session对象：

是HttpSession接口的一个对象

session最主要的作用是区分不同的用户，每一个session就表示一个用户

最主要的应用就是判断用户是否登录，常用方法如下：

|  |  |  |
| --- | --- | --- |
| NO | 方法 | 描述 |
| 1 | boolean isNew() | 判断是否是新的session |
| 2 | String getId() | 获取session值 |
| 3 | void invalidate() | 让session失效 |

session中的主要操作：

对于服务器来说，每个浏览器都是通过session id进行区分，浏览器在第一次连接的时候就会分配此id ,不会要手工设置

·取得session id

|  |
| --- |
| <h1><%=session.getId() %></h1> |

·让session失效

每次执行invaliddate ()方法的时候，就表示session失效

|  |
| --- |
| <%@page contentType=*"text/html; charset=GBK"*%>  <%  session.setAttribute("name","Wanczy");  %>  <h1><%=session.getId() %></h1>  <%  session.invalidate();  %>  <h1><%=session.getAttribute("name")%></h1> |

·最终要的作用在于系统登录

在所用的系统中，session对象使用最多的操作就是设置和取得属性

·判断是否是新的session

当客户端第一次连接到服务器的时候，可以通过isNew方法来判断此用户是否是新用户

|  |
| --- |
| <%@page contentType=*"text/html; charset=GBK"*%>  <%  **if**(session.isNew()){  %>  <h1>欢迎新用户</h1>  <%  }**else**{    %>  <h1>你是老人</h1>  <%  }  %> |

session与Cookie的关系：

|  |
| --- |
| <%@page contentType=*"text/html; charset=GBK"*%>  <%  Cookie c[] = request.getCookies();  **if**(c!=**null**){  **for**(**int** i =0;i<c.length;i++){  %>  <h1><%=c[i].getName() %> ------<%=c[i].getValue() %></h1>  <%  }  }  %>  <h1><%=session.getId() %></h1> |

通过代码可以看出，取得的session ID 和 Cookie里面的 JSESSIONID 是同一个值，实际上第一次访问的时候，服务器分配的session 就是存放到了Cookie里面，就是说session使用Cookie的处理机制