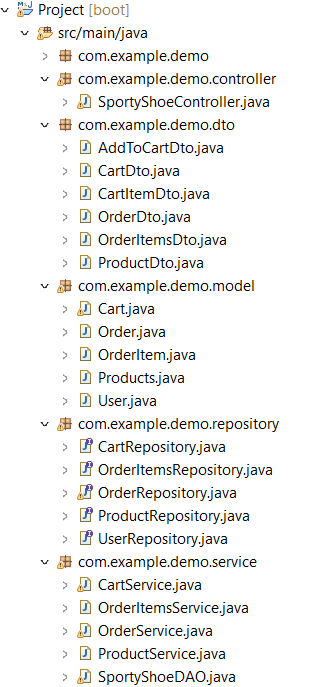
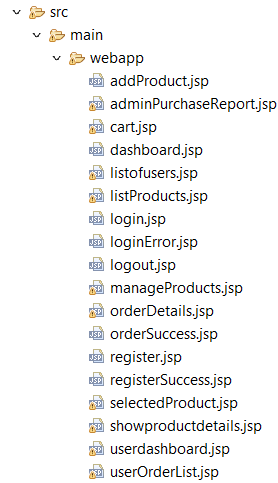
Phase 3 - Implement Frameworks the DevOps way

**Project Source code –**

**Project Structure:**

****

****

**SportyShoeController:**

package com.example.demo.controller;

import java.util.List;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpServletResponse;

import javax.servlet.http.HttpSession;

import com.example.demo.dto.AddToCartDto;

import com.example.demo.dto.CartDto;

import com.example.demo.model.Cart;

import com.example.demo.model.Order;

import com.example.demo.model.OrderItem;

import com.example.demo.model.Products;

import com.example.demo.model.User;

import com.example.demo.repository.CartRepository;

import com.example.demo.service.CartService;

import com.example.demo.service.OrderItemsService;

import com.example.demo.service.OrderService;

import com.example.demo.service.SportyShoeDAO;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Controller;

import org.springframework.web.bind.annotation.RequestMapping;

import org.springframework.web.servlet.ModelAndView;

@Controller

public class SportyShoeController {

@Autowired

SportyShoeDAO dao;

@Autowired

private CartService cartService;

@Autowired

private CartRepository cartRepository;

@Autowired

private OrderService orderService;

@Autowired

private OrderItemsService orderItemService;

@RequestMapping("/login")

public ModelAndView login(HttpServletRequest req, HttpServletResponse res) {

ModelAndView mv = new ModelAndView();

String userName = req.getParameter("username");

String userPassword = req.getParameter("password");

if (dao.findbyuser(userName).equals(dao.findbypassword(userPassword))) {

int id = dao.getUserID(req.getParameter("username"));

HttpSession session = req.getSession();

session.setAttribute("userName", userName);

session.setAttribute("ID", id);

if (id == 4) {

mv.setViewName("dashboard.jsp");

return mv;

} else {

List<Products> list = dao.getAll();

req.setAttribute("list", list);

mv.setViewName("userdashboard.jsp");

return mv;

}

} else {

mv.setViewName("loginError.jsp");

return mv;

}

}

// Dashboard page for Users

@RequestMapping("/userdashboard")

public ModelAndView userDashboard(HttpServletRequest req, HttpServletResponse res) {

ModelAndView mv = new ModelAndView();

List<Products> list = dao.getAll();

req.setAttribute("list", list);

mv.setViewName("userdashboard.jsp");

return mv;

}

@RequestMapping("/register")

public ModelAndView registerUser(HttpServletRequest req, HttpServletResponse res) {

String userFirstName = req.getParameter("fname");

String userLastName = req.getParameter("lname");

String userEmail = req.getParameter("useremail");

String userName = req.getParameter("username");

String userPassword = req.getParameter("password");

ModelAndView mv = new ModelAndView();

User pojo = new User();

pojo.setFirstName(userFirstName);

pojo.setLastName(userLastName);

pojo.setEmail(userEmail);

pojo.setUserName(userName);

pojo.setPassword(userPassword);

User pJ = dao.registerUser(pojo);

if (pJ != null) {

mv.setViewName("registerSuccess.jsp");

}

return mv;

}

@RequestMapping("/getproductlist")

public ModelAndView getProductList(HttpServletRequest req, HttpServletResponse res) {

ModelAndView mv = new ModelAndView();

List<Products> list = dao.getAll();

req.setAttribute("list", list);

mv.setViewName("listProducts.jsp");

return mv;

}

// Displaying the selected product details to another page

@RequestMapping("/productTypeSelection")

public ModelAndView showProduct(HttpServletRequest req, HttpServletResponse res) {

ModelAndView mv = new ModelAndView();

int id = Integer.parseInt(req.getParameter("id"));

List<Products> list = dao.getByProductID(id);

req.setAttribute("list", list);

mv.setViewName("selectedProduct.jsp");

return mv;

}

// Add item to the cart

@RequestMapping("/addtocart")

public ModelAndView addToCart(HttpServletRequest req, HttpServletResponse res) {

HttpSession session = req.getSession();

int productID = Integer.parseInt(req.getParameter("productID"));

String productname = req.getParameter("productName");

Products product = new Products();

product.setId(productID);

product.setName(productname);

int price = Integer.parseInt(req.getParameter("productPrice"));

int quantity = Integer.parseInt(req.getParameter("quantity"));

int total = quantity \* price;

AddToCartDto addToCartDto = new AddToCartDto(productID, quantity, total);

User user = new User();

user.setId((int) session.getAttribute("ID"));

ModelAndView mv = new ModelAndView();

cartService.addToCart(addToCartDto, product, user);

mv.setViewName("/userdashboard");

return mv;

}

// List items in the cart

@RequestMapping("/listcart")

public ModelAndView getCartItems(HttpServletRequest req, HttpServletResponse res) {

HttpSession session = req.getSession();

ModelAndView mv = new ModelAndView();

User user = new User();

user.setId((int) session.getAttribute("ID"));

CartDto cartDto = cartService.listCartItems(user);

List<Cart> cartList = cartRepository.findAllByUserOrderByCreatedDateDesc(user);

int totalCost = cartDto.getTotalCost();

req.setAttribute("list", cartList);

req.setAttribute("totalCost", totalCost);

mv.setViewName("cart.jsp");

return mv;

}

// place order after checkout

@RequestMapping("/purchaseOrder")

public ModelAndView placeOrder(HttpServletRequest req, HttpServletResponse res) {

HttpSession session = req.getSession();

// retrieve user

ModelAndView mv = new ModelAndView();

User user = new User();

user.setId((int) session.getAttribute("ID"));

// place the order

orderService.placeOrder(user);

mv.setViewName("orderSuccess.jsp");

return mv;

}

// get all orders of User

@RequestMapping("/listPurchaseOrder")

public ModelAndView getAllOrders(HttpServletRequest req, HttpServletResponse res) {

ModelAndView mv = new ModelAndView();

HttpSession session = req.getSession();

// retrieve user

int id = (int) session.getAttribute("ID");

// get orders

List<Order> orderDtoList = orderService.listOrders(id);

req.setAttribute("orderDtoList", orderDtoList);

mv.setViewName("userOrderList.jsp");

return mv;

}

// get orderitems for an order

@RequestMapping("/getOrderItems{id}")

public ModelAndView getOrderById(HttpServletRequest req, HttpServletResponse res) {

ModelAndView mv = new ModelAndView();

int id = Integer.parseInt(req.getParameter("id"));

Order order = new Order();

order.setId(id);

List<OrderItem> orderItemList = orderItemService.listOrders(order);

req.setAttribute("orderItemList", orderItemList);

mv.setViewName("orderDetails.jsp");

return mv;

}

// Admin access - listing signed-up users

@RequestMapping("/listusers")

public ModelAndView listUsers(HttpServletRequest req, HttpServletResponse res) {

ModelAndView mv = new ModelAndView();

List<User> list = dao.listofUsers();

req.setAttribute("list", list);

mv.setViewName("listofusers.jsp");

return mv;

}

// Admin access - Add a product

@RequestMapping("/adminAddProduct")

public ModelAndView addProduct(HttpServletRequest req, HttpServletResponse res) {

String productName = req.getParameter("productName");

String price = req.getParameter("productPrice");

int productPrice = Integer.parseInt(price);

Products product = new Products();

product.setName(productName);

product.setPrice(productPrice);

Products pro = dao.addproduct(product);

ModelAndView mv = new ModelAndView();

if (pro != null) {

mv.setViewName("manageProducts.jsp");

}

return mv;

}

// Admin access- Get all purchased order

@RequestMapping("/listAllPurchaseOrder")

public ModelAndView getAllPurchaseOrders(HttpServletRequest req, HttpServletResponse res) {

ModelAndView mv = new ModelAndView();

// get orders

List<Order> allOrderList = orderService.listAllOrders();

req.setAttribute("allOrderList", allOrderList);

mv.setViewName("adminPurchaseReport.jsp");

return mv;

}

}

**AddToCartDto:**

package com.example.demo.dto;

import lombok.Data;

@Data

public class AddToCartDto {

private int id;

private int productId;

private int quantity;

private int totalPrice;

public AddToCartDto() {

}

public AddToCartDto(int productId, int quantity, int total) {

this.productId = productId;

this.totalPrice = total;

this.quantity = quantity;

}

}

**CartDto:**

package com.example.demo.dto;

import lombok.Data;

import java.util.List;

@Data

public class CartDto {

private List<CartItemDto> cartItems;

private int totalCost;

public CartDto(List<CartItemDto> cartItemDtoList, int totalCost) {

this.cartItems = cartItemDtoList;

this.totalCost = totalCost;

}

public CartDto() {

}

}

CartItemDto:

package com.example.demo.dto;

import com.example.demo.model.Cart;

import com.example.demo.model.Products;

import lombok.Data;

@Data

public class CartItemDto {

private int id;

private int quantity;

private Products product;

private String productName;

public CartItemDto() {

}

public CartItemDto(Cart cart) {

this.setId(cart.getId());

this.setQuantity(cart.getQuantity());

this.setProduct(cart.getProduct());

this.setProductName(cart.getProductName());

}

}

**OrderDto:**

package com.example.demo.dto;

import com.example.demo.model.Order;

import lombok.Data;

@Data

public class OrderDto {

private int id;

private int userId;

public OrderDto(Order order) {

this.setId(order.getId());

}

}

**OrderItemsDto**:

package com.example.demo.dto;

import lombok.Data;

@Data

public class OrderItemsDto {

private int price;

private int quantity;

private int orderId;

private int productId;

public OrderItemsDto () {}

public OrderItemsDto(int price, int quantity, int orderId, int productId) {

this.price = price;

this.quantity = quantity;

this.orderId = orderId;

this.productId = productId;

}

}

**ProductDto**:

package com.example.demo.dto;

import com.example.demo.model.Products;

import lombok.Data;

@Data

public class ProductDto {

private Integer id;

private String name;

private int price;

public ProductDto(Products product) {

this.setId(product.getId());

this.setName(product.getName());

this.setPrice(product.getPrice());

}

public ProductDto(String name, int price) {

this.name = name;

this.price = price;

}

}

**Cart**:

package com.example.demo.model;

import java.util.Date;

import javax.persistence.\*;

import com.example.demo.dto.AddToCartDto;

import org.springframework.beans.factory.annotation.Autowired;

import lombok.Data;

@Entity

@Data

@Table(name = "cart")

public class Cart {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int id;

private Date createdDate;

private String productName;

private int quantity;

private int total;

@ManyToOne

@JoinColumn(name = "product\_id", referencedColumnName = "id")

private Products product;

@OneToOne(targetEntity = User.class, fetch = FetchType.EAGER)

@JoinColumn(nullable = false, name = "user\_id")

private User user;

public Cart() { }

public Cart(User user) {

this.user=user;

}

public Cart(Products product, String name, int quantity,int total, User user){

this.user = user;

this.product = product;

this.productName = name;

this.quantity = quantity;

this.total = total;

this.createdDate= new Date();

}

}

**Order**:

package com.example.demo.model;

import java.util.Date;

import java.util.List;

import javax.persistence.Entity;

import javax.persistence.FetchType;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.JoinColumn;

import javax.persistence.ManyToOne;

import javax.persistence.OneToMany;

import javax.persistence.Table;

import lombok.Data;

@Entity

@Data

@Table(name="orders")

public class Order {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int id;

private Date createdDate;

private int totalPrice;

@OneToMany(mappedBy = "order", fetch = FetchType.LAZY)

private List<OrderItem> orderItems;

@ManyToOne()

@JoinColumn(name = "userId", referencedColumnName = "id")

private User user;

public Order() {

}

}

**OrderItem**:

package com.example.demo.model;

import java.util.Date;

import javax.persistence.Column;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.JoinColumn;

import javax.persistence.ManyToOne;

import javax.persistence.OneToOne;

import javax.persistence.Table;

import lombok.Data;

@Entity

@Data

@Table(name = "orderitems")

public class OrderItem {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Integer id;

@Column(name = "product\_name")

private String productName;

@Column(name = "quantity")

private int quantity;

@Column(name = "price")

private int price;

@Column(name = "created\_date")

private Date createdDate;

@ManyToOne

@JoinColumn(name = "order\_id", referencedColumnName = "id")

private Order order;

@OneToOne

@JoinColumn(name = "product\_id", referencedColumnName = "id")

private Products product;

public OrderItem(){}

public OrderItem(Order order, Products product, int quantity, int price) {

this.product = product;

this.productName = product.getName();

this.quantity = quantity;

this.price = price;

this.order= order;

this.createdDate = new Date();

}

}

**Products**:

package com.example.demo.model;

import java.util.List;

import javax.persistence.Entity;

import javax.persistence.FetchType;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.OneToMany;

import javax.persistence.Table;

import com.example.demo.dto.ProductDto;

import lombok.Data;

@Entity

@Data

@Table(name = "products")

public class Products {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Integer id;

private String name;

private int price;

@OneToMany(fetch = FetchType.LAZY, mappedBy = "product")

private List<Cart> carts;

public Products() {

}

public Products(int id,String name) {

this.id=id;

this.name = name;

}

public Products(ProductDto productDto) {

this.name = productDto.getName();

this.price = productDto.getPrice();

}

}

**User**:

package com.example.demo.model;

import java.util.List;

import javax.persistence.Entity;

import javax.persistence.FetchType;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.OneToMany;

import javax.persistence.Table;

import lombok.Data;

@Entity

@Data

@Table(name = "users")

public class User {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private int id;

private String firstName;

private String lastName;

private String email;

private String userName;

private String password;

@OneToMany(mappedBy = "user",

fetch = FetchType.LAZY)

private List<Order> orders;

}

**CartRepository**:

package com.example.demo.repository;

import java.util.List;

import javax.transaction.Transactional;

import com.example.demo.model.Cart;

import com.example.demo.model.User;

import org.springframework.data.jpa.repository.JpaRepository;

public interface CartRepository extends JpaRepository<Cart, Integer> {

List<Cart> findAllByUserOrderByCreatedDateDesc(User user);

@Transactional

List<Cart> deleteByUser(User user);

}

**OrderItemsRepository**:

package com.example.demo.repository;

import java.util.List;

import com.example.demo.model.Order;

import com.example.demo.model.OrderItem;

import org.springframework.data.jpa.repository.JpaRepository;

import org.springframework.data.jpa.repository.Query;

public interface OrderItemsRepository extends JpaRepository<OrderItem, Integer>{

@Query("select orderItem from OrderItem orderItem where orderItem.order=?1")

List<OrderItem> findAllByOrderId(Order order);

}

**OrderRepository**:

package com.example.demo.repository;

import java.util.List;

import com.example.demo.model.Order;

import com.example.demo.model.User;

import org.springframework.data.jpa.repository.JpaRepository;

public interface OrderRepository extends JpaRepository<Order, Integer>{

List<Order> findAllByUserIdOrderByCreatedDateDesc(int id);

}

**ProductRepository**:

package com.example.demo.repository;

import java.util.List;

import com.example.demo.model.Products;

import org.springframework.data.jpa.repository.JpaRepository;

import org.springframework.data.jpa.repository.Query;

public interface ProductRepository extends JpaRepository<Products, Integer> {

@Query("select product from Products product where product.id=?1")

List<Products> findByProductID(int productType);

}

**UserRepository**:

package com.example.demo.repository;

import java.util.List;

import com.example.demo.model.User;

import org.springframework.data.jpa.repository.JpaRepository;

import org.springframework.data.jpa.repository.Query;

public interface UserRepository extends JpaRepository<User, Integer>{

@Query("select pojo from User pojo where pojo.userName=?1")

public User findByUser(String userName);

@Query("select pojo from User pojo where pojo.password=?1")

public User findByPassword(String userPassword);

@Query("select pojo.id from User pojo where pojo.userName=?1")

public int findByUserName(String userName);

@Query("select user from User user where id<>4")

List<User> findAllUsers();

}

**CartService**:

package com.example.demo.service;

import com.example.demo.repository.CartRepository;

import com.example.demo.dto.\*;

import java.util.List;

import java.util.ArrayList;

import com.example.demo.model.Cart;

import com.example.demo.model.Products;

import com.example.demo.model.User;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

@Service

public class CartService {

@Autowired

private CartRepository cartRepository;

public CartService() {

}

public CartService(CartRepository cartRepository) {

this.cartRepository = cartRepository;

}

public void addToCart(AddToCartDto addToCartDto, Products product, User user) {

Cart cart = new Cart(product, product.getName(), addToCartDto.getQuantity(), addToCartDto.getTotalPrice(),

user);

cartRepository.save(cart);

}

public CartDto listCartItems(User user) {

List<Cart> cartList = cartRepository.findAllByUserOrderByCreatedDateDesc(user);

List<CartItemDto> cartItems = new ArrayList<>();

for (Cart cart : cartList) {

CartItemDto cartItemDto = getDtoFromCart(cart);

cartItems.add(cartItemDto);

}

int totalCost = 0;

for (CartItemDto cartItemDto : cartItems) {

totalCost += (cartItemDto.getProduct().getPrice() \* cartItemDto.getQuantity());

}

return new CartDto(cartItems, totalCost);

}

public static CartItemDto getDtoFromCart(Cart cart) {

return new CartItemDto(cart);

}

public void deleteCartItems(int userId) {

cartRepository.deleteAll();

}

public void deleteUserCartItems(User user) {

cartRepository.deleteByUser(user);

}

}

**OrderItemsService**:

package com.example.demo.service;

import java.util.List;

import com.example.demo.model.Order;

import com.example.demo.model.OrderItem;

import com.example.demo.repository.OrderItemsRepository;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

@Service

public class OrderItemsService {

@Autowired

private OrderItemsRepository orderItemsRepository;

public void addOrderedProducts(OrderItem orderItem) {

orderItemsRepository.save(orderItem);

}

public List<OrderItem> listOrders(Order order) {

List<OrderItem> orderList = orderItemsRepository.findAllByOrderId(order);

return orderList;

}

}

**OrderService**:

package com.example.demo.service;

import java.util.Date;

import java.util.List;

import java.util.Optional;

import com.example.demo.dto.CartDto;

import com.example.demo.dto.CartItemDto;

import com.example.demo.model.Order;

import com.example.demo.model.OrderItem;

import com.example.demo.model.User;

import com.example.demo.repository.OrderItemsRepository;

import com.example.demo.repository.OrderRepository;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

@Service

public class OrderService {

@Autowired

private OrderRepository orderRepository;

@Autowired

private CartService cartService;

@Autowired

OrderItemsRepository orderItemsRepository;

public void placeOrder(User user) {

// first let get cart items for the user

CartDto cartDto = cartService.listCartItems(user);

List<CartItemDto> cartItemDtoList = cartDto.getCartItems();

// create the order and save it

Order newOrder = new Order();

newOrder.setCreatedDate(new Date());

newOrder.setUser(user);

newOrder.setTotalPrice(cartDto.getTotalCost());

orderRepository.save(newOrder);

for (CartItemDto cartItemDto : cartItemDtoList) {

// create orderItem and save each one

OrderItem orderItem = new OrderItem();

orderItem.setCreatedDate(new Date());

orderItem.setPrice(cartItemDto.getProduct().getPrice());

orderItem.setProductName(cartItemDto.getProductName());

orderItem.setProduct(cartItemDto.getProduct());

orderItem.setQuantity(cartItemDto.getQuantity());

orderItem.setOrder(newOrder);

// add to order item list

orderItemsRepository.save(orderItem);

}

cartService.deleteUserCartItems(user);

}

public List<Order> listOrders(int user) {

List<Order> orderList = orderRepository.findAllByUserIdOrderByCreatedDateDesc(user);

return orderList;

}

public List<Order> listAllOrders() {

List<Order> orderList = orderRepository.findAll();

return orderList;

}

public Order getOrder(Integer orderId) {

Optional<Order> order = orderRepository.findById(orderId);

return order.get();

}

}

**ProductService**:

package com.example.demo.service;

import java.util.\*;

import com.example.demo.dto.ProductDto;

import com.example.demo.model.Products;

import com.example.demo.repository.ProductRepository;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

@Service

public class ProductService {

@Autowired

private ProductRepository productRepository;

public List<ProductDto> listProducts() {

List<Products> products = productRepository.findAll();

List<ProductDto> productDtos = new ArrayList<>();

for(Products product : products) {

ProductDto productDto = getDtoFromProduct(product);

productDtos.add(productDto);

}

return productDtos;

}

public static ProductDto getDtoFromProduct(Products product) {

ProductDto productDto = new ProductDto(product);

return productDto;

}

public static Products getProductFromDto(ProductDto productDto) {

Products product = new Products(productDto);

return product;

}

public void addProduct(ProductDto productDto) {

Products product = getProductFromDto(productDto);

productRepository.save(product);

}

public void updateProduct(Integer productID, ProductDto productDto) {

Products product = getProductFromDto(productDto);

product.setId(productID);

productRepository.save(product);

}

public Products getProductById(Integer productId) {

Optional<Products> optionalProduct = productRepository.findById(productId);

return optionalProduct.get();

}

}

**SportyShoeDAO**:

package com.example.demo.service;

import java.util.List;

import java.util.logging.Logger;

import com.example.demo.model.Cart;

import com.example.demo.model.Products;

import com.example.demo.model.User;

import com.example.demo.repository.CartRepository;

import com.example.demo.repository.ProductRepository;

import com.example.demo.repository.UserRepository;

import org.springframework.beans.factory.annotation.Autowired;

import org.springframework.stereotype.Service;

@Service

public class SportyShoeDAO {

Logger log = Logger.getAnonymousLogger();

@Autowired

ProductRepository productRepo;

@Autowired

CartRepository cartRepo;

@Autowired

UserRepository userRepo;

public User findbyuser(String userName) {

return userRepo.findByUser(userName);

}

public User findbypassword(String userPassword) {

return userRepo.findByPassword(userPassword);

}

public int getUserID(String userName) {

return userRepo.findByUserName(userName);

}

public User registerUser(User pojo) {

return userRepo.save(pojo);

}

public List<User> listofUsers(){

return userRepo.findAllUsers();

}

public List<Products> getAll(){

log.info("Inside getAll()");

return productRepo.findAll();

}

public List<Products> getByProductID(int productType){

return productRepo.findByProductID(productType);

}

public Products addproduct(Products addProduct) {

return productRepo.save(addProduct);

}

public Cart addtoCart(Cart addProduct) {

return cartRepo.save(addProduct);

}

}

**addProduct**.jsp:

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<div align=*"center"*>

<form action=*"adminAddProduct"*>

<h1>Add Product</h1>

<br>

<table>

<tr>

<th>Product Name</th>

<td><input type=*"text"* name=*"productName"* size=*"45"* /></td>

</tr>

<tr>

<th>Product Price</th>

<td><input type=*"number"* name=*"productPrice"* size=*"5"* /></td>

</tr>

</table>

<br> <input type=*"submit"* value=*"Submit"* /> <br>

<a href=*"dashboard.jsp"*>Click here</a> to go back to the main menu<br>

</form>

</div>

</body>

</html>

**adminPurchaseReport.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ page import=*"java.util.\*"*%>

<%@page import=*"com.example.demo.model.\*"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Purchase report</title>

</head>

<body>

<div align=*"center"*>

<h2>Purchase Report</h2>

<br>

<%

List<Order> list = (List<Order>) request.getAttribute("allOrderList");

%>

<table border=*"1"*>

<tr>

<th>Order ID</th>

<th>Order Date</th>

<th>Item Total</th>

<th>Status</th>

</tr>

<%

**for** (Order order : list) {

%>

<tr>

<td><%=order.getId()%></td>

<td><%=order.getCreatedDate()%></td>

<td><%=order.getTotalPrice()%></td>

<td>Completed</td>

</tr>

<%

}

%>

</table>

<br> <a href=*"dashboard.jsp"*>Click here</a> to go back to the

main menu<br>

</div>

</body>

</html>

**Cart.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ page import=*"java.util.\*"*%>

<%@page import=*"com.example.demo.model.\*"*%>

<%@page import=*"com.example.demo.dto.\*"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<form action=*"purchaseOrder"*>

<div align=*"center"*>

<h2>Cart</h2>

<br>

<%

List<Cart> list = (List<Cart>) request.getAttribute("list");

%>

<table border=*"1"*>

<tr>

<th>Product Name</th>

<th>Product Quantity</th>

<th>Item Total</th>

</tr>

<%

**for** (Cart c : list) {

%>

<tr>

<td><%=c.getProductName()%></td>

<td><%=c.getQuantity()%></td>

<td><%=c.getTotal()%></td>

</tr>

<%

}

%>

</table>

<table border=*"1"*>

<tr>

<th>Total</th>

</tr>

<tr>

<td><%=request.getAttribute("totalCost")%></td>

</tr>

</table>

<br> <input type=*"submit"* value=*"Checkout"*> <br>

<a href=*"userdashboard"*>Click here</a> to go back to the main menu<br>

</div>

</form>

</body>

</html>

**dashboard.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Admin Dashboard</title>

</head>

<body>

<div align=*"center"*>

<h1>Sporty Shoes</h1>

<br> <br>

<h4>

Welcome

<%=session.getAttribute("userName")%></h4>

<br> <a href=*"logout.jsp"*>Logout</a> <br>

<p>Select an option from below -</p>

<table>

<tr>

<td><a href=*"manageProducts.jsp"*>Manage Products</a></td>

</tr>

<tr>

<td><a href=*"/listusers"*>View the list of Users</a></td>

</tr>

<tr>

<td><a href=*"/listAllPurchaseOrder"*>View purchase reports</a></td>

</tr>

</table>

</div>

</body>

</html>

**listofusers.jsp:**

<%@page import=*"com.example.demo.model.User"*%>

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ page import=*"java.util.\*"*%>

<%@ page import=*"com.example.demo.\*"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Signed-up user list</title>

</head>

<body>

<div align=*"center"*>

<%

List<User> list = (List<User>) request.getAttribute("list");

%>

<h1>List of Signed-up User</h1>

<br>

<table border=*"1"*>

<tr>

<th>User ID</th>

<th>First Name</th>

<th>Last Name</th>

<th>Username</th>

<th>Email ID</th>

</tr>

<%

**for** (User s : list) {

%>

<tr>

<td><%=s.getId()%></td>

<td><%=s.getFirstName()%></td>

<td><%=s.getLastName()%></td>

<td><%=s.getUserName()%></td>

<td><%=s.getEmail()%></td>

</tr>

<%

}

%>

</table> <br>

<a href=*"dashboard.jsp"*>Click here</a> to back to the main Menu

</div>

</body>

</html>

**listProducts.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ page import=*"java.util.\*"*%>

<%@page import=*"com.example.demo.model.Products"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<div align=*"center"*>

<h1>List of Products</h1>

<br>

<%

List<Products> list = (List<Products>) request.getAttribute("list");

%>

<table border=*"1"*>

<tr>

<th>Product Name</th>

<th>Product Price</th>

</tr>

<%

**for** (Products s : list) {

%>

<tr>

<td><%=s.getName()%></td>

<td><%=s.getPrice()%></td>

</tr>

<%

}

%>

</table> <br>

<a href=*"dashboard.jsp"*>Click here</a> to back to the main Menu

</div>

</body>

</html>

**login.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Login page for Student</title>

</head>

<body>

<div align=*"center"*>

<h1>Login</h1>

<form action=*"login"* method=*"post"*>

<table>

<tr>

<td>Username</td>

<td><input type=*"text"* name=*"username"*

placeholder=*"Enter username"* required/></td>

</tr>

<tr>

<td>Password</td>

<td><input type=*"password"* name=*"password"*

placeholder=*"Enter password"* required/></td>

</tr>

</table>

<input type=*"submit"* value=*"Login"* /> <br> <br>

</form>

New User? <a href=*"register.jsp"*>Register here</a>

</div>

</body>

</html>

**loginError.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<div align=*"center"*>

<h2>Invalid Credentials</h2> <br>

<a href=*"login.jsp"*>click here</a> to Login

</div>

</body>

</html>

**logout.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<div align=*"center"*>

<%session.invalidate(); %>

<h2>Logout Successfull</h2>

<br> <br> <a href=*"login.jsp"*>click here</a> to Login

</div>

</body>

</html>

**manageProducts.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ page import=*"java.util.\*"*%>

<%@ page import=*"com.example.demo.\*"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Product Management</title>

</head>

<body>

<div align=*"center"*>

<h1>Manage Products</h1>

<br>

<h2>

<form onclick="this.form.action" action=*"addProduct.jsp"*>

<input type=*"submit"* value=*"Add a product"*>

</form>

&nbsp;&nbsp;&nbsp;

<form onclick="this.form.action" action=*"/getproductlist"* method=*"get"*>

<input type=*"submit"* value=*"List products"*>

</form>

</h2>

</div>

</body>

</html>

**orderDetails.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ page import=*"java.util.\*"*%>

<%@page import=*"com.example.demo.model.\*"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<div align=*"center"*>

<h2>Order Details</h2>

<br>

<%

List<OrderItem> orderList = (List<OrderItem>) request.getAttribute("orderItemList");

%>

<table border=*"1"*>

<tr>

<th>Order Date</th>

<th>Product Name</th>

<th>Product Quantity</th>

<th>Product Price</th>

</tr>

<%

**for** (OrderItem orderItem : orderList) {

%>

<tr>

<td><%=orderItem.getCreatedDate()%></td>

<td><%=orderItem.getProductName()%></td>

<td><%=orderItem.getQuantity()%></td>

<td><%=orderItem.getPrice()%></td>

</tr>

<%

}

%>

</table>

<br> <a href=*"userdashboard"*>Click here</a> to go back to the

main menu<br>

</div>

</body>

</html>

**orderSuccess.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Order Status</title>

</head>

<body>

<form action=*"productTypeSelection"*>

<div align=*"center"*>

<h4>Order placed successfully</h4>

<br>

<a href=*"listPurchaseOrder"*>Click here</a> to view your order<br>

<a href=*"userdashboard"*>Click here</a> to go back to the main menu<br>

</div>

</form>

</body>

</html>

**register.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>User Registration</title>

</head>

<body>

<div align=*"center"*>

<h1>User Registration</h1>

<form action=*"register"*>

<table>

<tr>

<td>First Name</td>

<td><input type=*"text"* name=*"fname"*

placeholder=*"Enter First Name"* /></td>

</tr>

<tr>

<td>Last Name</td>

<td><input type=*"text"* name=*"lname"*

placeholder=*"Enter Last Name"* /></td>

</tr>

<tr>

<td>Email ID</td>

<td><input type=*"email"* name=*"useremail"*

placeholder=*"Enter your Email ID"* /></td>

</tr>

<tr>

<td>Username</td>

<td><input type=*"text"* name=*"username"*

placeholder=*"Enter a username"* /></td>

</tr>

<tr>

<td>Password</td>

<td><input type=*"password"* name=*"password"*

placeholder=*"Enter a password"* /></td>

</tr>

</table>

<br> <input type=*"submit"* value=*"Register"* />

</form>

</div>

</body>

</html>

**registerSuccess.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<div align=*"center"*>

<h2>Registeration Successfull</h2>

<br> <br> <a href=*"login.jsp"*>click here</a> to Login

</div>

</body>

</html>

**selectedProduct.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ page import=*"java.util.\*"*%>

<%@page import=*"com.example.demo.model.Products"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<form action=*"addtocart"*>

<div align=*"center"*>

<h4>Product Selection</h4>

<br>

<%

List<Products> list = (List<Products>) request.getAttribute("list");

%>

<table border=*"1"*>

<tr>

<th>Product ID</th>

<th>Product Name</th>

<th>Product Price</th>

<th>Product Quantity</th>

</tr>

<%

**for** (Products p : list) {

%>

<tr>

<td><input type=*"text"* name=*"productID"* size=*"20"*

value=*"*<%=p.getId()%>*"* /></td>

<td><input type=*"text"* name=*"productName"*

value=*"*<%=p.getName()%>*"* /></td>

<td><input type=*"text"* name=*"productPrice"*

value=*"*<%=p.getPrice()%>*"* /></td>

<td><select name=*"quantity"* id=*"quantity"*>

<option disabled value=*""* selected>Select a quantity</option>

<option value=*"1"*>1</option>

<option value=*"2"*>2</option>

<option value=*"3"*>3</option>

<option value=*"4"*>4</option>

</select></td>

</tr>

<%

}

%>

</table>

<br> <input type=*"submit"* value=*"Add to Cart"*> <br>

<a href=*"userdashboard"*>Click here</a> to go back to the main menu

</div>

</form>

</body>

</html>

**showproductdetails.jsp:**

<%@page import=*"com.example.demo.model.Products"*%>

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ page import=*"java.util.\*"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<form action=*"addtoCart"*>

<div align=*"center"*>

<%

List<Products> list = (List<Products>) request.getAttribute("list");

%>

<h1>Product Selection</h1>

<br>

<table border=*"1"*>

<tr>

<th>Product Name</th>

<th>Product Price</th>

<th>Product Category</th>

<th>Product Type</th>

<th>Add Quantity</th>

<th>Action</th>

</tr>

<%

**for** (Products s : list) {

%>

<tr>

<td><input type=*"text"* name=*"pName"* value=*"*<%=s.getName()%>*"*></td>

<td><input type=*"text"* name=*"pPrice"* value=*"*<%=s.getPrice()%>*"*></td>

<td><select name=*"pQuantity"* id=*"pQuantity"*>

<option disabled value=*""* selected>Select a Quantity</option>

<option value=*"1"*>1</option>

<option value=*"2"*>2</option>

<option value=*"3"*>3</option>

<option value=*"4"*>4</option>

</select></td>

<td><input type=*"submit"* value=*"Add to Cart"*></td>

</tr>

<%

}

%>

</table>

</div>

</form>

</body>

</html>

**userdashboard.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ page import=*"java.util.\*"*%>

<%@page import=*"com.example.demo.model.Products"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>User Dashboard</title>

</head>

<body>

<form action=*"productTypeSelection"*>

<div align=*"center"*>

<h1>Sporty Shoes</h1> <br>

<h3>

Welcome

<%=session.getAttribute("userName")%></h3>

<br> <a href=*"logout.jsp"*>Logout</a> <br>

<a href=*"listcart"*>Cart</a> <br> <a

href=*"listPurchaseOrder"*>View Order</a> <br>

<h4>Product Selection</h4>

<br>

<%

List<Products> list = (List<Products>) request.getAttribute("list");

%>

<table border=*"1"*>

<tr>

<th>Product Name</th>

<th>Product Price</th>

</tr>

<%

**for** (Products s : list) {

%>

<tr>

<td><%=s.getName()%></td>

<td><%=s.getPrice()%></td>

<td><a href=*"productTypeSelection?id=*<%=s.getId()%>*"*>Show

Details</a></td>

</tr>

<%

}

%>

</table>

<br> <br>

</div>

</form>

</body>

</html>

**userOrderList.jsp:**

<%@ page language=*"java"* contentType=*"text/html; charset=ISO-8859-1"*

pageEncoding=*"ISO-8859-1"*%>

<%@ page import=*"java.util.\*"*%>

<%@page import=*"com.example.demo.model.\*"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"ISO-8859-1"*>

<title>Insert title here</title>

</head>

<body>

<div align=*"center"*>

<h2>Order Details</h2>

<br>

<%

List<Order> orderList = (List<Order>) request.getAttribute("orderDtoList");

%>

<table border=*"1"*>

<tr>

<th>Order ID</th>

<th>Order Date</th>

<th>Item Total</th>

<th>Action</th>

</tr>

<%

**for** (Order order : orderList) {

%>

<tr>

<td><%=order.getId()%></td>

<td><%=order.getCreatedDate()%></td>

<td><%=order.getTotalPrice()%></td>

<td><a href=*"getOrderItems?id=*<%=order.getId()%>*"*>View Details</a></td>

</tr>

<%

}

%>

</table>

<br> <a href=*"userdashboard"*>Click here</a> to go back to the

main menu<br>

</div>

</body>

</html>