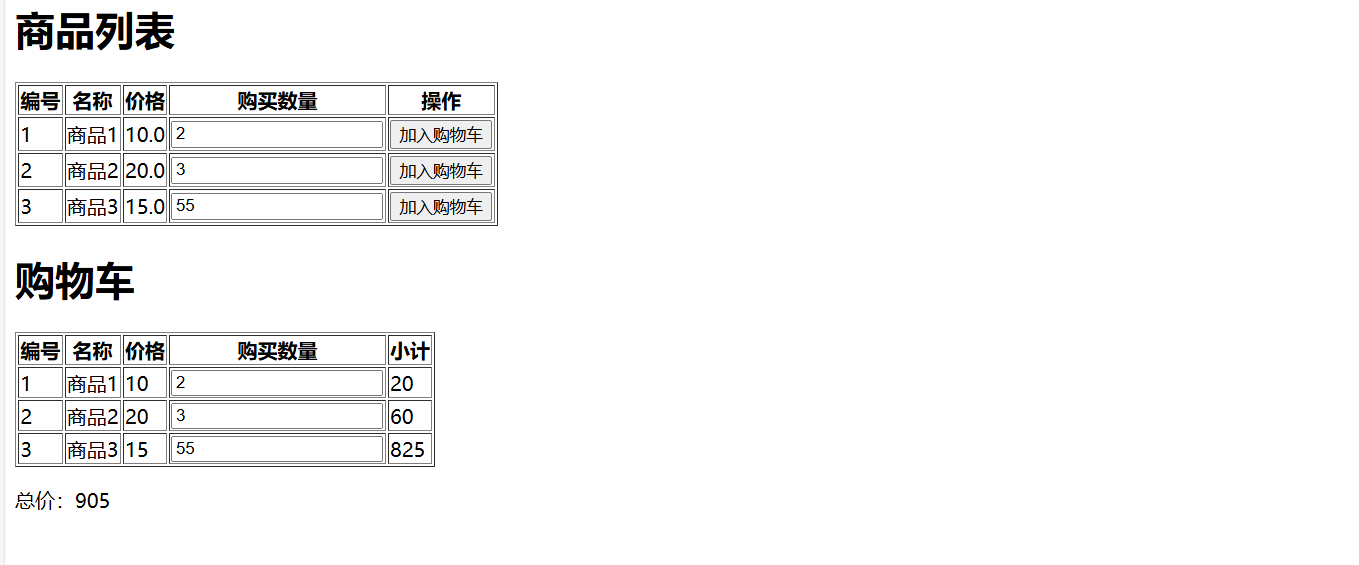
主要代码：

<%@ page import="com.example.javaee.Product" %>  
<%@ page import="java.util.List" %>  
<%@ page import="java.util.ArrayList" %>  
<%@ page contentType="text/html;charset=UTF-8" language="java" %>  
<html>  
<head>  
 <title>商品列表与购物车</title>  
 <script>  
 // 定义函数用于更新总价  
 function updateTotalPrice() {  
 var rows = document.querySelectorAll('table#cart-table tbody tr');  
 rows.forEach(function (row) {  
 var price = parseFloat(row.cells[2].innerText);  
 var quantity = parseInt(row.cells[3].querySelector('input').value);  
 totalPrice += price \* quantity;  
 });  
 document.getElementById('total-price').innerText = totalPrice;  
 }  
  
 // 模拟购物车数据，使用数组存储已加入购物车的商品信息（格式为 {product: 商品对象, quantity: 购买数量}）  
 var cartProducts = [];  
  
 // 加入购物车函数，将商品及购买数量添加到模拟购物车数据中，并更新购物车展示和总价  
 function addToCart(productId, productName, productPrice, quantity) {  
 var existingProductIndex = cartProducts.findIndex(function (item) {  
 return item.product.getId() === productId;  
 });  
 if (existingProductIndex !== -1) {  
 cartProducts[existingProductIndex].quantity += quantity;  
 } else {  
 cartProducts.push({  
 product: {  
 getId: function () { return productId; },  
 getName: function () { return productName; },  
 getPrice: function () { return productPrice; }  
 },  
 quantity: quantity  
 });  
 }  
 updateCartUI();  
 updateTotalPrice();  
 }  
  
 // 更新购物车界面展示函数，根据模拟购物车数据重新渲染购物车表格内容  
 function updateCartUI() {  
 var cartTableBody = document.querySelector('table#cart-table tbody');  
 cartTableBody.innerHTML = '';  
 cartProducts.forEach(function (item) {  
 var product = item.product;  
 var quantity = item.quantity;  
 var row = document.createElement('tr');  
 var idCell = document.createElement('td');  
 idCell.innerText = product.getId();  
 var nameCell = document.createElement('td');  
 nameCell.innerText = product.getName();  
 var priceCell = document.createElement('td');  
 priceCell.innerText = product.getPrice();  
 var quantityCell = document.createElement('td');  
 var quantityInput = document.createElement('input');  
 quantityInput.type = 'number';  
 quantityInput.min = 0;  
 quantityInput.value = quantity;  
 quantityInput.addEventListener('change', function () {  
 item.quantity = parseInt(quantityInput.value);  
 updateTotalPrice();  
 });  
 quantityCell.appendChild(quantityInput);  
 var subtotalCell = document.createElement('td');  
 subtotalCell.innerText = product.getPrice() \* quantity;  
 row.appendChild(idCell);  
 row.appendChild(nameCell);  
 row.appendChild(priceCell);  
 row.appendChild(quantityCell);  
 row.appendChild(subtotalCell);  
 cartTableBody.appendChild(row);  
 });  
 }  
  
 // 绑定加入购物车按钮  
 function bindAddToCartButtons() {  
 var addToCartButtons = document.querySelectorAll('.add-to-cart-btn');  
 addToCartButtons.forEach(function (button) {  
 button.addEventListener('click', function () {  
 var row = button.closest('tr');  
 var productId = parseInt(row.cells[0].innerText);  
 var productName = row.cells[1].innerText;  
 var productPrice = parseFloat(row.cells[2].innerText);  
 var quantity = parseInt(row.cells[3].querySelector('input').value);  
 if (quantity > 0) {  
 addToCart(productId, productName, productPrice, quantity);  
 } else {  
 alert("请选择购买数量");  
 }  
 });  
 });  
 }  
  
 document.addEventListener('DOMContentLoaded', function () {  
 bindAddToCartButtons();  
 });  
 </script>  
</head>  
<body>  
<!-- 商品列表展示部分 -->  
<h1>商品列表</h1>  
<table border="1">  
 <thead>  
 <tr>  
 <th>编号</th>  
 <th>名称</th>  
 <th>价格</th>  
 <th>购买数量</th>  
 <th>操作</th>  
 </tr>  
 </thead>  
 <tbody>  
 <%  
   
 com.example.javaee.Product product1 = new com.example.javaee.Product(1, "商品1", 10.0);  
 com.example.javaee.Product product2 = new com.example.javaee.Product(2, "商品2", 20.0);  
 com.example.javaee.Product product3 = new com.example.javaee.Product(3, "商品3", 15.0);  
 List<Product> products = new ArrayList<>();  
 products.add(product1);  
 products.add(product2);  
 products.add(product3);  
 for (com.example.javaee.Product product : products) {  
 %>  
 <tr>  
 <td><%= product.getId() %></td>  
 <td><%= product.getName() %></td>  
 <td><%= product.getPrice() %></td>  
 <td>  
 <input type="number" min="0" value="0" name="quantity\_<%= product.getId() %>">  
 </td>  
 <td>  
 <button class="add-to-cart-btn">加入购物车</button>  
 </td>  
 </tr>  
 <%  
 }  
 %>  
 </tbody>  
</table>  
  
<!-- 购物车展示部分 -->  
<h1>购物车</h1>  
<table id="cart-table" border="1">  
 <thead>  
 <tr>  
 <th>编号</th>  
 <th>名称</th>  
 <th>价格</th>  
 <th>购买数量</th>  
 <th>小计</th>  
 </tr>  
 </thead>  
 <tbody>  
 </tbody>  
</table>  
<p>总价：<span id="total-price">0</span></p>  
</body>  
</html>

## 结果：





问题：尝试使用java后端加载并且加减商品的数量，然后发现后端连接出现问题，和前端的函数联系先后也有问题，初始化products总是失败，最后成功但是依旧不稳定，难以做进一步的更新，最后选择使用内嵌脚本的方式完成，有待完善。2025.1.10