## 爬虫HTTP下载模块文档

# 模块功能

为了规范下载过程，作为爬虫的核心，特将下载和代理封装在一起，方便管理和减少代理重复，该模块包含http下载和proxy代理更换，减少爬取数据流程中对代理的处理，简化处理流程，清晰逻辑，其中http主管下载，proxy负责代理的更新，怎么更新。

## http

该方法负责接受http的参数，包括url，cookies，proxy，headers，data，callback，errback, \*\*kwargs等等参数, set\_cookie和上次的cookie封装在一起，然后通过requests包的get和post方法获取下载页面，获取response对象，通过twisted将每次访问derfer后形成异步访问，通过twisted的reactor循环异步快速访问数据，此HTTP相当于一个HTTP的服务，根据提供的url,cookie返回response。

http接口参数：

**def** request(self,url, method=**'get'**, callback=None, errback=None, meta=None, \*\*kwargs):

*"""Constructs and sends a :class:`Request <Request>`.  
Returns :class:`Response <Response>` object.***:param** *method: method for the new :class:`Request` object.***:param** *url: URL for the new :class:`Request` object.***:param** *params: (optional) Dictionary or bytes to be sent in the query string for the :class:`Request`.***:param** *data: (optional) Dictionary, bytes, or file-like object to send in the body of the :class:`Request`.***:param** *headers: (optional) Dictionary of HTTP Headers to send with the :class:`Request`.***:param** *cookies: (optional) Dict or CookieJar object to send with the :class:`Request`.***:param** *files: (optional) Dictionary of ``'name': file-like-objects`` (or ``{'name': ('filename', fileobj)}``) for multipart encoding upload.***:param** *auth: (optional) Auth tuple to enable Basic/Digest/Custom HTTP Auth.***:param** *timeout: (optional) How long to wait for the server to send data  
 before giving up, as a float, or a (`connect timeout, read timeout  
 <user/advanced.html#timeouts>`\_) tuple.***:type** *timeout: float or tuple***:param** *allow\_redirects: (optional) Boolean. Set to True if POST/PUT/DELETE redirect following is allowed.***:type** *allow\_redirects: bool***:param** *proxies: (optional) Dictionary mapping protocol to the URL of the proxy.***:param** *verify: (optional) if ``True``, the SSL cert will be verified. A CA\_BUNDLE path can also be provided.***:param** *stream: (optional) if ``False``, the response content will be immediately downloaded.***:param** *cert: (optional) if String, path to ssl client cert file (.pem). If Tuple, ('cert', 'key') pair.  
"""*

调用方法如下：

request(**'http://gsxt.lngs.gov.cn/saicpub/commonsSC/loginDC/securityCode.action?tdate=0'**,callback=self.yzm,meta={**'company\_name'**:**u"铁岭市薄香花"**},errback=self.err)

然后回调函数yzm的输入就是request的输出：

def yzm(res):

do\_yzm(res)

## 对http的监控

对每次http请求一个监控程序，如是否成功，失败次数，连续失败次数，在筛选优质代理时候，连续更换次数，下载次数，在哪一步失败率比较大，等等，方便后面对网站进行分析，

# 二、流程图



# 三、相关技术

1、封装requests的访问网络的模块

3、通过twisted封装http下载，形成异步下载模式

4、对访问返回错误的处理