[Python3网络爬虫实战 1](#_Toc1283)

[安装常用库 1](#_Toc15472)

[爬虫基本原理 2](#_Toc4315)

[爬虫基础 2](#_Toc10103)

[HTTP基本原理 2](#_Toc23607)

[无状态HTTP 2](#_Toc30641)

[什么是爬虫 3](#_Toc19910)

[爬虫的基本原理 3](#_Toc4333)

[爬虫基本流程 3](#_Toc31346)

[Request与Response的概念 4](#_Toc23395)

[Request包含的内容 4](#_Toc31937)

[Response中包含的内容 5](#_Toc13828)

[能抓怎样的数据 5](#_Toc13401)

[解析方式 6](#_Toc29803)

[为什么我抓到的和浏览器看到的不一样? 6](#_Toc5812)

[怎样解决JavaScript渲染的问题 6](#_Toc3411)

[可以怎样保存数据? 7](#_Toc16950)

[代理的基本原理 7](#_Toc1468)

[代理的作用 7](#_Toc20301)

[爬虫代理 7](#_Toc17951)

[代理分类 7](#_Toc30536)

[常见的代理设置 8](#_Toc12900)

[Urllib库详解 8](#_Toc9974)

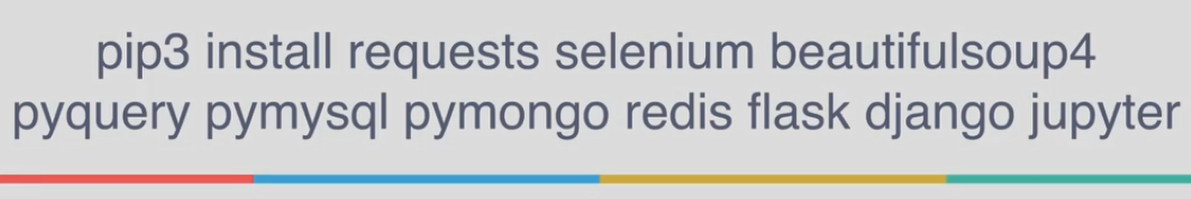
[什么是Urllib 8](#_Toc25214)

[相关方法 9](#_Toc862)

[Urlopen 9](#_Toc2252)

# Python3网络爬虫实战

## 安装常用库



用于自动化测试

Chromedriver

Geckodriver

异步http请求库

Aiohttp

字符编码检测库

Cchardet

DNS解析库

Aiodns

光学字符识别

Tesserocr (tesseract软件的python API访问层，所以需要先下载tesseract,pip install tesserocr pillow)

APP爬取相关库的安装

Charles (作用域fiddler相似，好用，但是收费,安装代理证书，手机也是，然后手机代理电脑对外访问抓包)

Mitmproxy (与Fiddler和mitmproxy作用差不多，但是是命令行的内容)

Appium(手机浏览器的selenium,模拟点击，滑动，等操作)

爬虫框架

Pyspider

Scrapy

Scrapy-Splash (Scrapy中支持JavaScript的渲染工具)

Scrapy-Redis(分布式专用)

Scrapyd(部署与运行的Scrapy项目的工具)

Scrapy-Client(将Scrapy代码部署到远程的Scrapyd,通过集中式的远程管理)

Scrapy-API(通过API获取内容的Scrapyd的运行情况)

Scrapyrt(为scrapy提供一个调度的HTTP接口,通过HTTP来调度Scrapy任务)

Gerapy(Scrapy的分布式管理模块)

## 爬虫基本原理

### 爬虫基础

#### HTTP基本原理

URI: 统一资源标识符

URL: 统一资源定位符

URN: 统一资源名称

URI = URL + URN

#### 无状态HTTP

无状态访问时HTTP的特点，所以我们需要会话(Session)和Cookies的辅助才能变为有状态访问，避免多次进行登录操作

### 什么是爬虫

**请求**网站并**提取**数据的**自动化**程序

### 爬虫的基本原理

概述

获取网页

提取信息

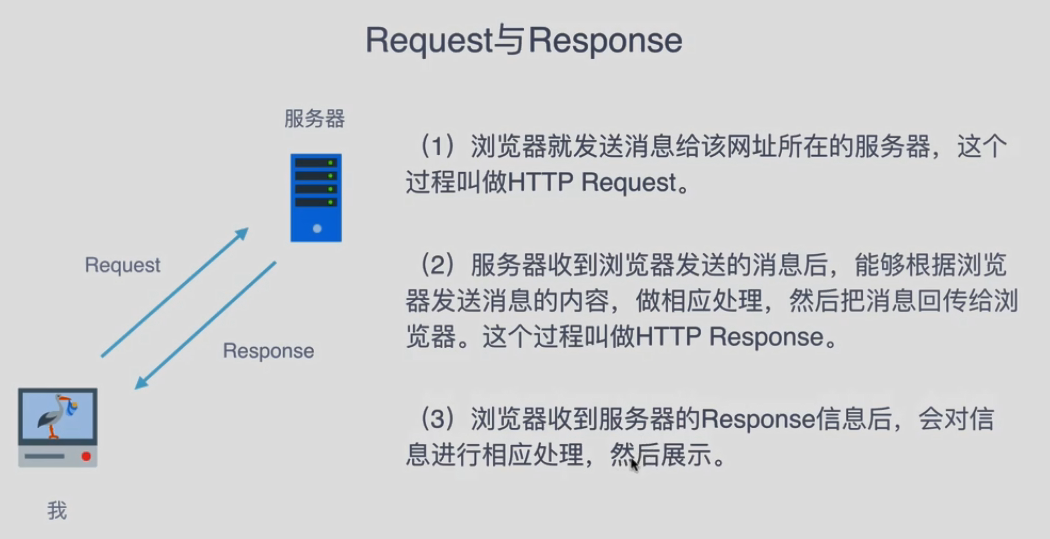
保存数据

自动化程序

### 爬虫基本流程



### Request与Response的概念



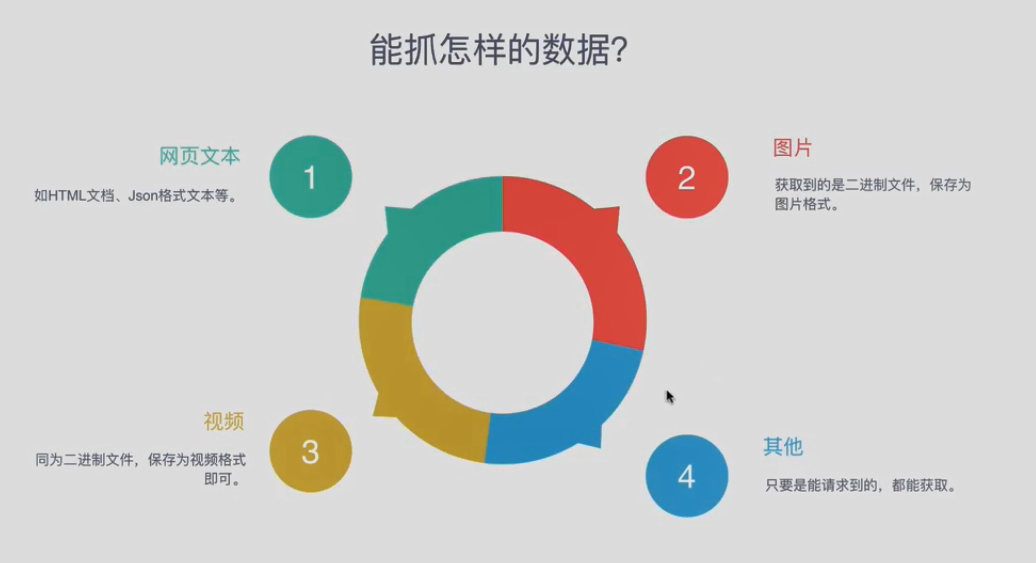
### Request包含的内容



### Response中包含的内容



### 能抓怎样的数据



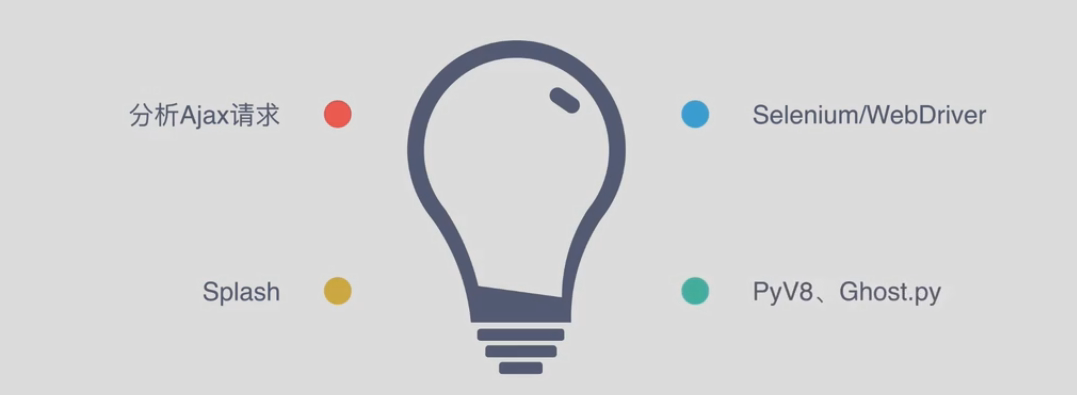
### 解析方式



### 为什么我抓到的和浏览器看到的不一样?

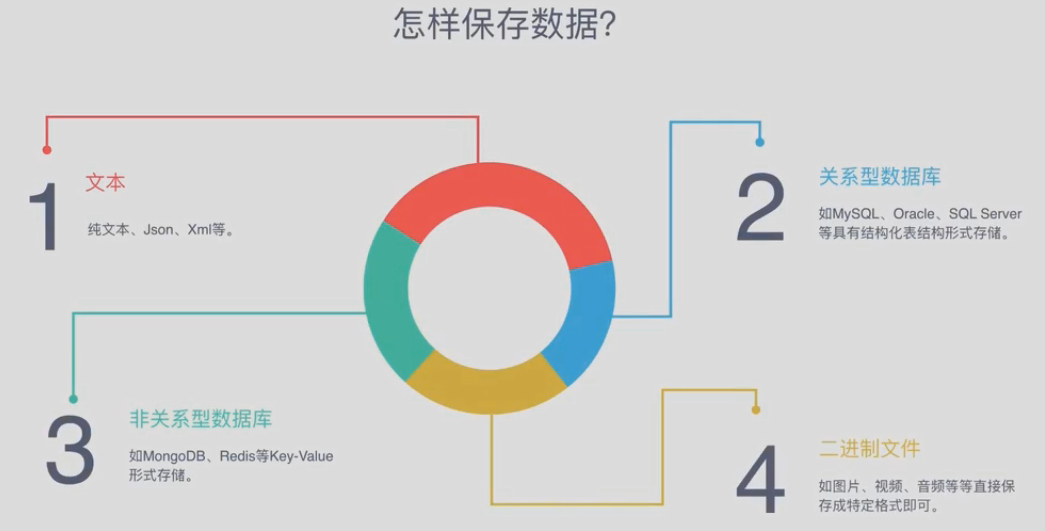
因为页面都是经过JavaScript后期加载渲染的，并不是单纯request请求回来的第一次内容

### 怎样解决JavaScript渲染的问题



使用selenium获取的页面原代码不是request当前URL的原代码，而是审查元素一样的，你能看到的内容组织起来的源代码

### 可以怎样保存数据?



### 代理的基本原理

由于爬虫目的网站因为访问频率太高而封禁IP访问，出现如403 Forbidden的问题，为了解决服务器的对某个IP单位时间的请求次数，我们采用代理IP的形式来做IP伪装.

代理，实际上指的的是代理服务器,功能是代理网络用户去取得网络信息。目的地服务器识别出的IP地址不再是我们本机的真实IP地址，而是代理服务器的IP地址，这样能够成功实现伪装.

#### 代理的作用

1. 突破自身IP访问限制，访问平时不能访问的站点(如:翻墙)
2. 访问单位和团体内部资源(如:教育网FTP)
3. 提高访问速度(代理服务器一般设置一个较大的硬盘缓冲区，用于保存曾经访问过的内容，如果当前访问的内容是在缓冲区中存在的，那么直接读取缓冲内容并返回)
4. 隐藏真实的IP,防止自身的IP被封禁

#### 爬虫代理

爬取过程中通过不能跟换代理，突破网站的IP封锁，能达到很好的爬虫效果

#### 代理分类

根据协议区分

1. FTP代理服务器(端口一般是:21,2121)
2. HTTP代理服务器(端口一般是:8080,3128)
3. SSL/TLS代理(主要访问加密网站，一般有SSL/TLS加密功能，端口一般为443)
4. RTSP代理(主要用于访问Real流媒体服务群，端口一般为554)
5. Telnet代理(一般用于telnet远程控制(黑客隐藏身份常用，端口23)
6. SOCK代理(单纯传递数据包，不关心协议和用法，常用端口1080，SOCK4仅支持TCP，SOCK5支持TCP和UDP，还支持身份校验)

根据匿名程度区分

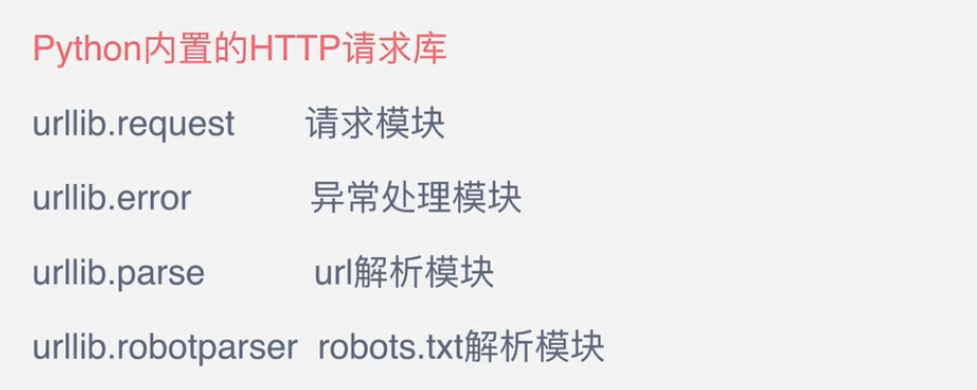
1. 高匿名代理(会将数据包原封不动的转发，目标服务器记录的IP是代理服务器的IP)
2. 普通匿名代理(会在数据包中做改动，让目标服务器发现到访问的是代理服务器，有几率查到真实IP)
3. 透明代理(不但改动数据包，还会告诉目标服务器的真实IP，一般用于透传与加速)
4. 间谍代理(指组织或者个人创建的用于记录用户传输的数据，然后进行研究，监控等目的的代理服务器，这种就是木马的目的器)

#### 常见的代理设置

1. 使用网上的免费代理(可用的不多，最好使用高匿名代理，需要在使用前筛选一下，进一步的话可以维护一个代理池)
2. 使用付费代理服务(互联网上存在许多代理商)
3. ADSL拨号(拨号一次换一次IP,稳定性高，也是一种比较有效的解决方案)

## Urllib库详解

### 什么是Urllib



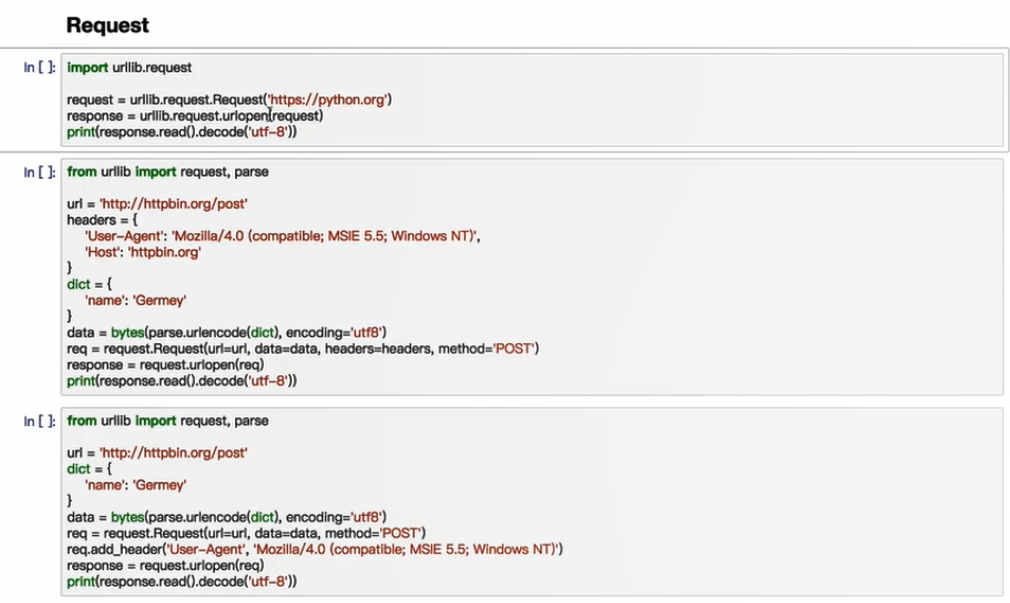
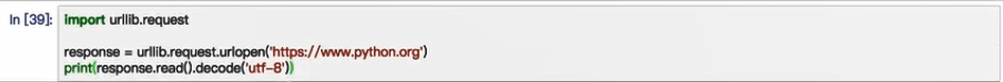
### 相关方法

Urlopen



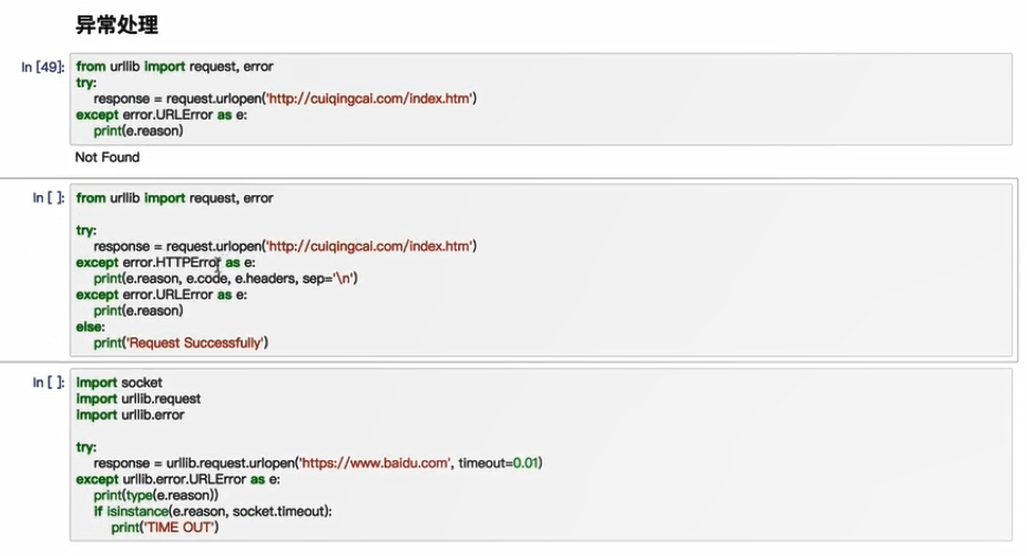


Decode设定制定的字符解析集

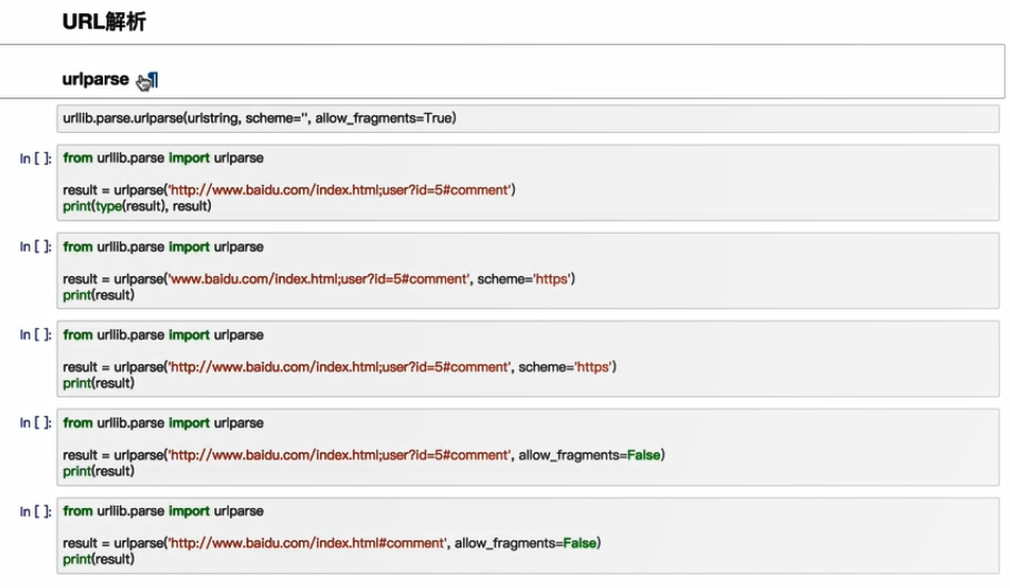
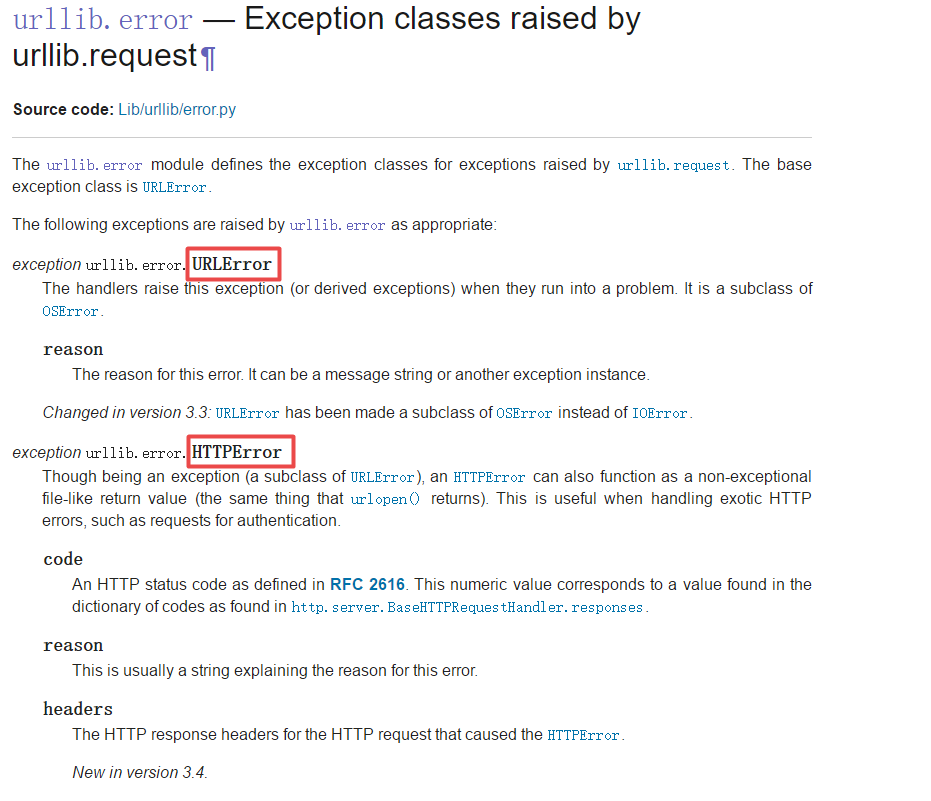


Cookies可以维持有状态访问

第三第四个例子是使用别的cookiejar来保存和读取cookie两个例子

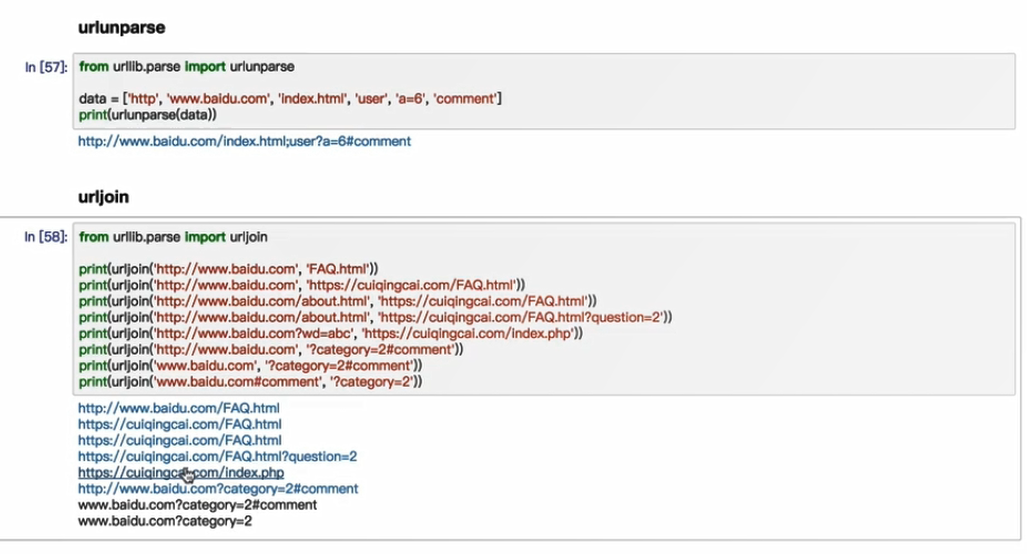


一般来说异常有两个，URLError和它的子类HTTPError



上图备注:

1. scheme：如果地址中不含有协议信息的时候，scheme的选项才有作用，若原来的请求字符串已经包含请求协议，那么scheme字段自动忽略



上图备注:

1. urljoin方法的中地址以后面地址为基准



上图备注:

1. urlencode方法是将params参数拼合成为get方法的请求参数,由于很多的请求参数都是使用字典的形式进行存储的，所以在get方法是需要参数拼接，urlencode方法可以解决这个问题

