

目录

前言	1.1
Python爬虫简介	1.2
抓包分析	1.3
裸写Python爬虫代码	1.4
用Python库写爬虫代码	1.5
用Python框架写爬虫代码	1.6
附录	1.7
参考资料	1.7.1

如何用Python写爬虫

- 最新版本： v0.7
- 更新时间： 20190329

简介

总结如何用Python去写爬虫，包括如何裸写爬虫代码，如何用Python库去写爬虫，如何用Python爬虫框架写爬虫，并给出实例详细解释具体的操作过程。

源码+浏览+下载

本书的各种源码、在线浏览地址、多种格式文件下载如下：

Gitbook源码

- [crifan/use_python_write_spider: 如何用Python写爬虫](#)

如何使用此Gitbook源码去生成发布为电子书

详见：[crifan/gitbook_template: demo how to use crifan gitbook template and demo](#)

在线浏览

- [如何用Python写爬虫 book.crifan.com](#)
- [如何用Python写爬虫 crifan.github.io](#)

离线下载阅读

- [如何用Python写爬虫 PDF](#)
- [如何用Python写爬虫 ePUB](#)
- [如何用Python写爬虫 Mobi](#)

crifan.com, 使用[知识署名-相同方式共享4.0协议](#)发布 all right reserved, powered by Gitbook该文件修订时间： 2019-03-29 22:59:32

Python爬虫简介

[爬取你要的数据：爬虫技术](#)中已经解释了爬虫的核心步骤了和相关涉及内容，也提到了很多语言都可以实现爬虫，都能爬取到你要的数据。

不过不同语言有自己的侧重点，而其中爬虫领域，最方便的，要数 Python。其在爬虫领域，有很多程序的库，框架，可供使用，便于高效的实现爬虫的功能。

用Python写爬虫的不同方式

正如[爬取你要的数据：爬虫技术](#)中所整理的，用Python去写爬虫，也有三种方式：

- 裸写Python爬虫代码
 - 下载
 - python的内置http网络库
 - [urllib](#)
 - [crifanLibPython](#)中的[getUrlRespHtml](#)
 - 提取
 - [re](#)模块
 - [Python中的正则表达式：re模块详解](#)
 - 保存
 - [txt](#)
 - [csv / excel](#)
 - [Python心得：操作CSV和Excel](#)
 - 用各种Python库组合去写爬虫代码
 - 下载
 - 选择第三方的、更强大的、更好用的网络库
 - [Python心得：http网络库](#)
 - [Requests](#)
 - [aiohttp](#)
 - 提取
 - [BeautifulSoup](#)
 - [Python专题教程：BeautifulSoup详解](#)
 - v3 -> Python2
 - v4 -> Python3
 - [PyQuery](#)
 - [Python心得：HTML解析库PyQuery](#)
 - [lxml](#)
 - [【记录】Python中尝试用lxml去解析html – 在路上](#)
 - 保存
 - [csv / excel](#)
 - [PyMySQL](#)
 - [主流关系数据库：MySQL](#)
 - [PyMongo](#)
 - [主流文档型数据库：MongoDB](#)
 - 用爬虫框架去写爬虫代码
 - 常见Python爬虫框架

- PySpider
 - Python爬虫框架：PySpider
- Scrapy
 - 主流Python爬虫框架：Scrapy
- 其他相关
 - 【整理】pyspider vs scrapy

crifan.com, 使用[知识署名-相同方式共享4.0协议](#)发布 all right reserved, powered by Gitbook该文件修订时间: 2019-03-29 22:08:12

抓包分析

不论用那种方式去写爬虫代码，都要先知道：

具体需要爬取哪些url（或api等），url地址是多少

以及需要传递什么参数，才能返回我们希望的数据

而要搞清楚，要抓取哪些数据，这些数据怎么获取到，即：

- 需要访问的网页url地址是什么
 - 以及需要传递什么参数
- 对于返回数据，需要抓取具体哪一部分
 - 对应的数据的提取规则是什么

这一抓取网络请求的数据包的过程，叫做： 抓包

抓包常用辅助工具

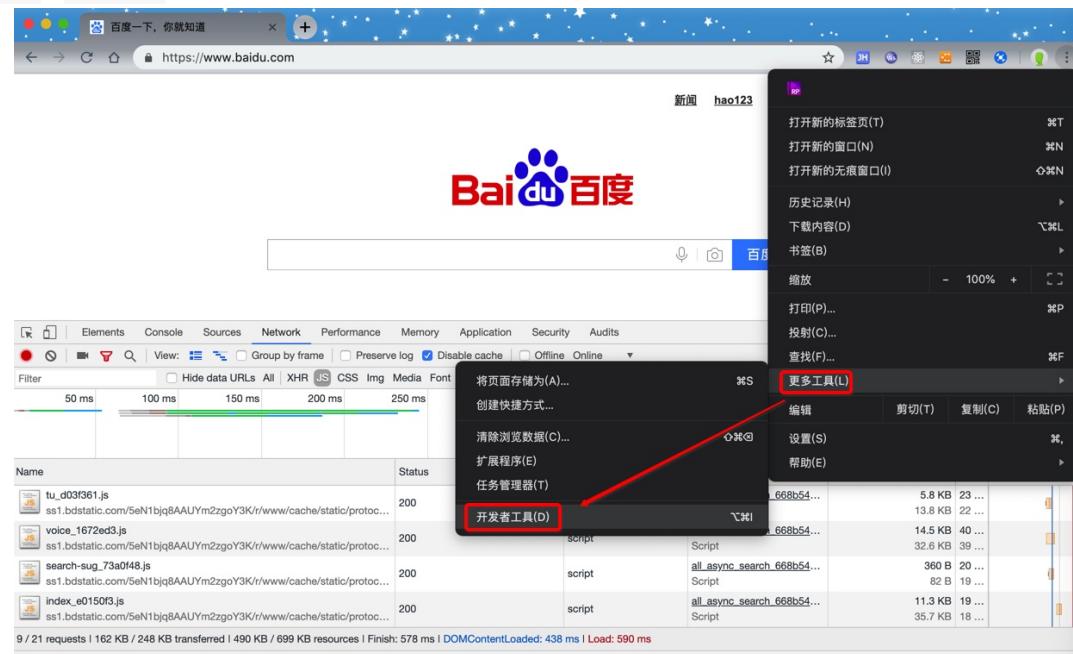
我们要写爬虫去爬取的数据，从数据源的形态分，大概分两类：

- 网站 =网页=网站中的各种网页
- app =app中内部发出的请求设计的api接口

根据要抓取的数据的源不同，常用的一些辅助分析工具有：

- 网站抓包分析

- Chrome 的 开发者工具



■ 快捷键：

■ Windows: Ctrl + Shift + I

■ Mac: Command + Option + I

■ 如何使用

■ 官网资料：[Chrome 开发者工具](#)

- IE 的 F12
 - 如何使用
 - 【整理】各种浏览器中的开发人员工具Developer Tools: IE9的F12, Chrome的Ctrl+Shift+J, Firefox的Firebug
 - 【总结】浏览器中的开发人员工具（IE9的F12和Chrome的Ctrl+Shift+I） -网页分析的利器
 - 【教程】如何利用IE9的F12去分析网站登陆过程中的复杂的（参数, cookie等）值（的来源）
 - 【教程】手把手教你如何利用工具(IE9的F12)去分析模拟登陆网站(百度首页)的内部逻辑过程
 - Firefox 的 firebug
- app抓包分析
 - Charles
 - app抓包利器：Charles

具体怎么抓包

先要搞清楚自己想要抓取什么数据，然后再去用工具辅助分析出网页或app等数据源中，如何一步步的获取对应数据，找到期间所要依次访问哪些url或api，传递什么参数，最终获取到所要的数据。

以抓取汽车之家中车型车型数据为例解释如何抓包

下面就以，想要抓取汽车之家网站中的车型车系数据为例，来解释，如何用抓包工具辅助分析，依次访问哪些页面，之后如何提取，才能得到我们要的数据。

TODO:

用Chrome浏览器分析过程，并截图，添加解释。

crifan.com, 使用[知识署名-相同方式共享4.0协议](#)发布 all right reserved, powered by Gitbook该文件修订时间: 2019-03-29 22:50:11

裸写Python爬虫代码

TODO:

用python内置urllib去裸写代码，去下载，再用re正则去提取，汽车之家车型车系数据。

crifan.com, 使用[知识署名-相同方式共享4.0协议](#)发布 all right reserved, powered by Gitbook该文件修订时间: 2019-03-29 22:30:55

用Python库写爬虫代码

TODO:

用python的第三方http库，比如requests，去下载，再去用BeautifulSoup去提取，汽车之家车型车系数据。

crifan.com, 使用[知识署名-相同方式共享4.0协议](#)发布 all right reserved, powered by Gitbook该文件修订时间: 2019-03-29 22:30:40

用Python框架写爬虫代码

用Python爬虫框架PySpider去爬取汽车之家的车型车系数据

此处举例说明，用PySpider这个Python爬虫框架去爬取汽车之家的车型车系数据

详细过程参见：

[【已解决】写Python爬虫爬取汽车之家品牌车系车型数据 – 在路上](#)

期间包括：

- [【记录】Mac中安装和运行pyspider](#)
- [【已解决】pyspider中如何写规则去提取网页内容](#)
- [【已解决】pyspider中如何加载汽车之家页面中的更多内容](#)
- [【已解决】PySpider如何把json结果数据保存到csv或excel文件中](#)
- [【已解决】PySpider中如何清空之前运行的数据和正在运行的任务](#)

TODO:

把 `autohomeCarData` 代码上传到GitHub，并在此贴出地址

而关于PySpider更多的介绍，详见：

[Python爬虫框架：PySpider](#)

crifan.com, 使用[知识署名-相同方式共享4.0协议](#)发布 all right reserved, powered by Gitbook该文件修订时间：2019-03-29 22:48:50

附录

下面列出相关参考资料。

crifan.com, 使用[知识署名-相同方式共享4.0协议](#)发布 all right reserved, powered by Gitbook该文件修订时间: 2019-03-29 21:30:12

参考资料

- 爬取你要的数据：爬虫技术
- crifanLibPython
- getUrlRespHtml
- re
- Python中的正则表达式：re模块詳解
- Python心得：操作CSV和Excel
- Python心得：http网络库
- Requests
- aiohttp
- Python专题教程：BeautifulSoup詳解
- Python心得：HTML解析库PyQuery
- 【记录】Python中尝试用lxml去解析html – 在路上
- 主流关系数据库：MySQL
- 主流文档型数据库：MongoDB
- Python爬虫框架：PySpider
- 主流Python爬虫框架：Scrapy
- 【整理】pyspider vs scrapy
- PyMySQL
- PyMongo
- urllib
- BeautifulSoup
- PyQuery
- lxml
- PySpider
- Scrapy
- Chrome 开发者工具 | Tools for Web Developers
- 【整理】各种浏览器中的开发人员工具Developer Tools：IE9的F12，Chrome的Ctrl+Shift+J，Firefox的Firebug
- 【总结】浏览器中的开发人员工具（IE9的F12和Chrome的Ctrl+Shift+I） -网页分析的利器
- 【教程】如何利用IE9的F12去分析网站登陆过程中的复杂的（参数， cookie等）值（的来源）
- 【教程】手把手教你如何利用工具(IE9的F12)去分析模拟登陆网站(百度首页)的内部逻辑过程
- app抓包利器：Charles
- 【已解决】写Python爬虫爬取汽车之家品牌车系车型数据 – 在路上
- 【记录】Mac中安装和运行pyspider
- 【已解决】pyspider中如何写规则去提取网页内容
- 【已解决】pyspider中如何加载汽车之家页面中的更多内容
- 【已解决】PySpider如何把json结果数据保存到csv或excel文件中
- 【已解决】PySpider中如何清空之前运行的数据和正在运行的任务
- rmax/scrapy-redis: Redis-based components for Scrapy.
- grangier/python-goose: Html Content / Article Extractor, web scrapping lib in Python
- Bloom Filters by Example
- Bloom Filters by Example 中文
- Scrapy入门教程 — Scrapy 0.24.6 文档
- Scrapy爬虫框架教程（一）-- Scrapy入门

