

## 图解HTTP(3) HTTP报文内的HTTP信息1

1 HTTP报文：用于 HTTP协议交互 的信息。

一般分为 请求报文 、 响应报文 2种。

2 HTTP报文 = 报文首部【内含 请求行 或 响应行 + 各种首部（请求、响应、通用、实体）】 + 报文主体（也可能不存在）。2者中间 用 空行（CR+LF）来划分。

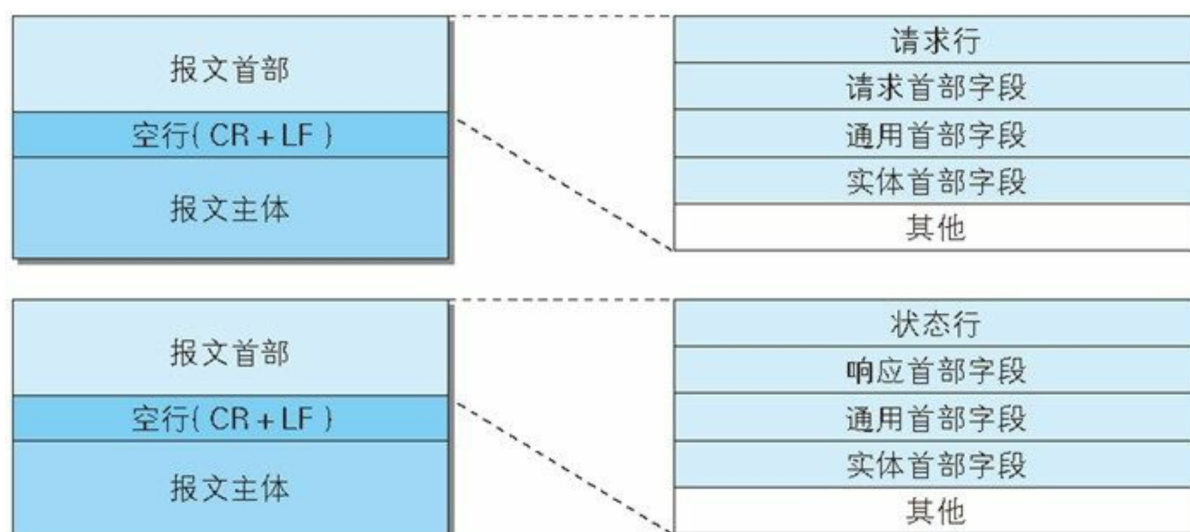
??！：HTTP报文 = 请求行 + 各种首部（“前面两个合起来就是 报文首部！！”） + 报文主体 这样更好些吧??！

HTTP报文 =

【内含 请求行 或 响应行 + 各种首部（请求、响应、通用、实体）】

+

报文主体（也可能不存在）。



3 请求行 = 请求方法、请求URI、HTTP版本

响应行 = HTTP版本、状态码、原因短语

4 首部字段【含 各种条件和各种首部！】：含 请求和响应的 各种条件 和 属性的各类首部！！

5 4种首部： 请求 / 响应首部、通用首部、实体首部。

还可能含 未定义的首部（Cookie等！！）

6 报文主体 和 实体主体 的差异【传输中进行编码，才会让它们变得不同！！】：

HTTP报文的主体用于传输 请求 / 响应 的实体主体。

通常 报文主体 就是 实体主体。只有当 传输中 进行编码操作时，实体主体的内容发生变化，才导致它和报文主体产生差异。

7 常用的内容编码方式：

gzip (GNU zip)

compress (UNIX系统的标准压缩)

deflate (zlib)

identity (不进行编码)

8 分块传输编码：把 实体主体【为啥不是报文主体??】 分块的功能。

这样，在传输大容量数据时，通过将数据分割，能够让浏览器 逐步显示页面！！

8.1 分块传输编码会将 实体主体分成多个部分（块）。每一块都会用 十六进制标记 该块的大小。而实体主体的最后一块会用“0 (CR+LF)”来标记！！

9 发送多种数据 的多部分对象集合。

可以往邮件里面写入文字添加多份附件，是因为 其用到了 MIME【多用途互联网邮件拓展】机制，它允许 邮件处理文本、图片、视频 等多个不同类型的 数据。

MIME拓展中会使用一种称为 多不分对象集合（Multipart）的方法，来容纳不同类型的数  
据。

9.1

multipart / form-data: web表单文件上传时 使用。

```
Content-Type: multipart/form-data; boundary=AaB03x

--AaB03x
Content-Disposition: form-data; name="field1"

Joe Blow
--AaB03x
Content-Disposition: form-data; name="pics"; filename="file1.txt"
Content-Type: text/plain

... (file1.txt的数据) ...
--AaB03x--
```

multipart /byteranges【字节范围，部分内容的传输】： 状态码206（部分内容）响应报文包含了多个范围的内容时使用。

```
... (范围指定的数据) ...
--THIS_STRING_SEPARATES
Content-Type: application/pdf
Content-Range: bytes 7000-7999/8000

... (范围指定的数据) ...
--THIS_STRING_SEPARATES--
```

9.2 HTTP报文中使用 多部分对象集合时， 需要在首部字段里加上 Content-type!!

9.3 范围请求， 响应码会返回 206【部分内容。如果服务器不支持响应范围请求， 则会返回状态码 200 OK 和 完整的实体内容】。

3种范围请求的写法：

- **5001~10 000 字节**

Range: bytes=5001-10000

- 从 **5001** 字节之后全部的

Range: bytes=5001-

- 从一开始到 **3000** 字节和 **5000~7000** 字节的多重范围

Range: bytes=-3000, 5000-7000

10 内容协商返回最适合的内容。

内容协商机制： 客户端 和 服务端 就响应的资源内容进行交涉， 然后提供 给客户端最适合的资源。内容协商 会以响应资源的语言、 字符集、 编码方式等 作为判断的基准。

10.2 3种内容协商技术

服务器驱动协商： 以请求的首部字段为参考， 在服务端自动处理。

客户端驱动协商： 用户从 浏览器显示 的 可选项列表中 手动选择。

透明协商： 服务器驱动 和 客户端驱动 的结合体。是由服务端、客户端各自进行内容协商的一种方法。

完