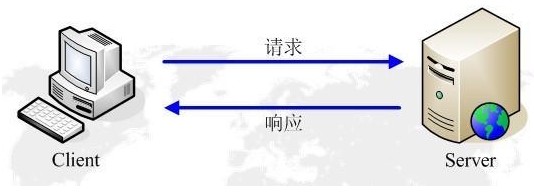
**HTTP简介：**

**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（WWW:World Wide Web ）服务器传输超文本到本地浏览器的传送协议。**

**HTTP协议工作于客户端-服务端架构上。浏览器作为HTTP客户端通过URL向HTTP服务端即WEB服务器发送请求。**

**Web服务器根据接收到的请求后，向客户端发送响应信息。**



**HTTP之URL：**

**HTTP使用统一资源标识符（Uniform Resource Identifiers, URI）来传输数据和建立连接。URL是一种特殊类型的URI，包含了用于查找某个资足源的够的信息。**

**URL,全称是Uniform Resource Locator, 中文叫统一资源定位符,是互联网上用来标识某一处资源的地址。以下面这个URL为例，介绍下普通URL的各部分组成：**

**http://www.aspxfans.com:8080/news/index.asp?boardID=5&ID=24618&page=1#name**

**从上面的URL可以看出，一个完整的URL包括以下几部分：**

**1.协议部分：该URL的协议部分为“http：”在"HTTP"后面的“//”为分隔符**

**2.域名部分：该URL的域名部分为“www.aspxfans.com”。一个URL中，也可以使用IP地址作为域名使用**

**3.端口部分：跟在域名后面的是端口，域名和端口之间使用“:”作为分隔符。端口不是一个URL必须的部分，如果省略端口部分，将采用默认端口**

**4.虚拟目录部分：从域名后的第一个“/”开始到最后一个“/”为止，是虚拟目录部分。**

**5.文件名部分：从域名后的最后一个“/”开始到“？”为止，是文件名部分，如果没有“?”,则是从域名后的最后一个“/”开始到“#”为止，是文件部分，如果没有“？”和“#”，那么从域名后的最后一个“/”开始到结束，都是文件名部分。**

**6.锚部分：从“#”开始到最后，都是锚部分。**

**7.参数部分：从“？”开始到“#”为止之间的部分为参数部分，又称搜索部分、查询部分。本例中的参数部分为“boardID=5&ID=24618&page=1”。参数可以允许有多个参数，参数与参数之间用“&”作为分隔符。**

**HTTP之请求消息Request：**

**客户端发送一个HTTP请求到服务器的请求消息包括以下格式：**

**请求行（request line）、请求头部（header）、空行和请求数据四个部分组成。**



**在火狐浏览器的开发者工具中看看Get请求例子部分消息**

**在火狐浏览器的开发者工具中看看POST请求例子部分消息**

**HTTP之响应消息Response**

**一般情况下，服务器接收并处理客户端发过来的请求后会返回一个HTTP的响应消息。**

**HTTP响应也由四个部分组成，分别是：状态行、消息报头、空行和响应正文（HTML文档往往就在正文里头）。**

**状态行里有HTTP之状态码,以后在开发中用的到：**

**状态代码有三位数字组成，第一个数字定义了响应的类别，共分五种类别:**

**1xx：指示信息--表示请求已接收，继续处理**

**2xx：成功--表示请求已被成功接收、理解、接受**

**3xx：重定向--要完成请求必须进行更进一步的操作**

**4xx：客户端错误--请求有语法错误或请求无法实现**

**5xx：服务器端错误--服务器未能实现合法的请求**

**GET和POST请求的区别**

**GET请求**

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

**Host: www.wrox.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**

**注意最后一行是空行**

**POST请求**

**POST / HTTP/1.1**

**Host: www.wrox.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**

**name=Professional%20Ajax&publisher=Wiley**

**1、GET提交，请求的数据会附在URL之后（就是把数据放置在HTTP协议头中），以?分割URL和传输数据，多个参数用&连接；这些数据都是键值对的格式。**

**POST提交：把提交的数据放置在是HTTP包的包体中。**

**因此，GET提交的数据会在地址栏中显示出来，而POST提交，地址栏不会改变**

**2、传输数据的大小：首先声明：HTTP协议没有对传输的数据大小进行限制，HTTP协议规范也没有对URL长度进行限制。但是部分浏览器对URL长度有限制。**

**GET:特定浏览器和服务器对URL长度有限制，例如 IE对URL长度的限制是2083字节(2K+35)。**

**POST:由于不是通过URL传值，理论上数据不受限。**

**3、GET方式提交数据，会带来安全问题，比如一个登录页面，通过GET方式提交数据时，用户名和密码将出现在URL上，如果页面可以被缓存或者其他人可以访问这台机器，就可以从历史记录获得该用户的账号和密码.**