



链滴

HttpServletRequest HTTP 请求中文乱码以及 HttpServletResponse 响应中文乱码处理

作者: [hjljy](#)

原文链接: <https://ld246.com/article/1635240358811>

来源网站: 链滴

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)



在项目的开发过程当中，请求乱码是一件非常常见的事情，特此记录下如何处理请求和响应中文乱码问题，尽管问题很容易处理，但是有时一个不注意半天都搞不定。

HttpServletRequest HTTP请求中文乱码

当读取请求参数出现中文乱码的时候，首先确定传过来的参数是否乱码，然后在继续处理，不然搞半天发现前端参数是乱码，就很难受了。

例如：HTTP 的Header参数不支持中文，传递中文参数的时候需要进行转码。

处理方式：JS: `encodeURIComponent(param)` JAVA: `URLDecoder.decode("待解析字符串", "UTF-8")`

当确定是后台读取的时候乱码的时候，处理方式非常简单就是设置请求的编码方式即可。

```
request.setCharacterEncoding("UTF-8");
```

ps: 注意事项，根据官方文档的说明，该方法如果在读取请求参数或者`getReader()`方法调用之后设置是不会生效的。

ps: 官方文档是通过IDEA点击方法，进入到方法当中，然后点击右上角的Download sources进行下载的。

ps: [springboot controller中请求参数中文乱码](#) 这篇文章算是一个非常特别的例子，乱码的原因还和启动时的文件编码有关系。

HttpServletResponse 响应中文乱码处理

当返回给前端的数据出现乱码的时候，首先通过断点确定在数据封装到`HttpServletResponse`里面的时候是否乱码，然后查看请求返回时的编码是什么。

如果没有乱码，或者返回的编码不对的时候，基本上也是通过设置`HttpServletResponse` 编码进行处理

。

```
response.setCharacterEncoding("UTF-8");
```

ps: 注意事项, 根据官方文档的说明, 该方法如果在getWriter()方法调用之后设置, 是不会生效的。

统一处理方式

```
/**
 * 编码过滤器
 *
 * @author hjljy
 * @date 2021/10/26
 */
@WebFilter(urlPatterns = "/*")
@Order(Ordered.HIGHEST_PRECEDENCE)
public class EncodeFilter implements Filter {
    @Override
    public void doFilter(ServletRequest servletRequest, ServletResponse servletResponse, Filter
    chain) throws IOException, ServletException {
        HttpServletRequest request = (HttpServletRequest) servletRequest;
        HttpServletResponse response = (HttpServletResponse) servletResponse;
        System.out.println(123);
        request.setCharacterEncoding("UTF-8");
        response.setCharacterEncoding("UTF-8");
        chain.doFilter(request, response);
    }
}
```

然后在启动类上加@ServletComponentScan注解即可。这样可以很大程度避免在调用getWriter()者getReader()折后设置。