

JSP常用的内置对象pageContext request response session application

POST:

```
request.setCharacterEncoding("UTF-8");  
String name = request.getParameter("name");  
String pwd = request.getParamter("pwd");
```

GET:

```
String name = request.getParameter("name");  
name = new String(name.getBytes("ISO-8859-1"), "UTF-8");  
String pwd = request.getParameter("pwd");  
pwd = new String(pwd.getBytes("ISO-8859-1"), "UTF-8");
```

response常用方法:

- sendRedirect(String path) //重定向, 完成页面跳转

```
request.getRequestDispatcher(String path).forward(request, response); //转发  
  
response.sendRedirect(String path); //重定向
```

转发和重定向的区别:

- 1.通过转发的形式完成页面跳转, 是同一个请求。A —》 B
- 2.通过重定向的方式完成页面跳转, 是两个请求。A —〉 B
- 3.通过转发的形式完成页面跳转, 地址栏url不变。A--->B 地址栏显示A, 结果是B。
- 4.通过重定向的方式完成页面跳转, 地址栏url改变。A--->B 地址栏开始显示A, 然后切换为B。
- 5.转发称之为服务器跳转。
- 6.重定向称之为客户端跳转。

如果在跳转页面的同时需要传递数据, 同时数据通过request作为载体来传输, 只能使用转发, 不能使用重定向, 因为重定向会创建新的request, 导致无法取出之前request中的数据。

如果不需要传输数据, 可以使用重定向。

- setContentType("text/html;charset=UTF-8"); //解决响应时的中文乱码问题

```
resp.setContentType("text/html;charset=UTF-8");  
PrintWriter printWriter = resp.getWriter();  
printWriter.write(message);
```

session

会话，会话产生的前提？

服务器可以接收到客户端请求，并且对这些请求进行处理，将处理结果响应回客户端。但是服务器无法识别每一次http请求是从哪个客户端发出的，它只会接收到一个请求信号。需要一种技术来区分不同的请求，以便将响应返回给正确的请求，这种技术就是会话。

会话：就是客户端和服务器之间发生的一系列连续的请求和响应的过程，打开浏览器进行操作到关闭浏览器的过程。

会话状态是指服务器和浏览器在会话过程中产生的状态信息，借助于会话状态，服务器就能够把属于同一会话的一系列请求和响应的过程联系起来。

Java WEB开发中实现会话有两种方式：

1.session

2.cookie

服务器每天要接收大量的请求，如何区分哪些请求属于同一个会话呢？即如何能识别出来自同一个浏览器的访问请求，这就需要浏览器对其发出的每一个请求都进行识别，属于同一个会话中的请求都有相同的标识，这就是sessionID。

session常用的方法：

1.String getId() //获取sessionID

2.void setMaxInactiveInterval(int interval) //设置session失效时间，单位为秒

3.int getMaxInactiveInterval() //获取session失效时间

4.void invalidate() //设置session立即失效

5.void setAttribute(String key,Object value) //以键值对的形式用session保存数据

6.Object getAttribute(String key) //通过key值获取数据

7.void removeAttribute(String key) //通过key值删除数据

