要求，那么用什么语言差异不大。

当然要是页面结构复杂，正则表达式写得巨复杂，尤其是用过那些支持xpath的类库/爬虫库后，就会发现此种方式虽然入门门槛低，但扩展性、可维护性等都奇差。

因此此种情况下还是推荐采用一些现成的爬虫库，诸如xpath、多线程支持还是必须考虑的因素。

2、如果是定向爬取，且主要目标是解析js动态生成的内容这时候，页面内容是有js/ajax动态生成的，用普通的请求页面->解析的方法就不管用了，需要借助一个类似firefox、chrome浏览器的js引擎来对页面的js代码做动态解析。

此种情况下，推荐考虑casperJS+phantomjs或slimerJS+phantomjs，当然诸如selenium之类的也可以考虑。

3、如果爬虫是涉及大规模网站爬取，效率、扩展性、可维护性等是必须考虑的因素时候大规模爬虫爬取涉及诸多问题：多线程并发、I/O机制、分布式爬取、消息通讯、判重机制、任务调度等等，这时候语言和所用框架的选取就具有极大意义了。

PHP对多线程、异步支持较差，不建议采用。

NodeJS：对一些垂直网站爬取倒可以，但由于分布式爬取、消息通讯等支持较弱，根据自己情况判断。

Python：强烈建议，对以上问题都有较好支持。

尤其是Scrapy框架值得作为第一选择。

优点诸多：支持xpath；

基于twisted，性能不错；  
有较好的调试工具；  
此种情况下，如果还需要做js动态内容的解析，casperjs就不适合了，只有基于诸如chrome V8引擎之类自己做js引擎。  
至于C、C++虽然性能不错，但不推荐，尤其是考虑到成本等诸多因素；  
对于大部分公司还是建议基于一些开源的框架来做，不要自己发明轮子，做一个简单的爬虫容易，但要做一个完备的爬虫挺难的。

### 三、java 如何实现 获取实时股票数据

一般有三种方式：1. 网页爬虫。  
采用爬虫去爬取目标网页的股票数据，去GitHub或技术论坛（如CSDN、51CTO）上找一下别人写的爬虫集成到项目中。  
2. 请求第三方API。  
会有专门的公司（例如百度API市场）提供股票数据，你只需要去购买他们的服务，使用他们提供的SDK，仿照demo开发实现即可。  
如下图所示：3. 请求第三方WebService接口。  
一般用的较多的网站是聚合数据，如下图所示：

### 四、如何利用爬虫技术来辅助老妈炒股票

炒股赚钱是一个极度复杂的综合体，不可能单凭某种技术就能炒股赚钱的。要是那样，股市就不是1赚1平8亏了。

### 五、python 爬虫 爬什么数据

主要就是爬一些网页内容。  
比如百度、google，就是靠着上万个爬虫服务器去爬取所有静态网页内容，然后缓存在自己的服务器，以便网民搜索。  
再比如，A网站有很多比较不错的图片、文章等信息，B网站自己没能力出原创，就通过爬虫去A把图片、文章爬下来后，直接发布在B网站。

等等等等.....

## 六、如何系统的学习从网上爬取数据，文本，以及分析

说起这个话题，我上半年很重要的一部分工作就是开发了一个大型的分布式爬虫系统和一个新闻热点分析工具。

因此，你的问题我最有发言权，我简单说说我的做法，我是把主要的导航网站找来，把它们关于新闻资讯的网站地址全拿来筛选一遍，选出有代表性的主流媒体网站和地方资讯网站，然后让爬虫系统去爬取4~5层深度的页面数据回来做分析测试数据，为避免无关数据干扰和加快爬取速度，我在爬行时，采用正则表达式和关键词过滤链接地址和正文内容。

数据爬回来后，可用的分析算法很多，分词处理后，我一般用聚类算法、规则关联算法这些。

## 参考文档

[下载：爬虫爬股票哪些信息.pdf](#)

[《股票理财资金追回需要多久》](#)

[《新的股票账户多久可以交易》](#)

[《股票多久能涨起来》](#)

[下载：爬虫爬股票哪些信息.doc](#)

[更多关于《爬虫爬股票哪些信息》的文档...](#)

#!NwL!#

声明：

本文来自网络，不代表

【股识吧】立场，转载请注明出处：

<https://www.gupiaozhishiba.com/article/6208813.html>