## 购物车实现cookie+redis：

一、未登录状态下添加商品到购物车

在不登陆的情况下也可以添加购物车。把购物车信息写入cookie。

优点：

1、不占用服务端存储空间

2、用户体验好。

3、代码实现简单。

缺点：

1、cookie中保存的容量有限。最大4k

2、把购物车信息保存在cookie中，更换设备购物车信息不能同步

实现思路：

（1） 从cookie中获取商品列表信息（单独提出来写成个通用的方法）

（2） 遍历购物车列表，判断需要添加的商品在购物车列表是否存在

（3） 商品存在的话，那么取出该商品原来的数量+添加的数量作为该商品现在的数量

（4） 如果商品不存在，那么调用服务，根据传来的商品id查询商品数量，设置商品的数量为页面传来的数量，取商品的第一张图片(购物车列表只展示一张图片)。

（5） 把修改后的购物车列表重新存入到cookie中

（6） 返回逻辑视图”cartSuccess”

2、展示购物车列表

单击“去购物车结算按钮”向服务端发送请求，服务端应该返回逻辑视图”cart”

请求地址:8090/cart/cart.html

返回逻辑视图:”cart”也就是购物车列表页面

3、未登录状态下购物车列表页面修改商品数量

购物车列表页面单击”+”,”-”会向服务端发送ajax请求。

页面需要根据调整的数量重新显示商品总计（已经实现了也就是输入框的值\*价格）和小计（用js，待实现）

服务端要求修改cookie中对应商品的数量

4、未登录状态下删除购物车商品

（1）从cookie中获取购物车列表   
 （2）遍历，查找到要删除的商品   
 （3）将该商品从购物车列表移除   
 （4）更新后的购物车列表重新写入cookie   
 （5）重定向到购物车列表页面

**二、登录状态下购物车功能的实现**

功能分析：   
 1、购物车数据保存的位置：   
 未登录状态下，把购物车数据保存到cookie中。   
 登录状态下，需要把购物车数据保存到服务端。需要永久保存，可以保存到数据 库中。可以把购物车数据保存到redis中。   
 2、redis使用的数据类型   
 a) 使用hash数据类型   
 b) Hash的key应该是用户id。Hash中的field是商品id，value可以是把商品信息转换成json   
 3、添加购物车   
 登录状态下直接把商品数据保存到redis中。   
 未登录状态保存到cookie中。   
 4、如何判断是否登录？   
 a) 从cookie中取token   
 b) 取不到未登录   
 c) 取到token，到redis中查询token是否过期。   
 d) 如果过期，未登录状态   
 e) 没过期登录状态。

实现：

1、登录拦截器   
 几乎在购物车所有功能执行 都要判断用户是否登录。利用aop思想，应该编写个拦截器，来判断用户是否登录。登录的话用户信息需要存在request域中   
 （1） 从cookie中取token   
 （2） 判断token是否存在   
 （3） 不存在，说明用于未登录，放行   
 （4） 如果token存在，调用服务，根据token从redis中取用户信息   
 （5） 取不到用户信息，说明已经过期，放行   
 （6） 取到了用户信息，说明用户已经登录，用户信息存到request中   
 （7） 放行

2、登录状态下，商品添加功能实现

向redis中添加购物车

比未登录多一个登录判断而已

3、登录状态下，商品列表展示   
分析：   
（1）从cookie中取购物车列表   
（2）判断用户是否登录   
（3）用户已经登录的话，则调用服务层，合并cookie中的列表和redis中的列表。存入到redis中。   
（4）同时删除cookie中的购物车列表   
（5）根据用户id，调用服务查询redis中所有的商品，返回购物车列表。   
（6）未登录状态还是跟前面一样   
（7）将列表绑定到参数，返回购物车列表页面

4、登录状态下修改购物车商品数量   
分析   
单击”+”,”-”修改商品的数量的时候,要求redis中该商品的数量发生改变   
（1） 根据用户id，商品id从redis中取出对应的商品   
（2） 设置商品的数量   
（3） 该商品更新到redis中

5、登录状态下，删除购物车商品   
分析   
单击删除的时候，删除redis中该商品。重定向到列表页面   
（1） 直接用jedisClient的del的方法根据用户id跟商品id 商品   
（2） 返回成功