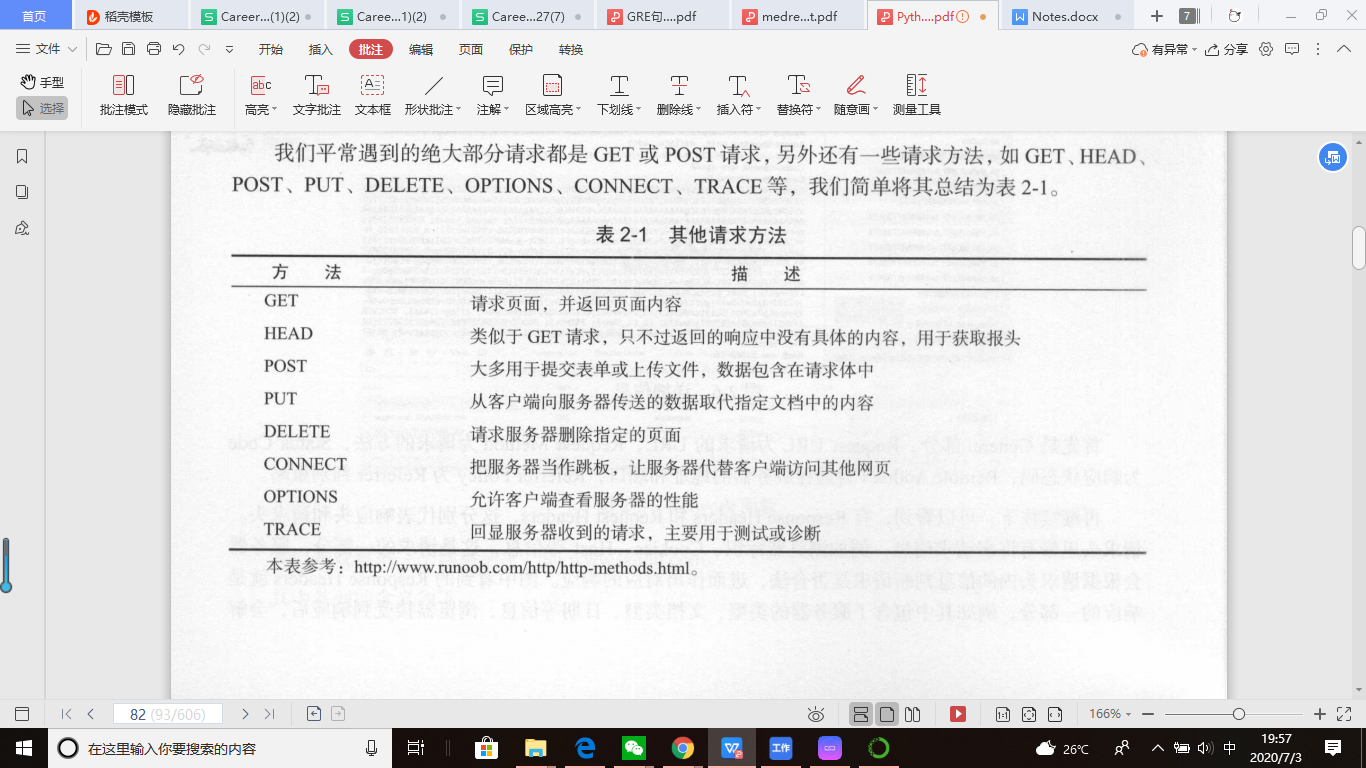
Python3网络爬虫教学笔记

**第一章**

URI:URL(Location）&URN（name）

Https：多了一个ssl加密

网站请求：

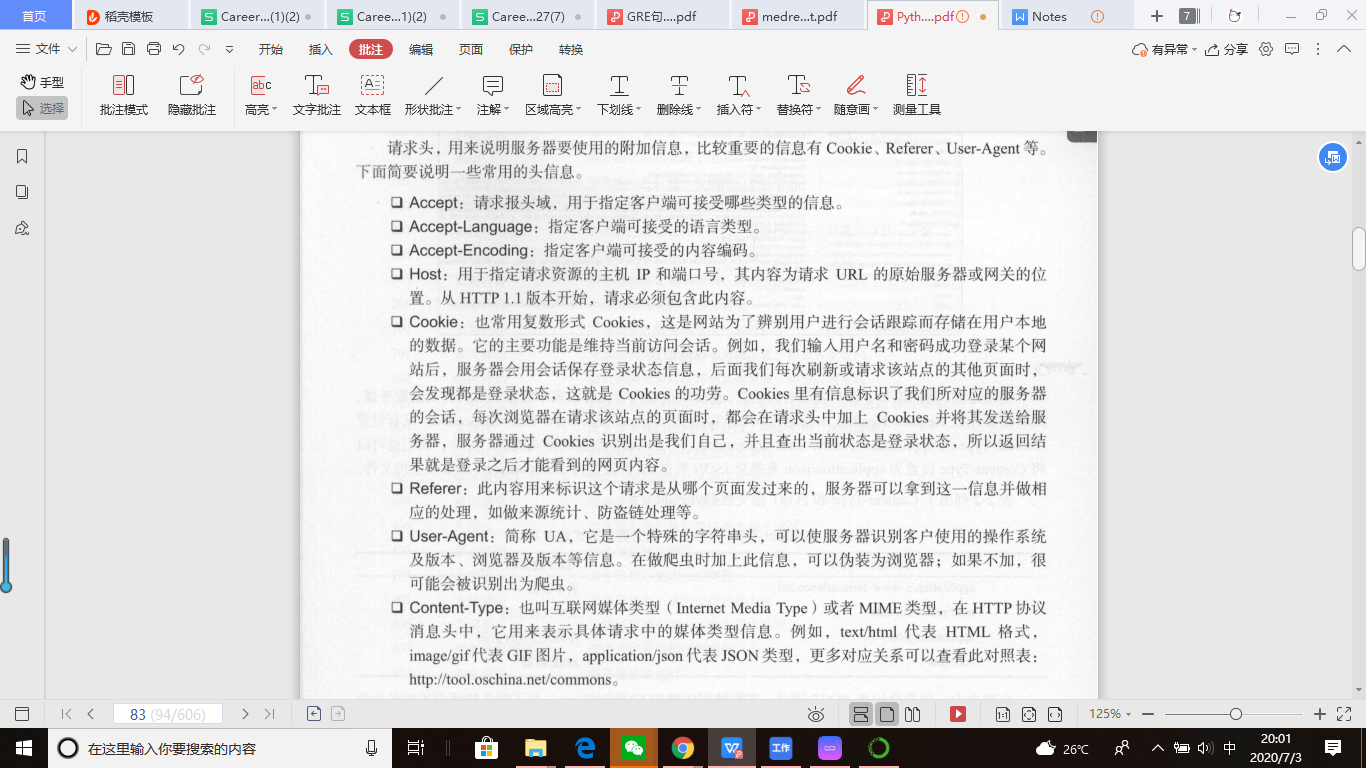
客户端向服务端发出，可以分为：请求方法（ Request Method 请求的网址( Request URL ）、请求头（ Request Headers） 请求体（ Request Body）

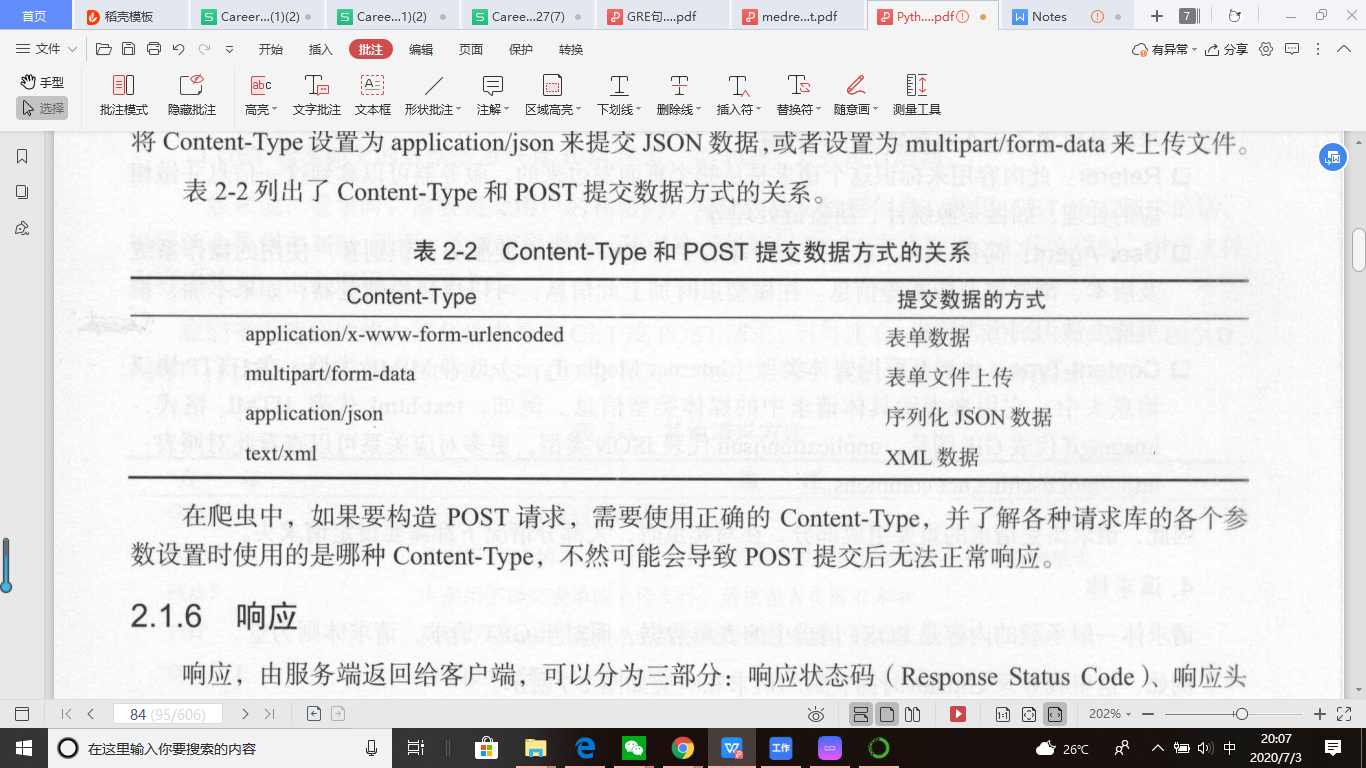
请求方法：

Get常用于直接到页面，反映在url里，有字节限制， Lost常用于表单填写和登录，用户名密码不会显示在url里，没有字节限制。

请求头：

用来说明服务器需要哪些附加信息

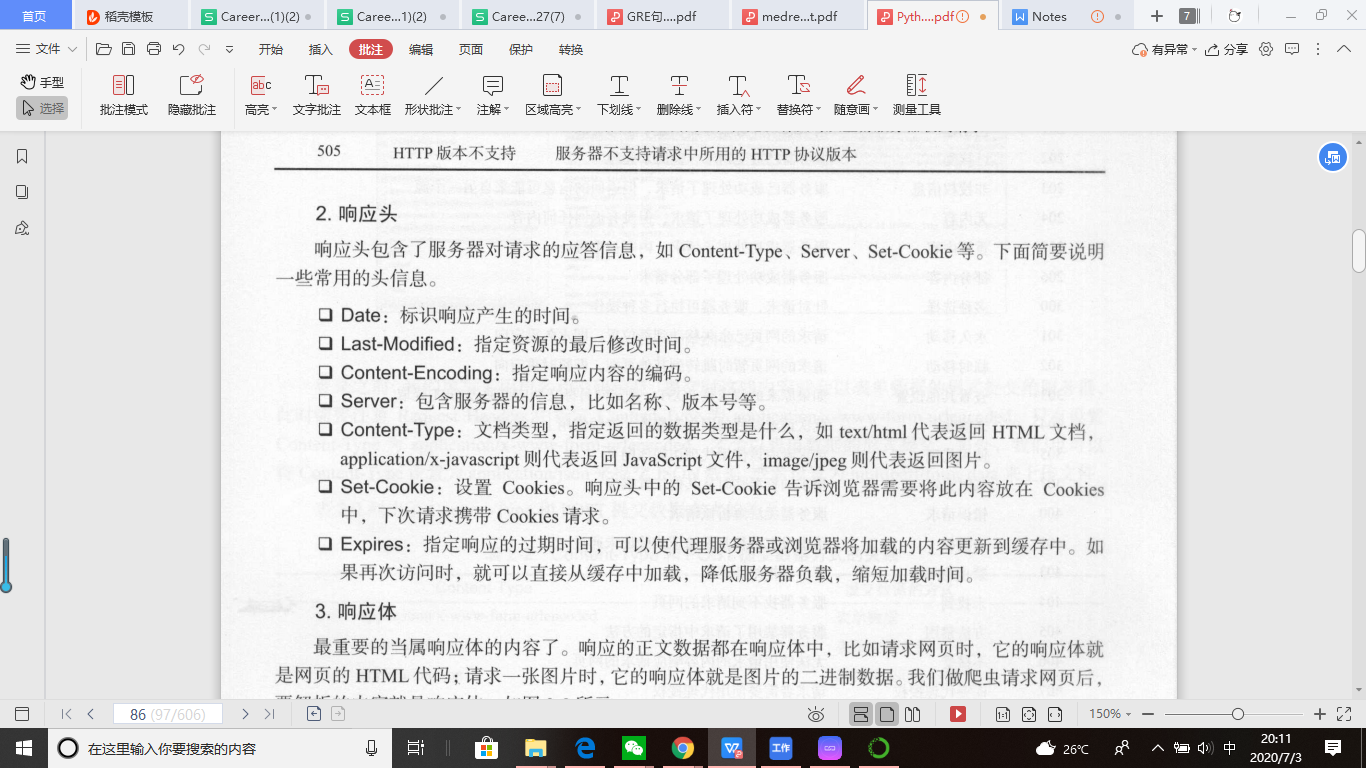


在写爬虫的时候，通常都是需要设定请求头的。

请求体通常contain POST请求里的表单信息，其他时候则为空

响应，由服务端返回给客户端，可以分为 部分：响应状态码（ Response Status Code ）、响应头

( Response Headers ）和响应体（ Response Body )



响应状态码表示服务器的响应状态: 表可以单独查到

响应头包含了服务器对请求的应答信息

响应的正文数据都包含在响应体中

网页基础：

HTML 相当于骨架， JavaScript 当于肌肉， css 当于皮肤

css ，全称叫作 Cascading Style Sheets ，即层叠样式表 “层叠”是指当在 HTML 中引用了数个样式文件，并且样式发生冲突时，浏览器能依据层叠顺序处理 “样式”指网页中文字大小颜色色 、元素间距、排列等格式

JavaScript ，简称 JS 是一种脚本语言，提供网页的交互内容

HTML 中，所有标签定义的内容都是节点，它们构成了一个 HTML DOM（Document Object Model）

CSS选择器来定位节点

（除此之外还有好多）

爬虫概述：

获取网页并提取和保存信息的自动程序

动态网页： 动态解析 URL 数的变化，关联数据库井动态 现不同的页面内容，非常灵活多变

HTTP 无状态是指 HTTP 协议对事务处理是没有记忆能力的，也就是说服务器不知道客户端是

什么状态

会话：服务器端保存，Cookie，客户端保存， 两个协作

代理与IP伪装：

Proxy Server， 中转站

按照协议和匿名程度分类

**第二章基本库的使用**

Urllib：

Request：发送请求 Error：异常处理

Parse：拆分解析合并URL RobotParser:识别robot.txt 分辨哪些可以爬哪些不能

Request：

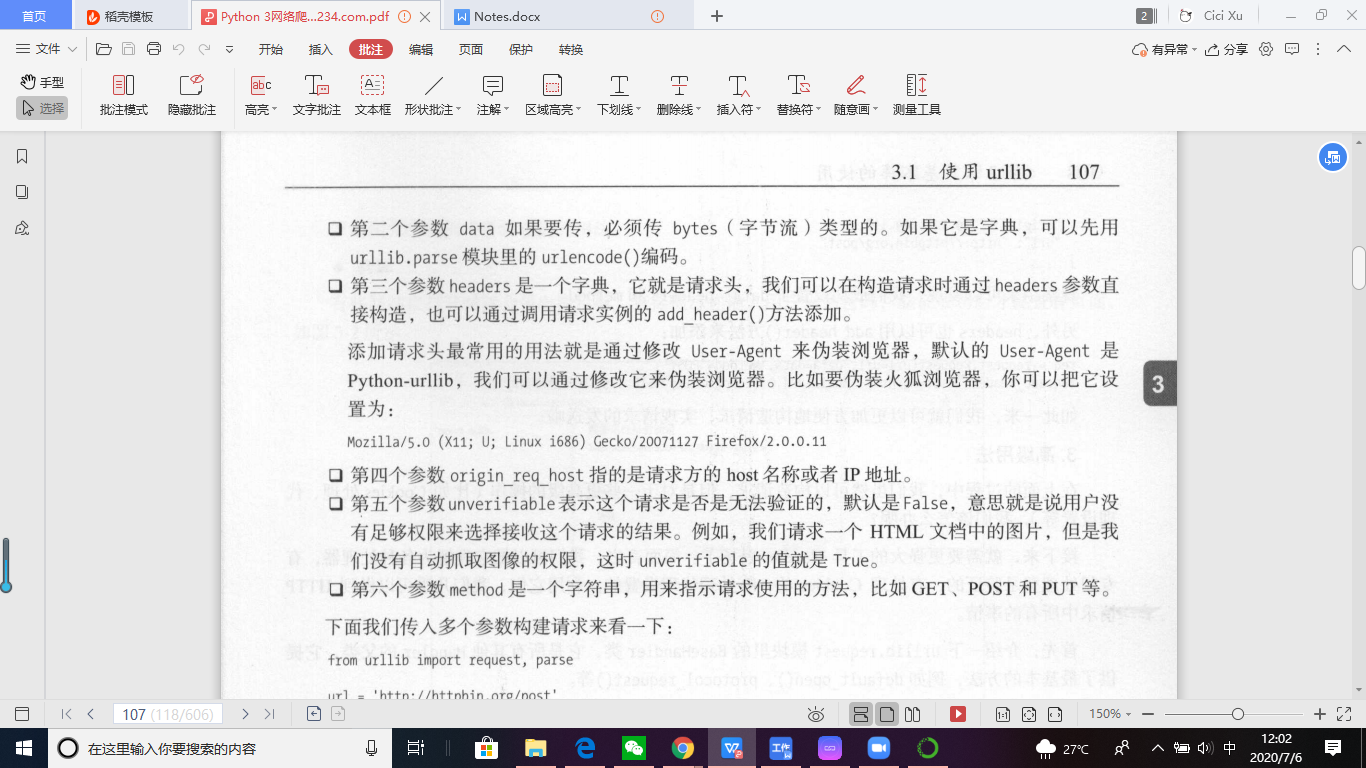
1. Urlopen， 直接找到源码，可以进行read （） readinto、 getheader(name）、Getheader，fileno等方法，以及 msg version status reason debuglevel closed 等属性

*urlopen(url,data=None,[timeout, ]\*,cafile=None,capath=None,cadefault=False,context=None)*

Data:传输的data需要通过bytes转换，改为POST因为传输了数据

Timeout设置超时时间，单位是秒， 结合try/except跳过时间长的抓取

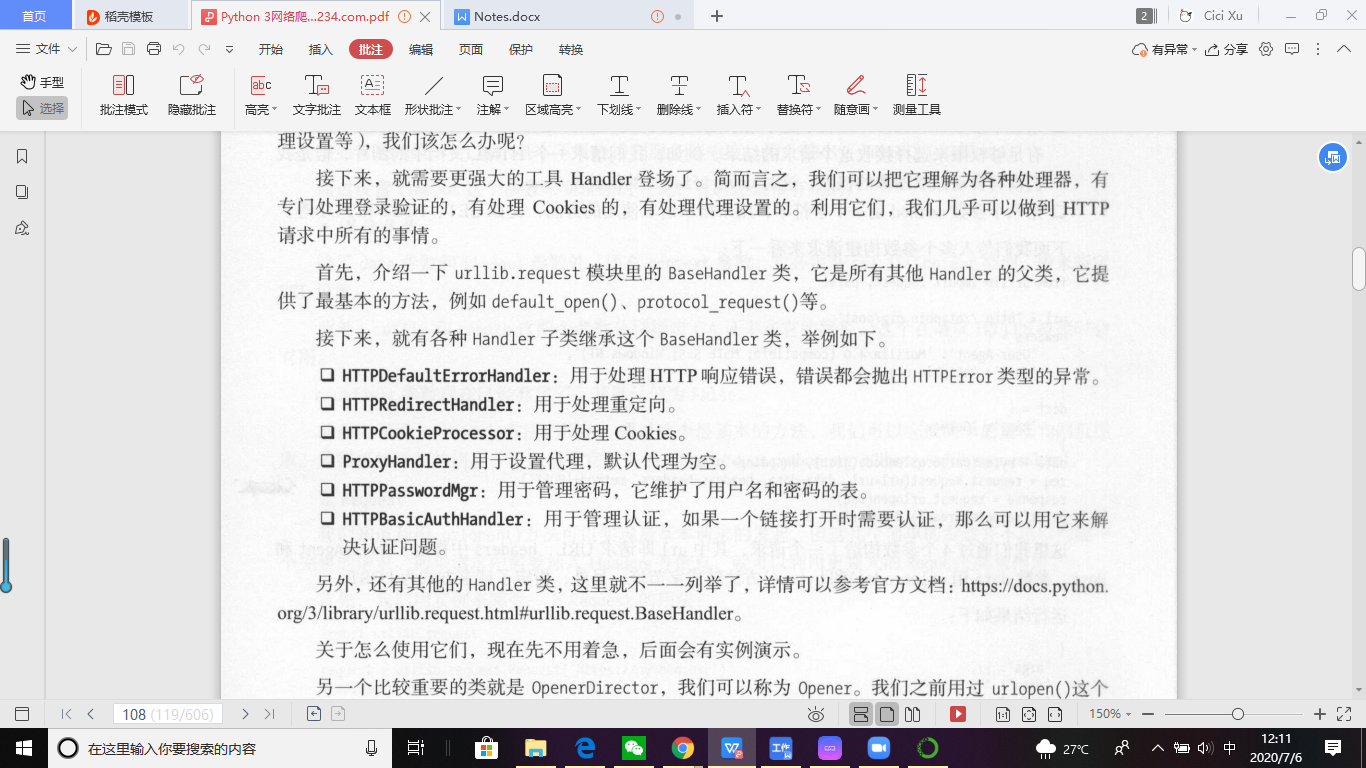
context 参数，它必须是 ssl.SSLContext 类型，用来指定 SSL cafile capath 这两个参数分别指定 CA 证书和它的路径，这个在请求 HTTPS 时有用， cadefault 参数已经弃用，默认为 False



1. Request：requet作为一个对象，用urlopen打开

Class urllib.request.Request(url,data=None,headers={}, origin\_req\_host=None,unverifiable=False, method=None)

Req.add\_header()

Opener, 比urlopen更强大的open方法

HTTPError：URLError子集，处理请求时的error，code返回状态码，reason，header