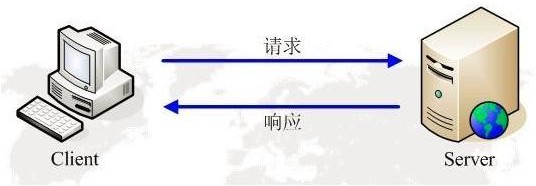
# Http协议课程介绍

互联网的发展非常迅速，但是万变不离其宗。学习 web 开发，需要我们对互联的交互机制有一定的了解。为了更好的理解并掌握 Servlet，在正式学习Servlet 之前需要对 web 开发中客户端和服务器的交互机制以及所涉及的规范(HTTP)进行学习。

# HTTP 协议详解

## 1、 web 交互的基本流程



图片介绍：

客户端根据用户输入的地址信息请求服务器，服务器在接收到用户的请求后进行处理，然后将处理结果响应给客户端，客户端将响应结果展示给用户。

|  |
| --- |
|  |

TCP (Transmission Control Protocol) 中文：传输控制协议,就是由[网络层](https://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=16964&ss_c=ssc.citiao.link)的IP协议和[传输层](https://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=16962)的[TCP协议](https://baike.sogou.com/lemma/ShowInnerLink.htm?lemmaId=20113)组成的

面向连接（三次握手），数据安全，速度略低。分为客户端和服务端。

三次握手: 客户端先向服务端发起请求, 服务端响应请求, 传输数据

举例：

|  |
| --- |
| 打电话:1.呼叫2.接听,3.数据传输  网站的访问:1.客户端发送请求(输入网址,回车),2.连接服务器3.数据的传输 |

## 专业术语:

请求：客户端根据用户地址信息将数据发送给服务器的过程

响应：服务器将请求的处理结果发送给浏览器的过程

## 问题：

客户端也就是浏览器的版本是有很多的，服务器的版本也是有很多的，如何实现不同版本的浏览器和不同版本的服务器之间的数据交互呢？

## 解决：

规范浏览器和服务器的数据交互的格式。

## 实现：

HTTP 协议

## 2、 HTTP 的概念和介绍

概念：

超文本传输协议（Hyper Text Transfer Protocol）

作用：

## 规范了浏览器和服务器的数据交互

## 特点：

简单快速：客户向服务器请求服务时，只需传送请求方法和路径。

请求方法常用的有 GET、HEAD、POST。每种方法规定了客户与服务器联系的类型不同。由于 HTTP 协议简单，使得 HTTP 服务器的程序规模小，因而通信速度很快

灵活：HTTP 允许传输任意类型的数据对象。正在传输的类型由Content-Type 加以标记。

无连接：无连接的含义是限制每次连接只处理一个请求。服务器处理完客户的请求，并收到客户的应答后，即断开连接。采用这种方式可以节省传输时间。

无状态：HTTP 协议是无状态协议。无状态是指协议对于事务处理没有记忆能力。缺少状态意味着如果后续处理需要前面的信息， 则它必须重传，这样可能导致每次连接传送的数据量增大。另一方面，在服务器不需要先前信息时它的应答就较快。

支持 B/S 及 C/S 模式。 GUI

HTTP1.1 版本后支持可持续连接

|  |
| --- |
| 再一次请求之后，在一定的时间内如果没有新的请求，那么此次http请求，立马进入断开的倒计时  优点：提高一次http连接的请求效率 |

## 3、 HTTP 的交互流程：

### HTTP 的交互流程一般分为四个步骤(一次完整的请求)：

步骤一:

客户端和服务器端建立连接步骤二：

客户端发送请求数据到服务器端(HTTP 协议) 步骤三：

服务器端接收到请求后，进行处理，然后将

处理结果响应客户端(HTTP 协议) 步骤四：

关闭客户端和服务器端的连接(HTTP1.1 后不会立即关闭)

## 4、 HTTP 协议之请求格式

### 请求格式的结构：

请求头：请求方式、请求的地址和 HTTP 协议版本

请求行：消息报头，一般用来说明客户端要使用的一些附加信息空行： 位于请求行和请求数据之间，空行是必须的。

请求数据：非必须。

## 注意： 一张网页的内容是极其丰富的，浏览器会遵循

HTTP 请求的格式将有效数据发送给服务器。

Http请求头数据解释

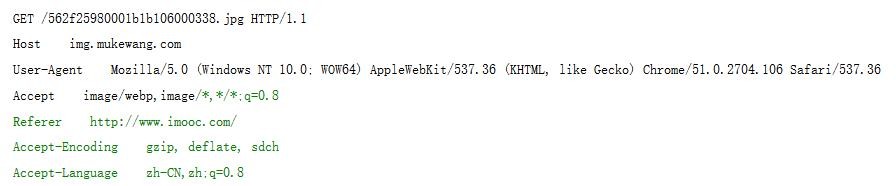
|  |
| --- |
|  |
| Accept:告诉服务器,客户端能够支持响应的数据类型  Accept-Enconding:告诉服务器,客户端支持响应的压缩算法  Accept-Language:客户端支持的语言  Connection:保持连接  Cookie:是服务器发送给客户端的一块牌牌,这个牌记录了客户端前面访问过服务器,并可以获取之前产生的一些数据信息  Host:访问的主机名  User-Agent:客户端(浏览器)的信息 |

响应头：

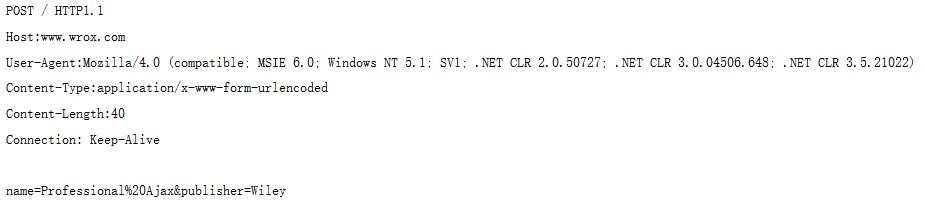
|  |
| --- |
|  |
| HTTP/2:连接状态,200表示连接成功,404:表示找不到资源,505:服务器的程序有问题,302:请求的方式不支持  content-type:响应的数据类型  cache-control:缓存的控制方式  Expires:缓存的失效时间  Server:服务器的型号 |

消息头：

|  |
| --- |
|  |
|  |

示例(get 请求方式)：

示例(post 请求方式):



|  |
| --- |
| get请求:没有请求体,提交的数据含在url后面 |
| post请求:有请求体,提交的数据在请求体中 |

5、 HTTP 协议之请求方式

根据 HTTP 标准，HTTP 请求可以使用多种请求方法。

**HTTP1.0** 定义了三种请求方法： **GET, POST** 和 **HEAD** 方法。**HTTP1.1** 新增了五种请求方法：**OPTIONS, PUT, DELETE, TRACE 和CONNECT** 方法。



### get 和post 请求方式的区别: get 请求方式：

请求数据会以？的形式隔开拼接在请求头中，不安全，没有请求实

体部分。

HTTP 协议虽然没有规定请求数据的大小，但是浏览器对 URL 的长度是有限制的，所以 get 请求不能携带大量的数据。url会出现在浏览器的地址栏中，没有请求体浏览器

|  |
| --- |
| **Microsoft Internet Explorer (Browser)**  IE浏览器对URL的最大限制为2083个字符，如果超过这个数字，提交按钮没有任何反应。 **Firefox (Browser)**  对于Firefox浏览器URL的长度限制为65,536个字符。  **Safari (Browser)**  URL最大长度限制为 80,000个字符。  **Opera (Browser)**  URL最大长度限制为190,000个字符。  **Google (chrome)**  URL最大长度限制为8182个字符。 |
| 服务器 **Apache (Server)**  能接受最大url长度为8,192个字符。  **Microsoft Internet Information Server(IIS)**  能接受最大url的长度为16,384个字符。 |

### post 请求方式：

请求数据在请求实体中进行发送，在 URL 中看不到具体的请求数据， 安全。适合数据量大的数据发送。用户提交的数据会出现在请求体中

## 6、 HTTP 协议之响应

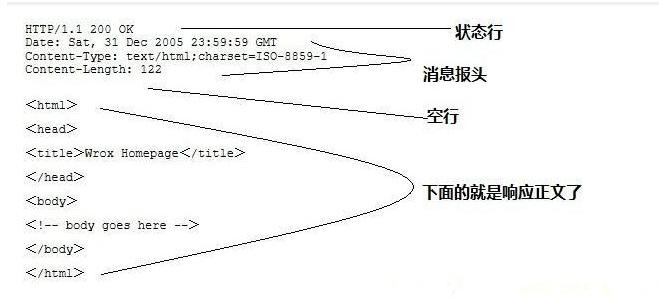
### 响应格式的结构：

响应行(状态行)：HTTP 版本、状态码、状态消息响应头：消息报头，客户端使用的附加信息

空行：响应头和响应实体之间的，必须的。

响应实体：正文，服务器返回给浏览器的信息

### 示例：

HTTP 常见响应状态码含义：

HTTP 状态码由三个十进制数字组成，第一个十进制数字定义了状态码的类型，后两个数字没有分类的作用。HTTP 状态码共分为 5 种类型：



常见状态码：

200 OK //客户端请求成功

400 Bad Request //客户端请求有语法错误，不能被服务器所理解

401 Unauthorized //请求未经授权，这个状态代码必须和WWW-Authenticate 报头域一起使用

403 Forbidden //服务器收到请求，但是拒绝提供服务

404 Not Found //请求资源不存在，eg：输入了错误的 URL

500 Internal Server Error //服务器发生不可预期的错误（java代码编码有错误）

503 Server Unavailable //服务器当前不能处理客户端的请求，一段时

间后可能恢复正常