requests库

2020年3月6日 8:42

Request 和 Response

请求 和 响应

浏览器（360极速模式，Chrome，Firefox，Edge）

在网页上右键-审查元素/检查，快捷键F12

可以看到网页源代码等信息

网页（大家可以看成是HTML）是由一组组标签构成的，

<起始标签>.....<终止标签> eg. 董老师讲课真好

HTML：超文本，归根结底还是文本，一个字符串

如何把这个字符串变成五彩斑斓的网页呢？都是靠浏览器的解释、渲染等工作的功劳

Request url:

Url http://www.baidu.com

DNS -> IP （ipv4 192.168.0.1） 或 （ipv6）

获取这个服务器上的资源（包括网页、图片....）

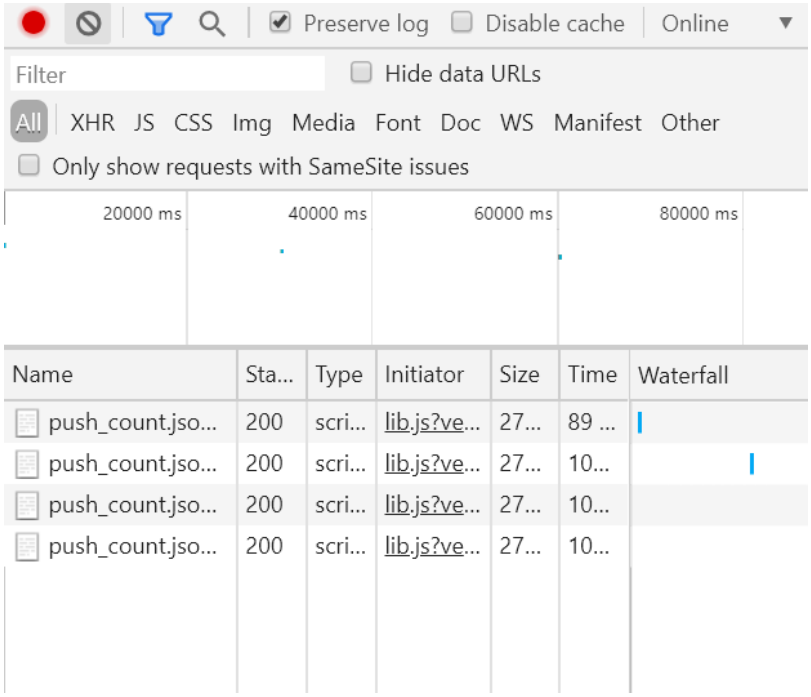
我们可以去干预浏览器的处理过程，eg.

但是，这种都是假象，刷新一下就没了

浏览器F12 【Network/网络】选项卡

是爬虫最重要的辅助功能
不会【Network】=盲人

我们打开了weibo.com，我们清空了记录（clear）
发现过了一段时间，变成了这样



明明我们没有动这个网页，但还是有数据包的传输
说明，这个网页有一个类似定时器的东西，再定时地、悄悄地与服务器发现交流

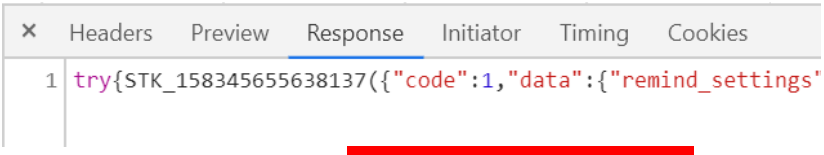
网页，通常情况涉及3个东西
HTML：描述的是一篇文章，HTML标签代表的是一个身份，说明被这个标签括起来的部分是什么
CSS（含style）：掌管样子（样式），这个标签（元素）长得什么样
JavaScript（简称JS）：虽然名字里有Java，但是和Java没有半毛钱关系。掌管动作。

状态码：200 正常
2020=404+404+404+404+404 程序员间的一个梗
404 Not Found 没找到 最常见的网页报错的状态码
5xx 服务器错误

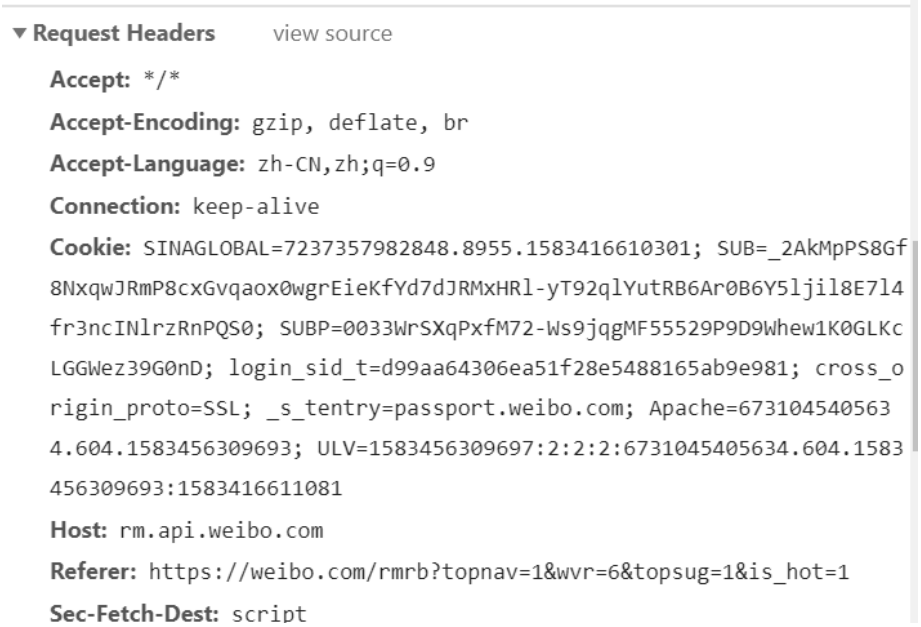
常见的服务器软件： apache、nginx、iis（微软，在装win系统，推荐安装旗舰版，家庭版没有办法安装IIS）
HTML：头<head> 身体<body>
文件：查看它的二进制，可以看到头和身体。明显的例子是：安装或copy一个文件耗时特别长，但是删除一个文件就特别快，原理就在于，copy时是复制的头和身体，删除时只是删掉了头，但

身体还保留在电脑的硬盘上。当我们误删了文件，我们可以使用diskrecovery等类似的还原软件，去把头给他补上，因为作为文件主体的身体还在。那么，身体就这样一直占用着硬盘的空间吗？不是！我们在新建、copy、安装一个文件时，发现这段空间是没有头（也就是无主的）的，电脑就是在这段空间上执行写操作（抹掉了它）。

Response：也有头header和身体

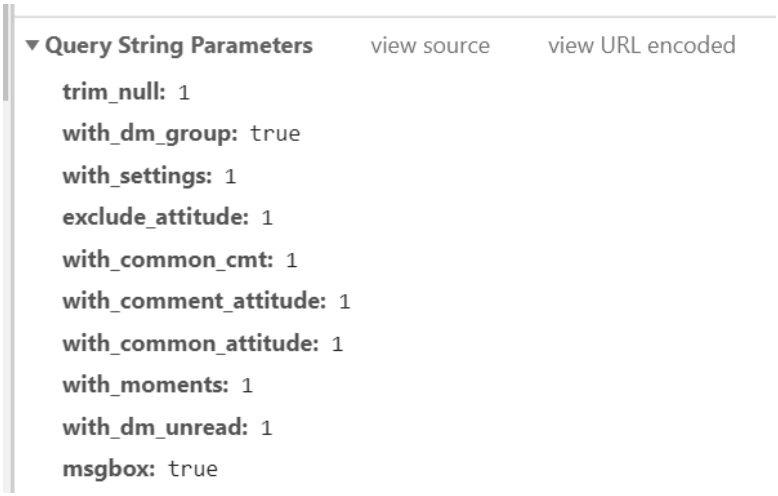


Request：也有头和身体 （重中之重：request header）



GET方法

https://rm.api.weibo.com/2/remind/push_count.json?trim_null=1&with_dm_group=true&with_settings=1&exclude_attitude=1&with_common_cmt=1&with_comment_attitude=1&with_common_attitude=1&with_moments=1&with_dm_unread=1&msgbox=true&with_page_group=1&with_chat_group=1&with_chat_group_notice=1&_pid=1&count=11&source=351354573&status_type=0&callback=STK_158345655638137



作业：看mooc

https://www.icourse163.org/learn/BIT-1001870001?tid=1206951268#/learn/announce