# 解决中文乱码问题

## 字符集

1. 字符(Character)是各种文字和符号的总称，包括各个国家文字、标点符号、图形符号、数字等。字符集(Character set)是多个字符的集合。
2. 任何数据在计算机内部实际上都是以二进制的形式保存的，所以我们看得见的字符本质上都是以二进制的形式保存的。而字符集实际上就是看得见的字符和二进制数之间的一张对应表。

|  |  |  |  |
| --- | --- | --- | --- |
| 汉字 | 编码方式 | 编码 | 二进制 |
| ‘中’ | GB2312 | D6D0 | 1101 0110-1101 0000 |
| UTF-16 | 4E2D | 0100 1110-0010 1101 |
| UTF-8 | E4B8AD | 1110 0100-1011 1000-1010 1101 |

## 编码与解码

1. 编码：将字符转换为二进制数

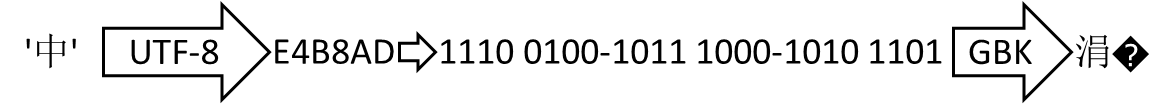
‘中’ → E4B8AD → 1110 0100-1011 1000-1010 1101

1. 解码：将二进制数转换为字符

1110 0100-1011 1000-1010 1101 → E4B8AD → ‘中’

1. 乱码：编码与解码使用的字符集不一致就会产生乱码。

如下图：使用 UTF-8 编码，使用GBK 解码



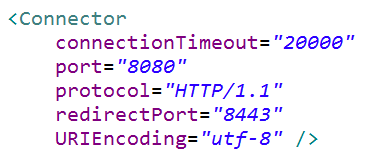
**所以**，解决乱码问题的根本方法就是**统一编码和解码的字符集**。

## 在Web应用中统一字符集

1. 请求参数中含有中文

* **请求由浏览器发送给服务器**
  + 由浏览器编码
    - 浏览器使用HTML页面指定的字符集编码，只需要将HTML页面的字符集指定为UTF-8即可
  + 由服务器解码
    - 默认字符集是IOS-8859-1
  + 解决方案
* **对于POST请求**：post请求在Servlet中解码，默认的字符集是ISO-8859-1，不支持中文，所以我们需要修改Servlet的字符集为UTF-8。使用request.setCharacterEncoding("UTF-8");设置request对象使用UTF-8字符集对**请求体**中的内容进行解码。**但需要注意的是**，这个操作一定要在调用getParameter()方法之前进行。
* **对于GET请求**：GET请求是在URL地址栏中传递请求参数的，它会被Tomcat服务器自动解码，而Tomcat服务器默认的字符集也是ISO-8859-1，

所以我们需要修改Tomcat服务器的字符集为UTF-8。由于请求参数是包含在**请求行内**的，无法通过request对象进行设置，所以需要在Server.xml文件中修改Connector标签，添加URIEncoding="utf-8"属性。



在eclipse中的Tomcat镜像中修改即可生效。但如果将来删除当前Tomcat镜像，重新关联Tomcat，则会使设置失效，所以在Tomcat配置文件中最好也进行配置。

1. 服务器响应中含有中文

* **响应由服务器发送给浏览器**
  + 由服务器编码
    - 默认使用ISO-8859-1进行编码
  + 由浏览器解码
    - 默认使用GBK进行解码
  + 解决方案
* 方法1：设置响应头

response.setHeader("Content-Type","text/html;charset=utf-8");

* 方法2：设置响应的内容类型

response.setContentType("text/html;charset=utf-8");

* **通过这种方式可以在响应头中告诉浏览器响应体的编码方式是UTF-8；同时服务器也会采用该字符集进行编码**

**但需要注意的是**，两种方法一定要在response.getWriter()之前进行。