

# Ajax 相关知识介绍

## 一、HTTP

### 1.HTTP 概述

- HTTP 就是超文本传输协议，也就是 HyperText Transfer Protocol
- HTTPS 协议是由 HTTP 和 SSL 协议构建的可进行加密传输和身份认证的网络协议，比 HTTP 协议的安全性更高。

以下拆分“超文本”，“传输”，“协议”

#### (1) 协议

- “协议”是需要至少两个或两个遇上的参与方，指定的一种“规范后的约定”
- HTTP 是一个用在计算机世界里的协议。它使用计算机能够理解的语言确立了一种计算机之间交流通信的规范，以及相关的各种控制和错误处理方式。



#### (2) 传输

- 将某些“要求（请求）”从“请求方”发到“响应方”，之后处理后再反馈给“请求方”。中间
- HTTP 是一个在计算机世界里专门用来在两点之间传输数据的约定和规范。

#### (3) 超文本

- 超文本是文字、图片、音频和视频等的连接。
- 经过浏览器的解读，展现的内容多样的页面。
- html 就是超文本标记语言

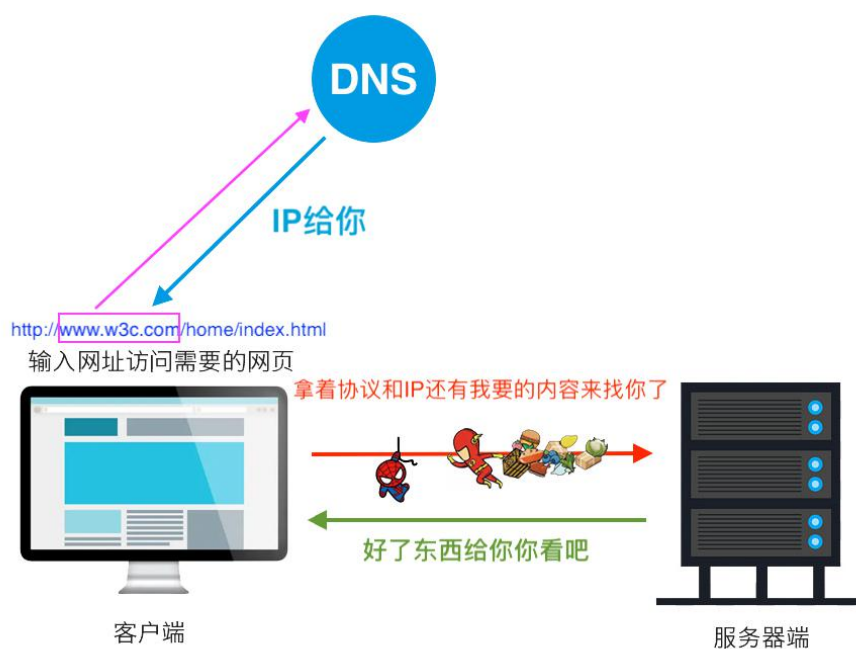
## 2.HTTP 与 Internet 的关系

互联网 Internet，我们通常所说的“上网”实际上访问的只是互联网的一个子集“万维网”（World Wide Web），它基于 HTTP 协议，传输 HTML 等超文本资源，能力也就被限制在 HTTP 协议之内。

## 3.HTTP 协议基本工作流程

HTTP 协议基本工作流程，就是“请求 - 应答”“一发一收”的模式。

在这个过程中参与请求和应答的是服务器端、客户端、http 协议本身。

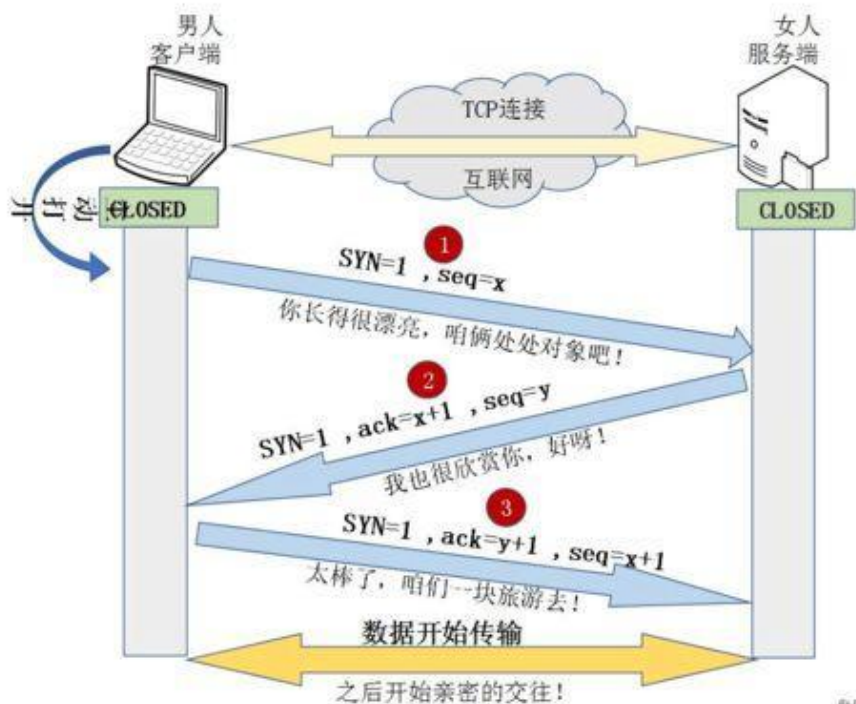


最简单的浏览器 HTTP 请求过程：

1. 浏览器从地址栏的输入中获得服务器的“IP”（后面有名词解释）地址和“端口号”（域名访问需要 DNS 解析）；
2. 浏览器用“TCP”（后面有名词解释）的三次握手与服务器建立连接；
3. 浏览器向服务器发送拼好的报文；
4. 服务器收到报文后处理请求，同样拼好报文再发给浏览器；
5. 浏览器解析报文，渲染输出页面。

## (1) TCP 协议

- TCP 传输控制协议 Transmission Control Protocol
- HTTP 协议是基于 TCP 协议的，TCP 协议是字节流传输，可以保证数据完整
- TCP 位于传输层，作用是提供可靠的字节流服务，为了准确无误地将数据送达目的地，TCP 协议采纳三次握手策略，以下为三次握手的原理：
  - 1/发送端首先发送一个带有 SYN (synchronize) 标志的数据包给接收方。(客户端呼叫服务器端：你好，可以听到吗？)
  - 2/接收方接收后，回传一个带有 SYN/ACK 标志的数据包传递确认信息，表示我收到了。(服务器端回复客户端：收到，你听到我说话了吗？)
  - 3/最后，发送方再回传一个带有 ACK 标志的数据包，代表我知道了，表示‘握手’结束。(客户端：我们都能听到彼此的话，现在可以开始通信了)



## (2) IP 协议

- IP 协议 Internet Protocol
- IP 协议使用 IP 地址的概念来定位互联网上的每一台计算机
- IP 协议 v4，地址是四个用 "." 分隔的数字，例如 "127.0.0.1"，v6，使用 8 组 ":" 分隔的数字作为地址

### (3)DNS

- 域名系统 Domain Name System
- 用有意义的名字来作为 IP 地址的等价替代,假如淘宝的 IP 地址是 192.0.168.1, 用域名 taobao.com 绑定这个 IP。用户访问时访问域名则指向它的 IP 地址

### (4)URL

URL 统一资源定位符 Uniform Resource Locator, 我们俗称的“网址”

- 协议名: 即访问该资源应当使用的协议, `http` 或 `https`、`ftp` 等`
- 主机名: 即互联网上主机的标记, 可以是域名或 IP 地址
- 端口号: 代表服务器上的资源或者特定服务的门牌号
  - 80 HTTP 协议的标准端口号
  - 443 HTTPS
  - 21 FTP 文件传输协议
  - 22 SSH 安全登录、文件传送(SCP)和端口重定向
  - 3306 mysql
  - 5001-65535 用来给用户自定义端口
- 资源路径: 即资源在主机上的位置, 使用 `/` 分隔多级目录
- 参数: 是提供给服务器的额外参数 (要求), 这些参数用 `&` 分隔, 键值对列表
- 资源锚点: 资源中的书签

`http://www.w3cweb.cn:80/poke/query/list.html?id=001&name=妙蛙种子#card`

→ 协议名, http协议 (https安全版协议) 也可以是其他协议

`http://www.w3cweb.cn:80/poke/query/list.html?id=001&name=妙蛙种子#card`

→ 域名, 由DNS服务器将IP转化为域名, 可以在网络上使用, 在互联网上, 域名可以代替IP

`http://www.w3cweb.cn:80/poke/query/list.html?id=001&name=妙蛙种子#card`

→ 端口号 (:必要), 代表服务器上的资源或特定服务的门牌号, http协议的标准端口为80

`http://www.w3cweb.cn:80/poke/query/list.html?id=001&name=妙蛙种子#card`

→ 资源路径, 代表需要访问的资源在服务器上的物理位置

`http://www.w3cweb.cn:80/poke/query/list.html?id=001&name=妙蛙种子#card`

参数, 是提供给网络服务器的额外参数 (要求), 这些参数是用 & 符号分隔的键/值对列表

`http://www.w3cweb.cn:80/poke/query/list.html?id=001&name=妙蛙种子#card`

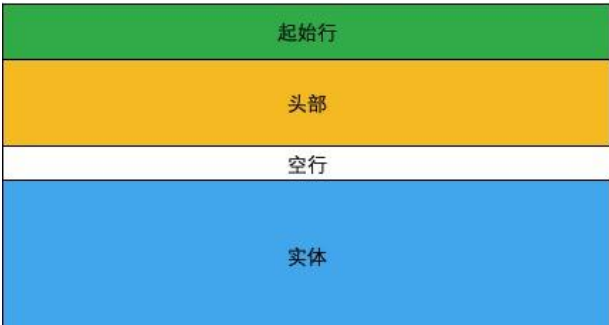
资源本身的另一部分的锚点。锚点表示资源中的一种“书签”, 例如, 在HTML文档上, 浏览器将滚动到定义锚点的位置;在视频或音频文档上, 浏览器将尝试转到锚代表的时间, #后面的部分 (也称为片段标识符) 从来没有发送到请求的服务器。

#### 4. HTTP 协议的报文

HTTP 协议的核心部分就是它传输的“报文”内容, HTTP 协议在规范文档里详细定义了报文的格式, 组成部分, 解析规则, 还有处理策略。

(1) 报文结构:

- 起始行
- 头部
- 空行
- 实体



(3) 起始行

请求行 (request line), 它简要地描述了客户端想要如何操作服务器端的资源。

请求行 :	请求方法	空格	请求目标	空格	版本号	换行		
GET	/cnpc/xhtml/images/public/zyzx011.jpg		HTTP/1.1					

```
Host: www.cnpc.com.cn
Connection: keep-alive
Pragma: no-cache
Cache-Control: no-cache
User-Agent: Mozilla/5.0 (Macintosh; Intel Mac OS X 10_14_6)
AppleWebKit/537.36 (KHTML, like Gecko) Chrome/91.0.4472.114 Safari/537.36
DNT: 1
Accept: image/avif,image/webp,image/apng,image/svg+xml,image/*,*/*;q=0.8
Referer: http://www.cnpc.com.cn/cnpc/index.shtml
Accept-Encoding: gzip, deflate
Accept-Language: en,zh-CN;q=0.9,zh;q=0.8
```

请求行由三部分构成:

1. 请求方法: 是一个动词, 如 GET/POST, 表示对资源的操作;
2. 请求目标: 通常是一个 URI, 标记了请求方法要操作的资源;
3. 版本号: 表示报文使用的 HTTP 协议版本。

状态行 (status line), 是服务器响应的状态。

状态行 :	版本号	空格	状态码	空格	原因	换行		
HTTP/1.1	200		OK					

```
Server: *****
Date: Wed, 23 Jun 2021 13:53:47 GMT
Content-Type: image/jpeg
Content-Length: 19822
Last-Modified: Tue, 15 Jun 2021 01:50:09 GMT
Connection: keep-alive
ETag: "60c80751-4d6e"
Content-Security-Policy: frame-ancestors http://*.cnpc/
```



- 1. 版本号：表示报文使用的 HTTP 协议版本；
- 2. 状态码：一个三位数，用代码的形式表示处理的结果，比如 200 是成功，404 没找到资源，500 是服务器错误；
- 3. 原因：作为数字状态码补充，是更详细的解释文字，帮助人理解原因。如果状态行返回`HTTP/1.1 404 Not Found`意思是没找到你要的资源，你自己处理吧。

(4) 头部字段

请求行或状态行再加上头部字段集合就构成了 HTTP 报文里完整的请求头或响应头。

请求头和响应头的结构是基本一样的，唯一的区别是起始行。

请求行：	请求方法	空格	请求目标	空格	版本号	换行		状态行：	版本号	空格	状态码	空格	原因	换行
请求头	属性1	:	值			换行		响应头	属性1	:	值			换行
	属性2	:	值			换行			属性2	:	值			换行
	属性3	:	值			换行			属性3	:	值			换行
	*****								*****					
空行	空行							空行	空行					

5. HTTP 请求方法

(1) 浏览器请求方法

HTTP/1.1 规定了八种方法，由客户端“请求”服务器来完成。注意，单词都必须是大写的形式。

- GET：从服务器获取资源，这个资源既可以是静态的文本、页面、图片、视频，也可以是由 PHP、Java 动态生成的页面或者其他格式的数据。
- HEAD：是 GET 方法的“简化版”，也是请求从服务器获取资源，服务器的处理机制也是一样的，但服务器不会返回请求的实体数据，只会传回响应头，也就是资源的“元信息”。比如，想要检查一个文件是否存在，只要发个 HEAD 请求。
- POST：向资源提交数据，相当于写入或上传数据；
- PUT：类似 POST，通常 POST 表示的是“新建”的含义，而 PUT 则是“修改”的含义。
- DELETE：删除资源；
- CONNECT：建立特殊的连接隧道；
- OPTIONS：列出可对资源实行的方法；
- TRACE：追踪请求 - 响应的传输路径。

需要注意的是：

- 其中常用方法是：GET、POST。
- GET 和 HEAD 方法是“只读”操作。
- POST、PUT、DELETE 操作会修改服务器上的资源。

## (2) 服务器拒绝请求

浏览器的要求服务器可以同意可以拒绝。比如，浏览器发起了一个 GET 请求，想获取“/mimi”这个文件，但这个文件保密级别比较高，服务器就可以有如下的几种响应方式：

1. 假装这个文件不存在，直接返回一个 404 Not found 没找到资源；
2. 稍微友好一点，明确告诉你有这个文件，但不允许访问，返回一个 403 Forbidden 被禁止的；
3. 再宽松一些，返回 405 Method Not Allowed，不允许的方法。

**不是你（浏览器）要我（服务器）就会给你**

## (3) 状态码

HTTP 状态码表示客户端 HTTP 请求的返回结果、标记服务器端的处理是否正常或者是出现的错误，能够根据返回的状态码判断请求是否得到正确的处理很重要。

状态码由 3 位数字和原因短语组成，数字中的第一位指定了响应类别，后两位无分类，响应类别有以下 5 种：

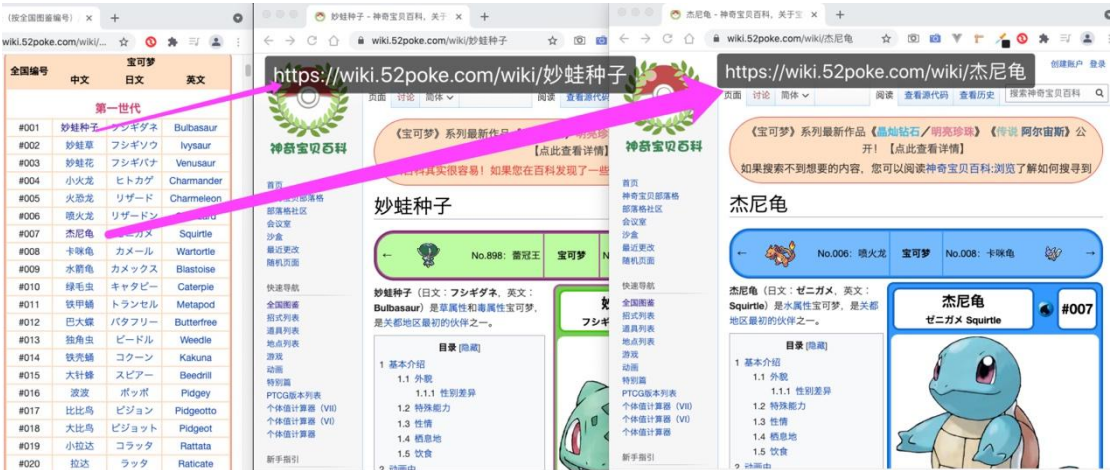
- 1××：提示信息，表示目前是协议处理的中间状态，还需要后续的操作；
- 2××：成功，报文已经收到并被正确处理；
- 3××：重定向，资源位置发生变动，需要客户端重新发送请求；
- 4××：客户端错误，请求报文有误，服务器无法处理；
- 5××：服务器错误，服务器在处理请求时，服务器内部发生了错误。

## 6. 请求和响应

### (1) Request 请求主体

请求主体中包括了要发送给服务器的数据，比如：我有一个庞大的小精灵数据库，但我  
只想看其中的一个小精灵，点击列表向服务要“妙蛙种子”这个数据接口。

RESTful 风格的接口命名规



query 参数



详见以下网站：

<https://guahao.zjol.com.cn/home>

(2) Response 响应

响应的主体部分包括很多内容，如：String (text/html 或 text/plain)、Buffer、Object  
(包括普通的对象 { foo: 'bar' } 和数组 ['foo', 'bar']) 等等。  
其中我们最需要等就是`Object`，俗称数据。



见豆瓣热门影视接口：

[https://movie.douban.com/j/search\\_subjects?type=movie&tag=%E7%83%AD%E9%97%A8&sort=recommend&page\\_limit=20&page\\_start=0](https://movie.douban.com/j/search_subjects?type=movie&tag=%E7%83%AD%E9%97%A8&sort=recommend&page_limit=20&page_start=0)

## 二、前后台交互

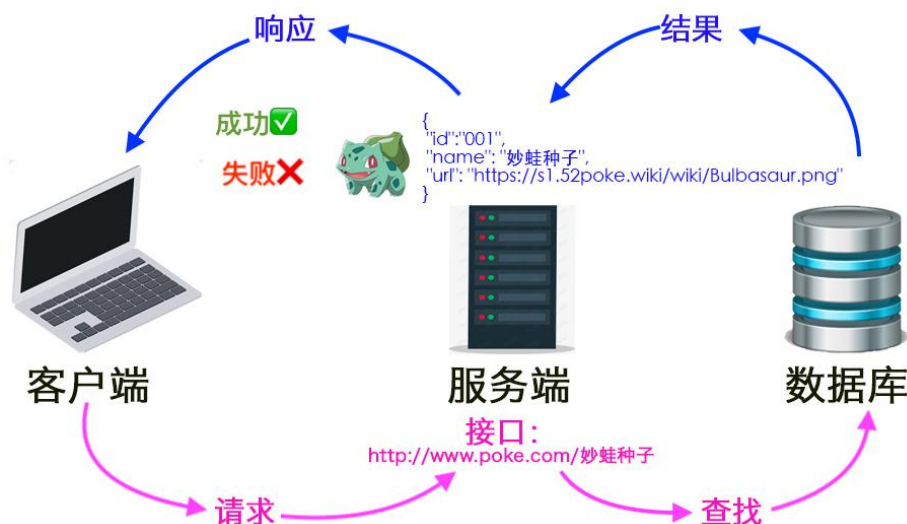
### 1. 静态和动态界面

静态是不可存储的状态，动态是可以读取以及存储数据。静态页面现在的项目是基本不需要的，需要动态数据的参与。如：电商页面的数据，投票网站的随时更新数据，视频网站的弹目等。

### 2. 交互过程

据交互的过程可简单理解为，前台想要获得某些数据，将传入参数通过 URL 接口地址，传递给服务器，服务器根据传入的参数了解到前台要获得什么数据，去数据库查询获取数据，然后将所需数据返回给前台，前台拿到数据做相应的页面展示。

成功，除了成功的状态码和成功的信息之外还有客户端需要的“数据”。而失败，将会传递失败的状态码和失败的信息，客户端需要的没有数据。



### 3. 请求数据接口

接口就是提供具体能力的一个标准和抽象，是一些预先定义的函数，包括接口地址、传入参数和返回参数和数据。

可以简单理解为，当需要访问某些数据，正常状态下传入合格参数，会收到该数据范围内的返回参数。前后台的交互都是通过程序接口实现的。

如以下后台提供的接口地址（由后台的路由分配）：

接口 url	描述
<a href="http://www.w3cweb.com/query">http://www.w3cweb.com/query</a>	所有小精灵数据
<a href="http://www.w3cweb.com/del">http://www.w3cweb.com/del</a>	删除一个小精灵
<a href="http://www.w3cweb.com/add">http://www.w3cweb.com/add</a>	添加一个小精灵

url 地址选择接口后可以传递参数,服务器可以通过参数在该接口访问的所有数据中进行筛选，如：

http://www.w3cweb.com/query?name=妙蛙种子

### 3.响应数据的格式

#### (1) 返回参数

返回参数是具备一些规范的，在企业开发中，尽量使用规范的返回格式。尽量使用标准的返回参数格式。

字段	说明	类型	备注	是否必填
code	接口状态码	Number	成功：1 失败：0	是
message	接口信息	String	成功：sucess 失败：提示信息	是
data		Object		是

#### (2) 返回数据格式

接口支持返回 XML 和 JSON 两种格式的数据，目前主流的数据返回格式都是以 JSON 的形式，JSON 的字段是以 { key: value } 的形式成对出现的。

类型	特征	例子
String	value 套在双引号中	{"name":"皮卡丘"}
Number	value 为整数或浮点数	{"age":18}
Boolean	value 为 true 或 false	{"isMam":true}
Array	value 套在方括号中	{"attr":["火系", "水系", "电系"]}
Object	value 套在方花号中	{"poke":{"uid": "004", "name": "小火龙"}}
Null	value 套在方花号中	{"isActive":null}

#### 4.同步和异步

没用 AJAX 的网页，点一个按钮就要刷新一下页面，比如表单提交写错了要刷新页面重新写。用了 AJAX 之后，你点击后，页面上的某个部分变化了，页面本身不用刷新。

##### (1) 同步

同步(Synchronous)，同步方法调用一旦开始，调用者必须等到方法调用返回后，才能继续后续的行为。

- 等待验证身份才能进入高考选择学校的界面
- 挂号系统，缴费成功后出现挂号单
- 扫二维码身份信息通过出现正常

##### (2) 异步

异步(Asynchronous)，异步方法调用更像一个消息传递，一旦开始，方法调用就会立即返回，调用者就可以继续后续的操作。而，异步方法通常会在另外一个线程中，“真实”地执行着。整个过程，不会阻碍调用者的工作。

- 注册用户时鼠标失去焦点，会自动出现验证
- 弹目同时出现在视频区域
- 购买成功或者抽奖成功短信通知

