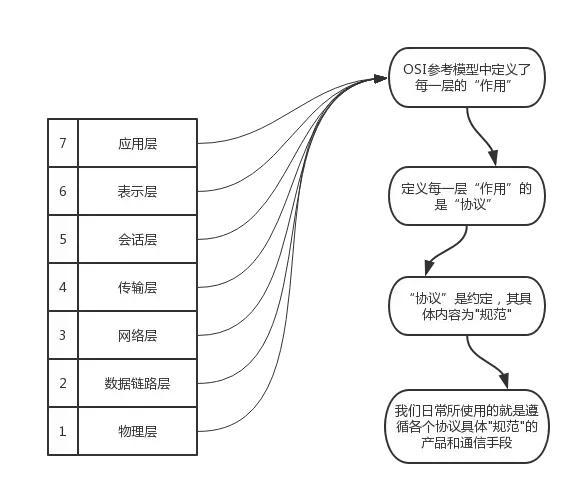
一、HTTP协议简介（注意不要跟计算机网络的底层格式混了，这个是应用层的协议）

HTTP协议是Hyper Text Transfer Protocol（超文本传输协议）的缩写，是用于从万维网服务器传输超文本到本地浏览器的传送协议。HTTP 是基于 TCP/IP 协议通信协议来传递数据（HTML 文件，图片文件，查询结果等）。它不涉及数据包（packet）传输，主要规定了客户端和服务器之间的通信格式，默认使用80端口。



1、通信，就是信息的传递和交换。

通信三要素：通信的主体、通信的内容、通信的方式

2、什么是通信协议

通信协议（Communication Protocol）是指通信的双方完成通信所必须遵守的**规则**和**约定**。

通俗的理解：通信双方采用**约定好的格式**来发送和接收消息，这种事先约定好的通信格式，就叫做通信协议。

3、http协议

网页内容又叫做**超文本**，因此网页内容的**传输协议**又叫做**超文本传输协议**（HyperText Transfer Protocol） ，简称 HTTP 协议。

（1）HTTP协议的交互模型



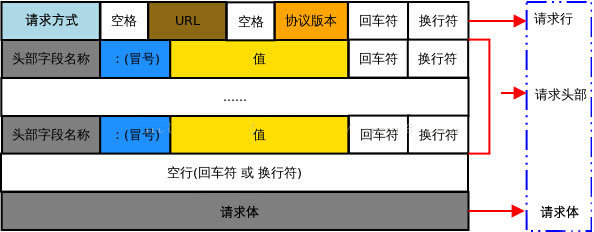
二、HTTP请求

由于 HTTP 协议属于客户端浏览器和服务器之间的通信协议。因此，客户端发起的请求叫做 **HTTP 请求**，客户端发送到服务器的消息，叫做 **HTTP 请求消息**。

注意：HTTP 请求消息又叫做 HTTP **请求报文**。

1、HTTP请求消息的组成部分

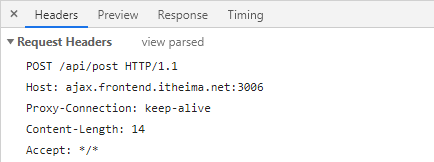
HTTP 请求消息由**请求行**（request line）、**请求头部**（ header ）、**空行** 和 **请求体** 4 个部分组成。



2、请求行

请求行由请求方式、URL 和 HTTP 协议版本 3 个部分组成，他们之间使用空格隔开。





3、请求头部（使用view source，F12的显示是经过优化的）

请求头部用来描述客户端的基本信息，从而把客户端相关的信息告知服务器。

User-Agent 用来说明当前是什么类型的浏览器；

Content-Type 用来描述发送到服务器的数据格式；

Accept 用来描述客户端能够接收什么类型的返回内容；

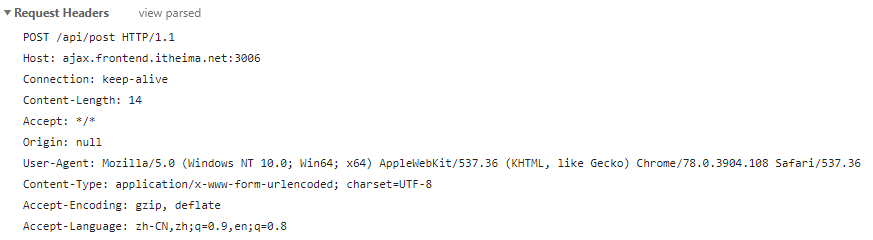
Accept-Language 用来描述客户端期望接收哪种人类语言的文本内容。

请求头部由多行 键/值对 组成，每行的键和值之间用英文的冒号分隔。



（1）常见的请求头字段

|  |  |
| --- | --- |
| **头部字段** | **说明** |
| Host | 要请求的服务器域名 |
| Connection | 客户端与服务器的连接方式(close 或 keepalive) |
| Content-Length | 用来描述请求体的大小 |
| Accept | 客户端可识别的响应内容类型列表 |
| User-Agent | 产生请求的浏览器类型 |
| Content-Type | 客户端告诉服务器实际发送的数据类型 |
| Accept-Encoding | 客户端可接收的内容压缩编码形式 |
| Accept-Language | 用户期望获得的自然语言的优先顺序 |



4、空行

最后一个请求头字段的后面是一个空行，通知服务器请求头部至此结束。

请求消息中的空行，用来分隔请求头部与请求体。



5、请求体

请求体中存放的，是要通过 **POST** 方式提交到服务器的数据。

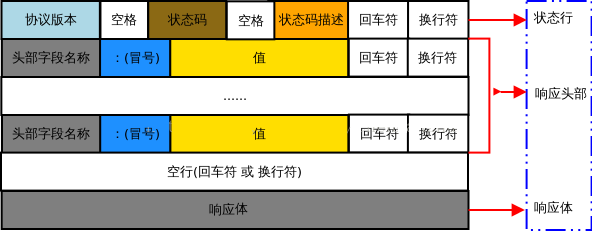


**注意**：只有 POST 请求才有请求体，GET 请求没有请求体！

三、HTTP响应消息

响应消息就是服务器响应给客户端的消息内容，也叫**作响应报文**。

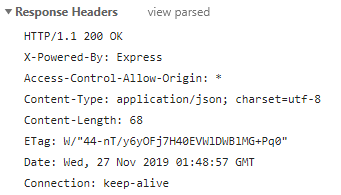
HTTP响应消息由状态行、响应头部、空行 和 响应体 4 个部分组成，如下图所示：



1、状态行

状态行由 HTTP 协议版本、状态码和状态码的描述文本 3 个部分组成，他们之间使用空格隔开

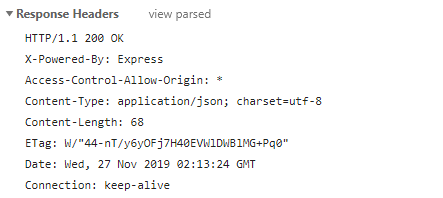




2、响应头部

响应头部用来描述服务器的基本信息。响应头部由多行 键/值对 组成，每行的键和值之间用英文的冒号分隔。

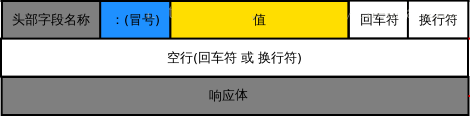




3、空行

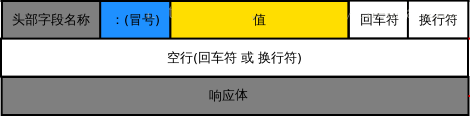
在最后一个响应头部字段结束之后，会紧跟一个空行，用来通知客户端响应头部至此结束。

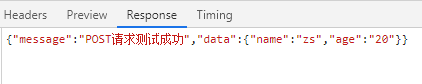
响应消息中的空行，用来分隔响应头部与响应体。



4、响应体

响应体中存放的，是服务器响应给客户端的资源内容。





四、HTTP请求方法

HTTP 请求方法，属于 HTTP 协议中的一部分，请求方法的作用是：用来表明要**对服务器上的资源执行的操作**。最常用的请求方法是 GET 和 POST。

1、HTTP的请求方法

|  |  |  |
| --- | --- | --- |
| **序号** | **方法** | **描述** |
| 1 | GET | (查询)发送请求来获得服务器上的资源，请求体中不会包含请求数据，请求数据放在协议头中。 |
| 2 | POST | (新增)向服务器提交资源（例如提交表单或上传文件）。数据被包含在请求体中提交给服务器。 |
| 3 | PUT | (修改)向服务器提交资源，并使用提交的新资源，替换掉服务器对应的旧资源。 |
| 4 | DELETE | (删除)请求服务器删除指定的资源。 |
| 5 | HEAD | HEAD 方法请求一个与 GET 请求的响应相同的响应，但没有响应体。 |
| 6 | OPTIONS | 获取http服务器支持的http请求方法，允许客户端查看服务器的性能，比如ajax跨域时的预检等。 |
| 7 | CONNECT | 建立一个到由目标资源标识的服务器的隧道。 |
| 8 | TRACE | 沿着到目标资源的路径执行一个消息环回测试，主要用于测试或诊断。 |
| 9 | PATCH | 是对 PUT 方法的补充，用来对已知资源进行局部更新 。 |

五、HTTP响应状态代码

HTTP 响应状态码（HTTP Status Code），也属于 HTTP 协议的一部分，**用来标识响应的状态。**

响应状态码会随着响应消息一起被发送至客户端浏览器，浏览器根据服务器返回的响应状态码，就能知道这次 HTTP 请求的结果是成功还是失败了。

