1. 请求报文

HTTP 协议是以 ASCII 码传输，建立在 TCP/IP 协议之上的应用层规范。规范把 HTTP 请求分为三个部分：状态行、请求头、消息主体。类似于下面这样：

<method> <request-URL> <version>

<headers>

<entity-body>

* 1. GET请求

GET请求报文示例：

状态行

GET /books/?sex=man&name=Professional HTTP/1.1

请求头

Host: www.example.com

User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.7.6)

Gecko/20050225 Firefox/1.0.1

Connection: Keep-Alive

* 1. POST请求

状态行

POST / HTTP/1.1

请求头

Host: www.example.com

User-Agent: Mozilla/5.0 (Windows; U; Windows NT 5.1; en-US; rv:1.7.6)

Gecko/20050225 Firefox/1.0.1

Content-Type: application/x-www-form-urlencoded

Content-Length: 40

Connection: Keep-Alive

消息主题

sex=man&name=Professional

* 1. POST提交数据方式

Content-Type和消息主题编码方式

application/x-www-form-urlencoded

这是最常见的 POST 数据提交方式。浏览器的原生 <form> 表单，如果不设置 enctype 属性，那么最终就会以 application/x-www-form-urlencoded 方式提交数据

multipart/form-data

我们使用表单上传文件时，必须让 <form> 表单的 enctype 等于 multipart/form-data

------------------------------------post提交文本和上传文件-----------------------------------

POST http://www.example.com HTTP/1.1

Content-Type:multipart/form-data; boundary=----WebKitFormBoundaryrGKCBY7qhFd3TrwA

------WebKitFormBoundaryrGKCBY7qhFd3TrwA

Content-Disposition: form-data; name="text"

title

------WebKitFormBoundaryrGKCBY7qhFd3TrwA

Content-Disposition: form-data; name="file"; filename="chrome.png"

Content-Type: image/png

PNG ... content of chrome.png ...

------WebKitFormBoundaryrGKCBY7qhFd3TrwA—

注释：首先生成了一个 boundary 用于分割不同的字段，为了避免与正文内容重复，boundary 很长很复杂。然后 Content-Type 里指明了数据是以 multipart/form-data 来编码，本次请求的 boundary 是什么内容。消息主体里按照字段个数又分为多个结构类似的部分，每部分都是以 --boundary 开始，紧接着是内容描述信息，然后是回车，最后是字段具体内容（文本或二进制）。如果传输的是文件，还要包含文件名和文件类型信息。消息主体最后以 --boundary-- 标示结束。

1. 响应报文
   1. 包含：

* 状态行
* 响应头(Response Header)
* 响应正文

状态行由：协议版本、数字形式的状态代码、相应的状态描述，亦空格分隔

常见状态码：

* 200 OK 客户端请求成功
* 301 Moved Permanently 请求永久重定向
* 302 Moved Temporarily 请求临时重定向
* 304 Not Modified 文件未修改，可以直接使用缓存的文件。
* 400 Bad Request 由于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。服务器通常会在响应正文中给出不提供服务的原因
* 404 Not Found 请求的资源不存在，例如，输入了错误的URL
* 500 Internal Server Error 服务器发生不可预期的错误，导致无法完成客户端的请求。
* 503 Service Unavailable 服务器当前不能够处理客户端的请求，在一段时间之后，服务器可能会恢复正常。

一个响应实例：

HTTP/1.1 200 OK

Server:Apache Tomcat/5.0.12

Date:Mon,6Oct2003 13:23:42 GMT

Content-Length:112

<html>...