爬虫基础

主讲：孙国元

华信培训

本章要点

- Web基础
- 爬虫基本原理

1

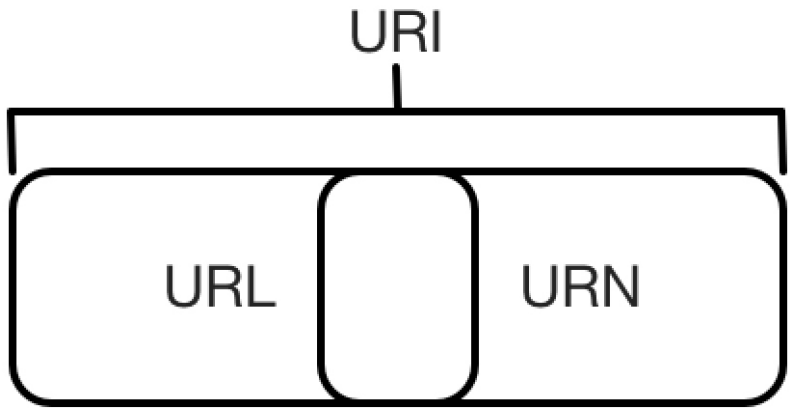
Web基础

URI和URL

- URI的全称为Uniform Resource Identifier，即统一资源标志符，URL的全称为Universal Resource Locator，即统一资源定位符。https://github.com/favicon.ico是GitHub的网站图标链接，它是一个URL，也是一个URI。即有这样的一个图标资源，我们用URL/URI来唯一指定了它的访问方式，这其中包括了访问协议https、访问路径（/即根目录）和资源名称favicon.ico。通过这样一个链接，我们便可以从互联网上找到这个资源，这就是URL/URI。
- URL是URI的子集，也就是说每个URL都是URI，但不是每个URI都是URL。那么，怎样的URI不是URL呢？URI还包括一个子类叫作URN，它的全称为Universal Resource Name，即统一资源名称。URN只命名资源而不指定如何定位资源，比如urn:isbn:0451450523指定了一本书的ISBN，可以唯一标识这本书，但是没有指定到哪里定位这本书，这就是URN。

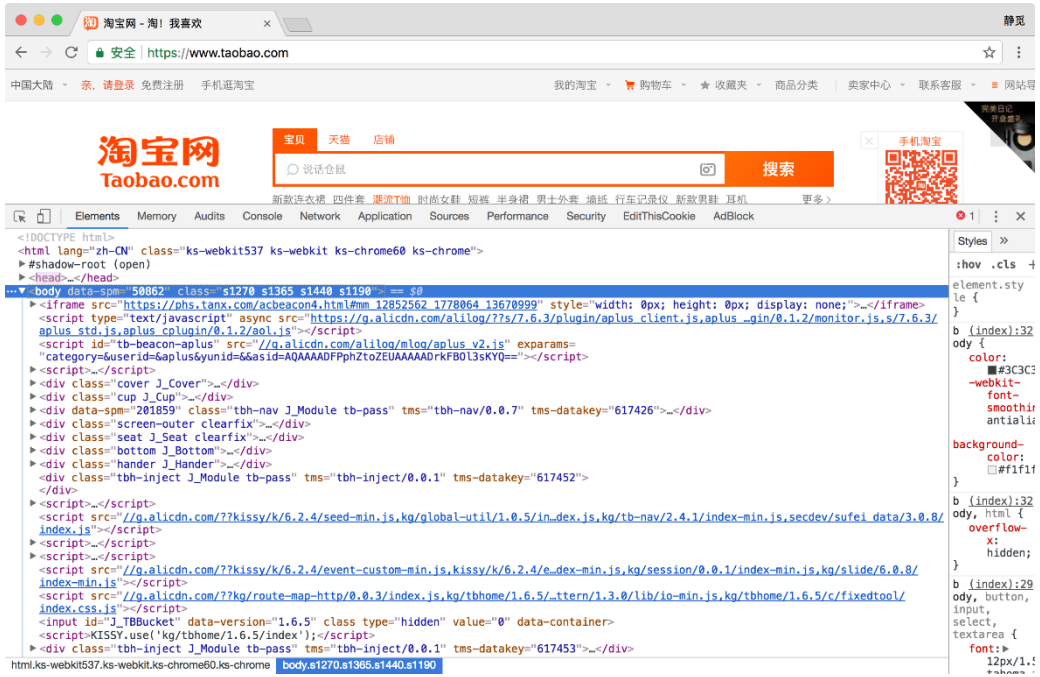
URI和URL

- 在目前的互联网中，URN用得非常少，所以几乎所有的URI都是URL，一般的网页链接我们既可以称为URL，也可以称为URI。



超文本

- 超文本，其英文名称叫作hypertext，我们在浏览器里看到的网页就是超文本解析而成的，其网页源代码是一系列HTML代码，里面包含了一系列标签，比如img显示图片，p指定显示段落等。浏览器解析这些标签后，便形成了我们平常看到的网页，而网页的源代码HTML就可以称作超文本。



HTTP和HTTPS

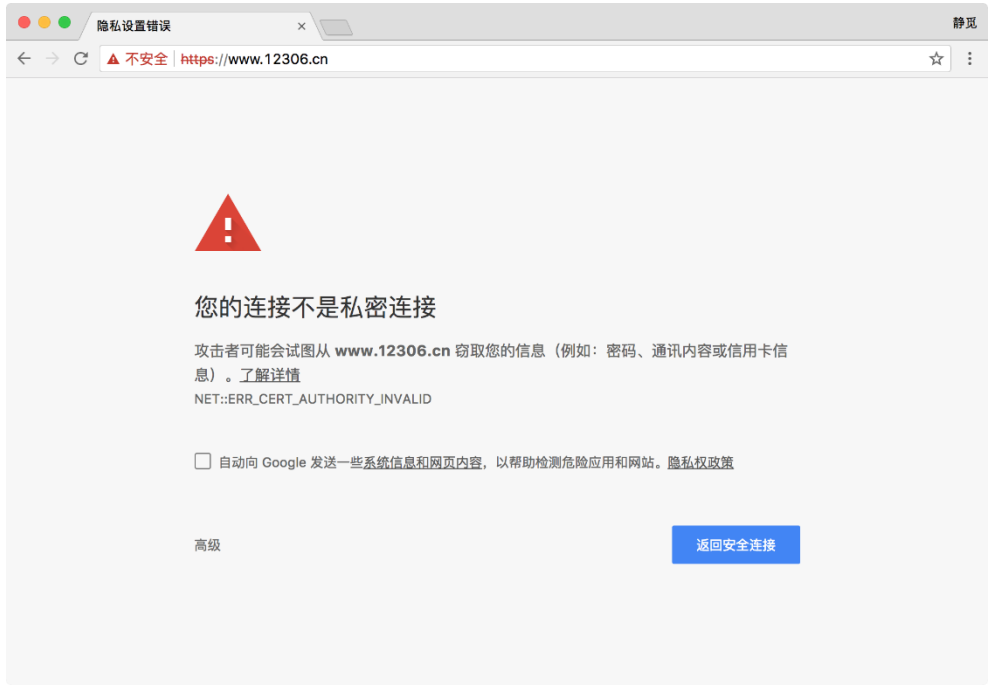
- 在淘宝的首页https://www.taobao.com/中，URL的开头会有http或https，这就是访问资源需要的协议类型。有时，我们还会看到ftp、sftp、smb开头的URL，它们都是协议类型。
- HTTP的全称是Hyper Text Transfer Protocol，中文名叫作超文本传输协议。HTTP协议是用于从网络传输超文本数据到本地浏览器的传送协议，它能保证高效而准确地传送超文本文档。HTTP由万维网协会（World Wide Web Consortium）和Internet工作小组IETF（Internet Engineering Task Force）共同合作制定的规范，目前广泛使用的是HTTP 1.1版本。

HTTP和HTTPS

- HTTPS的全称是Hyper Text Transfer Protocol over Secure Socket Layer，是以安全为目标的HTTP通道，简单讲是HTTP的安全版，即HTTP下加入SSL层，简称为HTTPS。
- HTTPS的安全基础是SSL，因此通过它传输的内容都是经过SSL加密的，它的主要作用可以分为两种：
  - 建立一个信息安全通道来保证数据传输的安全。
  - 确认网站的真实性，凡是使用了HTTPS的网站，都可以通过点击浏览器地址栏的锁头标志来查看网站认证之后的真实信息，也可以通过CA机构颁发的安全签章来查询。

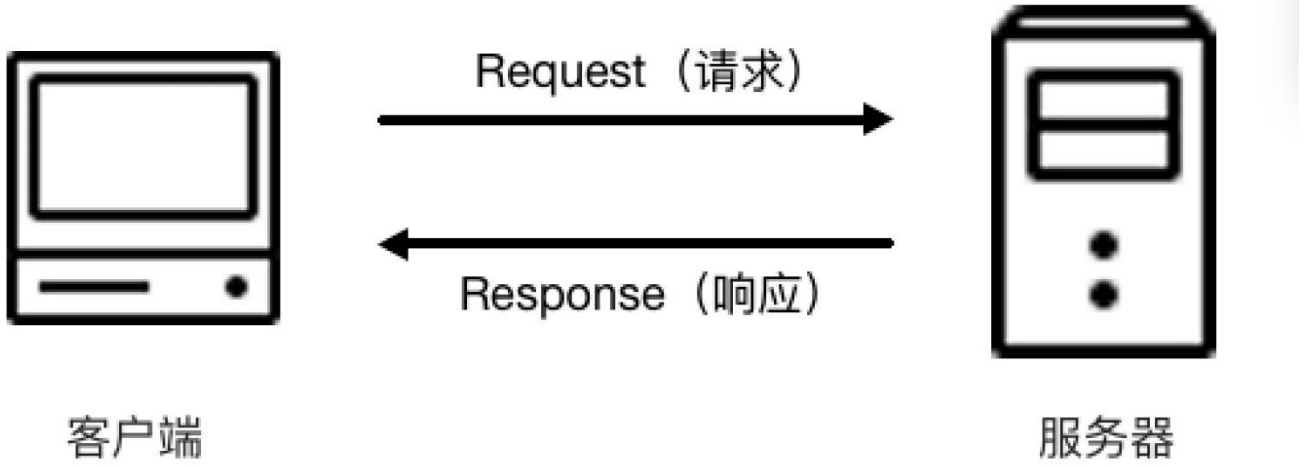
HTTP和HTTPS

- 某些网站虽然使用了HTTPS协议，但还是会被浏览器提示不安全，例如我们在Chrome浏览器里面打开12306，链接为：https://www.12306.cn/，这时浏览器就会提示“您的连接不是私密连接”这样的话，这是因为12306的CA证书是中国铁道部自行签发的，而这个证书是不被CA机构信任的，所以这里证书验证就不会通过而提示这样的话，但是实际上它的数据传输依然是经过SSL加密的。如果要爬取这样的站点，就需要设置忽略证书的选项，否则会提示SSL链接错误。



HTTP请求过程

- 在浏览器中输入一个URL，回车之后便会在浏览器中观察到页面内容。实际上，这个过程是浏览器向网站所在的服务器发送了一个请求，网站服务器接收到这个请求后进行处理和解析，然后返回对应的响应，接着传回给浏览器。响应里包含了页面的源代码等内容，浏览器再对其进行解析，便将网页呈现了出来



请求

- 请求，由客户端向服务端发出，可以分为4部分内容：请求方法（Request Method）、请求的网址（Request URL）、请求头（Request Headers）、请求体（Request Body）。

- (1) 请求方法
  - 常见的请求方法有两种：GET和POST。
  - GET请求中的参数包含在URL里面，数据可以在URL中看到，而POST请求的URL不会包含这些数据，数据都是通过表单形式传输的，会包含在请求体中。
  - GET请求提交的数据有限制，而POST方式没有限制。

请求

- (2) 请求的网址
  - 请求的网址，即统一资源定位符URL，它可以唯一确定我们想请求的资源。

请求

- (3) 请求头
  - 请求头，用来说明服务器要使用的附加信息，比较重要的信息有Cookie、Referer、User-Agent等。下面简要说明一些常用的头信息。

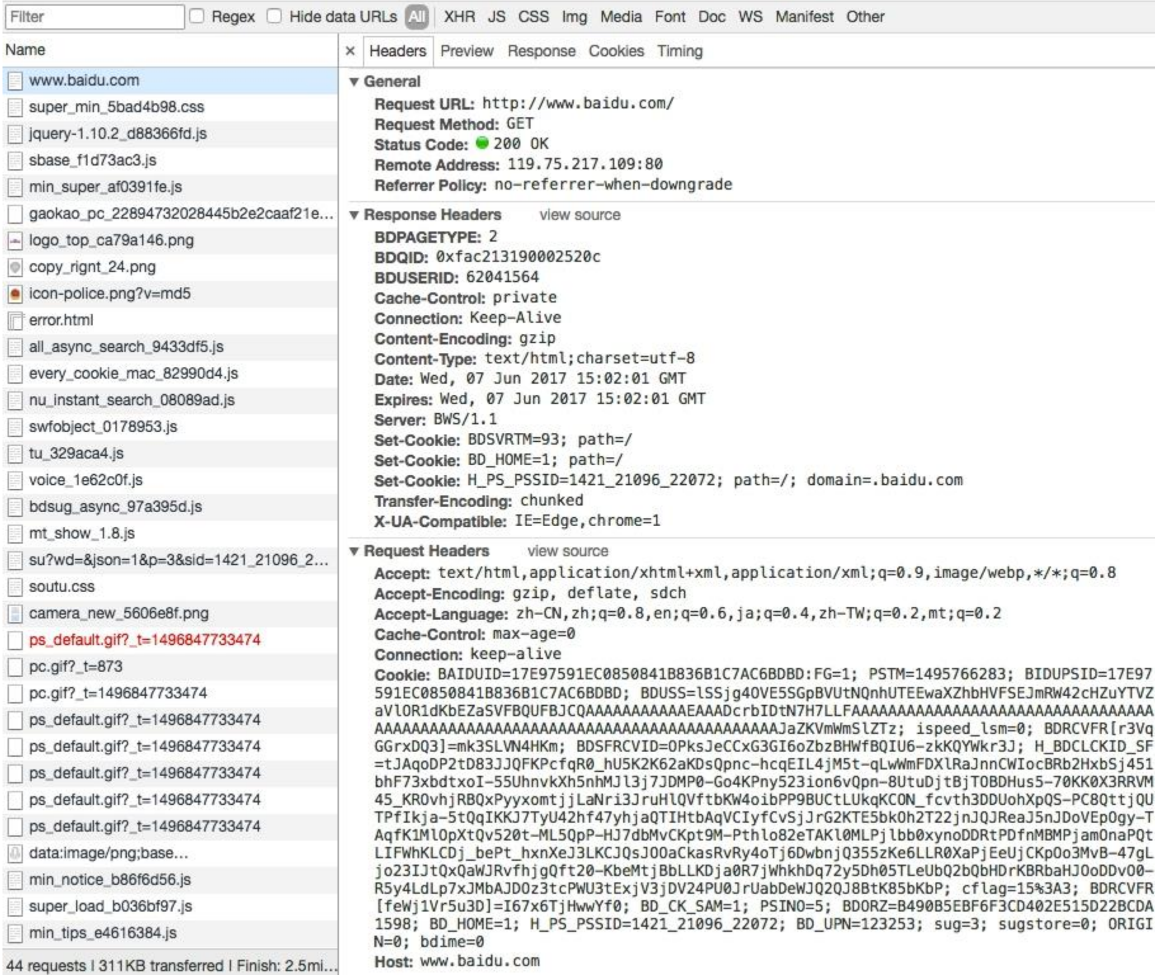
  - Accept：请求报头域，用于指定客户端可接受哪些类型的信息。
  - Accept-Language：指定客户端可接受的语言类型。
  - Accept-Encoding：指定客户端可接受的内容编码。
  - Host：用于指定请求资源的主机IP和端口号。
  - Cookies：为了辨别用户进行会话跟踪而存储在用户本地的数据。
  - Referer：此内容用来标识这个请求是从哪个页面发过来的，服务器可以拿到这一信息并做相应的处理，如作来源统计、防盗链处理等。

请求

- (3) 请求头
  - User-Agent：简称UA，它是一个特殊的字符串头，可以使服务器识别客户使用的操作系统及版本、浏览器及版本等信息。在做爬虫时加上此信息，可以伪装为浏览器；如果不加，很可能会被识别出为爬虫。
  - Content-Type：也叫互联网媒体类型（Internet Media Type）或者MIME类型，在HTTP协议消息头中，它用来表示具体请求中的媒体类型信息。例如，text/html代表HTML格式，image/gif代表GIF图片，application/json代表JSON类型，更多对应关系可以查看此对照表：http://tool.oschina.net/commons。

请求

- (4) 请求体
  - 请求体一般承载的内容是POST请求中的表单数据，而对于GET请求，请求体则为空。



请求

- Content-Type和POST提交数据方式的关系

Content-Type	提交数据的方式
application/x-www-form-urlencoded	表单数据
multipart/form-data	表单文件上传
application/json	序列化JSON数据
text/xml	XML数据

响应

- 响应，由服务端返回给客户端，可以分为三部分：响应状态码（Response Status Code）、响应头（Response Headers）和响应体（Response Body）。

- (1) 响应状态码
  - 响应状态码表示服务器的响应状态，如200代表服务器正常响应，404代表页面未找到，500代表服务器内部发生错误。

响应

- (2) 响应头
  - 响应头包含了服务器对请求的应答信息，如Content-Type、Server、Set-Cookie等。下面简要说明一些常用的头信息。

  - Date：标识响应产生的时间。
  - Last-Modified：指定资源的最后修改时间。
  - Content-Encoding：指定响应内容的编码。
  - Server：包含服务器的信息，比如名称、版本号等。
  - Content-Type：文档类型。
  - Set-Cookie：设置Cookies。。
  - Expires：指定响应的过期时间。

响应

- (3) 响应体
  - 响应的正文数据都在响应体中，比如请求网页时，它的响应体就是网页的HTML代码；请求一张图片时，它的响应体就是图片的二进制数据。

网页基础-HTML

- HTML是用来描述网页的一种语言，其全称叫作Hyper Text Markup Language，即超文本标记语言。网页包括文字、按钮、图片和视频等各种复杂的元素，其基础架构就是HTML。不同类型的文字通过不同类型的标签来表示，如图片用img标签表示，视频用video标签表示，段落用p标签表示，它们之间的布局又常通过布局标签div嵌套组合而成，各种标签通过不同的排列和嵌套才形成了网页的框架。

网页基础-CSS

- HTML定义了网页的结构，但是只有HTML页面的布局并不美观，可能只是简单的节点元素的排列，为了让网页看起来更好看一些，这里借助了CSS。
- 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、元素间距、排列等格式。

```
#head_wrapper.s-ps-islite .s-p-top {
    position: absolute;
    bottom: 40px;
    width: 100%;
    height: 181px;
}
```

网页基础-JavaScript

- avaScript，简称JS，是一种脚本语言。HTML和CSS配合使用，提供给用户的只是一种静态信息，缺乏交互性。我们在网页里可能会看到一些交互和动画效果，如下载进度条、提示框、轮播图等，这通常就是JavaScript的功劳。它的出现使得用户与信息之间不只是一种浏览与显示的关系，而是实现了一种实时、动态、交互的页面功能。
- JavaScript通常也是以单独的文件形式加载的，后缀为js，在HTML中通过script标签即可引入

```
<script src="jquery-2.1.0.js"></script>
```

网页基础-基本结构

```
<!DOCTYPE html>
<html>
    <head>
        <meta charset="UTF-8">
        <title>This is a Demo</title>
    </head>
    <body>
        <div id="container">
            <div class="wrapper">
                <h2 class="title">Hello World</h2>
                <p class="text">Hello, this is a paragraph.</p>
            </div>
        </div>
    </body>
</html>
```

网页基础-DOM

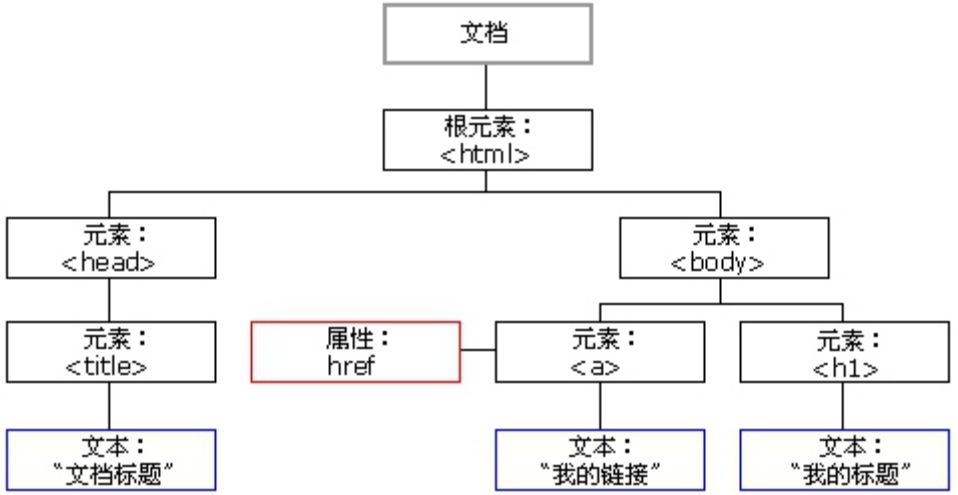
- 在HTML中，所有标签定义的内容都是节点，它们构成了一个HTML DOM树。DOM是W3C（万维网联盟）的标准，其英文全称Document Object Model，即文档对象模型。它定义了访问HTML和XML文档的标准：W3C文档对象模型（DOM）是中立于平台和语言的接口，它允许程序和脚本动态地访问和更新文档的内容、结构和样式。
- W3C DOM标准被分为3个不同的部分。
  - 核心DOM： 针对任何结构化文档的标准模型。
  - XML DOM：针对XML文档的标准模型。
  - HTML DOM：针对HTML文档的标准模型。

网页基础-DOM

- 根据W3C的HTML DOM标准，HTML文档中的所有内容都是节点。
  - 整个文档是一个文档节点；
  - 每个HTML元素是元素节点；
  - HTML元素内的文本是文本节点；
  - 每个HTML属性是属性节点；
  - 注释是注释节点。

网页基础-DOM

- HTML DOM将HTML文档视作树结构，这种结构被称为节点树
- 通过HTML DOM，树中的所有节点均可通过JavaScript访问，所有HTML节点元素均可被修改，也可以被创建或删除。
- 节点树中的节点彼此拥有层级关系。我们常用父（parent）、子（child）和兄弟（sibling）等术语描述这些关系。父节点拥有子节点，同级的子节点被称为兄弟节点。



2

爬虫基本原理

爬虫概述

- 可以把互联网比作一张大网，而爬虫（即网络爬虫）便是在网上爬行的蜘蛛。把网的节点比作一个个网页，爬虫爬到这就相当于访问了该页面，获取了其信息。可以把节点间的连线比作网页与网页之间的链接关系，这样蜘蛛通过一个节点后，可以顺着节点连线继续爬行到达下一个节点，即通过一个网页继续获取后续的网页，这样整个网的节点便可以被蜘蛛全部爬行到，网站的数据就可以被抓取下来了。

爬虫概述

- (1) 获取网页
  - 爬虫首先要做的工作就是获取网页，这里就是获取网页的源代码。源代码里包含了网页的部分有用信息，所以只要把源代码获取下来，就可以从中提取想要的信息了。
  - Python提供了许多库来帮助我们实现这个操作，如urllib、requests等。我们可以用这些库来帮助我们实现HTTP请求操作，请求和响应都可以用类库提供的数据结构来表示，得到响应之后只需要解析数据结构中的Body部分即可，即得到网页的源代码，这样我们可以用程序来实现获取网页的过程了。

爬虫概述

- (2) 提取信息
  - 获取网页源代码后，接下来就是分析网页源代码，从中提取我们想要的数据。首先，最通用的方法便是采用正则表达式提取，这是一个万能的方法，但是在构造正则表达式时比较复杂且容易出错。
  - 另外，由于网页的结构有一定的规则，所以还有一些根据网页节点属性、CSS选择器或XPath来提取网页信息的库，如Beautiful Soup、pyquery、lxml等。使用这些库，我们可以高效快速地从中提取网页信息，如节点的属性、文本值等。

爬虫概述

- (3) 保存数据
  - 提取信息后，我们一般会将提取到的数据保存到某处以便后续使用。这里保存形式有多种多样，如可以简单保存为TXT文本或JSON文本，也可以保存到数据库，如MySQL和MongoDB等。

能抓怎样的数据

- 在网页中我们能看到各种各样的信息，最常见的便是常规网页，它们对应着HTML代码，而最常抓取的便是HTML源代码。
- 另外，可能有些网页返回的不是HTML代码，而是一个JSON字符串（其中API接口大多采用这样的形式），这种格式的数据方便传输和解析，它们同样可以抓取，而且数据提取更加方便。
- 此外，我们还可以看到各种二进制数据，如图片、视频和音频等。利用爬虫，我们可以将这些二进制数据抓取下来，然后保存成对应的文件名。
- 另外，还可以看到各种扩展名的文件，如CSS、JavaScript和配置文件等，这些其实也是最普通的文件，只要在浏览器里面可以访问到，就可以将其抓取下来。

静态网页和动态网页

- 网页的内容是HTML代码编写的，文字、图片等内容均通过写好的HTML代码来指定，这种页面叫作静态网页。
- 动态网页可以动态解析URL中参数的变化，关联数据库并动态呈现不同的页面内容，非常灵活多变。我们现在遇到的大多数网站都是动态网站，它们不再是一个简单的HTML，而是可能由JSP、PHP、Python等语言编写。

无状态HTTP

- HTTP的无状态是指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。当我们向服务器发送请求后，服务器解析此请求，然后返回对应的响应，服务器负责完成这个过程，而且这个过程是完全独立的，服务器不会记录前后状态的变化，也就是缺少状态记录。
- 这时两个用于保持HTTP连接状态的技术就出现了，它们分别是会话和Cookies。会话在服务端，也就是网站的服务器，用来保存用户的会话信息；Cookies在客户端，也可以理解为浏览器端，有了Cookies，浏览器在下次访问网页时会自动附带上它发送给服务器，服务器通过识别Cookies并鉴定出是哪个用户，然后再判断用户是否是登录状态，然后返回对应的响应。

会话

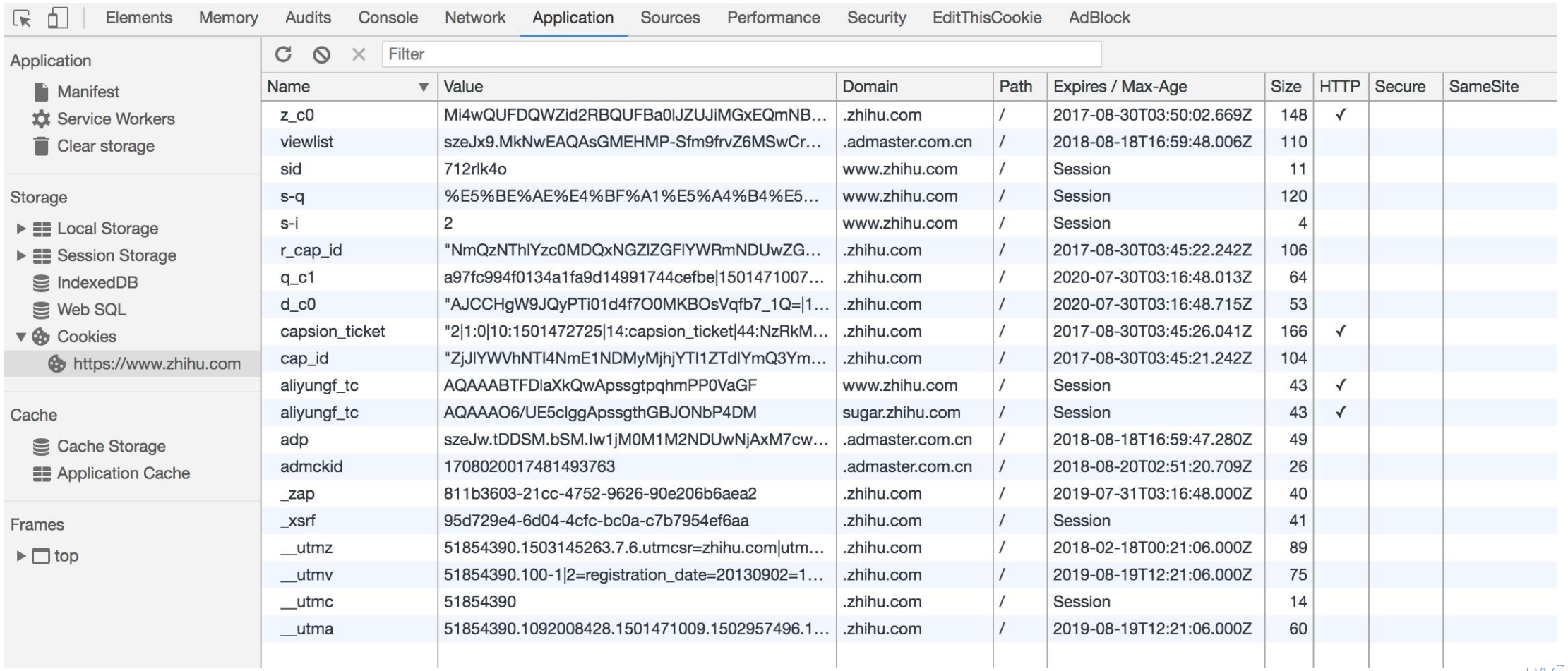
- 会话，其本来的含义是指有始有终的一系列动作/消息。
- 在Web中，会话对象用来存储特定用户会话所需的属性及配置信息。这样，当用户在应用程序的Web页之间跳转时，存储在会话对象中的变量将不会丢失，而是在整个用户会话中一直存在下去。当用户请求来自应用程序的Web页时，如果该用户还没有会话，则Web服务器将自动创建一个会话对象。当会话过期或被放弃后，服务器将终止该会话。

Cookies

- Cookies指某些网站为了辨别用户身份、进行会话跟踪而存储在用户本地终端上的数据。
- 当客户端第一次请求服务器时，服务器会返回一个请求头中带有Set-Cookie字段的响应给客户端，用来标记是哪一个用户，客户端浏览器会把Cookies保存起来。当浏览器下一次再请求该网站时，浏览器会把此Cookies放到请求头一起提交给服务器，Cookies携带了会话ID信息，服务器检查该Cookies即可找到对应的会话是什么，然后再判断会话来以此来辨认用户状态。

Cookies

- 属性结构



Name	Value	Domain	Path	Expires / Max-Age	Size	HTTP	Secure	SameSite
z_c0	Mi4wQUFDQWZid2RBQUFBa0lJZUJiMGxEQmNB…	.zhihu.com	/	2017-08-30T03:50:02.669Z	148	✓		
viewlist	szeJx9.MkNwEAQAsGMEHMP-Sfm9frvZ6MSwCr…	.admaster.com.cn	/	2018-08-18T16:59:48.006Z	110			
sid	712rlk4o	www.zhihu.com	/	Session	11			
s-q	%E5%BE%AE%E4%BF%A1%E5%A4%B4%E5…	www.zhihu.com	/	Session	120			
s-i	2	www.zhihu.com	/	Session	4			
r_cap_id	"NmQzNThlYzc0MDQxNGZlZGFlYWRmNDUwZG…	.zhihu.com	/	2017-08-30T03:45:22.242Z	106			
q_c1	a97fc994f0134a1fa9d14991744cefbe|1501471007…	.zhihu.com	/	2020-07-30T03:16:48.013Z	64			
d_c0	"AJCCHgW9JQyPTi01d4f7O0MKBOsVqfb7_1Q=|1…	.zhihu.com	/	2020-07-30T03:16:48.715Z	53			
capsion_ticket	"2|1:0|10:1501472725|14:capsion_ticket|44:NzRkM…	.zhihu.com	/	2017-08-30T03:45:26.041Z	166	✓		
cap_id	"ZjJlYWVhNTI4NmE1NDMyMjhjYTI1ZTdlYmQ3Ym…	.zhihu.com	/	2017-08-30T03:45:21.242Z	104			
aliyungf_tc	AQAAABTFDlaXkQwApssgtpqhmPP0VaGF	www.zhihu.com	/	Session	43	✓		
aliyungf_tc	AQAAAO6/UE5clggApssgthGBJONbP4DM	sugar.zhihu.com	/	Session	43	✓		
adp	szeJw.tDDSM.bSM.Iw1jM0M1M2NDUwNjAxM7cw…	.admaster.com.cn	/	2018-08-18T16:59:47.280Z	49			
admckid	1708020017481493763	.admaster.com.cn	/	2018-08-20T02:51:20.709Z	26			
_zap	811b3603-21cc-4752-9626-90e206b6aea2	.zhihu.com	/	2019-07-31T03:16:48.000Z	40			
_xsrf	95d729e4-6d04-4cfc-bc0a-c7b7954ef6aa	.zhihu.com	/	Session	41			
__utmz	51854390.1503145263.7.6.utmcsr=zhihu.com|utm…	.zhihu.com	/	2018-02-18T00:21:06.000Z	89			
__utmv	51854390.100-1|2=registration_date=20130902=1…	.zhihu.com	/	2019-08-19T12:21:06.000Z	75			
__utmc	51854390	.zhihu.com	/	Session	14			
__utma	51854390.1092008428.1501471009.1502957496.1…	.zhihu.com	/	2019-08-19T12:21:06.000Z	60			

Cookies

- 属性结构
  - Name：该Cookie的名称。
  - Value：该Cookie的值。
  - Domain：可以访问该Cookie的域名。
  - Max Age：该Cookie失效的时间，单位为秒，也常和Expires一起使用，通过它可以计算出其有效时间。Max Age如果为正数，则该Cookie在Max Age秒之后失效。如果为负数，则关闭浏览器时Cookie即失效，浏览器也不会以任何形式保存该Cookie。
  - Path：该Cookie的使用路径。如果设置为/path/，则只有路径为/path/的页面可以访问该Cookie。如果设置为/，则本域名下的所有页面都可以访问该Cookie。
  - Size字段：此Cookie的大小。
  - HTTP字段：Cookie的httponly属性。
  - Secure：该Cookie是否仅被使用安全协议传输。默认为false。

代理的基本原理

- 在做爬虫的过程中经常会遇到这样的情况，最初爬虫正常运行，正常抓取数据，一切看起来都是那么美好，然而一杯茶的功夫可能就会出现错误，比如403 Forbidden，这时候打开网页一看，可能会看到“您的IP访问频率太高”这样的提示。出现这种现象的原因是网站采取了一些反爬虫措施。比如，服务器会检测某个IP在单位时间内的请求次数，如果超过了这个阈值，就会直接拒绝服务，返回一些错误信息，这种情况可以称为封IP。

代理的作用

- 代理实际上指的就是代理服务器，英文叫作proxy server，它的功能是代理网络用户去取得网络信息。形象地说，它是网络信息的中转站。在我们正常请求一个网站时，是发送了请求给Web服务器，Web服务器把响应传回给我们。如果设置了代理服务器，实际上就是在本机和服务器之间搭建了一个桥，此时本机不是直接向Web服务器发起请求，而是向代理服务器发出请求，请求会发送给代理服务器，然后由代理服务器再发送给Web服务器，接着由代理服务器再把Web服务器返回的响应转发给本机。

代理的作用

- 突破自身IP访问限制，访问一些平时不能访问的站点。
- 访问一些单位或团体内部资源：比如使用教育网内地址段免费代理服务器，就可以用于对教育网开放的各类FTP下载上传，以及各类资料查询共享等服务。
- 提高访问速度：通常代理服务器都设置一个较大的硬盘缓冲区，当有外界的信息通过时，同时也将其保存到缓冲区中，当其他用户再访问相同的信息时，则直接由缓冲区中取出信息，传给用户，以提高访问速度。
- 隐藏真实IP：上网者也可以通过这种方法隐藏自己的IP，免受攻击。对于爬虫来说，我们用代理就是为了隐藏自身IP，防止自身的IP被封锁。

根据匿名程度区分

- 高度匿名代理：会将数据包原封不动地转发，在服务端看来就好像真的是一个普通客户端在访问，而记录的IP是代理服务器的IP。
- 普通匿名代理：会在数据包上做一些改动，服务端上有可能发现这是个代理服务器，也有一定几率追查到客户端的真实IP。代理服务器通常会加入的HTTP头有HTTP_VIA和HTTP_X_FORWARDED_FOR。
- 透明代理：不但改动了数据包，还会告诉服务器客户端的真实IP。这种代理除了能用缓存技术提高浏览速度，能用内容过滤提高安全性之外，并无其他显著作用，最常见的例子是内网中的硬件防火墙。
- 间谍代理：指组织或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。

常见代理设置

- 使用网上的免费代理：最好使用高匿代理，另外可用的代理不多，需要在使用前筛选一下可用代理，也可以进一步维护一个代理池。
- 使用付费代理服务：互联网上存在许多代理商，可以付费使用，质量比免费代理好很多。
- 宽带拨号：拨一次号换一次IP，稳定性高，也是一种比较有效的解决方案。

本章小结

- Web基础
- 爬虫基本原理

华信培训