

# JSP 客户端请求

当浏览器请求一个网页时，它会向网络服务器发送一系列不能被直接读取的信息，因为这些信息是作为HTTP信息头的一部分来传送的。您可以查阅HTTP协议来获得更多的信息。

下表列出了浏览器端信息头的一些重要内容，在以后的网络编程中将会经常见到这些信息：

信息	描述
Accept	指定浏览器或其他客户端可以处理的MIME类型。它的值通常为 <b>image/png</b> 或 <b>image/jpeg</b>
Accept-Charset	指定浏览器要使用的字符集。比如 ISO-8859-1
Accept-Encoding	指定编码类型。它的值通常为 <b>gzip</b> 或 <b>compress</b>
Accept-Language	指定客户端首选语言，servlet会优先返回以当前语言构成的结果集，如果servlet支持这种语言的话。比如 en，en-us，ru等等
Authorization	在访问受密码保护的网页时识别不同的用户
Connection	表明客户端是否可以处理HTTP持久连接。持久连接允许客户端或浏览器在一个请求中获取多个文件。 <b>Keep-Alive</b> 表示启用持久连接
Content-Length	仅适用于POST请求，表示 POST 数据的字节数
Cookie	返回先前发送给浏览器的cookies至服务器
Host	指出原始URL中的主机名和端口号
If-Modified-Since	表明只有当网页在指定的日期被修改后客户端才需要这个网页。 服务器发送304码给客户端，表示没有更新的资源
If-Unmodified-Since	与If-Modified-Since相反， 只有文档在指定日期后仍未被修改过，操作才会成功
Referer	标志着所引用页面的URL。比如，如果你在页面1，然后点了个链接至页面2，那么页面1的URL就会包含在浏览器请求页面2的信息头中
User-Agent	用来区分不同浏览器或客户端发送的请求，并对不同类型的浏览器返回不同的内容

## HttpServletRequest类

request对象是javax.servlet.http.HttpServletRequest类的实例。每当客户端请求一个页面时，JSP引擎就会产生一个新的对象来代表这个请求。

request对象提供了一系列方法来获取HTTP信息头，包括表单数据，cookies，HTTP方法等等。

接下来将会介绍一些在JSP编程中常用的获取HTTP信息头的方法。详细内容请见下表：

序号	方法& 描述
1	<b>Cookie[] getCookies()</b> 返回客户端所有的Cookie的数组
2	<b>Enumeration getAttributeNames()</b> 返回request对象的所有属性名称的集合
3	<b>Enumeration getHeaderNames()</b> 返回所有HTTP头的名称集合
4	<b>Enumeration getParameterNames()</b> 返回请求中所有参数的集合
5	<b>HttpSession getSession()</b> 返回request对应的session对象，如果没有，则创建一个
6	<b>HttpSession getSession(boolean create)</b> 返回request对应的session对象，如果没有并且参数create为true，则返回一个新的session对象
7	<b>Locale getLocale()</b> 返回当前页的Locale对象，可以在response中设置
8	<b>Object getAttribute(String name)</b> 返回名称为name的属性值，如果不存在则返回null。
9	<b>ServletInputStream getInputStream()</b> 返回请求的输入流
10	<b>String getAuthType()</b> 返回认证方案的名称，用来保护servlet，比如 "BASIC" 或者 "SSL" 或 null 如果 JSP没设置保护措施
11	<b>String getCharacterEncoding()</b> 返回request的字符编码集名称
12	<b>String getContentType()</b> 返回request主体的MIME类型，若未知则返回null
13	<b>String getContextPath()</b> 返回request URI中指明的上下文路径

14	<b>String getHeader(String name)</b> 返回name指定的信息头
15	<b>String getMethod()</b> 返回此request中的HTTP方法，比如 GET, , POST, 或PUT
16	<b>String getParameter(String name)</b> 返回此request中name指定的参数，若不存在则返回null
17	<b>String getPathInfo()</b> 返回任何额外的与此request URL相关的路径
18	<b>String getProtocol()</b> 返回此request所使用的协议名和版本
19	<b>String getQueryString()</b> 返回此 request URL包含的查询字符串
20	<b>String getRemoteAddr()</b> 返回客户端的IP地址
21	<b>String getRemoteHost()</b> 返回客户端的完整名称
22	<b>String getRemoteUser()</b> 返回客户端通过登录认证的用户，若用户未认证则返回null
23	<b>String getRequestURI()</b> 返回request的URI
24	<b>String getRequestedSessionId()</b> 返回request指定的session ID
25	<b>String getServletPath()</b> 返回所请求的servlet路径
26	<b>String[] getParameterValues(String name)</b> 返回指定名称的参数的所有值，若不存在则返回null
27	<b>boolean isSecure()</b> 返回request是否使用了加密通道，比如HTTPS
28	<b>int getContentLength()</b>

	返回request主体所包含的字节数，若未知的返回-1
29	<b>int getIntHeader(String name)</b> 返回指定名称的request信息头的值
30	<b>int getServerPort()</b> 返回服务器端口号

## HTTP信息头示例

在这个例子中，我们会使用HttpServletRequest类的getHeaderNames()方法来读取HTTP信息头。这个方法以枚举的形式返回当前HTTP请求的头信息。

获取Enumeration对象后，用标准的方式来遍历Enumeration对象，用hasMoreElements()方法来确定什么时候停止，用nextElement()方法来获得每个参数的名字。

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ page import="java.io.*,java.util.*" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="utf-8">
<title>菜鸟教程(runoob.com)</title>
</head>
<body>
<h2>HTTP 头部请求实例</h2>
<table width="100%" border="1" align="center">
<tr bgcolor="#949494">
<th>Header Name</th><th>Header Value(s)</th>
</tr>
<%
    Enumeration headerNames = request.getHeaderNames();
    while(headerNames.hasMoreElements()) {
        String paramName = (String)headerNames.nextElement();
        out.print("<tr><td>" + paramName + "</td>\n");
        String paramValue = request.getHeader(paramName);
        out.println("<td> " + paramValue + "</td></tr>\n");
    }
%>
</table>
</body>
</html>
```

访问main.jsp，将会得到以下结果：

# HTTP 头部请求实例

Header Name	Header Value(s)
host	localhost:8080
connection	keep-alive
cache-control	max-age=0
upgrade-insecure-requests	1
user-agent	Mozilla/5.0 (Macintosh; Intel Mac OS X 10_11_1) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/51.0.2704.103 Safari/537.36
accept	text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,*/*;q=0.8
accept-encoding	gzip, deflate, sdch
accept-language	zh-CN,zh;q=0.8
cookie	JSESSIONID=EA4479FBC215E2F4F873E02C4D5A774B; CNZZDATA1254569789=662054999-1452676649-%7C1454050881

您可以在上面代码中尝试HttpServletRequest类的其它方法。

← JSP 隐式对象

JSP 服务器响应 →

 点我分享笔记