

# 乐字节教育高级架构课程

正所谓“授人以鱼不如授人以渔”，你们想要的 **Java 学习资料** 来啦！

不管你是学生，还是已经步入职场的同行，希望你们都要珍惜眼前的学习机会，奋斗没有终点，知识永不过时。

扫描下方二维码即可领取



乐字节晓啡



乐字节官方交流群



## rpc服务系统编写购物车返回对象

shop-rpc的CartResult.java

```
package com.xxxx.rpc.pojo;

import com.xxxx.rpc.vo.CartVo;

import java.io.Serializable;
import java.math.BigDecimal;
import java.util.List;

/**
 * 购物车结果对象
 *
 * @author zhoubin
 * @since 1.0.0
 */
public class CartResult implements Serializable {
    private static final long serialVersionUID = -8936811376383018291L;
    // 购物车列表
    private List<CartVo> cartList;

    // 购物车总金额
    private BigDecimal totalPrice;

    public List<CartVo> getCartList() {
        return cartList;
    }

    public void setCartList(List<CartVo> cartList) {
        this.cartList = cartList;
    }

    public BigDecimal getTotalPrice() {
        return totalPrice;
    }

    public void setTotalPrice(BigDecimal totalPrice) {
        this.totalPrice = totalPrice;
    }

    @Override
    public String toString() {
        return "CartResult{" +
            "cartList=" + cartList +
            ", totalPrice=" + totalPrice +
            '}';
    }
}
```

## rpc服务系统修改购物车服务

shop-rpc的CartService.java

```
/**
 * 获取购物车列表
 * @param admin
 * @return
 */
CartResult getCartList(Admin admin);
```

shop-rpc的CartServiceImpl.java

```
/**
 * 获取购物车列表
 *
 * @param admin
 * @return
 */
@Override
public CartResult getCartList(Admin admin) {
    //1.判断用户是否存在
    if (null == admin || null == admin.getAdminId()) {
        return null;
    }
    //初始化返回对象
    CartResult cartResult = null;
    List<CartVo> cartList = null;
    BigDecimal totalPrice = new BigDecimal("0");
    //从redis里获取购物车信息
    hashOperations = redisTemplate.opsForHash();
    Map<String, String> cartMap = hashOperations.entries(userCart + ":" +
admin.getAdminId());
    //判断是否存在购物车信息
    if (!CollectionUtils.isEmpty(cartMap)) {
        //如果存在
        cartResult = new CartResult();
        cartList = new ArrayList<>();
        for (Map.Entry<String, String> entry : cartMap.entrySet()) {
            CartVo cartVo = JsonUtil.jsonStr2Object(entry.getValue(), CartVo.class);
            //将购物车对象放入list
            cartList.add(cartVo);
            //计算总价
            /**
             * 先算单价*数量
             * 在把所有价格相加
             */
            BigDecimal singePrice =
                cartVo.getMarketPrice().multiply(new
BigDecimal(String.valueOf(cartVo.getGoodsNum())));
            totalPrice = totalPrice.add(singePrice);
        }
    }
}
```

```
//将购物车list放入返回结果对象
cartResult.setCartList(cartList);
//格式化2位小数, 四舍五入
totalPrice.setScale(2, BigDecimal.ROUND_HALF_UP);
cartResult.setTotalPrice(totalPrice);
}
return cartResult;
}
```

## 前台系统修改Controller

shop-portal的CartController.java

```
/**
 * 跳转至购物车列表页面
 *
 * @param request
 * @param model
 * @return
 */
@RequestMapping("getCartList")
public String getCartList(HttpServletRequest request, Model model) {
    Admin admin = (Admin) request.getSession().getAttribute("user");
    CartResult cartResult = cartService.getCartList(admin);
    model.addAttribute("cartResult", null == cartResult ? new CartResult() : cartResult);
    return "cart/list";
}
```

