

自动程序研究

数学-教师-外汇交易-网站开发-Python-爬虫-深度学习-数据分析-自动化测试-算法-数据挖掘工程师-数据科学家

MONTH: FEBRUARY 2018

FEBRUARY 14, 2018

Python数据采集（爬虫）浅谈

1. WEB请求

第一段就难倒我了，非要讲述Web8种http请求方式，

无异于孔乙己知道茴字的N种写法(关键我也不会啊)。

先说说我学爬虫的过程，搬板凳，带瓜子，讲故事了：

每一个Python初学者大都绕不开爬虫的坑，

那年我玩CPC网赚，按键精灵、国产编程语言，大漠插件，靠谱助手，知道的举个手？

了解了Web前端的标签，网页像素级点击的方法，后台发送数据的技巧.....

吾爱破解，小众软件社区跑多了觉得能破解、写软件看起来挺牛逼的，

偶然进了我乎，发现程序猿大牛真特么厉害，三条汪都还在，炒鸡偶像！

程序员们在撕逼PHP是宇宙内第一语言的时候，也顺便提到了人类未来Python，人生苦短，我用Python，逼格当时就爆棚了，

没错，@ 廖雪峰，为人不识陈近南啊，照着他2.X教程撸3.X还是痛苦的，之后他出了3.X，我就顺手爬了他的教程，做成的带标签的PDF，

现在还是吾爱破解的热帖（捂脸！）

如何爬技术博客专题，并将其作为带标签的PDF，请自行研究！

当然，那个时候@ 静觅 | 崔庆才的系列博客也不能少。【操蛋，扯远了！】

2.WEB请求流程及器具

第一步，IE准备HttpWatch(来自易语言启蒙教程，嘿嘿)；火狐浏览器安装FireBug插件；通杀软件Fiddler，连费德勒都解决不了的，祭出神器Shark。

这个阶段，先了解一个网络请求是Get还是Post，请求协议头长什么样子，Post数据是什么格式、来源，响应都包含哪些，Html标签含义，JS的作用等等。挨个说起来就话长了。对了，你要抓APP，还的自备随身WIFI，设置Fiddler或Shark，无外乎获取canshu。

第二步，由于我坚定学习3.x，而Scrapy只支持2.7，所以我就从底层自己构建，包括后来的多线程，IP池，验证码，都一个个解决。所以我的建议是先不要用框架，等你熟悉了再用不迟。

网络请求不要纠结，用Requests：快速上手 – Requests 2.18.1 文档，别人的文档写的这么好了，我就不做搬运工，以后用实例慢慢解析。

第三步，GET请求成功（200）之后，相应就是Html源码，如何提取？Re正则肯定是万能的神器，但是结合BeautifulSoup你会觉得可以更简单，我个人的编程风格就是混搭风，英文、中文、pinyin混合使用，不拘泥于形式，以最短的路径实现功能，有点像大数据的梯度下降，越快越好。同样Beautiful Soup 4.2.0 文档也已经很完善，稍后用实例解析。

第四步，提取的数据保存，最开始自然是TXT格式，顺便熟悉以下TXT格式的读取，这个在任何时候都可能用到；然后是CSV，别以为import csv你就可以随便存了，里面还有不少坑，尤其是行头，嗯嗯，稍后实例解析。妹子高清图除外，图片是字节集，要用file保存。

题外话：我没有说POST是因为他除了用来登录以外，常常被用作各种论坛群发机，用作营销推广（留言板、站内信、顶帖），当然，账号被封也是家常便饭，所以本主不打算在我乎细说这个。呵呵，感兴趣的朋友可以看看这个：xchaoinfo/fuck-login

如果这个大牛的登陆源码你都系数看懂了，那你就可以自成一脉，专杀各种论坛、博客营销。如果想更深度的发扬这个体系，你还的搞定IP代理，验证码识别、自然语言NLP，路慢慢洗！

胶水语言博大精深，
本主只得一二为新人带路，
老鸟可以无视存在，

FEBRUARY 14, 2018

浅谈一下最近使用八爪鱼采集器遇到的坑（还有对比其他采集软件和爬虫）

首先八爪鱼采集器功能比较多，而且对比主流的采集软件八爪鱼采集器不落下风。我个人比较熟悉的主流采集工具有火车采集器、火车浏览器、八爪鱼、GooSeeker等。而这些软件之中，火车采集器效率极高，爬取速度快，但是规则配置烦琐，有一些高级功能必须付费版，甚至高级付费版才能使用，烦死.....火车浏览器功能强大，但是和采集器不一样，它以模拟网页操作的形式来采集或者操作网页，效率低于火车采集器，和八爪鱼差不多，功能各有千秋，费用就不怎么样了，贵到死。GooSeeker只是简单用了一下，基本功能还可以，规则编制有些烦琐，部分其他采集工具能实现的高级功能它不太好实现，至少我没直接看懂怎么实现。八爪鱼作为可视化型的采集工具，算是最折中和方便的，配置规则速度快，比费劲分析写爬虫代码容易。

首先有一些很麻烦的坑我要说一下为什么其他几种采集方法会很费劲。**AJAX**加载技术大家已经比较清楚了，而且已经运用到了很多网站之中，这种技术对傻瓜采集工具和小白用户来说是致命的，首先你采集不到合适的数据，其次你也无法很好地使用傻瓜软件翻页和下拉到页面底端，麻烦到死。

火车采集器没有网页操作能力，能支持的只是通过**POST**和**GET**等方式获得内容详情页，再进行采集，这就需要使用到抓包工具**Fiddler**来抓包分析，好拼配组合出需要采集的网址。中途过程之酸爽，估计让很多小萌新吐血到死。苦逼的是，一个网站一个样，也没什么太好的通用型教程，小白只能吐血而死。还有就是，如果你只花了几百块钱买了个版本，那么.....**JSON**数据无法解析啊，我就**XXX**它们.....不能解析的话，很多主流新闻网站的评论都废掉了。火车浏览器就不说了，功能很好，就是设置规则繁琐，免费版只能运行**30**分钟，适合有钱和有技术人员的中小企业。

网络爬虫技术**JAVA**的我不懂，我只熟悉**Python**的，基本静态页面爬取和分析那简直是爽歪歪，轻松搞定多线程啪啪啪！但是对**AJAX**最烦人的，最容易遇到的几个问题：

A，下拉到页面底端才会加载数据

B，网址不会改变，点击上下页只会异步加载数据（多用于分页文章或者评论）

C，里面有加密或者需要js代码中获取某些数据之类的

如果偷懒，那就用**Selenium+PhantomJS**，采取执行部分页面**JS**脚本，或者通过**DOM**定位等形式来加载出来需要的数据。如果想要速度，可以通过抓包工具分析，然后程序拼出网址，最终采集的方式。**Python**对**JSON**数据解析还是很给力的，虽然对于数据里带“,”的格式解析容易出问题，我到现在也不是太会处理。需要再研究一下。总之程序是万能的，就看程序猿水平如何了..... _(:3|_/_ (哦，我不是程序猿.....我是产品汪啊.....)

八爪鱼这一点完全是小白化、智能化、傻瓜化的，也许一些公司的领导会觉得这么简单就能采集了，找个实习生就解决了，没技术含量。对此我只能说，这群自以为是领导就不思进取到处忽悠骗钱的**SB**，他们哪里知道这软件的好玩之处。虽然我很想买个企业版，不过公司里有技术人员，又会写爬虫，实在不值得掏这个成本了，所以算了。

所谓简单只是表面的，实际上八爪鱼想用好了需要知道HTML语言结构、XPATH规则、正则表达式等基本知识。以下是我遇到和发现的一些问题，简单分享一下。

我自己最近用云采集也经常遇到一些问题，采集运行一段时间就停止了，一开始也郁闷，八爪鱼云采集怎么这么不给力。但是后来我仔细分析了一下规则之后，我感觉，除了一些特殊情况，比如八爪鱼后台在调整之类的，云采集出问题的常见原因还是以下的：

- 1，小白傻瓜化点点点，这类的不会写准确的XPATH，也不会分析页面源码，所以一旦页面数据格式变化了，死板的点击生成的规则就很容易废掉；
- 2，没有足够的判断语句来跳出循环，有些页面的下一页直接点击会出问题，还有一些的下一页，点击生成规则就会无限循环，必须有判断语句判断是否到了最后一页，比如汽车之家新闻评论采集就会如此。所以最好的方法就是看源代码判断是否可以用傻瓜方式写翻页循环，否则就需要自己XPATH定位元素；
- 3，官方的教程很实用，基本都看完一遍掌握以后，大部分的问题就都能解决了，有一些解决不了的，大多数是不懂HTML结构和XPATH定位原理导致的。蛋疼的帮很多人改了代码问题以后的总结就是：好好学看源码，好好学看源码，好好学看源码.....重要的事情说三遍！

下面说一个简单的实例：

采集需求：对汽车之家新闻列表里所有包含“丰田”关键词的新闻，采集所有下面的评论。

采集分析：需要从新闻列表页-进入新闻文章页-进入评论列表页-采集。形成循环，最终采集一共3300多页的数据。

如果用点点点和循环大法，很快就可以写出来代码，单机执行首先出错，云采集更出错。各种重复采集、重复加载、无限循环。为什么呢？因为页面变化了，虽然变化不多，万幸！首先XPATH定位要精准，需要点击的元素都需要精准定位，确保循环执行不出错。其次翻页循环会遇到无限循环问题，需要看清下一页A标签的class内容变化，从而精准定位。

再有就是加入判断语句，多重嵌套判断语句没有问题，虽然看着有些乱，但是最后判断语句可以帮助流程顺利执行下去。非要说起来的话，这种判断语句比

写python代码简单太多了，根本不叫事。

最后我的规则就变成了一大堆框框合并在一起的“东西”.....但是这样的好处就是，我的云采集完整的执行下来了，而且编写脚本的时间比python快得多，虽然功能有缺失，但是对于临时采集项目来说很方便，我对于小型数据分析一般都用这种方法直接采集数据。python主要用于并行多线程采集和复杂数据清洗。

最后，总结一下八爪鱼里的坑：

云采集不是那么好用的，需要注意规则，尽量别太复杂

云采集是多ip多节点处理的，所以尽量别采集需要登录的网站，那是作死

如果没有ajax加载的页面，尽量勾选优化非ajax页面速度选项

单机测试一下，如果可以的话就不加载图片来采集

cookies有很多网站是有效时间并不长，所以别期待cookies可以一直使用

网速和计算机响应速度很大程度决定了单机采集的速度

云采集会漏数据的，精度要求高的话，尽量规则上做完善，比如加入加载失败的重试或者判断是否出现数据的流程

热门网站比如淘宝、阿里巴巴、企查查、天眼查等，基本上对云采集的ip都屏蔽了，别想那么容易.....

单机采集可以使用网上的vps服务器，便宜的大概1天1块钱，好点的还能自动切换ip，小规模使用应该能解决不少问题

给页面留一些加载时间，或者循环延迟1秒，别让请求速度太快，容易漏数据

八爪鱼数据是没有计算功能的，所以只能采集和判断

学好Xpath，否则你就是从入门到放弃！

FEBRUARY 14, 2018

八爪鱼采集器的优缺点

首先，我本人使用过八爪鱼、火车头采集器、火车头浏览器、集搜客采集器、神箭手采集器、其他一些不太知名的采集器（比如发源地、前嗅数据等等），还有我自己会用Python写基本的爬虫（包括Scrapy框架和简单的分布式，我是产品狗，coding能力很一般），综上所述，大部分主流数据采集工具都尝试

了，也不同程度的感受了。所以我的观点角度来说，不会像小白用户，也不像技术大牛，属于折中的产品汪角度。

下面来说八爪鱼优点：

1，小白用户福音（简单易用、规则好找、可视化界面、容易学习和模仿）

如果我是小白用户，我不太懂Html和Http协议，那么我看完所有上面所说的工具之后，我可能会做出这样的选择：

直接去某数据平台购买数据（比如数多多和发源地等）

直接去淘宝买规则（火车头采集器和八爪鱼居多）

小白用户是啥：

领导让我采集这些数据，臣妾不会啊，臣妾做不到啊.....

老师让我采集数据做论文啊，臣妾做不到啊.....

数据看样子可以买到，但是没人给我掏钱啊，臣妾没钱啊.....

综上所述，一老子没钱，二老子不懂技术，三老子还想要数据。最简单便宜省钱的方法就是用八爪鱼，几乎没有其他可选的。为啥？

火车采集器（499元起）你得用旧版才能免费导出csv出来，excel都没戏，新版你连这个都导不出来。火车浏览器本来就得花钱买（2180元起），否则也没法用，而且学习成本有些高。其他几个采集工具很难或者没有导入规则的方法，网上也没什么卖规则的，神箭手采集虽然可以免费采集一些数据，但是导出也是要收费的，而且数据多了还得单独下载数据导出工具。

八爪鱼怎么省钱？完善信息1000积分，每天签到还30积分，网上下个规则或者买个规则，剩下的靠积分就足够搞定初期的采集了，再不行充值一些积分也能随需随用，导出excel、数据库都可以。

2，直观看到网页变化，不管是测试还是采集的时候都容易避坑

很多人说，火车采集器采集速度快啊，八爪鱼好慢啊，弱爆了。不可否认的是，火车采集器是无需完整显示页面就可以采集数据的，这样的好处就是它介

于爬虫和八爪鱼中间，速度确实快一些。但是不好的地方就是，它必须自己拼网址啊！拼网址啊！拼网址啊！本来八爪鱼点一下就可以搞定的东西，因为火车头看不见，只能自己去浏览器里看页面代码才行。采集的时候因为看不到网页变化，只能看到说采集了xxx条数据，说实话每次测试火车头我都头疼，谁知道采集了一大堆，是否都是正确数据，又看不到页面的具体变化.....

你说一开始写火车头规则的时候测试可以看到，嗯，对，可以看到，但是网站都是傻子么，让你随便采集的网站已经越来越少了，动不动就给你403或者加验证了。条件判断呢？火车头哪有这玩意啊.....八爪鱼虽然慢点，但是测试的时候可以看到页面是如何变化的，出了问题可以追溯和调整，否则慢慢哭去吧。

另外一个优点就是看着数据唰唰的采集上来的时候，我才不会告诉你感觉很爽.....

3, 写规则速度快

先不管小白用户，我们这种有一定经验的用户，写八爪鱼规则，比如采集汽车之家某车型全部文章全部评论的规则，第一次写需要40分钟。也许你会问，这玩意还要这么久，你弱爆了.....我的解释是汽车之家用了ajax加载，而且有的页面需要分析一下“下一页”的跳转链接定位，避免无限循环翻页还有页面丢失的问题。第一次写规则需要一个一个页面查看一下xpath定位，并且单机测试一下。所以实际上大部分时间用在分析页面上，写规则的时间其实也就十几分钟就足够了。如果用火车头或者别的工具，那就头疼了.....网址如何拼配琢磨10分钟，ajax加载的问题，火车头的老版免费版根本解决不了，新版还得买高级的json解析才好弄，即使如此还得抓包自己搞清楚网址如何组合.....除非是老鸟，否则谁敢说第一次写三四十分钟就能搞定这个规则？神箭手就更不用说了，写之前先去把js学一遍，然后开发文档看一遍.....

其他优点不说了，请各位看官自行摸索，说多了就有软文嫌疑了。

下面进入大家喜闻乐见的批斗环节：

1, 莫名其妙的错误，简单粗暴的解决办法

如果我只是从0开始写一个规则，通常问题不大，但是当我修改或者复制规则里的某些部分再添加到另一个规则里的时候，有时候就会出现一些莫名其妙问

题。比如规则逻辑结构显示乱套了、规则执行出错等等，尤其是逻辑判断加入比较多的规则，就容易乱套。

解决办法倒是很简单，全部删了按照新的想法重新做一个规则.....

2，云采集不是号称10倍速，多IP么？怎么感觉还那么慢

这个是很多人的误解，云采集是10个节点跑，但是应该不是什么时候都能到10个节点，也不是10个节点就是10个IP。所以速度并不一定是10倍速，但是我是真心希望可以10个节点，10个IP，最好还可以选择多长时间换IP，这样才能解决很多烦人的采集问题，这些功能多收钱也是正常的。

3，云采集丢数据问题

因为看不见云采集的具体运行过程，也没有追溯方法，所以丢了什么数据，哪些页面没采集到完全不知道。最好能有个云采集的详细每个节点的运行日志允许用户导出查看。

4，自动IP代理啥时候才能有

目前除了云采集（还不确定多长时间IP切换），本机的单机采集只能自己写代理服务器IP和端口进来，于是只能网上先买个自动切换的代理IP再填进来，不方便啊。八爪鱼官网说快要添加好这些功能了，我们拭目以待吧，目前反正切换IP不好弄。

这里说一个邪恶的办法.....网上买一个快速切换IP的VPS主机，然后让八爪鱼在上面跑单机，就可以实现IP自动切换了，记得买IP自动切换型的，PPPOE拨号切换的不行，因为八爪鱼没这个自动拨号的功能。

5，最后的批斗，那就是某些情况下没有神箭手采集器省事

以下神箭手采集器的功能，八爪鱼的产品经理需要想一下八爪鱼现在是否有，如何简化操作。

神箭手可以购买专门人员开发的规则，并且规则还可以更新，确保采集不会出错（而且是一键更新）

神箭手可以采集图片，并且还可以托管到临时存储空间里
神箭手可以一键勾选是否IP代理，采集节点数量还可以随时增减
神箭手可以输出日志，还可以查看详细的日志信息
神箭手可以设置自动采集（每天几点到几点，间隔多长时间，是否检测页面变化，变化的数据是更新还是增量采集等等，一个页面就搞定）
神箭手可以自动导出数据到数据库，后台自动的，不需要打开软件，因为它就是个平台而已
神箭手可以自动发布数据到博客、论坛，同理，因为就是个平台，不用开电脑也可以继续运行

所谓人无完人，机无完机，采集器也没有最理想的。如果让我选，我会八爪鱼优先用，神箭手弥补，Python代码搞定剩下的。火车头？我用Py写代码又不花钱，我用火车头这种差不多的东西有啥用.....

FEBRUARY 13, 2018

对比了 18000 个 Python 项目，这 TOP45 值得学习！

热门资源博客 Mybridge AI 比较了 18000 个关于 Python 的项目，并从中精选出 45 个最具竞争力的项目。我们进行了翻译，在此一并送上。

这份清单中包括了各不相同的 20 个主题，以及一些资深程序员分享使用 Python 的经验，值得收藏。Mybridge AI 的排名结合了内部机器评估的内容质量和各种人为因素，包括阅读次数和阅读时长等。



对于 Python 的初学者，我们推荐以下这些课程：

REST API：使用 Python，Flask，Flask-RESTful 和 Flask-SQLAlchemy 构建

专业的 REST API [12,602 个推荐, 4.6 / 5 星]

链接:

<https://www.udemy.com/rest-api-flask-and-python/>

算法交易: 用于财务分析和算法交易的 Python, 主要学习包括
numpy, pandas, matplotlib, quantopian, finance [8,077 个推荐, 4.6 / 5
星]

链接:

<https://www.udemy.com/python-for-finance-and-trading-algorithms/>

年度开源 Python 项目 [平均 4,078 星]

链接:

<https://medium.mybridge.co/30-amazing-python-projects-for-the-past-year-v-2018-9c310b04cdb3>

机器学习年度最佳文章

链接:

<https://medium.mybridge.co/learn-to-build-a-machine-learning-application-from-top-articles-of-2017-cdd5638453fc>

Python 算法

回顾 Python 交互式编码中所要面临的挑战 (算法和数据结构)

本文对算法编码和数据结构中的问题提出了简单易懂又切实可行的方案。

作者: Donne Martin; [github - 11811 星]

链接:

<https://github.com/donnemartin/interactive-coding-challenges>

Python 中算法和数据结构的最小样本

如何让 Python 中的数据结构和算法最小、最干净？

作者：keon； [github - 10271 星]

链接：

<https://github.com/keon/algorithms>

最重要的 Python 算法—Pygorithm

Pygorithm 是一个纯 Python 风格编写的模块，通过导入所需的算法，获得相应的代码、时间复杂性等。这是一个开始学习 Python 编程的好方法，能够帮助初学者学习并实现 Python 中所有算法。

作者：Satwik Kansal； [github - 3156 星]

链接：

<http://pygorithm.readthedocs.io/en/latest/>

Python 综合指南

一个有趣又鲜为人知的 Python 代码片段集合—wtfPython

作者：Satwikkansal； [github - 4,933 星]

链接：

<https://github.com/satwikkansal/wtfPython>

Python 的脚本结构

一个关于如何从 Python 脚本到打包项目的标准化指南

作者: Courtesy of Vicki

链接:

<http://veekaybee.github.io/2017/09/26/python-packaging>

Python 中的列表

Python 列表生成器的教程

在这份教程中, 你将能够学习到如何在 Python 中有效地使用列表生成器来创建列表, 替换(嵌套) for 循环以及使用 map(), filter(), reduce() 函数等。

文章首先简单回顾了 Python 中列表的基本概念, 并与 Python 中其他的数据结构进行比较。接着讲解了列表生成器的学习。文章还讲解了 Python 列表背后的数学知识, 创建列表生成器的方法, 以及如何在 for 循环或 lambda 隐函数中重写它们。

作者: Karlijn Willems

链接:

<https://www.datacamp.com/community/tutorials/python-list-comprehension>

Python 中的类

如何使用 Dunder (Magic、Special) 方法来丰富你的 Python 类

Dunker 是 Python 中的一种特殊方法, 通过双下划线开始和结束的形式存在, 例如 __init__ 来丰富类的预定义方法。

作者: Dan Bader

链接:

<https://dbader.org/blog/python-dunder-methods>

Python 中的网页抓取

如何使用 Python 中的 Scrapy、SQL 和 Matplotlib 等库进行网页抓取，并获取网页数据分析

你可以通过这篇文章学习到网页爬取知识，并用于实践中。

作者：ScrapingAuthority

链接：

<http://www.scrapingauthority.com/python-scrapy-mysql-and-matplotlib-to-gain-web-data-insights/>

高级的网页抓取教程：绕过“403 禁止”，验证码等问题

作者：Evan Sangaline

链接：

<http://sangaline.com/post/advanced-web-scraping-tutorial/>

掌握 Python 的网页抓取技巧来获取你所需要的数据

作者：Lauren Glass 和 Hackernoon

链接：

<https://hackernoon.com/mastering-python-web-scraping-get-your-data-back-e9a5cc653d88>

Python 中的自动化操作

如何使用 Twilio、Python 和 Google 自动化婚礼的进程

作者: Thomas Curtis

链接:

<https://www.twilio.com/blog/2017/04/wedding-at-scale-how-i-used-twilio-python-and-google-to-automate-my-wedding.html>

如何用 Python 在 Medium 上找到有趣的人

作者: Radu Raicea 和 freeCodeCamp。

链接:

<https://medium.freecodecamp.org/how-i-used-python-to-find-interesting-people-on-medium-be9261b924b0>

Python 中的 Bot

制作 Reddit+Facebook 的信息箱

作者: Yasoob Khalid

链接:

Making a Reddit + Facebook Messenger Bot

我在 Instagram 上用 Python 写的开源机器人 (让我拥有了 2500 个粉丝, 所花的服务器成本只有 5 美元)

作者: TimG

链接:

<https://medium.freecodecamp.org/my-open-source-instagram-bot-got-me-2-500-real-followers-for-5-in-server-costs-e40491358340>

Python 中的电子表格

权威指南：Python 的 Excel 教程

通过这个教程，你可以了解如何使用 Python 读取和导入 Excel 文件，如何将数据写入这些电子表格。

作者：Karlijn Willems

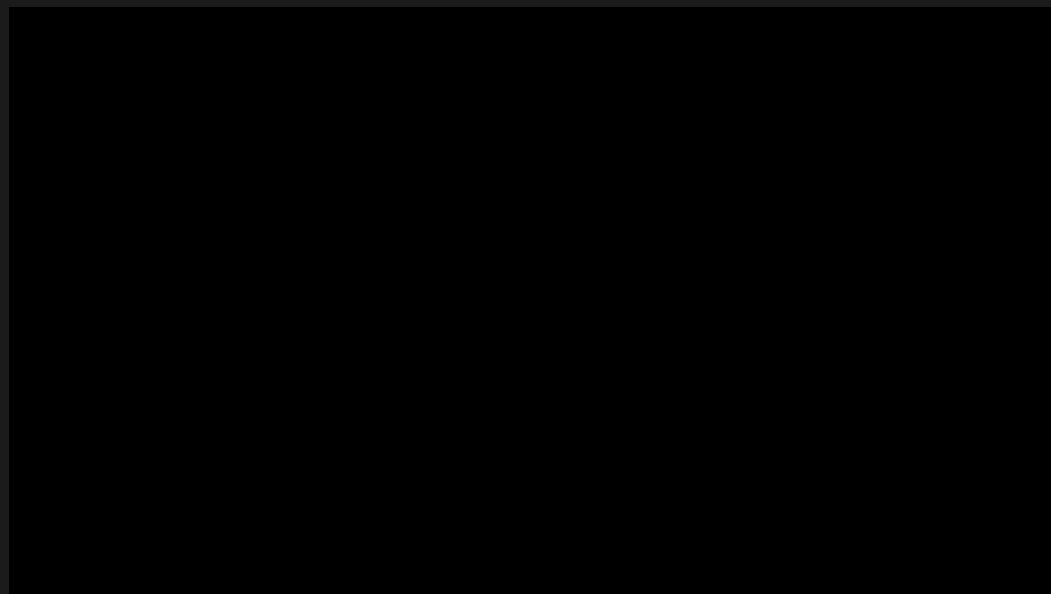
链接：

<https://www.datacamp.com/community/tutorials/python-excel-tutorial>

Python 和 Google 电子表格

作者：Brent Schooley

链接：



Python 中的金融应用

Python 中的金融：算法交易

这是一份 Python 与金融应用的教程，在此你能学习到算法交易的基本知识及相关内容。

作者: Karlijn Willems

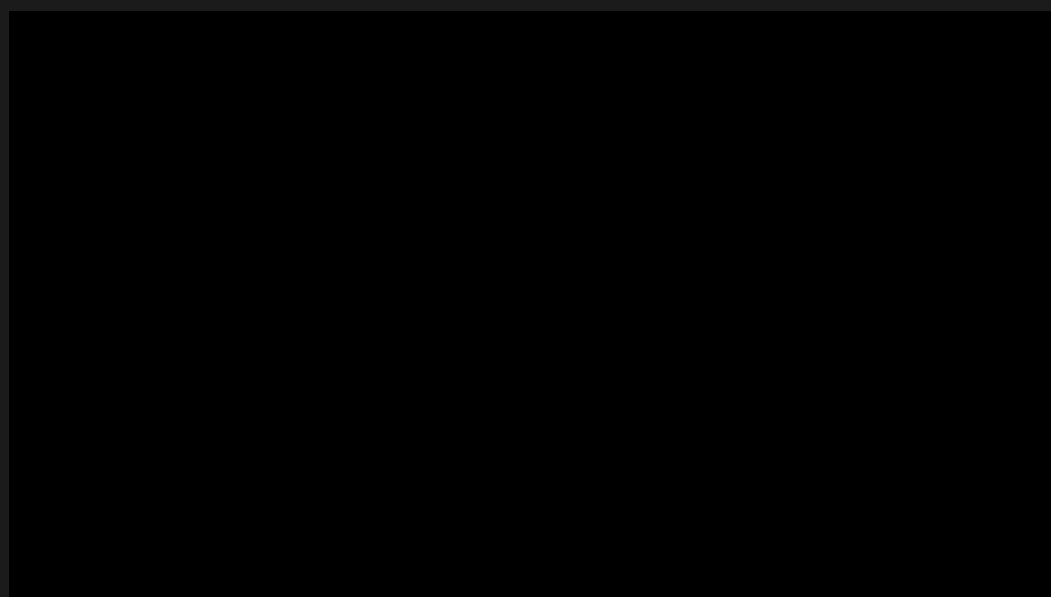
链接:

<https://www.datacamp.com/community/tutorials/finance-python-trading>

Python 中的金融教程—股票价格及相关数据介绍

作者: Harrison Kinsley

链接:



用 Python 分析加密货币市场

比特币市场是如何表现? 加密货币价值突然出现高峰和低谷的原因是什么? 不同 altcoins 市场是不可分割的或基本独立的? 我们如何预测接下来会发生什么?

这篇文章将简单地介绍如何使用 **Python** 来分析加密货币。文章通过一个简单的 **Python** 脚本来检索, 分析和可视化不同加密货币上的数据。在这个过程中, 文章还将揭示这些波动剧烈的市场行为以及一个有趣的演变趋势。

作者: Patrick Triest

链接:

<https://blog.patricktriest.com/analyzing-cryptocurrencies-python/>

Python 中的区块链

学习并构建一个区块链

毫无疑问，区块链这项新颖的技术是计算的奇迹。区块链技术的出现引发了新的全数字货币，如比特币和莱特币，而这些货币并非由中央当局发行或管理。区块链也以 **Ethereum** 这样的技术形式革命化了分布式计算，并引入了智能合约等有趣的概念。

这篇文章将会帮助你学习并理解区块链的工作原理。通过这篇教程，你将学习到一个功能强大的区块链，并掌握它们的工作流程。

作者：Daniel van Flymen 和 Hackernoon

链接：

<https://hackernoon.com/learn-blockchains-by-building-one-117428612f46>

② 如何构建一个最小的区块链

本文将用少于 50 行的代码（Python2）来创建一个最简单、最小的区块链。

作者：Gerald Nash

链接：

<https://medium.com/crypto-currently/lets-build-the-tiniest-blockchain-e70965a248b>

Python 中的视频合成

② 用 Python 构建一个视频合成器

视频合成器是利用音频输入来创建视觉信号的设备，自上世纪 60 年代以来，已

有很长的历史。

这篇文章将用 **Python** 编写一个基本的视频合成器，并使用 **aubio** 进行 Onset 目标检测。

作者：Kirk Kaiser

链接：

<https://www.makeartwithpython.com/blog/video-synthesizer-in-python/>

Python 的性能

②③ 用 **Python** 处理每秒 100 万个请求

用 **Python** 每秒能够达到 100 万个请求吗？为了节省服务器价格，最近很多公司正在从 **Python** 向其他编程语言中迁移。但实际并不需要。

Python 社区最近在性能提升方面做了很多工作。**CPython 3.6** 通过新的字典提高了整体解释器的性能。由于引入了更快的调用约定和字典查找缓存，**CPython 3.7** 将会更快。

对于数字处理任务，你可以使用 **PyPy** 进行代码编译。你还可以运行 **NumPy** 的测试套件，该测试套件现在已经改进了 **Python** 与 **C** 语言扩展的整体兼容性。在随后的更新版本中，**PyPy** 预计将与 **Python 3.5** 兼容。

作者：Paweł Piotr Przeradowski。

链接：

<https://medium.freecodecamp.org/million-requests-per-second-with-python-95c137af319>

②④ “**Python** 很慢，但我不在乎”

这篇文章将介绍一些关于 **Python** 中 **asyncio** 的内容，并讨论有关 **Python** 速度的问题。

作者：Nick Humrich

链接：

<https://hackernoon.com/yes-python-is-slow-and-i-dont-care-13763980b5a1>

②⑤ Python 中的缓存：如何缓存函数的结果

文章将介绍一种快捷的方法来加速 Python 记忆代码。你将看到何时以及如何运用 Python 记忆代码。记忆代码优化你的程序，在某些情况下会加速你的代码运行。

链接：

<https://dbader.org/blog/python-memoization>

Python 中的 Django 框架

②⑥ 七步骤带你完整地学习 Django

Django 是用 Python 编写的一个 Web 框架。这篇文章是介绍 Django 基础知识的系列教程，共分为七个部分，将分别从安装，准备开发环境，模型，视图，模板，URL 到更高级的主题（如迁移，测试和部署）出发，详细探讨所有的基本概念。

作者：Vitor Freitas

链接：

<https://simpleisbetterthancomplex.com/series/2017/09/04/a-complete-beginners-guide-to-django-part-1.html>

②⑦ 使用 Django 构建 REST API 的测试驱动方法：第一部分

这篇文章将介绍如何利用 Django 来构建一个 REST API 的测试驱动，并详细介

绍了每个步骤。

作者：Jee Githinji Gikera 和 Scotch Development

链接：

<https://scotch.io/tutorials/build-a-rest-api-with-django-a-test-driven-approach-part-1>

Python 中的 Flake

⑳ Ol'List 规则

这篇博文中我们将介绍 Flake8 (pyflakes, pycodestyle 和 mccabe) 中的每个规则及相对应的示例。

链接：

<https://lintlyci.github.io/Flake8Rules/>

㉑ 使用 Python 和 Flask 开发 RESTful API

本文包括以下几部分：

为什么用 Python?

什么是 Flask?

引导 Flask 应用程序

用 Flask 创建一个 RESTful 端点

用 Python 类映射模型

用 Marshmallow 进行序列化和反序列化对象

Dockerizing Flask 应用程序

用 Auth0 保护 Python API

作者：Bruno Krebs

链接：

<https://auth0.com/blog/developing-restful-apis-with-python-and-flask/>

Python 中的 Numpy

⑩ 从 Python 到 Numpy

本文通过一种新颖的方式，向量化地集中讲解了如何从 Python 迁移到 Numpy 的学习。另外，本文还包括一些很少提到的使用技巧。

链接：

<http://www.labri.fr/perso/nrougier/from-python-to-numpy/>

⑪ 探索 Python 每种工具包的行长度

本文探索了 Python 的流行包，如 NumPy, SciPy, Pandas, Scikit-Learn, Matplotlib, AstroPy 等。

作者：Jake VanderPlas

链接：

<http://jakevdp.github.io/blog/2017/11/09/exploring-line-lengths-in-python-packages/>

Python 中的 NashPy

⑫ NashPy 教程—建立并找到一种简单的游戏平衡

博弈论是用来研究理性主体之间的战略互动：当双方试图采用对各自最有益的

方式来完成某件事情时，对双方互动行为的研究。这篇文章将采用 Python 中的 NashPy 来研究这种双方博弈的互动行为。

链接：

<http://nashpy.readthedocs.io/en/latest/tutorial/>

Python 中的马尔可夫过程

③ 用 Python 模拟 Chutes 和 Ladders

这篇文章将通过 Chutes 和 Ladders 游戏的例子，建立模型并阐述马尔可夫过程的原理。整个分析过程附有 Python 源码，感兴趣的读者可以尝试一下。

作者：Jake VanderPlas

链接：

<http://jakevdp.github.io/blog/2017/12/18/simulating-chutes-and-ladders/>

Python 中的数据分析

④ 用 Python 分析美国联邦政治行为

科学、政治、个人意见和社会政策的交集可能呈现相当复杂的情况。思想和学科的交汇点通常充斥着有争议的观点和基于信仰但缺乏经验证据的议程。这时，数据科学在这方面就显得特别重要，因为它提供了一种以实际事实为基础的考察世界的方法，能够深入了解我们今天所面临的一些最重要的问题。

这篇文章我们将用 Python 来分析美国联邦政府的一些政治行为，深入了解政治背后所隐藏的故事。

作者：Patrick Triest

链接：

<https://blog.patricktriest.com/police-data-python/>

③⑤ 用 Python 分析 1000+ 的希腊葡萄酒

作者: Florents Tselai

链接:

<https://tselai.com/greek-wines-analysis.html>

③⑥ 如何用 Python 生成 FiveThirtyEight 图

这篇文章将用 Python 的 matplotlib 和 pandas, 来学习并查看 FiveThirtyEight (FTE) 可视化的核心部分, 并教会你使用 Python 来为自己的数据进行可视化。

作者: Josh Devlin

链接:

<https://www.dataquest.io/blog/making-538-plots/>

③⑦ 使用 Apache Spark 和 Python 为 8000 万 Amazon 产品进行评价打分

作者编写了一个简单的 Python 脚本, 将亚马逊产品评论数据集中的每类评分数据进行整合, 并对这些 Amazon 产品评论数据进行分析打分, 以发现用户的喜好。

作者: Max Woolf

链接:

<http://minimaxir.com/2017/01/amazon-spark/>

③⑧ 使用 Python 进行地理空间分析

作者: Matthew Rocklin

链接:

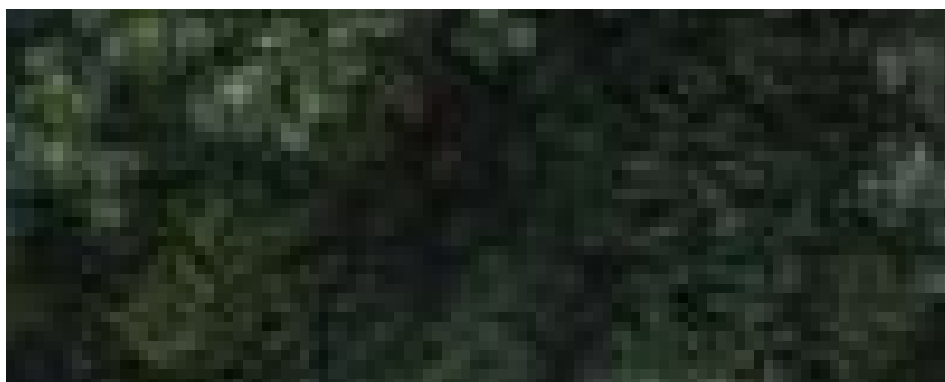
<https://matthewrocklin.com/blog/work/2017/09/21/accelerating-geopandas-1>

③ 星球：从太空中了解亚马逊，来自 Kaggle 头奖获奖者的采访

文章采访了 Kaggle 的“星球：从太空中了解亚马逊”竞赛的获奖者，内容包括他如何使用 11 个微调的卷积神经网络，标签关联的结构模型，以及如何避免过拟合现象等。

作者: Edwin Chen

链接:



Planet: Understanding the Amazon from Space, 1st Place Winner's Interview

In our recent Planet: Understanding the Amazon from Space competition, Planet challenged the Kaggle community to label satellite images from the Amazon basin, in order to better track and understand causes of deforestation. The competition contained over 40,000 training images, each of which could contain multiple labels, generally divided into the following groups: Atmospheric conditions: clear, partly cloudy, cloudy, and haze Common land cover and land use types: rainforest, agriculture, rivers, towns/cities, roads, cultivation, and bare ground Rare land cover ... Continue reading



No Free Hunch

9

④⑩ 从零开始学习 Python

Python 的创建者 Guido van Rossum 曾说过，“Python 是一个高级编程语言，其核心设计理念是让代码具有高度的可读性和简单的语法，程序员可以用几行代码表达自己的想法。”

作者：TK

链接：

<https://medium.freecodecamp.org/learning-python-from-zero-to-hero-120ea540b567>

④⑪ 重要的 Python 练习清单

这篇文章列出了一些重要的练习项目，包括 Python 语言本身和标准库的练习。文章中还有 Python 中不同主题模块的知识。

作者：Ynon Perek

链接：

<https://www.ynonperek.com/2017/09/21/python-exercises/>

④⑫ API 的设计原则：先思考再编码

API 是定义应用程序接口的通用术语，换句话说，就是用户（人或机器）与程序的交互接口。在 Web 开发世界中，API 通常是一个网站，其中包含一系列端点，用于响应客户端请求和结构化文本数据。这篇文章将告诉你为什么以及如何设计一个正确的 API，如何将自己的思想植入到 API 的设计中来构建属于你自己的 API。

作者：Jonatas Baldin

链接：

<https://www.ckl.io/blog/api-design-think-first-code-later>

④③ Python 机器学习指南

本文将通过清晰地解释和有效的练习，来帮助你深度理解相关的机器学习算法。

作者：Conor Dewey

链接：

<https://medium.freecodecamp.org/the-hitchhikers-guide-to-machine-learning-algorithms-in-python-bfad66adb378>

④④ 如何学习 Python 编程：6 位经验丰富的 Python 开发者分享了他们的观点

对于当下热门的 Python 语言，有太多的教程、书籍、视频和博客文章资源，然而如此多的冗余资料，你该如何选择最佳的方式开始你的 Python 学习之旅呢？这篇文章列出了 6 位 Python 专家分享的学习经验，相信这对于迷茫中的你来说，将受益匪浅。

链接：

Best way to Learn Python Programming- 6 Experts Share their Amazing Learning Tips.

④⑤ 如何用 Python 实现强大的数据分析

Python 是数据分析的最佳编程语言，这得益于它自带的依赖库。依赖库能够存储，操纵数据，并从数据中获得核心信息，因而在数据科学领域被广泛使用并展现出强大的功能。本文介绍了 Python 在数据科学领域的应用历史以及最新发展。

作者: Jake VanderPlas

链接:

<https://www.youtube.com/watch?v=9by46AAqz70>

<https://medium.mybridge.co/python-top-45-tutorials-for-the-past-year-v-2018-1b4d46c9e857>

Github 链接:

<https://github.com/Mybridge/learn-python>

FEBRUARY 13, 2018

如何用ZennoPoster工具来做自动化百度SEO的,ZennoPoster网页自动化工具能做什么?

从最开始接触SEO,到不断的建站、推广SEO,也已经有5年时间了。5年的时间说长不长,说短也不短。如果按鲁迅说的只要在一个领域积累10年的时间,定能才为这方面的专家,那我也应该算半个“砖家”了。最开始做SEO,圈子里的人都说“外链为王、内容为皇”。那时候就是拼了命的做外链,当初也取得了一定效果,当然最重要的是积累一些经验。

我们的SEO工作,很多都是重复性的工作,比如说分析网站排名、分析关键字、还有发外链。作为程序员出身的SEOer,我一直都在思考,怎么才能将这些工作实现自动化,将重复性的工作用工具来代替,而将更多的时间和精力用在SEO的策略和思路上。

2012年底的时候,ZennoPoster出现在我的眼前。由于自己也在做英文SEO,所以对国外的SEO软件也比较关注。当时是看到ZennoPoster实现自动注册Twitter和Facebook的一个视频,惊叹这款软件的高度定制化,他应该可以把我的一些重复性的工作实现自动化。

后面才了解到,ZennoPoster出自于俄罗斯人之手,俄罗斯在黑客领域的地位不容小觑,在SEO方面自然也是引领技术潮流。这个软件也有点年头了,不过因为直到出了英文版,才会为大家所知。而今年4月份,ZennoPoster推出了中文

版，更加适合国人使用了。

中文版出来后，我开始用**ZennoPoster**做我的国内大型博客的分享脚本。思路就是能把我在博客上写的文章同步到我注册的各大博客站上，比如点点、新浪、163、百度空间等。这样来提高我的文章的曝光度。在后期我还想做一个投稿同步，只要写了的文章，能自动投稿到指定平台上。这样就能省去我不少的时间，要知道，如果要同时同步到几十个博客平台上，我要登录、复制、粘贴多少次啊？而自动化的实现，我只要轻点鼠标就行了。

现在这个脚本已经做出来了，暂时做了10个比较大型的博客平台，同步成功率也在90%以上，大家可以看下我做的演示视频，就知道**ZennoPoster**是有多么强大了！

根据官方（zennolab.cn）的几个实例脚本，**ZennoPoster**是很容易入门的。最突出的特征就是这个软件是完全图形化的，很容易上手。制作脚本的时候基本上就是录制型的，他会自动把你在浏览器中的操作记录到脚本编辑器中。

现在我已经离不开**ZennoPoster**了，通过它我实现了多个百度SEO相关的脚本，例如：百度排名自动跟踪、国内大型博客（点点、新浪、163、百度空间等）文章同步、百度关键字扩展、百度问答自动化等等。可以说，只要能想得到的，**ZennoPoster**都可以帮我实现！

=====

ZennoPoster网页自动化工具能做什么？

ZennoPoster到底能做什么？

一，中文版出来后，我开始用**ZennoPoster**做我的国内大型博客的分享脚本。思路就是能把我在博客上写的文章同步到我注册的各大博客站上，比如点点、新浪、163、百度空间等。这样来提高我的文章的曝光度。在后期我还想做一个投稿同步，只要写了的文章，能自动投稿到指定平台上。这样就能省去我不少的时间，要知道，如果要同时同步到几十个博客平台上，我要登录、复制、粘贴多少次啊？而自动化的实现，我只要轻点鼠标就行了。

现在这个脚本已经做出来了，暂时做了10个比较大型的博客平台，同步成功率也在90%以上，大家可以看下我做的演示视频，就知道**ZennoPoster**是有多么强大了！

根据官方的几个实例脚本，**ZennoPoster**是很容易入门的。最突出的特征就是这

个软件是完全图形化的，很容易上手。制作脚本的时候基本上就是录制型的，他会自动把你在浏览器中的操作记录到脚本编辑器中。

现在我已经离不开ZennoPoster了，通过它我实现了多个百度SEO相关的脚本，例如：百度排名自动跟踪、国内大型博客（点点、新浪、163、百度空间等）文章同步、百度关键字扩展、百度问答自动化等等。可以说，只要能想得得到的，**ZennoPoster**都可以帮我实现！

二，打造的自动发布squidoo lens的工具

自动注册邮箱

自动注册squidoo账号，自动邮件确认

自动根据关键词采集ezinearticles文章，打乱，重排，通过the best spinner api改写。

自动增加squidoo文本，视频，链接等模块，自动发布lens，如果通不过，自动删除

根据设定的列表随机选择不同的利基项目，对应每一个项目的文件夹，随机挑选需要插入到文本模块的链接

不同squidoo账号随机互踩，互评，提高分数和排名

自动提取已经发布成功的lens链接和标题，以各种形式保存，用于其他网站的推广

一个晚上可以自动发布几十个lens，而且是通过的，当然后期有些会被locked，这些就不管他了

还有一些功能就不一一列举了，呵呵

三，有了zennoposter，你可以制作很多自动化机器，比如自动注册，验证，采集，发布，自动评论，自动关注，提取用户，甚至可以打成一个自动邮件群发系统。非常强大，只是价格也不便宜，我买了两个专业版，有两台电脑从

不关机，自动帮我完成网络推广的任务。等于雇佣个机器人帮我干活，而且它还不知疲倦，从不罢工。

这个工具真的为我节省了很多时间和劳动力，等于帮我雇了几个网络小工。我睡觉的时候，它也照样运行，帮我完成指定的任务。比如，我用zennoposter打造了一个自动邮件群发机，用它的强大功能自动模拟网页登陆邮箱发送邮件，而且用随机变量插入邮件，让每一封邮件都不相同，发送的效果非常不错，我用它来卖自己的产品。不用管推广，深更半夜都会有订单，发送到手机邮箱。

基本每天都有订单，这都是zennoposter为我带来的，我每天的工作是看邮件，联系客户发货就行了。

四，我所在的部门，每天都需要给其它部门呈交各种数据报表，其中大家最常见的就是，网页当天排名、收录、Google bot最后来访时间、域名年龄、域名权重等，在我没有来到部门之前，数据的收集工作是每天来回打开几个可供查询的工具网站（类似站长工具），把关键词输入进去，填写验证码，提交查询，周而复始。每天需要查询的数据在600行左右，负责这个录入工作的是一位MM，由于该工作重复性较高，加上长时间录入，容量产生视觉疲劳，所以某些数据可能会有些出入。

于是我弄了一份数据样本来看了一下，需要查询的数据类型不多，其它类型分散在三个网站上，1、网页权重，2、google bot最后来访时间，3、网页的排名和收录。简单的在这几个工具查询网站看了一下，第一个工具站需要做一道算术题才能提交查询，但是不幸的是算术题做了处理，转化成图片显示的。第二个工具在登录状态下，查询无限制，第三个，无任何限制。对于熟悉ZP的人来说，做一个登录脚本，实在是太easy了，唯一棘手的是需要做算术题的那个工具，题目的范围是2位数和2位数的运算，其实可以直接通过捕抓验证码图片，做一个弹出，人工来输入就可以了，这已经可以极大程度的减少了人工介入的成本了，或者可以通过第三方打码服务商，实现无人值守。但是，好像支持算术题的打码服务商并不多，而且单次打码价格相比其它的类型的验证码单价要高。不甘心，在网上找了一下，居然找到一款OCR程序，通过把验证码图片保存到本地，通过外部程序调用OCR来识别，发现成功率非常高，测试了十几次，大概有80%的成功率，不过这个没关系，添加一个验证码重试的流程就可以了，最后只需要把OCR识别的结果存储到指定的文件，用ZenoPoster去读取就可以完成全自动无人值守了。如果网站源码和数据不出问题的话，是无需人工介绍的。

自动化赚钱真的存在吗？

人的天性都是懒惰的，都希望以最少的努力获得最大的收获。做网络营销也是如此，做网络推广本身也是很繁琐和苦逼的事，很多事情都是需要重复去做的，有时候感觉自己就是个网络民工。

但也别泄气，这样想的不是你一个人，也自然有人填补这个需求。网络上一直都不缺自动化的软件，但真正好的软件不多。有以前用过**Magic Submitter**，按月计费，69美金一个月，但效果一般，但有一个软件我真的是膜拜和推崇，那就是俄罗斯人做的**Zennoposter**，真TM一个帅，基本上能模拟网页上的任何操作，自动填写表单那是基本的，键盘动作那也是小菜一碟，包括拖拉一个网页元素都可以，自动进入邮箱点击确认链接也没问题。

所以，有了**zennoposter**，你可以制作很多自动化机器，比如自动注册，验证，采集，发布，自动评论，自动关注，提取用户，甚至可以打造成自动邮件群发系统。非常强大，只是价格也不便宜，我买了两个专业版，有两台电脑从不关机，自动帮我完成网络推广的任务。等于雇佣个机器人帮我干活，而且它还不知疲倦，从不罢工。

这个工具真的为我节省了很多时间和劳动力，等于帮我雇了几个网络小工。我睡觉的时候，它也照样运行，帮我完成指定的任务。比如，我用**zennoposter**打造了一个自动邮件群发机，用它的强大功能自动模拟网页登陆邮箱发送邮件，而且用随机变量插入邮件，让每一封邮件都不相同，发送的效果非常不错，我用它来卖自己的产品。不用管推广，深更半夜都会有订单，发送到手机邮箱。

大家可以看看我发表在我另外一个博客的截图，基本每天都有订单，这都是**zennoposter**为我带来的，我每天的工作是看邮件，联系客户发货就行了。

这不就是自动化赚钱吗？当然绝对的自动化赚钱是不存在的，首先你要懂得利用自动化工具，你要投资于工具和学习，还有打造你为你赚钱的机器人。如果你自己打造机器人有困难，你可以买。

FEBRUARY 11, 2018

站群新技术

站群新技术？

泛站群：不推荐，一试便知。使用无限二级域名，并自动生成模板，自动随机汉字组合生成文章方式去生成。虽然有所改进，但效果依然是竹篮打水一场空，但市场上这类软件比比皆是，纯是忽悠新手的，这也是黑豹站群极力不推荐的方式，毫不夸张的说，只要是个初级的搜索引擎程序员只需一行代码即可判断出来，并k之，不多解释。行或不行，一试便知。

二级目录站群：不推荐。需要在大量一级域名网站下建立大量二级目录站，使用不同模板，轮链交叉。原理与无限二级域名无差别。一行代码秒杀了。有的用户说不行，也有很多被收录还有排名啊？这个只是你看到留存下来的，而没有看到总数量与留存比率。做100个一级域名有10个域名上去就不错了。而且无法积累，不能稳定维持，可能今天上去，明天就下了，浪费大量域名及服务器。

火星文站群：将各种汉字组合完全无法看懂的汉字组合。所以通俗的讲类似火星文一样。完全的不行，忽悠新手的方法。

正常的**cms**站群：这是现在最有效的站群方法。建立正常的一级域名网站，配合软件维护更新网站，1个月内会有明显效果，而且由于完全模拟正规网站，搜索引擎无法查觉，站群的流量和排名会逐步增加，稳定性根据现有黑豹会员站群统计，保守的讲这种站群网站可以稳定维持3至5年以上。越到后期流量越大。所以站群要养的。很多人都是不明白这个道理，浮躁只能让你东一榔头西一锤子的，最后什么事也干不成。

事实上以黑豹站群软件开发至今，已有5年了。依然是最有效的站群软件。这与当时建立黑豹站群软件的核心思想分不开。

黑豹软件在5年前当时的核心思想是“完全隐藏在正规网站中”“以完全模拟正常的网站为思想核心”去建立软件算法和程序。

现在想想这是已超过那时代的思想。

FEBRUARY 8, 2018

有人做垃圾站群，年入百万，这些站是怎么获得大量流量的呢？

曾经做过一个网站，通过一些非常规推广手段，一周就达到了日 IP 几十万，虽然没达到年入百万，通过 Google Adsense 日入几百美刀。先占坑等，有电脑了详细写.....

转眼半年过去了，我回来填个坑。

这绝对是我的一部黑历史，那是 2013 年的事，时隔这么久，不匿名了。

那年发生了一件事想必大家都知道，如家汉庭等酒店共 2000 万条开房信息泄露。

俗话说的好：哪里有热点，哪里就有流量。要么创造热点，要么跟随热点。

我没有能力创造热点，那只能跟着热点了。但是热点的时间窗口很短，于是我第一时间做了一个网页：中间一个搜索框，右边一个搜索按钮。如果不知道什么样子，可以脑补一下 Google。唯一不同的是，我的页面上下左右都是广告，而且 ~ 不能搜索！不能搜索！——因为我没有下载到（买到）这 2000W 条数据。蜜汁尴尬 ~ ~ ~

没有就没有吧。如果我不说，谁知道我没有呢。

于是我花了 10 分钟做了一个奇丑无比的网页，这个页面看起来大概是这样的：中间一个输入框被四周广告环绕，还有一个不能用的搜索按钮。

但是用户搜索时怎么办呢？干脆把搜索按钮去掉，当用户找不到搜索时，一般会有 2 种动作：一是按回车键，二是乱点页面的其他地方，但是我的页面其他地方除了广告，还是广告。

OK，把这个页面扩展名改成 **PHP**，给人一种动态网页的感觉，再把搜索关键词堆砌到页面里，开房查询系统**1.0**版本上线。

当用户访问到我的页面时，广告点击率肯定不低。

下一步的重点就是怎么让用户来我网站，说通俗点就是流量在哪里。

答案就在**2.0**版里。

（以下所有事都不是我干的，如有雷同，算你抄袭我）

1、获取**2000**万泄露信息。

2、推广。

既然网上找不到这**2000**万开房信息，那就得换个思路了。现在，肯定有人在买卖这些信息，而且肯定也有人在下载这些信息，那么在**DHT**网络中肯定有这些文件。所以，我可以直接从**DHT**网络上探测这个文件。

解决了数据问题，剩下的就是最难的了：流量。

那就玩心理学吧～

大部分人的使用习惯是，至少搜索**2**次，第一次搜自己，第二次搜对方。就算是单身狗，一般也会搜搜自己的女神。

于是我增加了注册功能。

想搜索吗？想。

不好意思，先注册。

注册时推荐他使用**QQ**邮箱。为什么？后面解释。

好，注册完了，搜索。吡，搜出结果了。再搜，系统提示，“积分不足”。

对，这就是套路，注册送积分，但是只够搜一次。怎么赚积分呢？分享。

但是分享完成后还要判断分享是否成功，需要接入第三方平台，开发难度是小事，但是接入周期太长，不能忍啊。

但是转念想，你在分享窗口弹出后点击过取消吗？

所以，我把判断逻辑写在了JS里，只要用户点击了分享按钮，就等待10秒钟发Ajax请求，给用户加积分。

2.0上线后流量依然不高，日pv才几千。

继续开发套路3.0——不不不，说错了——是查开房3.0。

在三一五这天更新，我的手都在抖~抖~抖~ ~ ~（以下主角换成我有一个朋友，机智.jpg）

我的这位朋友开发完2.0后，一刻不歇的着手开发3.0，从此打开了月入几万的大门。

如果2.0是套路版，那么3.0就是暗黑版。

依赖用户主动分享和诱导分享，效果还是太差。我的朋友突然冒出个想法：我来帮用户分享，而且我们做好事是从来不留名的，不用谢，我的朋友叫雷锋。

可是我们没有接入第三方平台，没有用户的openids，怎么分享呢。

一个歪主意就是：社工+钓鱼+撞库。

先说钓鱼，其实也算不上是钓鱼，因为这个网站是一个真实的网站。你也可以把“站”去了——这个网站是一个真实的网。我们不钓鱼，我们网鱼。

我们每个人都注册过几十个网站，但是我们一般使用的密码却只有三四个。根据爱因斯坦•门捷列夫的抽屉原理，你注册我朋友这个网站所使用的密码很可能是你的QQ密码。

读到这是不是先把QQ密码改了再继续读。

十五分钟过去了，你的密码改完了，我们继续。

哈哈，开个玩笑。其实QQ对于安全已经做的很好了。只要我们在其他设备上登录，或者异地登录都会有验证，要么需要手机扫码，要么需要发送短信。所以，即使我朋友知道了你的QQ号和密码也无法登录。

但是别忘了，腾讯家的QQ密码可以登录腾讯加的一切，而我朋友需要的只是：QQ空间、QQ分享、说说。

如果你注册网站时的密码是123456，可以断定这肯定不是QQ密码。所以需要写个代码根据密码复杂度进行筛选。

事实证明，分享到班级群、兴趣群的效果最佳，因为：这网站根本不是广告，而是一个很有用的工具。只要分享语写的好，你的朋友根本里不知道是别人在登录你的帐号。

没有人会反感这种垃圾网站的，就像那些页面只有广告没有下载链接的下载站一样，让用户觉得你的网站是有用的。

随着各大网站争相报道这次如家汉庭信息泄露事件，我朋友的网站一天之内平均要宕机三四次。流量堪比DDOS。

不到一周就冲到了百度首页，三个月之后热度就已经不在了。

如今网站已关，域名也不再续费。

=====

做垃圾站群年收入百万这个事...这得看什么样的站被界定为垃圾站了。

做信息聚合站的算是垃圾站么？做快播视频站的算是垃圾站么？做音乐、Iphone壁纸、旅游信息等等主题的站群算是垃圾站么？所以很难具体的回答楼主的问题。

获取流量：SEO咯，大量的域名，也就意味着有更多的展示机会，更多的流量。他们会钻搜索引擎的一些空子，通过在服务端判定什么样的流量是机器

（搜索引擎spider），哪些是真实的用户。再将真实用户流量统一导向某个站点统一提高该主站的广告展现机会与点击量。

主要盈利方式都是靠广告，抓到足够精准的流量，能够产生广告点击也是很靠谱的一件事。Adsense 英文方面有关医药、健康类的广告、信贷、金融、保险类的广告EPC(Earn per click) 单价都蛮高的，没记错的话有单个点击超过 15美金。

当然还有些做 CPA 的也闷头垄断不少财富，视频交友类网站一个注册就有1-3元不等（国内），如果有站群导量再配合插件（& ¥ # ¥）等等技术手段还是能赚不少钱的。

说两个我身边的个人站长实例，且不曝光他们所作网站的领域了，只能说他们做了一些信息的聚合，同为英文项目，两人每个月的收入都在 8-15万美金之间。

对你那两个做英文站的朋友挺感兴趣的，有时间的话，在不公开他们核心内容的情况，多聊聊他们做的英文站吧。谢了先！他们做的是哪些广告？是自己销售产品，还是做的类似adsense这样的广告？谢谢

回答：adsense

=====

看了上文的回答，感觉一知半解的占居多，没有说到核心的观点。来，我们就来聊聊何谓站群。而站群的利润到底有多少？

有的朋友说年入百万不少，其实说句实话，年入几百万的人都不少。但是我接触过的，包括大部分的做站群的朋友，基本上利润空间一年在50-80万左右。

好的，那么我来详细讲讲，站群，和一个独立的网站到底有什么不同。很多人第一次接触互联网的时候，最早想到的东西，就是“我要做一个网站。”诚然，做一个网站非常简单，找到一个合适的服务器或者空间，找到一个域名，找到比较好的内容，就可以开始运作来。但是，当网站真正运作起来的时候，才会发

现百度通过搜索引擎过来的流量少之又少。无论说图片加ALT属性，还是优化TITLE，还是不停地放外链。（这也催生来一大批专门卖外链的行业）。好像网站的整体收入始终上不来，那么，我来解析一下，我们做站群需要具备哪些条件？

-不少于30W的现金储备。

-有懂得相关技术的技术人才。

-能给自己的站群找到合适的定位，去挖掘内容。

-懂得一点基础的SEO知识，实际上既然成了“群”，这相关的知识也就不是非常重要了。

-不少于30W的现金储备。

1，站群服务器的购买，租赁。因为一般来说，所谓的虚拟空间VPS的服务器只有一条IP(也就是说只能挂一个域名)，而既然要成群，所配置的服务器的数量绝对不能少。一般来说，购买这类服务器的渠道有几个，一个是淘宝等平台，另一个说站长之家这种二道贩子，还有就是通过谷歌搜索，直接购买相应的服务器，当然，我也有朋友自己买硬件在工作室弄的。反正一套服务器的费用，大概要5W-10W左右。为什么呢，因为站群很容易招到攻击（现在好多了），所以需要搭配一些镜像服务器实时备份。以上就是花钱，一般的技术员都能搞定。2，如果要成群，就要准备无数个域名。如果是国内的站群的话，还得准备备案过的域名。（百度权重会略高）。如果不是国内的站群的话，英文域名也要批量购买几千个。这一套费用，差不多就在3W-10W左右，费用之高，令人难以想象。3、最后的现金，就是看你的站群具体想做的收入是哪一块来，如果你是要卖产品，那就可以理解为货费。如果说要进联盟，赚点击费，那就可以理解为赚广告费。反正，流量上来来，变现的途径多种多样。

-有懂得相关技术的技术人才。

懂得相关技术的技术人才暂且不提，从技术层面讲，这个东西根本不难。一般来说，2线城市8K元配置2个程序员，就可以完成绝大部分的工作。

-能给自己的站群找到合适的定位，去挖掘内容。

这也是做站群的最核心的东西来。因为百度搜索的算法，他比较算法，既不是人，也不是人工智能。当一个热门的新闻或者事件在相关事件爆发的时候，百度首先判断的是有多少站群收录了。首先，我来举个例子吧，比如今天我们搜一个新闻或事件吧。比如《斗鱼主播回应脱衣门》。你们会发现，当点开第一页，或者第二页中的个别网站，会有一个不大不小的新闻网站。

等你点开的时候，马上页面就弹到相应的菠菜网站，也就是传说中的赌博网站。当然，这个新闻的标题不是很热，所以很多小站群基本上都在第一二页。

那我再搜一个《犹太人到底有多优秀》。这种类型的标题，都是很多常见站群喜闻乐见的内容。随便打开第一页，就可以看到网站。

你看右边就是接入来京东推广的东西，京东推广可以识别到访问者之前做京东所搜索的内容，然后推送产品。很显然，就是赚广告费了。你去搜索这个网站的域名，通过友链。可以很快反查出相关联的友链网站（我们不去臆测是不是一家的），反正大体来说，一般的站群网站，就是看上去神神叨叨的内容一大堆，里面各种联盟的小补丁广告各种各样。在这里，我要提一句，做站群，很多人的思维是，比如一个新闻，做无数个的新闻网站来进行增加权重，实际上现在这样的方法行不通来。现在一般的操作思路是，套着羊头卖狗肉，比如做一些日本旅游，什么音乐爱好者，天气查询之类的网站。做这些网站的二级页面，依然隐藏着许多实时新闻，咨询类的内容。有人说这么多网站的内容怎么维护，现在的站群，很多内容都是根据关键词来提取采集图片文字到后台。比如站群中的其中一个站说做巴厘岛旅游的，可能里面的所有旅游资讯，采集机器就会自己去百度搜索相关的文章扔到（主页面），然后里面子页面，什么“八卦新闻”，“最新资讯”，就是大家懂得的千篇一律的内容。《犹太人为什么很聪明》《二战的时候最先叩响扳机的是谁》《比基尼的诞生令人震惊了》等等，这种吸引眼球，搜索度高，又可以长期霸占百度排名的关键词。因为这些词没人会去投付费广告，所以很显然，就是利用百度爬虫的漏洞进行，要说这漏洞，永远补不上的。大家熟知的新浪，搜狐的网页平台，是不是也经常点到一些不相关的页面呢，而且域名也不是这些平台的。是的，没错，都是做通过搜索引擎来抢流量！

第二天种站群呢，说卖产品，也是我之前做过，和我朋友搞过的方式。主要卖的说两性壮阳类茶具烟斗电子烟之类的产品。你随便搜索一些男性相关知识，就会看到这类的网站。

。呵呵，不过现在很多做站群的，都把流量引入到公众号了，包括我自己，也自己做做公众号“号群”。还有一种卖产品，说卖外贸名牌假货，一般来说欧洲的服务器可以随便你挂各种假货的网页，然后呢，老外搜索比如GUCCI之类的，除了官网以下的，很大一部分都是做假货的网页，这个不太了解，不知道近几年怎么样。反正，无论做站群也好，公众号号群也罢。统一的就是相信一个思路，那就是：大部分的网民素质，还不足以以自己的评判眼光去挑选文章，越是吸引人的标题，他的转发率（阅读量），他就阅高。要想想，知乎的用户毕竟在中国网民里面是极小众，而且说句实话，知乎的问答很多人也喜欢看标题

党的东西。比如《有一个很漂亮的妹妹说什么体验》做互联网流量就是钱，这个道理显而易见。

总结：一般来说，投资这个比例，如果找到比较细分领域的内容，去多方面做站群的话，一般来说，年收入过百万不是很难的问题。我现在还有一个朋友在做，卖壮阳燕麦，一年也有很稳定六七十万。其实退一步讲，中国几亿网民，做互联网上的流量就是一个巨大的黑洞，只要找到合适的切入点都不会很难赚到钱的。比如我现在自己做的公众号，因为我有引流的技术，我今年就计划做100个左右。呵呵，虽然都是很挫的标题党公众号，可是抛开这一切面纱，还是赚钱最重要啊！

=====

搞站群运营8年了，有些经验与大家分享：

一、做站群，现在靠做google广告，百度广告，淘宝客等联盟广告根本赚不到钱了，因为那是靠大量的流量换点击费用，少的可怜

二、现在做站群，想要赚钱，必须自己有产品，借助于大量网站的流量来推广，可以省下很多推广费，以小博大

三、做站群，选择域名、服务器以及有效的站群软件是关键，现在做站群只能选择黑豹站群，克隆站群这样的专业站群软件来操作，之所以专业是因为他们的开发者是某家大型搜索引擎公司的研发人员，而且使用也简单，算法持续更新。域名最好选择新的com域名，服务器最好选多ip的美国服务器，这样既节省成本又不用备案域名，非常方便

四、做站群不会选产品怎么办？

做站群如果自己没有产品或者不会选择产品，就直接上百度找那些竞价推广的，直接复制广告页面，改个联系方式，上货就卖了。

=====

站群的暴利时代已经过去了，就像seo行业也慢慢没落了一样。

曾经我也有过数百个小语种网站组成的站群，现在只剩下十来个权威站了。

只有那些能够带来持续的行业内容价值输出的网站，才能有长久的生命力，靠迷惑搜索引擎取得好的排名的网站，终究会被历史淘汰！

目前唯一还有效的方法就是高端的mashup，通过针对性的采集大量零散信息，然后重新组装形成具有新价值的页面，这种页面对于访客是真正具有浏览价值的，那么对于SE而言，也是可以接受的。

比如，采集或通过其他渠道获得某种行业的详细数据库，通过自组织或进一步延伸形成更丰富的内容，然后再通过网站展示出来，那么这个网站就会具备较长久的生命力。

=====

有钱技术差点的可以用钱去堆原创结合PBN，有人6个月时间做到单站1w刀。没钱技术牛逼的可以做全自动采集结合人工智能编辑采集的内容让google收录，抓取近期趋势关键词然后自动建立相关博客，流量是非常可观的，但是技术要求很高。有钱又有技术的，您随意。

FEBRUARY 5, 2018

对国内首款伪原创工具作者泊君的访谈

提及SEO伪原创工具，想必大家都非常的熟悉！但是当谈及到SEO伪原创工具的鼻祖，估计很多人都非常的陌生，因为他在SEO行业非常的低调，很少在SEO界露面及参加各种SEO峰会；他的学生都是SEO行业中的精英；他的课程每年仅有一次；他精通搜索引擎内部算法；他通晓天文地理，精通奇门八卦。并将易经成功应用于SEO网站诊断；他就是国内著名职业SEO顾问——中国泊君。

今晚我们有幸邀请到泊君为我们做一次简短的访谈，来谈谈当时泊君开发天天伪原创工具的经历。

小编：想必很多站长都知道您是SEO行业中顶尖的高手之一，您是否能简单的介绍下自己呢？或者说下自己的发展历程呢？

泊君：高手谈不上，我只是一个SEO从业者，然后是程序员。闻道有先后，术业有专攻，可能是熟能生巧吧！

小编：都说您精通奇门八卦，这看来是真的！我们知道，在SEO行业中，伪原创工具是由您所开发的，说您是伪原创工具鼻祖也不为过；那当时您是处于一个什么样的情况下去开发这款软件呢？

泊君：嗯，我曾经在一线做SEO；当时手里管理很多中小的企业网站，文章更新都靠手动，当时想如果有一款工具可以一键替换修改文章 同时让搜索引擎所喜欢，那就再好不过了！然后就开发了国内第一款伪原创工具：天天SEO伪原创工具！

小编：是的，当时我记得伪原创工具风靡一时！在站长圈子里掀起了一股龙卷风，接着就是其他类型的伪原创工具横行互联网，后来在百度的大更新中，靠简单的替换语义也不再起作用，可以说伪原创工具在当时是给很多站长带来了福音。那在后来，您是否还开发了类似的伪原创工具呢？和之前的天天伪原创工具有什么不同？

泊君：后来有由于公司变更天天伪原创工具更名为泊君伪原创工具，由于伪原创工具可读性无法满足一些高质量文章编辑，后来又开发了泊君原创工具，这款原创工具只在公司内部用，没有对外销售！

小编：原来如此！对于广大站长来说，这确实是一件让人非常心痛的事情。如果说有时间的話，您是否还会针对百度搜索引擎的算法，再次开发一款伪原创工具对外销售呢？

泊君：伪原创工具目前生成的文章依然收录很好。唯一需要改进的是继续升级词库，让文章更具有可读性，又省力，还不影响客户体验，不过中国汉语博大精深，要做到完美依然是需要继续努力的事情。

小编：恩！中国汉语博大精深，做到完美也几乎是一件不可能的事情；唯一能做的就是现有的功能中去尽量完善。

泊君：是的！

小编：那能否请问下，开发一款这样的工具需要花费您多长时间呢？

泊君：当时开发的时候国内没有近义词库，所以只好去书店买了一本现代汉语大词典，然后分成好几份，聘请的在校大学文科生兼职十几个，经过近半年的编写和修改才制作完成第一个同义词替换词库，回想起来工程还是很浩大的。比如说近义词库：元月—一月、腊月—12月等；记得当时遇到一些很难去找同义词的词语，比如：什么。后来实在没办法只好对应了一个词：啥子！现在回想起来，有点很可笑的感觉。

小编：听您这么一说，确实开发一款软件是既耗时又费力。既然如此，那最早的一款伪原创工具是如何形成的呢？

泊君：最早的第一款伪原创工具，主要是在把中文提交到在线翻译，翻译为英文，然后再把英文翻译成中文得来的。后来由于速度太慢，网络不稳定导致的很多问题无法得到解决，所以就耗时半年编写了一个中文近义词替换词库。

小编：对于时下百度算法来说，您觉得伪原创工具应该具备哪些方面？

泊君：对于搜索引擎来说，核心是关注客户体验，当然对于伪原创工具来说也是要更多的关注客户体验，让用户既省时又省力，同时又可以作出利于收录不影响网站整体体验的文章，这是伪原创工具未来的发展方向！

小编：没想到这次访谈谈了半个多小时，那在最后结束的时候，您还有什么话想对各位站长说吗？

泊君：作为一个站长，要创造有价值的互联网信息，但是同时又要养家糊口，文章伪原创也是一个无奈之举，所以希望站长朋友能兼顾与两者之间，在省时省力的同时不要忘记创造有价值的信息，更多的去关注解决网站的用户体验，关注客户真正想要的信息，并给出他们想要的答案，解决他们的问题。这才是一个有价值的网站存在的意义。

小编：泊君的这番话说出了广大站长的心声，特别是最近百度打击垃圾文章的同时，我们的网站更要为互联网、为用户创造有用的信息，这才是一个良好的互联网秩序！

FEBRUARY 5, 2018

Github-没有做不到-只有想不到

Github-没有做不到-只有想不到

Github背景

体验过Github的几个产品，感觉开发Github的这批人真的很富有想像力；原来

一个托管代码的网站还能这么玩：没有做不到、只有想不到；

所以今天我来谈谈Github的创新；

Github能做什么

提起Github，可能大家第一印象便是：

1. 不就一个托管代码网站么；
2. 顺便存储些文件、图片，还能当一个网盘用；
3. 可以在上面找些开源工具来用；
4. 不注册一个Github账号，都不好意思跟别人打招呼；
5. 程序猿交友网站（同性）

对，上边理解都有一定道理；要说Github的创新，我们来看看Github给我们提供了哪些产品？

1. 写代码、托管代码，这点不用多说，但是Github一出场就能远甩SourceForge几条街，仅仅是因为它支持Git这个工具么？
2. 写博客，什么，这不是WordPress这类软件做的事情么？你竟然用Github写博客，不好意思，Github真的可以，并且能让你像写代码一样写博客，让博客回归写作本质；
3. 代码片段，这是什么东东？对不起，它有个正式的名字叫：Gist，我们稍后会介绍；
4. 编辑器，当你还在讨论神用的编辑器和编辑器之神的时候，Github告诉我们，其实你还有另一种选择，Sublime？No No No，虽然它长的很像Sublime，它有正式名字叫：Atom；
5. 电子书，这TM不是epub、mobi之流的特长么，你敢说Github也支持？不错，Gitbook，写电子书，为你而来；
6. 工具市场，什么？Github要卖东西，不错，如果你需要一些工具，可以直接到市场里去买，对于一些小创公司来说，节省成本，正如Github Marketplace宣传视频里所说那样；
7. Bug追踪，你是在逗我么？我们公司明明有用开源的BugFree软件，且慢，先来看看Github的Issues；
8. StackOverFlow，神经病吧，这不是我们程序猿写代码的灵感源泉么？Github也有这功能，有没有我不知道，总之以前技术难题都去stackoverflow上查，现在发现有些难题得去Github上查；
9. 招聘，你是想逼疯HR么，Github说：我也不想的，只是有HR这么干，

从Github上找候选人，怪我喽？

10. 在线简历，行行行，I服了You，你到底还有哪些事情不能做？

11.

别着急，且听我慢慢给你道来。

代码托管

曾几何时，代码托管不是SourceForge、GoogleCode这类产品的天下么，怎么突然出来一个Github，把市场搅的天翻地覆，如果仅仅做为支持Git工具的角度来看，Github还真不比前边那些网站有优势，毕竟人家要么家大业大、要么起步早积累厚，但是Github不跟你们拼这些，Github说：我除了长像小清新、支持版本控制工具Git外，我还支持PR、Fork等功能，我还是代码网站中的微博、我能让程序猿们Follow他们喜欢的猿、我让用户有粘性，你要硬说我是程序猿交友网站，我也不反对；当你们还在苦苦力拼谁支持的免费空间更大、更长的時候，我玩的是社交，懂么？要想有粘性、得靠互动、得有社交；这时候是来看一下这张网站流传的图了：

Github-SS

在代码托管的领域里，Github说：我先行一步，你们这些老古董们慢慢追。

写博客

既然代码能托管，那么文章为啥就不能托管呢？能、能、能，Github Pages就是干这个的；你可以用你喜欢的任一款编辑器写完文章，Push到Github便可，此时文章便可访问。

什么？这么简单？难道不需要租得VPS、建数据库、搭Nginx、买域名？用WordPress五分钟建站、或者直接上AWS套餐不也也简单么？如果你觉得网站版富文本编辑器好用，那我也无话可说。

用过多款建站工具之后，Github Pages这类静态模板网站，真的是专为开发人员而设；不错，本站正是用Github Pages，Fork一个你喜欢的代码仓库，便可以开始写博客了。

当然，如果非开发人员，只要你会Git几个命令，上手也很简单；

代码片段

如果你平时需要将一些片段代码保存、留待后用、查看，那么试试**Gist**，比如这个场景：开发人员在写文章的时候，要在文章中嵌入一些代码片段，用**Gist**就恰好

编辑器

如果你用过**Sublime**，那么看到**Atom**，你会觉得很亲切；对js、Node同学是另一种不错的选择；畅想一下，加上**Github**的托管功能，**Atom**是不是就一个**Evenote**、云笔记？另加上强大的插件功能、原来的一些单体应用比如**Mou**是不是可以退休了？

Atom是想发展成程序猿日常工具的全家桶么？

Atom

电子书

如果你经常看技术文章，那么对**Gitbook**形式的电子书一定不陌生；也有很多开源项目的文档就用**Gitbook**来写，比如**TensorFlow**；

当然，你如果想用**Gitbook**写日记、写博客，也没人反对；

工具市场

工具市场有什么用呢？直接上官方宣传视频，一眼明了；

MarketPlace

现在市场里已经有**Travis CI**、**Sentry**等工具可用，这完全是模仿**App Store**的套路来的；

问题追踪

如果你有一个开源项目，别人有问题，可以用**Issues**给你提，是像**Jira Task**？还是像**BugFree**？还是像论坛留言板？你说像什么就像什么吧？反正用着挺爽；

如果大家都针对相应的技术问题用**Issues**讨论，那么**StackOverFlow**是不是感觉有点小危机？

如果一个人在**Issues**回答的问题质量又高、又多，是不是可以称得上大V？是不

是也可以开一堂Github Live?

其它

用Github做一张简历? Resume

HR在上面筛选简历?

当然还有很多功能我没写出来的、也会一直不断有新功能加进来;

体会

能把一个代码托管网站玩出这么多花样, 你能想得到么?

两岸猿声啼不住, 轻舟已过万重山.

FEBRUARY 4, 2018

github-gist-可以做什麼

github 提供另一個有用的服務, 叫做 gist。

開發者可以用來展示 code snippets。

但它不只是給 geeks 及 coders。

github 是給每一個人都可以使用。

如果你有聽過 web apps (像是 Pastebin 或 Pastie),

那麼, gist 就是類似的東西但是更好, 它沒有廣告以及擁有更多特色。

以下列出幾個服務你可以使用, 你不必是個 geek。

1. 匿名寫些文章。

你不必建立 github 帳號, 就可以使用 gists。

只要到 gist.github.com, 在文字框寫下幾段文字並且建立一個 gist。

你可以選擇建立一個 secret gist (私密 gist) 這不會被搜尋引擎看到, 只會被知道這個 URL 的人看到。

2. 像 wiki 一樣追蹤文章內容變化

一個被發佈過的 `gist`，當你編輯寫的內容時，之前的內容會被保留。
你可以按 `revisions tab` 來追蹤曾經做過哪些內容編輯。
內建的 `diff` 引擎會標記兩個版本之間的差別之處。
這也可以用來比對文字檔。

3. 以格式化文字發佈 `gist`

`gist` 接受非格式化文字(plain text)，
也接受用 `markdown` 格式文字發佈，
它會自己轉成 `HTML` 格式。
你可以加列表(list)，圖像(放在外部)，甚至表格(原先不在 `markdown` 規格內)。
當你用 `markdown` 格式寫好文章，記得用 `.md` 這個副檔名存檔。

4. 把 `gist` 當做寫作平台

現在有很多寫作的引擎，從 `Blogger` 到 `Medium` 到 `Tumblr` 等等。
你也可以用 `github` 的 `gist` 當做一個平台快速發佈你的文章到 `web` 上。
建立一個 `gist` (使用 `plain text` 或 `markdown` 格式)，
然後用 `roughdraft.io` 來發佈，把 `gist` 做成一個 `web page` (網頁)
如果想增加可讀性，還可以使用表情符號。

5. 單頁網站的服務提供

`Gist` 有一個用有趣的應用叫做 `Bl.ocks`。
你可以寫 `HTML`, `CSS`, `javascript`，然後把 `gist` 存成 `index.html`。
然後用 `bl.ocks.org` 讓它變成 `html` 的網頁，看起來就像在瀏覽器上看。
例如這個連結就是個活生生的例子。
`http://bl.ocks.org/labnol/raw/122d4de95c6a127b1c9b/`
原始檔放在 `gist`： `https://gist.github.com/labnol/122d4de95c6a127b1c9b`

這招看起來就是會有流量的限制，但這仍是一個好工具。

另一招是用 `google drive`： `http://www.labnol.org/internet/host-website-on-google-drive/28178/`

6. 維設任務列表

你可以使用 **gist** 來追蹤「卡關任務」(範

例: <https://gist.github.com/labnol/8e1cdf64cd7b0c1a811e>)

這些任務在純文本中用特別語法寫成，但它們會被換成勾選框。

你可以勾選或不勾選這些項目，它們自己會更新原始文字。

如果你的 **gist** 是公開的，任何人可以看到你的待辦事項，但只有擁有者才能改變每個任務的狀態

7. 把 gist 當成網路剪貼簿

gistbox 是 **google chrome** 的外掛，可以讓你從網頁保留些小段文字，存成公開或私密的 **gist**。

你甚至可以加 **label**, **hashtag**，讓你的 **gist** 容易被查找。

8. 把 gist 嵌進網頁裡

你可以把任何 **gist** 嵌進自己的網頁中，只需要一行 **javascript** 的程式。

被嵌入的 **gist**，會維持所有的格式與語法標示。

你的網頁訪客可以容易地克隆或分支你的 **gist** 進入他們的 **github** 帳號。

在 **WordPress blog** 也有插件及短程式碼可嵌入 **gist**。



oEmbed Gist by Takayuki Miyauchi

□ □ □ □ □ (16)

Embed your source from gist into WordPress easily.

□ □ Tested with 4.9.8

Get this plugin

Gist

```
<script src="https://gist.github.com/username/gist-id.js"></script>
```

9. 量測流量

你可以使用 **google analytics** 來量測你的 **gist** 的流量。

因為 **github** 不允許在純文本的 **gist** 使用 **javascript**，我們可以用 **GA Beacon** 來記錄 **gist** 即時的訪問。

<https://github.com/igrigorik/ga-beacon>

加入以下的程式在你的 `gist`，並且存成 `markdown` 格式，這樣就會追蹤你的 `gist`。

10. 使用桌面軟體來管理 `gist`

Gisto(<http://www.gistoapp.com/>)是一個免費的桌面應用軟體，可以用來管理自己的 `gist` (不依靠瀏覽器)。

你可以搜尋 `gist`，編輯 `gist` 的內容，觀看歷史記錄，分享 `gist`。

這軟體有 Mac OSX，Windows，Linux 版本。

另外，GistBox (<http://www.gistboxapp.com/>)這個 web app 是另外一種選擇。

ARCHIVES

July 2018

June 2018

May 2018

April 2018

March 2018

February 2018

January 2018

CATEGORIES

Python

META

Log in

Entries RSS

Comments RSS

WordPress.org

PAGES

FxForex.ga List

HostNine

KVC

Namesilo

Python职业方向

计课Project

February 2018						
M	T	W	T	F	S	S
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28				
« Jan					Mar »	