目录

[Session的实际应用 1](#_Toc497462105)

[登陆和注销 1](#_Toc497462106)

[基于Session的购物车案例 2](#_Toc497462107)

[前台页面的实现 4](#_Toc497462108)

[验证码的使用 4](#_Toc497462109)

[防止表单重复提交 5](#_Toc497462110)

# Session的实际应用

## 登陆和注销

登陆验证实现的步骤:

后台:

1.创建一张用户表:t\_user

2.创建domain（实体）对象:User

3.创建dao(IUserDAO,UserDAOImpl)

4.创建DAO的测试类

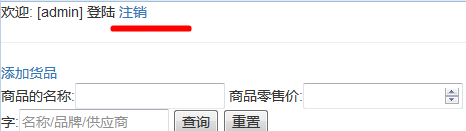
5.实现DAO的操作

前台:

1.创建登陆页面(登陆表单)

2.创建Servlet处理请求





## 基于Session的购物车案例

生活中的购物车:在商场中选中心仪的商品,丢进购物车,购物完了之后,推着购物车去结账

网络商城购物车:在商城中选中商品加入购物车

如果没有购物车,每次选中一个商品就要去结算

反之,就可以一次选中多个商品,最后一起结算

---------------------------------------------------------

网络中的购物车的实现方式:

1.使用Cookie来实现---->缺点在于是客户端的技术,不同的客户端不能共享数据

2.使用Session来实现---->缺点在于Session是会话Cookie,关闭浏览器就没了

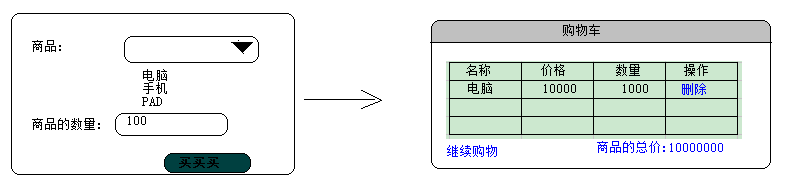
3.使用Cookie+数据库---->在用户登录之前使用Cookie暂时保存数据,登录之后就将数据保存到数据库中

------------------------------------------------------

这里使用Session来实现购物车的的流程





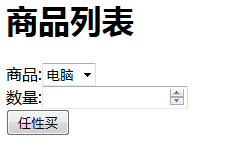


**购物车的实现(业务方法的实现):**

**1.封装商品对象**

**2.购物车的封装**

## 前台页面的实现

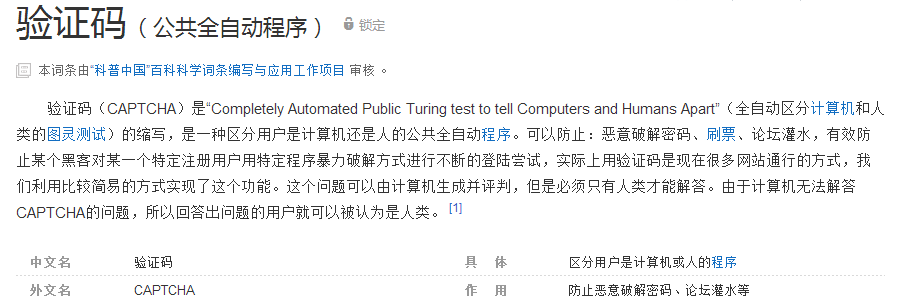


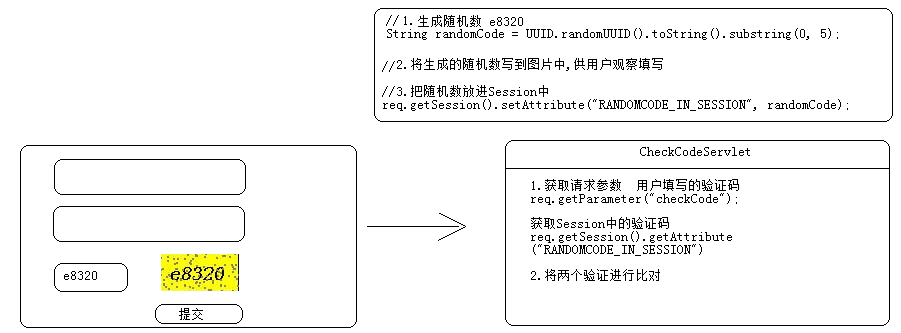
## 验证码的使用

**一次性验证码**的主要目的就是为了限制人们利用工具软件来暴力猜测密码，其原理与利用Session防止表单重复提交的原理基本一样，只是将表单标识号变成了验证码的形式，并且要求用户将提示的验证码手工填写进一个表单字段中，而不是通过表单的隐藏字段自动回传给服务器。

服务器程序接收到表单数据后，首先判断用户是否填写了正确的验证码，只有该验证码与服务器端保存的验证码匹配时，服务器程序才开始正常的表单处理流程。

密码猜测工具要逐一尝试每个密码的前题条件是先输入正确的验证码，而验证码是一次性有效的，这样基本上就阻断了密码猜测工具的自动地处理过程。







## 防止表单重复提交

没有完整的进行一次请求页面->提交页面的过程而完成数据提交

**造成重复提交的原因：**

1.由于服务器缓慢或网络延迟的原因,重复点击提交按钮

2.已经提交成功,刷新成功页面(forward)

3.已经提交成功,通过回退,再次点击提交按钮

注意:

回退后,刷新表单页面,再次提交这时不是重复提交,而是发送新的请求

在Firefox下，重复提交到同一地址无效

**能用URL重定向就使用URL重定向**

**表单重复提交的问题的实质:**

没有先请求页面,就直接在提交数据

**解决方案**:令牌机制

基本操作:

在后台先生成一串随机数,然后将随机数放到Session,同时在JSP页面中使用EL表达式将Session中的数据放在表单的隐藏域中

