# **客户端请求**

Http请求包含三部分：

* <请求方法> <统一资源标识符（URI）> <协议/版本>
* <请求头>
* <实体>

示例：

GET /example/hello.jsp HTTP/1.1

User-Agent: curl/7.16.3 libcurl/7.16.3

OpenSSL/0.9.7l zlib/1.2.3

Host: [www.example.com](http://www.example.com)

Accept-Language: en-gb

lastName=Franks&firstName=Micheal

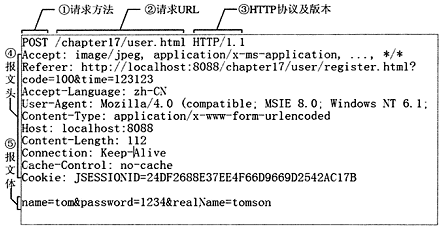
说明:

Get是请求方法，/examples/hello.jsp表示URI，Http/1.1表示请求使用的协议和版本。

请求头包含客户端环境和请求实体正文的相关信息。如：浏览器使用的语言，请求实体正文的长度，各个请求头之间用换行符间隔开。

请求实体正文与请求头之间有一个空行，该空行只有CRLF符。这个空行对Http请求格式非常重要。

**如下是一个具体的http请求信息：**



# **服务端响应**

Http响应包含三部分：

* <协议> <状态码> <描述>
* <响应头>
* <实体>

示例：

HTTP/1.1 200 OK

Date: Mon, 27 Jul 2009 12:28:53 GMT

Server: Apache

Last-Modified: Wed, 22 Jul 2009 19:15:56 GMT

Content-Length: 51

Content-Type: text/html

Hello World! My payload includes a trailing CRLF.

说明：

Http响应头的第一行与Http请求头的第一行类似。第一行指明了使用的协议是HTTP/1.1，请求发送成功（状态码200表示请求成功)。与请求头类似，响应头也包含一些有用信息。响应实体正文与响应头之间包含CRLF的一个空行分隔。

# **MIME类型详解**

。。。

# HTTP Request Header 请求头

**Accept**

请求报文可通过一个“Accept”报文头属性告诉服务端 客户端接受什么类型的响应。 如下报文头相当于告诉服务端，俺客户端能够接受的响应类型仅为纯文本数据啊，你丫别发其它什么图片啊，视频啊过来，那样我会歇菜的~~~：

示例：Accept: text/plain, text/html

Accept属性的值可以为一个或多个MIME类型的值，关于MIME类型，大家请参考：[http://en.wikipedia.org/wiki/MIME\_type](http://en.wikipedia.org/wiki/MIME_type" \t "http://blog.csdn.net/yuyunli1989/article/details/_blank)

**Accept-Charset**

表示浏览器可以接受的字符编码集

示例：Accept-Charset: iso-8859-5

| **Header** | **解释** | **示例** |
| --- | --- | --- |
| Accept | 表示客户端能够接收的内容类型 | Accept: text/plain, text/html |
| Accept-Charset | 浏览器可以接受的字符编码集。 | Accept-Charset: iso-8859-5 |
| Accept-Encoding | 指定浏览器可以支持的web服务器返回内容压缩编码类型。 | Accept-Encoding: compress, gzip |
| Accept-Language | 浏览器可接受的语言 | Accept-Language: en,zh |
| Accept-Ranges | 可以请求网页实体的一个或者多个子范围字段 | Accept-Ranges: bytes |
| Authorization | HTTP授权的授权证书 | Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ== |
| Cache-Control | 指定请求和响应遵循的缓存机制 | Cache-Control: no-cache |
| Connection | 表示是否需要持久连接。（HTTP 1.1默认进行持久连接） | Connection: close |
| Cookie | HTTP请求发送时，会把保存在该请求域名下的所有cookie值一起发送给web服务器。 | Cookie: $Version=1; Skin=new; |
| Content-Length | 请求的内容长度 | Content-Length: 348 |
| Content-Type | 请求的与实体对应的MIME信息 | Content-Type: application/x-www-form-urlencoded |
| Date | 请求发送的日期和时间 | Date: Tue, 15 Nov 2010 08:12:31 GMT |
| Expect | 请求的特定的服务器行为 | Expect: 100-continue |
| From | 发出请求的用户的Email | From: user@email.com |
| Host | 指定请求的服务器的域名和端口号 | Host: www.zcmhi.com |
| If-Match | 只有请求内容与实体相匹配才有效 | If-Match: “737060cd8c284d8af7ad3082f209582d” |
| If-Modified-Since | 如果请求的部分在指定时间之后被修改则请求成功，未被修改则返回304代码 | If-Modified-Since: Sat, 29 Oct 2010 19:43:31 GMT |
| If-None-Match | 如果内容未改变返回304代码，参数为服务器先前发送的Etag，与服务器回应的Etag比较判断是否改变 | If-None-Match: “737060cd8c284d8af7ad3082f209582d” |
| If-Range | 如果实体未改变，服务器发送客户端丢失的部分，否则发送整个实体。参数也为Etag | If-Range: “737060cd8c284d8af7ad3082f209582d” |
| If-Unmodified-Since | 只在实体在指定时间之后未被修改才请求成功 | If-Unmodified-Since: Sat, 29 Oct 2010 19:43:31 GMT |
| Max-Forwards | 限制信息通过代理和网关传送的时间 | Max-Forwards: 10 |
| Pragma | 用来包含实现特定的指令 | Pragma: no-cache |
| Proxy-Authorization | 连接到代理的授权证书 | Proxy-Authorization: Basic QWxhZGRpbjpvcGVuIHNlc2FtZQ== |
| Range | 只请求实体的一部分，指定范围 | Range: bytes=500-999 |
| Referer | 先前网页的地址，当前请求网页紧随其后,即来路 | Referer: http://www.zcmhi.com/archives/71.html |
| TE | 客户端愿意接受的传输编码，并通知服务器接受接受尾加头信息 | TE: trailers,deflate;q=0.5 |
| Upgrade | 向服务器指定某种传输协议以便服务器进行转换（如果支持） | Upgrade: HTTP/2.0, SHTTP/1.3, IRC/6.9, RTA/x11 |
| User-Agent | User-Agent的内容包含发出请求的用户信息 | User-Agent: Mozilla/5.0 (Linux; X11) |
| Via | 通知中间网关或代理服务器地址，通信协议 | Via: 1.0 fred, 1.1 nowhere.com (Apache/1.1) |
| Warning | 关于消息实体的警告信息 | Warn: 199 Miscellaneous warning |

# **HTTP Responses Header 响应头**

| **Header** | **解释** | **示例** |
| --- | --- | --- |
| Accept-Ranges | 表明服务器是否支持指定范围请求及哪种类型的分段请求 | Accept-Ranges: bytes |
| Age | 从原始服务器到代理缓存形成的估算时间（以秒计，非负） | Age: 12 |
| Allow | 对某网络资源的有效的请求行为，不允许则返回405 | Allow: GET, HEAD |
| Cache-Control | 告诉所有的缓存机制是否可以缓存及哪种类型 | Cache-Control: no-cache |
| Content-Encoding | web服务器支持的返回内容压缩编码类型。 | Content-Encoding: gzip |
| Content-Language | 响应体的语言 | Content-Language: en,zh |
| Content-Length | 响应体的长度 | Content-Length: 348 |
| Content-Location | 请求资源可替代的备用的另一地址 | Content-Location: /index.htm |
| Content-MD5 | 返回资源的MD5校验值 | Content-MD5: Q2hlY2sgSW50ZWdyaXR5IQ== |
| Content-Range | 在整个返回体中本部分的字节位置 | Content-Range: bytes 21010-47021/47022 |
| Content-Type | 返回内容的MIME类型 | Content-Type: text/html; charset=utf-8 |
| Date | 原始服务器消息发出的时间 | Date: Tue, 15 Nov 2010 08:12:31 GMT |
| ETag | 请求变量的实体标签的当前值 | ETag: “737060cd8c284d8af7ad3082f209582d” |
| Expires | 响应过期的日期和时间 | Expires: Thu, 01 Dec 2010 16:00:00 GMT |
| Last-Modified | 请求资源的最后修改时间 | Last-Modified: Tue, 15 Nov 2010 12:45:26 GMT |
| Location | 用来重定向接收方到非请求URL的位置来完成请求或标识新的资源 | Location: http://www.zcmhi.com/archives/94.html |
| Pragma | 包括实现特定的指令，它可应用到响应链上的任何接收方 | Pragma: no-cache |
| Proxy-Authenticate | 它指出认证方案和可应用到代理的该URL上的参数 | Proxy-Authenticate: Basic |
| refresh | 应用于重定向或一个新的资源被创造，在5秒之后重定向（由网景提出，被大部分浏览器支持） | Refresh: 5; url=  http://www.zcmhi.com/archives/94.html |
| Retry-After | 如果实体暂时不可取，通知客户端在指定时间之后再次尝试 | Retry-After: 120 |
| Server | web服务器软件名称 | Server: Apache/1.3.27 (Unix) (Red-Hat/Linux) |
| Set-Cookie | 设置Http Cookie | Set-Cookie: UserID=JohnDoe; Max-Age=3600; Version=1 |
| Trailer | 指出头域在分块传输编码的尾部存在 | Trailer: Max-Forwards |
| Transfer-Encoding | 文件传输编码 | Transfer-Encoding:chunked |
| Vary | 告诉下游代理是使用缓存响应还是从原始服务器请求 | Vary: \* |
| Via | 告知代理客户端响应是通过哪里发送的 | Via: 1.0 fred, 1.1 nowhere.com (Apache/1.1) |
| Warning | 警告实体可能存在的问题 | Warning: 199 Miscellaneous warning |
| WWW-Authenticate | 表明客户端请求实体应该使用的授权方案 | WWW-Authenticate: Basic |

# **HTTP的响应状态码**

**HTTP的响应状态码**由5段组成：

1xx 消息，一般是告诉客户端，请求已经收到了，**正在处理**，别急...

2xx **处理成功**，一般表示：请求收悉、我明白你要的、请求已受理、已经处理完成等信息.

3xx **重定向到其它地方。它让客户端再发起一个请求以完成整个处理**。

4xx **处理发生错误，责任在客户端**，如客户端的请求一个不存在的资源，**客户端未被授权，禁止访问**等。

5xx **处理发生错误，责任在服务端**，如服务端抛出异常，路由出错，HTTP版本不支持等

忽略：277187

特别注意：306已经不存在了

**1xx（临时响应）**  
表示临时响应并需要请求者继续执行操作的状态代码。

代码   说明   
100   （继续） 请求者应当继续提出请求。 服务器返回此代码表示已收到请求的第一部分，正在等待其余部分。    
101   （切换协议） 请求者已要求服务器切换协议，服务器已确认并准备切换。

**2xx （成功）**  
表示成功处理了请求的状态代码。

代码   说明   
**200   （成功）** 服务器已成功处理了请求。 通常，这表示服务器提供了请求的网页。   
201   （已创建）  请求成功并且服务器创建了新的资源。   
202   （已接受）  服务器已接受请求，但尚未处理。   
203   （非授权信息）  服务器已成功处理了请求，但返回的信息可能来自另一来源。   
204   （无内容）  服务器成功处理了请求，但没有返回任何内容。   
205   （重置内容） 服务器成功处理了请求，但没有返回任何内容。   
206   （部分内容）  服务器成功处理了部分 GET 请求。

**3xx （重定向）**  
表示要完成请求，需要进一步操作。 通常，这些状态代码用来重定向。

代码   说明   
300   （多种选择）  针对请求，服务器可执行多种操作。 服务器可根据请求者 (user agent) 选择一项操作，或提供操作列表供请求者选择。   
301   （永久移动）  请求的网页已永久移动到新位置。 服务器返回此响应（对 GET 或 HEAD 请求的响应）时，会自动将请求者转到新位置。   
**302   （临时移动）**  服务器目前从不同位置的网页响应请求，但请求者应继续使用原有位置来进行以后的请求。   
**303   （查看其他位置，跳转至其他页面，目标的url会根据响应的报文告诉你）**请求者应当对不同的位置使用单独的 GET 请求来检索响应时，服务器返回此代码。   
**304   （未修改,可以直接用本地的缓存）** 自从上次请求后，请求的网页未修改过。 服务器返回此响应时，不会返回网页内容。   
305   （使用代理） 请求者只能使用代理访问请求的网页。 如果服务器返回此响应，还表示请求者应使用代理。   
**307   （临时重定向）**  服务器目前从不同位置的网页响应请求，但请求者应继续使用原有位置来进行以后的请求。

**4xx（请求错误）**  
这些状态代码表示请求可能出错，妨碍了服务器的处理。

代码   说明   
400   （错误请求） 服务器不理解请求的语法。   
401   （未授权） 请求要求身份验证。 对于需要登录的网页，服务器可能返回此响应。   
403   （禁止） 服务器拒绝请求。   
404   （未找到） 服务器找不到请求的网页。   
405   （方法禁用） 禁用请求中指定的方法。   
406   （不接受） 无法使用请求的内容特性响应请求的网页。   
407   （需要代理授权） 此状态代码与 401（未授权）类似，但指定请求者应当授权使用代理。   
408   （请求超时）  服务器等候请求时发生超时。   
409   （冲突）  服务器在完成请求时发生冲突。 服务器必须在响应中包含有关冲突的信息。   
410   （已删除）  如果请求的资源已永久删除，服务器就会返回此响应。   
411   （需要有效长度） 服务器不接受不含有效内容长度标头字段的请求。   
412   （未满足前提条件） 服务器未满足请求者在请求中设置的其中一个前提条件。   
413   （请求实体过大） 服务器无法处理请求，因为请求实体过大，超出服务器的处理能力。   
414   （请求的 URI 过长） 请求的 URI（通常为网址）过长，服务器无法处理。   
415   （不支持的媒体类型） 请求的格式不受请求页面的支持。   
416   （请求范围不符合要求） 如果页面无法提供请求的范围，则服务器会返回此状态代码。   
417   （未满足期望值） 服务器未满足”期望”请求标头字段的要求。

**5xx（服务器错误）**   
这些状态代码表示服务器在尝试处理请求时发生内部错误。 这些错误可能是服务器本身的错误，而不是请求出错。

代码   说明   
500   （服务器内部错误）  服务器遇到错误，无法完成请求。   
501   （尚未实施） 服务器不具备完成请求的功能。 例如，服务器无法识别请求方法时可能会返回此代码。   
502   （错误网关） 服务器作为网关或代理，从上游服务器收到无效响应。   
**503   （服务不可用）** 服务器目前无法使用（由于超载或停机维护）。 通常，这只是暂时状态。   
504   （网关超时）  服务器作为网关或代理，但是没有及时从上游服务器收到请求。   
505   （HTTP 版本不受支持） 服务器不支持请求中所用的 HTTP 协议版本。