

Servlet HTTP 状态码

HTTP 请求和 HTTP 响应消息的格式是类似的，结构如下：

- 初始状态行 + 回车换行符（回车+换行）
- 零个或多个标题行+回车换行符
- 一个空自行，即回车换行符
- 一个可选的消息主体，比如文件、查询数据或查询输出

例如，服务器的响应头如下所示：

```
HTTP/1.1 200 OK
Content-Type: text/html
Header2: ...
...
HeaderN: ...
(Blank Line)
<!doctype ...>
<html>
<head>...</head>
<body>
...
</body>
</html>
```

状态行包括 HTTP 版本（在本例中为 HTTP/1.1）、一个状态码（在本例中为 200）和一个对应于状态码的短消息（在本例中为 OK）。

以下是可能从 Web 服务器返回的 HTTP 状态码和相关的信息列表：

代码	消息	描述
100	Continue	只有请求的一部分已经被服务器接收，但只要它没有被拒绝，客户端应继续该请求。
101	Switching Protocols	服务器切换协议。
200	OK	请求成功。
201	Created	该请求是完整的，并创建一个新的资源。

202	Accepted	该请求被接受处理，但是该处理是不完整的。
203	Non-authoritative Information	
204	No Content	
205	Reset Content	
206	Partial Content	
300	Multiple Choices	链接列表。用户可以选择一个链接，进入到该位置。最多五个地址。
301	Moved Permanently	所请求的页面已经转移到一个新的 URL。
302	Found	所请求的页面已经临时转移到一个新的 URL。
303	See Other	所请求的页面可以在另一个不同的 URL 下被找到。
304	Not Modified	
305	Use Proxy	
306	<i>Unused</i>	在以前的版本中使用该代码。现在已不再使用它，但代码仍被保留。
307	Temporary Redirect	所请求的页面已经临时转移到一个新的 URL。
400	Bad Request	服务器不理解请求。
401	Unauthorized	所请求的页面需要用户名和密码。
402	Payment Required	<i>您还不能使用该代码。</i>
403	Forbidden	禁止访问所请求的页面。
404	Not Found	服务器无法找到所请求的页面。
405	Method Not Allowed	在请求中指定的方法是不允许的。

406	Not Acceptable	服务器只生成一个不被客户端接受的响应。
407	Proxy Authentication Required	在请求送达之前，您必须使用代理服务器的验证。
408	Request Timeout	请求需要的时间比服务器能够等待的时间长，超时。
409	Conflict	请求因为冲突无法完成。
410	Gone	所请求的页面不再可用。
411	Length Required	"Content-Length" 未定义。服务器无法处理客户端发送的不带 Content-Length 的请求信息。
412	Precondition Failed	请求中给出的先决条件被服务器评估为 false。
413	Request Entity Too Large	服务器不接受该请求，因为请求实体过大。
414	Request-url Too Long	服务器不接受该请求，因为 URL 太长。当您转换一个 "post" 请求为一个带有长的查询信息的 "get" 请求时发生。
415	Unsupported Media Type	服务器不接受该请求，因为媒体类型不被支持。
417	Expectation Failed	
500	Internal Server Error	未完成的请求。服务器遇到了一个意外的情况。
501	Not Implemented	未完成的请求。服务器不支持所需的功能。
502	Bad Gateway	未完成的请求。服务器从上游服务器收到无效响应。
503	Service Unavailable	未完成的请求。服务器暂时超载或死机。
504	Gateway Timeout	网关超时。
505	HTTP Version Not Supported	服务器不支持"HTTP 协议"版本。

设置 HTTP 状态代码的方法

下面的方法可用于在 Servlet 程序中设置 HTTP 状态码。这些方法通过 *HttpServletResponse* 对象可用。

序号	方法 & 描述
1	public void setStatus (int statusCode) 该方法设置一个任意的状态码。 <code>setStatus</code> 方法接受一个 <code>int</code> (状态码) 作为参数。如果您的响应包含了一个特殊的状态码和文档, 请确保在使用 <i>PrintWriter</i> 实际返回任何内容之前调用 <code>setStatus</code> 。
2	public void sendRedirect(String url) 该方法生成一个 302 响应, 连同同一个带有新文档 URL 的 <i>Location</i> 头。
3	public void sendError(int code, String message) 该方法发送一个状态码 (通常为 404), 连同同一个在 HTML 文档内部自动格式化并发送到客户端的短消息。

HTTP 状态码实例

下面的例子把 407 错误代码发送到客户端浏览器, 浏览器会显示 "Need authentication!!!" 消息。

```
// 导入必需的 java 库
import java.io.*;
import javax.servlet.*;
import javax.servlet.http.*;
import java.util.*;
import javax.servlet.annotation.WebServlet;

@WebServlet("/showError")
// 扩展 HttpServletResponse 类
public class showError extends HttpServletResponse {

    // 处理 GET 方法请求的方法
    public void doGet(HttpServletRequest request,
                      HttpServletResponse response)
        throws ServletException, IOException
```



```
{
    // 设置错误代码和原因
    response.sendError(407, "Need authentication!!!" );
}
// 处理 POST 方法请求的方法
public void doPost(HttpServletRequest request,
                   HttpServletResponse response)
    throws ServletException, IOException {
    doGet(request, response);
}
}
```

现在，调用上面的 Servlet 将显示以下结果：

HTTP Status 407 - Need authentication!!!

type Status report

message Need authentication!!!

description The client must first authenticate itself with the proxy (Need authentication!!!).

Apache Tomcat/5.5.29