**Source Code**

1. coms.SportyShoes :

**SportyShoes.java**

package coms.SportyShoes;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication

public class SportyShoes {

public static void main(String[] args) {

SpringApplication.run(SportyShoes.class, args);

System.out.println("Server Started....");

}

}

1. coms.SportyShoes.model:-

**CartItem.java**

package coms.SportyShoes.model;

import java.math.BigDecimal;

/\*pojo class\*/

public class CartItem {

private Long productId;

private String name;

private BigDecimal rate; //per item

private int quantity;

private BigDecimal price; //total

private String image\_Url;

public Long getProductId() {return this.productId; }

public String getName() { return this.name;}

public BigDecimal getPrice() { return this.price;}

public BigDecimal getRate() { return this.rate;}

public int getQuantity() { return this.quantity;}

public void setProductId(Long id) { this.productId= id;}

public void setName(String value) { this.name = value;}

public void setPrice(BigDecimal value) { this.price = value;}

public void setRate(BigDecimal value) { this.rate = value;}

public void setQuantity(int value) { this.quantity = value;}

public String getImage\_Url() {

return image\_Url;

}

public void setImage\_Url(String image\_Url) {

this.image\_Url = image\_Url;

}

}

**Product java**

package coms.SportyShoes.model;

import java.math.BigDecimal;

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 lombok.Getter;

import lombok.NoArgsConstructor;

import lombok.Setter;

import lombok.ToString;

@Entity

@NoArgsConstructor

@Setter

@Getter

public class Product {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long productCode;

private String productName;

@ManyToOne(fetch = FetchType.LAZY)

@JoinColumn(name = "productCategory\_fk")

private ProductCategory productCategory;

private int size;

private BigDecimal price;

private String image\_Url;

public Long getProductCode() {

return productCode;

}

public void setProductCode(Long productCode) {

this.productCode = productCode;

}

public String getProductName() {

return productName;

}

public void setProductName(String productName) {

this.productName = productName;

}

public ProductCategory getProductCategory() {

return productCategory;

}

public void setProductCategory(ProductCategory productCategory) {

this.productCategory = productCategory;

}

public int getSize() {

return size;

}

public void setSize(int size) {

this.size = size;

}

public BigDecimal getPrice() {

return price;

}

public void setPrice(BigDecimal price) {

this.price = price;

}

public String getImage\_Url() {

return image\_Url;

}

public void setImage\_Url(String image\_Url) {

this.image\_Url = image\_Url;

}

@Override

public String toString() {

return "Product [productCode=" + productCode + ", productName=" + productName + ", productCategory="

+ productCategory + ", size=" + size + ", price=" + price + ", image\_Url=" + image\_Url + "]";

}

public Product(Long productCode, String productName, ProductCategory productCategory, int size, BigDecimal price,

String image\_Url) {

super();

this.productCode = productCode;

this.productName = productName;

this.productCategory = productCategory;

this.size = size;

this.price = price;

this.image\_Url = image\_Url;

}

}

**ProductCategory**

package coms.SportyShoes.model;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

@Entity

public class ProductCategory {

@Id

@GeneratedValue(strategy = GenerationType.AUTO)

private Long id;

private String brandName;

private String style; // wear hiking sneaker biking basketball

private String targetGroup;//gender M F kids- boys girl

public Long getId() {

return id;

}

public void setId(Long id) {

this.id = id;

}

public String getBrandName() {

return brandName;

}

public void setBrandName(String brandName) {

this.brandName = brandName;

}

public String getStyle() {

return style;

}

public void setStyle(String style) {

this.style = style;

}

public String getTargetGroup() {

return targetGroup;

}

public void setTargetGroup(String targetGroup) {

this.targetGroup = targetGroup;

}

@Override

public String toString() {

return "ProductCategory [id=" + id + ", brandName=" + brandName + ", style=" + style + ", targetGroup="

+ targetGroup + "]";

}

public ProductCategory(Long id, String brandName, String style, String targetGroup) {

super();

this.id = id;

this.brandName = brandName;

this.style = style;

this.targetGroup = targetGroup;

}

public ProductCategory() {

}

}

**PurchaseItem.java**

package coms.SportyShoes.model;

import java.math.BigDecimal;

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.Table;

@Entity

@Table

public class PurchaseItem {

@Id

@GeneratedValue(strategy = GenerationType.AUTO)

private Long ID;

@ManyToOne

@JoinColumn(name = "purchaseOrder\_fk")

private PurchaseOrder purchaseOrder;

@ManyToOne

@JoinColumn(name = "product\_fk")

private Product poduct;

@ManyToOne

@JoinColumn(name = "user\_fk")

private User user;

private BigDecimal rate;

private int quantity;

private BigDecimal price;

public PurchaseItem(Long iD, Product poduct, User user, BigDecimal rate, int quantity, BigDecimal price) {

super();

ID = iD;

this.poduct = poduct;

this.user = user;

this.rate = rate;

this.quantity = quantity;

this.price = price;

}

public PurchaseItem() {

}

public Long getID() {

return ID;

}

public void setID(Long iD) {

ID = iD;

}

public Product getPoduct() {

return poduct;

}

public void setPoduct(Product poduct) {

this.poduct = poduct;

}

public User getUser() {

return user;

}

public void setUser(User user) {

this.user = user;

}

public BigDecimal getRate() {

return rate;

}

public void setRate(BigDecimal rate) {

this.rate = rate;

}

public int getQuantity() {

return quantity;

}

public void setQuantity(int quantity) {

this.quantity = quantity;

}

public BigDecimal getPrice() {

return price;

}

public void setPrice(BigDecimal price) {

this.price = price;

}

public PurchaseOrder getPurchaseOrder() {

return purchaseOrder;

}

public void setPurchaseOrder(PurchaseOrder purchaseOrder) {

this.purchaseOrder = purchaseOrder;

}

@Override

public String toString() {

return "PurchaseItem [ID=" + ID + ", poduct=" + poduct + ", user=" + user + ", rate=" + rate + ", quantity="

+ quantity + ", price=" + price + "]";

}

}

**PurchaseOrder.java**

package coms.SportyShoes.model;

import java.math.BigDecimal;

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.Temporal;

import javax.persistence.TemporalType;

@Entity

public class PurchaseOrder {

@Id

@GeneratedValue(strategy = GenerationType.SEQUENCE)

@Column(name = "ID")

private Long ID;

@ManyToOne

@JoinColumn(name = "user\_fk")

private User user;

@Temporal(TemporalType.DATE)

private Date date;

@Column(name = "gross\_total")

private BigDecimal total;

public PurchaseOrder(Long iD, User user, Date date, BigDecimal total) {

super();

ID = iD;

this.user = user;

this.date = date;

this.total = total;

}

public Long getID() {

return ID;

}

public void setID(Long iD) {

ID = iD;

}

public User getUser() {

return user;

}

public void setUser(User user) {

this.user = user;

}

public Date getDate() {

return date;

}

public void setDate(Date date) {

this.date = date;

}

public BigDecimal getTotal() {

return total;

}

public void setTotal(BigDecimal total) {

this.total = total;

}

@Override

public String toString() {

return "PurchaseOrder [ID=" + ID + ", user=" + user + ", date=" + date + ", total=" + total + "]";

}

public PurchaseOrder() {

}

}

**User.java**

package coms.SportyShoes.model;

import java.util.Date;

import java.util.List;

import javax.persistence.Column;

import javax.persistence.DiscriminatorColumn;

import javax.persistence.DiscriminatorType;

import javax.persistence.DiscriminatorValue;

import javax.persistence.Entity;

import javax.persistence.Id;

import javax.persistence.Inheritance;

import javax.persistence.InheritanceType;

import javax.persistence.Table;

import javax.persistence.Temporal;

import javax.persistence.TemporalType;

import javax.persistence.Transient;

import lombok.Getter;

import lombok.NoArgsConstructor;

import lombok.Setter;

import lombok.ToString;

@Entity

@Table(name = "USER")

@Inheritance(strategy = InheritanceType.SINGLE\_TABLE)

@DiscriminatorColumn(discriminatorType = DiscriminatorType.STRING, name = "User\_Role")

@DiscriminatorValue("Customer")

/\*

\* @Setter

\*

\* @Getter

\*

\* @NoArgsConstructor

\*/

public class User {

public User() {

}

@Id

private String emailId;

@Column(length = 20)

private String password;

@Column

private String firstName;

@Column

private String lastName;

@Column

private String country;

@Column

private String gender;

@Temporal(TemporalType.DATE)

private Date dateOfBirth;

@Override

public String toString() {

return "User [emailId=" + emailId + ", password=" + password + ", firstName=" + firstName + ", lastName="

+ lastName + ", country=" + country + ", gender=" + gender + ", dateOfBirth=" + dateOfBirth + "]";

}

public User(String emailId, String password, String firstName, String lastName, String country, String gender,

Date dateOfBirth) {

super();

this.emailId = emailId;

this.password = password;

this.firstName = firstName;

this.lastName = lastName;

this.country = country;

this.gender = gender;

this.dateOfBirth = dateOfBirth;

}

public String getEmailId() {

return emailId;

}

public void setEmailId(String emailId) {

this.emailId = emailId;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

public String getFirstName() {

return firstName;

}

public void setFirstName(String firstName) {

this.firstName = firstName;

}

public String getLastName() {

return lastName;

}

public void setLastName(String lastName) {

this.lastName = lastName;

}

public String getCountry() {

return country;

}

public void setCountry(String country) {

this.country = country;

}

public String getGender() {

return gender;

}

public void setGender(String gender) {

this.gender = gender;

}

public Date getDateOfBirth() {

return dateOfBirth;

}

public void setDateOfBirth(Date dateOfBirth) {

this.dateOfBirth = dateOfBirth;

}

@Transient

public String getDiscriminatorValue() {

return this.getClass().getAnnotation(DiscriminatorValue.class).value();

}

/\*

\* @OneToMany private List<Purchase> listOfPurchases; // cartID

\*/}

1. **coms.SportyShoes.repository**

**ProductCategoryRepository.java**

package coms.SportyShoes.repository;

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

import org.springframework.stereotype.Repository;

import coms.SportyShoes.model.ProductCategory;

@Repository

public interface ProductCategoryRepository extends JpaRepository<ProductCategory,Long >{

}

**ProductRepository.java**

package coms.SportyShoes.repository;

import java.util.List;

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

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

import org.springframework.data.repository.query.Param;

import org.springframework.stereotype.Repository;

import coms.SportyShoes.model.Product;

import coms.SportyShoes.model.ProductCategory;

@Repository

public interface ProductRepository extends JpaRepository<Product, Long> {

List<Product>findByProductCategory(ProductCategory productCategory);

@Query(value = "SELECT \* FROM product JOIN product\_category ON product.product\_category\_fk=product\_category.id AND product\_category.brand\_name =:brand\_name",nativeQuery = true)

List<Product> findAllProductsByBrand(@Param("brand\_name") String brand\_name);

@Query(value = "SELECT \* FROM product JOIN product\_category ON product.product\_category\_fk=product\_category.id AND product\_category.style =:style",nativeQuery = true)

List<Product> findAllProductsByStyle(@Param("style") String style);

@Query(value = "SELECT \* FROM product JOIN product\_category ON product.product\_category\_fk=product\_category.id AND product\_category.target\_group =:target\_group", nativeQuery = true)

List<Product> findAllProductsByTargetGroup(@Param("target\_group") String target\_group);

}

**PurchaseItemRepository.java**

package coms.SportyShoes.repository;

import java.util.List;

import org.springframework.data.domain.Sort;

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

import org.springframework.stereotype.Repository;

import coms.SportyShoes.model.PurchaseItem;

import coms.SportyShoes.model.User;

@Repository

public interface PurchaseItemRepository extends JpaRepository<PurchaseItem, Long>{

List<PurchaseItem> findByUser(User user, Sort sort);

}

**PurchaseOrderRepository.java**

package coms.SportyShoes.repository;

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

import coms.SportyShoes.model.PurchaseOrder;

public interface PurchaseOrderRepository extends JpaRepository<PurchaseOrder, Long> {

}

**UserRepository.java**

package coms.SportyShoes.repository;

import java.util.List;

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

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

import org.springframework.stereotype.Repository;

import coms.SportyShoes.model.User;

@Repository

public interface UserRepository extends JpaRepository<User, String>{

User findByEmailIdAndPassword(String emailId, String password);

List<User> findByfirstNameContaining(String firstName);

List<User> findByfirstNameContainingIgnoreCase(String firstName);

@Query("from User")

public List<User> getUsers();

}

1. coms.SportyShoes.service:-

**ProductCategoryService.java**

package coms.SportyShoes.service;

import java.util.List;

import coms.SportyShoes.model.ProductCategory;

public interface ProductCategoryService {

public ProductCategory getCategoryById(Long id);

public void updateCategory(ProductCategory category);

public ProductCategory saveCategory(ProductCategory category);

public void deleteCategory(Long id) ;

public List<ProductCategory> fetchProductCategoryList();

}

**ProductCategoryServiceImpl.java**

package coms.SportyShoes.service;

import java.util.List;

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

import org.springframework.stereotype.Service;

import coms.SportyShoes.model.ProductCategory;

import coms.SportyShoes.repository.ProductCategoryRepository;

@Service(value = "productCategoryService")

public class ProductCategoryServiceImpl implements ProductCategoryService {

@Autowired

ProductCategoryRepository productCategoryRepository;

@Override

public ProductCategory getCategoryById(Long id) {

// TODO Auto-generated method stub

return productCategoryRepository.findById(id).get();

}

@Override

public void updateCategory(ProductCategory category) {

productCategoryRepository.save(category);

}

@Override

public void deleteCategory(Long id) {

productCategoryRepository.deleteById(id);

}

@Override

public ProductCategory saveCategory(ProductCategory category) {

return productCategoryRepository.save(category);

}

@Override

public List<ProductCategory> fetchProductCategoryList() {

return productCategoryRepository.findAll();

}

}

**ProductService.java**

package coms.SportyShoes.service;

import java.util.List;

import coms.SportyShoes.model.Product;

import coms.SportyShoes.model.ProductCategory;

public interface ProductService {

public Product getProductById(Long productCode);

public Product getProductByProductCategory(ProductCategory productCategory);

public void updateProduct(Product product);

public Product saveProduct(Product product);

public void deleteProduct(Long productCode);

public List<Product> getAllProducts();

}

**ProductServiceImpl.java**

package coms.SportyShoes.service;

import java.util.List;

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

import org.springframework.stereotype.Service;

import coms.SportyShoes.model.Product;

import coms.SportyShoes.model.ProductCategory;

import coms.SportyShoes.repository.ProductRepository;

@Service(value = "productService")

public class ProductServiceImpl implements ProductService{

@Autowired

private ProductRepository productRepository;

@Override

public Product getProductById(Long productCode) {

return productRepository.findById(productCode).get();

}

@Override

public void updateProduct(Product product) {

if(productRepository.findById(product.getProductCode()).get() != null) {// add conditions if not found

productRepository.save(product);

}

}

@Override

public void deleteProduct(Long productCode) {

productRepository.deleteById(productCode);

}

@Override

public List<Product> getAllProducts() {

return productRepository.findAll();

}

@Override

public Product getProductByProductCategory(ProductCategory productCategory) {

// TODO Auto-generated method stub

//productRepository.fin

return null;

}

@Override

public Product saveProduct(Product product) {

return productRepository.save(product);

}

}

**PurchaseItemService.java**

package coms.SportyShoes.service;

import java.util.List;

import coms.SportyShoes.model.PurchaseItem;

import coms.SportyShoes.model.PurchaseOrder;

import coms.SportyShoes.model.User;

public interface PurchaseItemService {

public PurchaseItem getPurchaseItemById(Long id);

public List<PurchaseItem> getAllItemsByPurchaseOrder(PurchaseOrder order);

public List<PurchaseItem> getAllPurchaseItemByUserId(User userId);

public PurchaseItem savePurchaseItem(PurchaseItem purchaseItem);

public List<PurchaseItem> getAllPurchaseItemList();

}

**PurchaseItemServiceImpl.java**

package coms.SportyShoes.service;

import java.util.List;

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

import org.springframework.data.domain.Sort;

import org.springframework.stereotype.Service;

import coms.SportyShoes.model.PurchaseItem;

import coms.SportyShoes.model.PurchaseOrder;

import coms.SportyShoes.model.User;

import coms.SportyShoes.repository.PurchaseItemRepository;

@Service(value = "purchaseItemService")

public class PurchaseItemServiceImpl implements PurchaseItemService {

@Autowired

PurchaseItemRepository purchaseItemRepository;

@Override

public PurchaseItem getPurchaseItemById(Long id) {

// TODO Auto-generated method stub

return null;

}

@Override

public List<PurchaseItem> getAllItemsByPurchaseOrder(PurchaseOrder order) {

// TODO Auto-generated method stub

return null;

}

@Override

public List<PurchaseItem> getAllPurchaseItemByUserId(User userId) {

// TODO Auto-generated method stub

return null;

}

@Override

public PurchaseItem savePurchaseItem(PurchaseItem purchaseItem) {

purchaseItemRepository.save(purchaseItem);

return purchaseItem;

}

@Override

public List<PurchaseItem> getAllPurchaseItemList() {

return purchaseItemRepository.findAll(Sort.by(Sort.Direction.ASC, "purchaseOrder\_ID"));

}

}

**PurchaseOrderService.java**

package coms.SportyShoes.service;

import java.util.List;

import coms.SportyShoes.model.PurchaseOrder;

public interface PurchaseOrderService {

public PurchaseOrder getPurchaseById(long id);

public List<PurchaseOrder> getAllPurchaseOrders();

public List<PurchaseOrder> getAllPurchaseOrderByUserId(Long userId);

public PurchaseOrder savePurchaseOrder(PurchaseOrder purchaseOrder);

}

**PurchaseOrderServiceImpl.java**

package coms.SportyShoes.service;

import java.util.List;

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

import org.springframework.stereotype.Service;

import coms.SportyShoes.model.PurchaseOrder;

import coms.SportyShoes.repository.PurchaseOrderRepository;

@Service(value = "purchaseOrderService")

public class PurchaseOrderServiceImpl implements PurchaseOrderService {

@Autowired

PurchaseOrderRepository purchaseOrderRepository;

@Override

public PurchaseOrder getPurchaseById(long id) {

// TODO Auto-generated method stub

return null;

}

@Override

public List<PurchaseOrder> getAllPurchaseOrders() {

// TODO Auto-generated method stub

return null;

}

@Override

public List<PurchaseOrder> getAllPurchaseOrderByUserId(Long userId) {

// TODO Auto-generated method stub

return null;

}

@Override

public PurchaseOrder savePurchaseOrder(PurchaseOrder purchaseOrder) {

purchaseOrderRepository.save(purchaseOrder);

return purchaseOrder;

}

}

**UserService.java**

package coms.SportyShoes.service;

import java.util.List;

import coms.SportyShoes.model.User;

public interface UserService {

public List<User> fetchUserList();

public User saveUser(User user);

public void deleteByEmailId(String emailId);

public User getUser(String emailId);

public void updateUser(User user);

public User getUser(String emailId, String password);

}

**UserServiceImpl.java**

package coms.SportyShoes.service;

import java.util.List;

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

import org.springframework.stereotype.Service;

import coms.SportyShoes.model.User;

import coms.SportyShoes.repository.UserRepository;

@Service(value = "userService")//service interface first letter is small

public class UserServiceImpl implements UserService{

@Autowired

private UserRepository userRepository;

@Override

public List<User> fetchUserList() {

System.out.println(userRepository.findAll().toString());

return (List<User>) userRepository.findAll();

}

@Override

public User saveUser(User user) {

System.out.println("Save User");

return userRepository.save(user);

}

@Override

public void deleteByEmailId(String emailId) {

userRepository.deleteById(emailId);

}

@Override

public User getUser(String emailId) {

return userRepository.findById(emailId).get();

}

@Override

public User getUser(String emailId, String password) {

return userRepository.findByEmailIdAndPassword(emailId, password);

}

@Override

public void updateUser(User user) {

System.out.println(user.getEmailId() + user.getFirstName());

if(userRepository.findById(user.getEmailId()).get() != null)

userRepository.save(user);

System.out.println(user.getEmailId() + user.getFirstName());

}

}

1. coms.SportyShoes.controller:-

**CartController.java**

package coms.SportyShoes.controller;

import java.math.BigDecimal;

import java.math.RoundingMode;

import java.time.LocalDate;

import java.time.ZoneId;

import java.util.ArrayList;

import java.util.List;

import java.util.Objects;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

import org.springframework.ui.ModelMap;

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

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

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

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

import coms.SportyShoes.model.CartItem;

import coms.SportyShoes.model.Product;

import coms.SportyShoes.model.PurchaseItem;

import coms.SportyShoes.model.PurchaseOrder;

import coms.SportyShoes.model.User;

import coms.SportyShoes.service.ProductService;

import coms.SportyShoes.service.PurchaseItemService;

import coms.SportyShoes.service.PurchaseOrderService;

import coms.SportyShoes.service.UserService;

@Controller

public class CartController {

@Autowired

ProductService productService;

@Autowired

UserService userService;

@Autowired

PurchaseOrderService purchaseOrderService;

@Autowired

PurchaseItemService purchaseItemService;

private int cartCount;

@SuppressWarnings("unchecked")

@GetMapping("/addToCart/{productId}")

public String cartAddItem(@PathVariable Long productId, javax.servlet.http.HttpServletRequest request) {

HttpSession session = request.getSession();

// if cart is already in session then retrieve it else create a new cart list

// and

// save it to session

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

if (session.getAttribute("cart\_items") != null)

cartItems = (List<CartItem>) session.getAttribute("cart\_items");

if (isItemInCart(cartItems, productId)) {

System.out.println("Item already in cart - quantity increased by 1");

} else {

Product product = productService.getProductById(productId);

CartItem item = new CartItem();

item.setProductId(productId);

item.setQuantity(1);

item.setRate(product.getPrice());

BigDecimal dprice = item.getRate().multiply(new BigDecimal(item.getQuantity()));

item.setPrice(dprice);

item.setName(product.getProductName());

item.setImage\_Url(product.getImage\_Url());

cartItems.add(item);

}

BigDecimal totalValue = getCartValue(cartItems);

session.setAttribute("cartCount", cartCount);

session.setAttribute("cartValue", totalValue);

session.setAttribute("cart\_items", cartItems);

return "redirect:/cart";

}

@SuppressWarnings("unchecked")

@GetMapping("/cart/deleteItem")

public String cartDeleteItem(ModelMap map, javax.servlet.http.HttpServletRequest request,

@RequestParam(value = "productId", required = true) Long productId) {

HttpSession session = request.getSession();

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

if (session.getAttribute("cart\_items") != null)

cartItems = (List<CartItem>) session.getAttribute("cart\_items");

for (CartItem item : cartItems) {

if (item.getProductId() == productId) {

if (item.getQuantity() == 1) {

cartItems.remove(item);

BigDecimal totalValue = getCartValue(cartItems);

session.setAttribute("cartValue", totalValue);

System.out.println("###Total " + totalValue);

break;

} else {

item.setQuantity(item.getQuantity() - 1);

BigDecimal totalValue = getCartValue(cartItems);

session.setAttribute("cartValue", totalValue);

System.out.println("###Total " + totalValue);

break;

}

}

}

session.setAttribute("cart\_items", cartItems);

return "redirect:/cart";

}

@SuppressWarnings("unchecked")

@GetMapping("/cart")

public String viewCart(ModelMap map, javax.servlet.http.HttpServletRequest request) {

// check if user is logged in

HttpSession session = request.getSession();

String useremailId = (String) session.getAttribute("userEmailId");

if (Objects.nonNull(useremailId)) {

session.removeAttribute("errMessage");

}

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

if (session.getAttribute("cart\_items") != null)

cartItems = (List<CartItem>) session.getAttribute("cart\_items");

// get total of all cart items

BigDecimal totalValue = getCartValue(cartItems);

System.out.println("###Total " + totalValue);

map.addAttribute("cartValue", totalValue);

map.addAttribute("cartItems", cartItems);

session.setAttribute("cartValue", totalValue);

session.setAttribute("cartItems", cartItems);

return "cart";

}

@GetMapping("/checkout")

public String checkout(Model model, javax.servlet.http.HttpServletRequest request) {

HttpSession session = request.getSession();

String useremailId = (String) session.getAttribute("userEmailId");

if (Objects.isNull(useremailId)) {

session.setAttribute("errMessage", "You need Login before checkout!!");

return "redirect:/cart";

}

BigDecimal cartValue = (BigDecimal) session.getAttribute("cartValue");

System.out.println(cartValue);

if (cartValue != null) {

model.addAttribute("cartTotal", session.getAttribute("cartValue"));

BigDecimal shipping = (BigDecimal) ((BigDecimal) session.getAttribute("cartValue"))

.multiply(new BigDecimal(0.10)).setScale(2, RoundingMode.CEILING);

model.addAttribute("shipping", shipping);

BigDecimal total = shipping.add(cartValue).setScale(2, RoundingMode.CEILING);

model.addAttribute("total", total);

} else {

model.addAttribute("cartTotal", 0);

model.addAttribute("shipping", 0);

model.addAttribute("total", 0);

}

return "checkout";

}

@PostMapping("/payNow")

public String completePurchase(ModelMap map, javax.servlet.http.HttpServletRequest request) {

HttpSession session = request.getSession();

String useremailId = (String) session.getAttribute("userEmailId");

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

if (session.getAttribute("cart\_items") != null)

cartItems = (List<CartItem>) session.getAttribute("cart\_items");

BigDecimal totalValue = getCartValue(cartItems);

User user = userService.getUser(useremailId);

PurchaseOrder purchaseOrder = new PurchaseOrder();

purchaseOrder.setUser(user);

purchaseOrder.setDate(java.sql.Date.valueOf(LocalDate.now(ZoneId.systemDefault())));

purchaseOrder.setTotal(totalValue);

purchaseOrder = purchaseOrderService.savePurchaseOrder(purchaseOrder);

for (CartItem item : cartItems) {

PurchaseItem pItem = new PurchaseItem();

pItem.setPurchaseOrder(purchaseOrder);

pItem.setPoduct(productService.getProductById(item.getProductId()));

pItem.setUser(user);

pItem.setRate(item.getRate());

pItem.setQuantity(item.getQuantity());

pItem.setPrice(item.getPrice());

purchaseItemService.savePurchaseItem(pItem);

}

map.addAttribute("cartValue", 0);

map.addAttribute("cartItems", null);

map.addAttribute("orderNo", purchaseOrder.getID());

session.setAttribute("cartValue", 0);

session.setAttribute("cartItems", null);

cartCount = 0;

session.setAttribute("cartCount", cartCount);

return "confirm";

}

/\*\*

\* Check if an item is already in the cart

\*

\* @param list

\* @param item

\* @return

\*/

private boolean isItemInCart(List<CartItem> list, long item) {

boolean retVal = false;

for (CartItem thisItem : list) {

if (item == thisItem.getProductId()) {

thisItem.setQuantity(thisItem.getQuantity() + 1);

retVal = true;

break;

}

}

return retVal;

}

/\*\*

\* Get total value of items in cart

\*

\* @param list

\* @return

\*/

private BigDecimal getCartValue(List<CartItem> list) {

BigDecimal total = new BigDecimal(0.0);

cartCount = 0;

for (CartItem item : list) {

cartCount = cartCount + item.getQuantity();

BigDecimal dprice = item.getRate().multiply(new BigDecimal(item.getQuantity()));

total = total.add(dprice);

}

return total;

}

}

**HomeController.java**

package coms.SportyShoes.controller;

import java.util.Objects;

import javax.servlet.http.HttpSession;

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

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

@Controller

public class HomeController {

@GetMapping({"/","/homePage"})

public String home(Model model,javax.servlet.http.HttpServletRequest request ) {

HttpSession session = request.getSession();

if(Objects.isNull(session.getAttribute("cartCount")))

session.setAttribute("cartCount", 0);

return "homePage";

}

}

**LoginController.java**

package coms.SportyShoes.controller;

import javax.servlet.ServletException;

import javax.servlet.http.HttpServletRequest;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.ModelMap;

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

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

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

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

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

import org.springframework.web.servlet.ModelAndView;

import coms.SportyShoes.model.User;

import coms.SportyShoes.service.UserService;

import coms.SportyShoes.validatator.EmailValidator;

@Controller

public class LoginController {

@Autowired

private UserService userService;

String errMessage;

@PostMapping("/login")

public String loginAction(ModelMap map, javax.servlet.http.HttpServletRequest request,

@RequestParam(value="emailId", required=true) String emailId,

@RequestParam(value="password", required=true) String password)

{

if(EmailValidator.isValid(emailId)) {

User userDB = userService.getUser(emailId,password);

if(userDB!=null) {

System.out.println("############# " + userDB.getFirstName());

String userRole;

if( userDB.getDiscriminatorValue().equals("Admin")) {

userRole = "Admin";

map.addAttribute(userRole);

HttpSession session = request.getSession();

session.setAttribute("userEmailId", emailId);

session.setAttribute("userName", userDB.getFirstName());

return "redirect:/admin";

}

else

{

userRole ="Customer";

map.addAttribute(userRole);

HttpSession session = request.getSession();

session.setAttribute("userEmailId", emailId);

session.setAttribute("userName", userDB.getFirstName());

return "redirect:/homePage";

}

}else {

errMessage = "Invalid Credentials";

map.addAttribute("errorMessage", errMessage);

return "redirect:/login";

}

}else {

errMessage ="Invalid email id";

map.addAttribute("errorMessage", errMessage);

System.out.println("################### "+ errMessage);

return "redirect:/login";

}

}

@GetMapping("/logout")

public String logout(javax.servlet.http.HttpServletRequest request) {

HttpSession session = request.getSession();

if(session !=null) {

session.invalidate();

return "redirect:/homePage";

}

return "redirect:/homePage";

}

@GetMapping("/login/changePassword/{emailID}")

public String changePassword(@PathVariable String emailID,javax.servlet.http.HttpServletRequest request ) {

request.setAttribute("emailIdVal", emailID);

return "changePassword";

}

@GetMapping("/login/changePassword")

public ModelAndView changePassword() {

ModelAndView modelAndView = new ModelAndView("changePassword");

//modelAndView.addObject("message", "test data");

modelAndView.addObject("emailIdVal", "dev@123"); return modelAndView;

}

@PostMapping("/login/changePassword")

public String saveChangePassword(@ModelAttribute("user") User user , HttpServletRequest request) {

User user1 = userService.getUser(user.getEmailId());

System.out.println("##"+ user.getEmailId() );

user1.setPassword(user.getPassword());

userService.updateUser(user1);

request.setAttribute("changePasswordMsg", "Password changed");

return "homePage";

}

@GetMapping("/register")

public String registerGet() {

return "register";

}

@PostMapping("/register/save")

public String registerPost(@ModelAttribute("user") User user , HttpServletRequest request) throws ServletException {

userService.saveUser(user);

return "homePage";

}

}

**ShopController.java**

package coms.SportyShoes.controller;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.Model;

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

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

import coms.SportyShoes.repository.ProductCategoryRepository;

import coms.SportyShoes.repository.ProductRepository;

import coms.SportyShoes.service.ProductCategoryService;

import coms.SportyShoes.service.ProductService;

@Controller

public class ShopController {

@Autowired

ProductCategoryService productCategoryService;

@Autowired

ProductService productService;

@Autowired

ProductRepository productRepository;

@Autowired

ProductCategoryRepository productCategoryRepository;

@GetMapping("/shop")

public String shop(Model model) {

model.addAttribute("products", productService.getAllProducts());

return "shop";

}

@GetMapping({ "/shop/Gender/{gender}", "/shop/Kids/{gender}" })

public String shopByMen(@PathVariable String gender, Model model) {

model.addAttribute("products", productRepository.findAllProductsByTargetGroup(gender));

return "shop";

}

@GetMapping("/shop/brand/{brandName}")

public String shopByBrand(@PathVariable String brandName, Model model) {

model.addAttribute("products", productRepository.findAllProductsByBrand(brandName));

return "shop";

}

@GetMapping("/shop/style/{styleVal}")

public String shopByStyle(@PathVariable String styleVal, Model model) {

model.addAttribute("products", productRepository.findAllProductsByStyle(styleVal));

return "shop";

}

@GetMapping("/shop/viewProduct/{productCode}")

public String viewProduct(@PathVariable Long productCode, Model model) {

System.out.println("## View Product ->Product Code " + productCode);

model.addAttribute("product", productService.getProductById(productCode));

return "viewProduct";

}

}

**UserController.java**

package coms.SportyShoes.controller;

import java.net.URI;

import java.util.List;

import javax.validation.Valid;

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

import org.springframework.http.ResponseEntity;

import org.springframework.ui.Model;

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

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

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

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

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

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

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

import org.springframework.web.servlet.ModelAndView;

import org.springframework.web.servlet.support.ServletUriComponentsBuilder;

import coms.SportyShoes.model.User;

import coms.SportyShoes.service.UserService;

@RestController

public class UserController {

@Autowired

private UserService userService;

// fetchAll users

// Http method : get

// uri : /users

@GetMapping("/users")

public List<User> getAllUsers() {

return userService.fetchUserList();

}

@RequestMapping("/user-list")

public List<User> userList(Model model) {

List<User> users = userService.fetchUserList();

model.addAttribute("users", users);

return users;

}

@GetMapping(path = "/users/{emailId}")

public User getUser(@PathVariable String emailId) {

User user = userService.getUser(emailId);

if (user == null) {

throw new RuntimeException("id null");// user not foundexception

}

return user;

}

@PostMapping(path = "/users") // Request body --> json data

public ResponseEntity<User> createUser(@RequestBody User user) {

// System.out.println("username"+ username);

System.out.println(user.getFirstName());

User SaveUser = userService.saveUser(user);

// response header : location

// status code :201

URI location = ServletUriComponentsBuilder.fromCurrentRequest().path("/{id}")

.buildAndExpand(SaveUser.getEmailId()).toUri();

return ResponseEntity.created(location).build();

}

// PUT

@PutMapping(value = "createUser")

public User createUser2(@RequestBody User userDataInput) {

return userService.saveUser(userDataInput);

}

// update

@PutMapping(value = "/users/{emailId}")

public void updateUser(@PathVariable String emailId, @RequestBody User userDataInput) {

userService.updateUser(userDataInput);

}

@GetMapping("/goToViewPage")

public ModelAndView passParametersWithModelAndView() {

ModelAndView modelAndView = new ModelAndView("UserList");

modelAndView.addObject("message", "test data");

return modelAndView;

}

@GetMapping("/login")

public ModelAndView login() {

ModelAndView modelAndView = new ModelAndView("login");

modelAndView.addObject("message", "test data");

return modelAndView;

}

}

1. coms.SportyShoes.validatator :-

**EmailValidator.java**

package coms.SportyShoes.validatator;

import java.util.regex.Matcher;

import java.util.regex.Pattern;

public class EmailValidator{

private static Pattern pattern;

private static Matcher matcher;

private static final String EMAIL\_PATTERN =

"^[\_A-Za-z0-9-]+(\\.[\_A-Za-z0-9-]+)\*@[A-Za-z0-9]+(\\.[A-Za-z0-9]+)\*(\\.[A-Za-z]{2,})$";

public EmailValidator(){

pattern = Pattern.compile(EMAIL\_PATTERN);

}

/\*\*

\* Validate with regular expression

\* @param email for validation

\* @return true valid email, false invalid email

\*/

public static boolean isValid(final String email){

pattern = Pattern.compile(EMAIL\_PATTERN);

matcher = pattern.matcher(email);

return matcher.matches();

}

}

**Views(.jsp Pages):-**

**addProduct.jsp**

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

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

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<%@ taglib prefix=*"form"* uri=*"http://www.springframework.org/tags/form"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*

>

<title>Add Product</title>

</head>

<body>

<jsp:include page=*"adminHeader.jsp"* />

<div class=*"container"*>

<h2>Add New Product</h2>

<form:form action=*"/admin/product/add"* modelAttribute=*"product"* method=*"post"*>

<label for=*"productName"*>Product Name </label>

<input type=*"text"* id=*"productName"* name=*"productName"* required>

<label for=*"size"*>Size</label>

<input type=*"text"* id=*"size"* name=*"size"* required>

<label for=*"price"*>Price </label>

<input type=*"text"* id= *"price"* name=*"price"* required>

<label for=*"productCategory"* >Product Category</label>

<select class=*"form-control"* id=*"productCategory"*

name=*"productCategory"*>

<c:forEach items=*"*${categories}*"* var=*"category"*>

<option value=*"*${category.id}*"*>${category.brandName} | ${category.style} | ${category.targetGroup}</option>

</c:forEach>

</select>

<label for=*"image\_Url"*>Select Product Image </label>

<!-- select from folder -->

<div class=*"custom-file"*>

<input type=*"file"* class=*"custom-file-input"* name=*"image\_Url"* accept=*"image/jpeg,image/jpg, image/png"* id=*"image\_Url"*/>

<label class=*"custom-file-label"* for=*"image\_Url"*>Choose file</label>

</div>

<div class=*"form-group"*>

<img src=*"#"* id=*"imgPreview"* height=*"100px"* width=*"100px"* src=*"/productImages/nike101.jpg"* style="margin-top: *20px*" alt=*" "*>

</div>

<input type=*"hidden"* name=*"imgName"* value=*"/productImages/nike101.jpg"*>

<input class=*"btn, btn-success btn-lg .btn-block"* type=*"submit"*

name=*"Add"* value=*"Add Product"* >

<div style="color:*red*">${errMessageAddCategory}</div>

</form:form>

</div>

</body>

<style>

/\* Style all input fields \*/

**label,** *.action-label* {

font-size: *x-large*;

color: *black*;

}

**input,** **select**{

width: *100%*;

padding: *12px*;

border: *3px solid #ccc*;

border-radius: *4px*;

box-sizing: *border-box*;

margin-top: *6px*;

margin-bottom: *16px*;

}

/\* Style the container for inputs \*/

*.container* {

/\* width: 50%;

margin-left: 25%;

margin-top: 8% \*/

position: *relative*;

z-index: *2*;

width: *500px*;

margin: *50px auto*;

padding: *20px*;

background: *white*;

opacity: *90%*;

}

*.btn* {

background-color: *#04AA6D*;

color: *#0099CC*;

border: *2px solid #0099CC*;

border-radius: *6px*;

text-transform: *uppercase*;

}

</style>

<script>

**function** readURL(input){

**if**(input.files && input.files[0]){

**var** reader = **new** FileReader();

reader.onload = **function**(e){

$('#imgPreview').attr('src', e.target.result).width(100).height(100);

}

reader.readAsDataURL(input.files[0])

}

}

$('#image\_Url').change(**function**(){

readURL(**this**);

});

$(".custom-file-input").on("change", **function**() {

**var** fileName = $(**this**).val().split("\\").pop();

$(**this**).siblings(".custom-file-label").addClass("selected").html(fileName);

});

</script>

</html>

**addProductCategory.java**

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

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

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*

>

<link rel=*"stylesheet"* href=*"https://use.fontawesome.com/releases/v5.7.0/css/all.css"*

>

<title>Add Product Category</title>

</head>

<jsp:include page=*"adminHeader.jsp"* />

<body>

<div class=*"container"*>

<h2>Add Product Category</h2>

<form action=*"/categories/add"* method=*"post"* >

<label for=*"brandName"*>Brand Name </label>

<input type=*"text"* id=*"brandName"* name=*"brandName"* required>

<label for=*"style"*>Style</label>

<input type=*"text"* id=*"style"* name=*"style"* required>

<label for=*"targetGroup"*>Target Group </label>

<input type=*"text"* id=*"targetGroup"* name=*"targetGroup"* required>

<input class=*"btn, btn-success btn-lg .btn-block"* type=*"submit"*

name=*"Add"* value=*"Add Category"* >

<div style="color:*red*">${errMessageAddCategory}</div>

</form>

</div>

</body>

<style>

/\* Style all input fields \*/

**label,** *.action-label* {

font-size: *x-large*;

color: *black*;

}

**input** {

width: *100%*;

padding: *12px*;

border: *3px solid #ccc*;

border-radius: *4px*;

box-sizing: *border-box*;

margin-top: *6px*;

margin-bottom: *16px*;

}

/\* Style the container for inputs \*/

*.container* {

/\* width: 50%;

margin-left: 25%;

margin-top: 8% \*/

position: *relative*;

z-index: *2*;

width: *500px*;

margin: *50px auto*;

padding: *20px*;

background: *white*;

opacity: *90%*;

}

*.btn* {

background-color: *#04AA6D*;

color: *#0099CC*;

border: *2px solid #0099CC*;

border-radius: *6px*;

text-transform: *uppercase*;

}

</style>

</html>

**adminDetails.jsp**

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

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

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*>

<link rel=*"stylesheet"* href=*"https://use.fontawesome.com/releases/v5.7.0/css/all.css"*>

<title>Sporty Shoes</title>

<style>

</style>

</head>

<body>

<jsp:include page=*"adminHeader.jsp"* />

<div class=*"jumbotron text-center"* style="height: *400px*;">

<h1 class=*"display-4"*>Welcome To Admin</h1>

<p>manage Sporty Shoes data</p>

</div>

<div class=*"container-fluid"*>

<div class=*"row"*>

<div class=*"col-sm-3 pt-3"* >

<div class=*"card"* style="background-color:*white*;">

<div class=*"card-body"*>

<h4 class=*"card-title"*>Categories</h4>

<p class=*"card-text"*>Manage the categories section here.</p>

<a href=*"/admin/categories"* class=*"card-link btn btn btn-primary"*>Manage</a>

</div>

</div>

</div>

<div class=*"col-sm-3 pt-3"* >

<div class=*"card"* style="background-color:*white*;">

<div class=*"card-body"*>

<h4 class=*"card-title"*>Products</h4>

<p class=*"card-text"*>Manage all the products here.</p>

<a href=*"/admin/products"* class=*"card-link btn btn-primary"*>Manage</a>

</div>

</div>

</div>

<div class=*"col-sm-3 pt-3"* >

<div class=*"card"* style="background-color:*white*;">

<div class=*"card-body"*>

<h4 class=*"card-title"*>Registered Users</h4>

<p class=*"card-text"*>See All Registered Users Here.</p>

<a href=*"/admin/userList"* class=*"card-link btn btn-primary"*>List Users</a>

</div>

</div>

</div>

<div class=*"col-sm-3 pt-3"* >

<div class=*"card"* style="background-color:*white*;">

<div class=*"card-body"*>

<h4 class=*"card-title"*>Purchase Report</h4>

<p class=*"card-text"*>See Purchase Report Here</p>

<a href=*"/admin/purchaseReport"* class=*"card-link btn btn-primary"*>Purchase Report</a>

</div>

</div>

</div>

</div>

</div>

</body>

</html>

**adminHeader.jsp**

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

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

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<!DOCTYPE html>

<html>

<meta name=*"viewport"* content=*"width=device-width, initial-scale=1"*>

<script src=*"https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"* type=*"text/javascript"*></script>

<script src=*"https://ajax.aspnetcdn.com/ajax/jquery.ui/1.12.1/jquery-ui.min.js"* type=*"text/javascript"*></script>

<link href=*"https://code.jquery.com/ui/1.12.1/themes/smoothness/jquery-ui.css"* rel=*"stylesheet"* type=*"text/css"*>

<link rel=*"stylesheet"*

href=*"https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"*>

<script

src=*"https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"*></script>

<style>

*.navbar* {

z-index: *2*;

}

**h3**{

margin-top:*-45px*;

margin-left:*60px*;

color: *rgb(37, 150, 190)*;

}

</style>

<nav class=*"navbar navbar-dark bg-dark navbar-expand-sm "* >

<div class=*"container-fluid"*>

<!-- <nav class="nabar navbar-dark bg-dark navbar-expand-sm"> -->

<a class=*"navbar-brand"* href=*"#"*>

<img src=*"/images/shoeslogo.png"* width=*"50"* height=*"50"* class=*"d-inline-block align-top"* alt=*"Sporty Shoes"*>

<h3>Sporty Shoes</h3>

</a>

<ul class=*"nav nav-pills"* role=*"tablist"*>

<li class=*"nav-item"*><a class=*"nav-link"* href=*"/admin"*>Dashboard</a></li>

<c:if test = *"*${**not empty** userEmailId }*"*>

<li><span class=*"badge badge-success"* style = "font-size: *x-large*;">${userName}</span></li>

<li class=*"nav-item"*><a class=*"nav-link"* href=*"/logout"*>

<img src=*"/images/logout.png"* width=*"30"* height=*"30"* class=*"d-inline-block align-top"* alt=*""*>

Logout</a>

</li>

</c:if>

</ul>

</div>

</nav>

</html>

**Cart.jsp**

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

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

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*>

<link rel=*"stylesheet"* href=*"https://use.fontawesome.com/releases/v5.11.2/css/all.css"*>

<title>Cart </title>

</head>

<body>

<jsp:include page=*"header.jsp"* />

<main>

<div class=*"container"*>

<div style="color:*red*">${errMessage}</div>

<!--Section: Block Content-->

<section class=*"mt-5 mb-4"*>

<!--Grid row-->

<div class=*"row"*>

<!--Grid column-->

<div class=*"col-lg-8"*>

<!-- Card -->

<div class=*"card wish-list mb-4"*>

<div class=*"card-body"*>

<h5 class=*"mb-4"*>Cart (<span >${cartItems.size()}</span> items)</h5>

<c:forEach items=*"*${cart\_items}*"* var=*"item"*>

<c:url var=*"deleteItem"* value=*"/cart/deleteItem"*>

<c:param name =*"productId"* value=*"*${item.productId}*"*/>

</c:url>

<div class=*"row mb-4"* >

<div class=*"col-md-5 col-lg-3 col-xl-3"*>

<div class=*"mb-3 mb-md-0"*>

<img class=*"img-fluid w-100"*

src=*'/productImages/*${item.image\_Url}*'*

src=*"https://mdbootstrap.com/img/Photos/Horizontal/E-commerce/Vertical/12a.jpg"* alt=*"Sample"*>

</div>

</div>

<div class=*"col-md-7 col-lg-9 col-xl-9"*>

<div>

<div class=*"d-flex justify-content-between"*>

<div>

<h5 >${item.name}</h5>

<p class=*"mb-3 text-uppercase small"*>Quantity: <span> ${item.quantity}</span> </p>

</div>

</div>

<div class=*" d-flex justify-content-between align-items-center"*>

<div>

<a href=*"*${deleteItem}*"* type=*"button"* class=*"card-link-secondary small text-uppercase mr-3"*><i

class=*"fas fa-trash-alt mr-1"*></i> Remove item </a>

</div>

<p class=*"mb-0 "*><span><strong>Item Price: $<span >${item.price}</span></strong></span></p>

</div>

</div>

</div>

</div>

</c:forEach>

</div>

</div>

</div>

<!--Grid column-->

<!--Grid column-->

<div class=*"col-lg-4"*>

<!-- Card -->

<div class=*"card mb-4"*>

<div class=*"card-body"*>

<h5 class=*"mb-3"*>The total amount </h5>

<ul class=*"list-group list-group-flush"*>

<li class=*"list-group-item d-flex justify-content-between align-items-center border-0 px-0 pb-0"*>

Amount Payable

<span>$<span >${cartValue}</span></span>

</li>

</ul>

<c:if test = *"*${cartItems.size() > 0}*"*>

<a href=*"/checkout"* class=*"btn btn-primary btn-block waves-effect waves-light"*>go to

checkout</a>

</c:if>

</div>

</div>

<!-- Card -->

</div>

<!--Grid column-->

</div>

<!--Grid row-->

</section>

<!--Section: Block Content-->

</div>

</main>

<!--Main layout-->

</body>

</html>

**changePassword.jsp**

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

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

<%@ taglib prefix=*"form"* uri=*"http://www.springframework.org/tags/form"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*/link>

</head>

<script type=*"text/javascript"*>

**var** check = **function**() {

**if** (document.getElementById('password').value ==

document.getElementById('confirm\_password').value) {

document.getElementById('message').style.color = 'green';

document.getElementById('message').innerHTML = 'Password matching';

} **else** {

document.getElementById('message').style.color = 'red';

document.getElementById('message').innerHTML = 'Password not matching';

}

}

</script>

<body>

<jsp:include page=*"header.jsp"* />

<div class=*"container"*>

<h2> Change Password </h2>

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

<%-- <form:form action="/login/changePassword/" method="post" modelAttribute="user">

<form:label path="emailId">Email Address </form:label> <form:input type="text" path="emailId" />

<form:label path="password">Password: </form:label>

<form:input type="text" path="password" pattern="(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z]).{8,}"

title="Must contain at least one number and one uppercase and lowercase letter, and at least 8 or more characters"

/>

<input type="submit" value="submit"/>

</form:form> --%>

<label for=*"usrname"*>Email Address </label>

<input type=*"text"* id=*"emailId"* name=*"emailId"* required>

<%-- <input type="text" id="emailId" placeholder= ${emailIdVal }

class="form-control"> --%>

<label for=*"psw"*>Password</label>

<input type=*"password"* id=*"password"* name=*"password"* pattern=*"(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z]).{8,}"*

title=*"Must contain at least one number and one uppercase and lowercase letter, and at least 8 or more characters"*

required>

<label for=*"psw"*>Confirm Password</label>

<input type=*"password"* id=*"confirm\_password"* name=*"confirm\_password"* pattern=*"(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z]).{8,}"*

title=*"Must contain at least one number and one uppercase and lowercase letter, and at least 8 or more characters"*

required onkeyup='check();'>

<span id=*'message'*></span>

<input class=*"btn, btn-success btn-lg .btn-block"* type=*"submit"*

name=*"Reset"* value=*"Reset Password"* >

<a class=*"action-label btn, .btn-link"*

href=*"/homePage"*>Back to Home Page</a>

<div style="color:*green*">${changePasswordMsg}</div>

</form>

</div>

</body>

<style>

/\* Style all input fields \*/

**label,** *.action-label* {

font-size: *x-large*;

color: *black*;

}

**input** {

width: *100%*;

padding: *12px*;

border: *3px solid #ccc*;

border-radius: *4px*;

box-sizing: *border-box*;

margin-top: *6px*;

margin-bottom: *16px*;

}

/\* Style the container for inputs \*/

*.container* {

/\* width: 50%;

margin-left: 25%;

margin-top: 8% \*/

position: *relative*;

z-index: *2*;

width: *500px*;

margin: *50px auto*;

padding: *20px*;

background: *white*;

opacity: *90%*;

}

*.btn* {

background-color: *#04AA6D*;

color: *#0099CC*;

border: *2px solid #0099CC*;

border-radius: *6px*;

text-transform: *uppercase*;

}

</style>

</html>

**Checkout.jsp**

<%-- <%@ page language="java" contentType="text/html; charset=UTF-8"

pageEncoding="UTF-8"%> --%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*

>

<!-- <link rel="stylesheet" href="https://use.fontawesome.com/releases/v5.11.2/css/all.css"> -->

<title>Checkout</title>

</head>

<body>

<jsp:include page=*"header.jsp"* />

<div class=*"container"*>

<!--Section: Block Content-->

<section class=*"mt-5 mb-4"*>

<!--Grid row-->

<div class=*"row"*>

<!--Grid column-->

<div class=*"col-lg-8 mb-4"*>

<!-- Card -->

<div class=*"card wish-list pb-1"*>

<div class=*"card-body"*>

<h5 class=*"mb-2"*>Billing details</h5>

<!-- Grid row -->

<div class=*"row"*>

<!-- Grid column -->

<div class=*"col-lg-6"*>

<!-- First name -->

<div class=*"md-form md-outline mb-0 mb-lg-4"*>

<label for=*"firstName"*>First name</label>

<input type=*"text"* id=*"firstName"* required class=*"form-control mb-0 mb-lg-2"*>

</div>

</div>

<!-- Grid column -->

<!-- Grid column -->

<div class=*"col-lg-6"*>

<!-- Last name -->

<div class=*"md-form md-outline"*>

<label for=*"lastName"*>Last name</label>

<input type=*"text"* id=*"lastName"* required class=*"form-control"*>

</div>

</div>

<!-- Grid column -->

</div>

<!-- Grid row -->

<!-- Country -->

<div class=*"d-flex flex-wrap"*>

<div class=*"select-outline position-relative w-100"*>

<label>Country - <strong>USA</strong></label>

</div>

</div>

<!-- Address Part 1 -->

<div class=*"md-form md-outline mt-2"*>

<label for=*"form14"*>Address (Line 1)</label>

<input type=*"text"* required id=*"form14"* placeholder=*"House number and street name"* class=*"form-control"*>

</div>

<!-- Address Part 2 -->

<div class=*"md-form md-outline mt-2"*>

<label for=*"form15"*>Address (Line 2)</label>

<input type=*"text"* id=*"form15"* placeholder=*"Apartment, suite, unit etc. (optional)"*

class=*"form-control"*>

</div>

<!-- Postcode / ZIP -->

<div class=*"md-form md-outline mt-3"*>

<label for=*"form16"*>Postcode / ZIP</label>

<input type=*"text"* maxlength=*"6"* minlength=*"6"* required id=*"form16"* class=*"form-control"*>

</div>

<!-- Town / City -->

<div class=*"md-form md-outline mt-2"*>

<label for=*"form17"*>Town / City</label>

<input type=*"text"* required id=*"form17"* class=*"form-control"*>

</div>

<!-- Phone -->

<div class=*"md-form md-outline mt-2"*>

<label for=*"form18"*>Phone</label>

<input type=*"number"* minlength=*"10"* maxlength=*"10"* required id=*"form18"* class=*"form-control"*>

</div>

<!-- Email address -->

<div class=*"md-form md-outline mt-2"*>

<label for=*"form19"*>Email address</label>

<input type=*"email"* required id=*"form19"* class=*"form-control"*>

</div>

<!-- Additional information -->

<div class=*"md-form md-outline mt-2"*>

<label for=*"form76"*>Additional information</label>

<textarea id=*"form76"* class=*"md-textarea form-control"* rows=*"4"*></textarea>

</div>

</div>

</div>

<!-- Card -->

</div>

<!--Grid column-->

<!--Grid column-->

<div class=*"col-lg-4"*>

<!-- Card -->

<div class=*"card mb-4"*>

<div class=*"card-body"*>

<h5 class=*"mb-3"*>The total amount </h5>

<ul class=*"list-group list-group-flush"*>

<li class=*"list-group-item d-flex justify-content-between align-items-center border-0 px-0 pb-0"*>

Amount Payable

<span>$${cartTotal}</span>

</li>

<li class=*"list-group-item d-flex justify-content-between align-items-center px-0"*>

Shipping

<span>$${shipping}</span>

</li>

<li class=*"list-group-item d-flex justify-content-between align-items-center border-0 px-0 mb-3"*>

<div>

<strong>The total amount of</strong>

<strong>

(including VAT)

</strong>

</div>

<span><strong>$${total}</strong></span>

</li>

</ul>

<form method=*"post"* action=*"/payNow"* >

<button type=*"submit"* class=*"btn btn-primary btn-block waves-effect waves-light"*>Pay Now</button>

</form>

</div>

</div>

<!-- Card -->

<!-- <!-- Card -->

<!-- <div class="card mb-4">

<div class="card-body">

<a class="dark-grey-text d-flex justify-content-between" data-toggle="collapse" href="#collapseExample"

aria-expanded="false" aria-controls="collapseExample">

Add a discount code (optional)

<span><i class="fas fa-chevron-down pt-1"></i></span>

</a>

<div class="collapse" id="collapseExample">

<div class="mt-3">

<div class="md-form md-outline mb-0">

<input type="text" id="discount-code" class="form-control font-weight-light"

placeholder="Enter discount code">

</div>

</div>

</div>

</div>

</div> -->

<!-- Card -->

</div>

<!--Grid column-->

</div>

<!--Grid row-->

</section>

<!--Section: Block Content-->

</div>

</body>

</html>

**Confirm.jsp**

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

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

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*>

<link rel=*"stylesheet"* href=*"https://use.fontawesome.com/releases/v5.7.0/css/all.css"*>

<title>Order Successful</title>

</head>

<jsp:include page=*"header.jsp"* />

<body>

<div class=*"container"* >

<table class=*"table table-borderless table-striped"*>

<tr>

<td><h5> Your Order has been Confirmed !!</h5> </td>

</tr>

<tr>

<td><h5> Order No: ${orderNo}</h5></td>

</tr>

<tr>

<td>

<a href=*"/shop"*>

<button type=*"button"* class=*"btn btn-primary btn-lg"*>Continue shopping</button>

</a>

</td>

</tr>

</table>

</div>

</body>

</html>

**deleteProduct.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>Delete Product</title>

</head>

<body>

</body>

</html>

**editProduct.jsp**

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

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

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<%@ taglib prefix=*"form"* uri=*"http://www.springframework.org/tags/form"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*>

<link rel=*"stylesheet"* href=*"https://use.fontawesome.com/releases/v5.7.0/css/all.css"*>

<title>Edit Product</title>

</head>

<body>

<jsp:include page=*"adminHeader.jsp"* />

<div class=*"container"*>

<h2>Edit Product Detail</h2>

<form:form action=*"/admin/product/Edit/save"* modelAttribute=*"product"* method=*"post"*>

<form:hidden path=*"productCode"*/>

<label for=*"productName"*>Product Name </label>

<form:input path=*"productName"*/>

<label for=*"size"*>Size</label>

<form:input path=*"size"*/>

<label for=*"price"*>Price </label>

<form:input path=*"price"*/>

<label for=*"productCategory"* >Product Category</label>

<select

name=*"productCategory"*>

<c:forEach items=*"*${categories}*"* var=*"category"*>

<c:choose>

<c:when test=*"*${category.id == product.productCategory.id}*"*>

<option value=*"*${category.id}*"* selected=*"selected"*>

${category.brandName} | ${category.style} | ${category.targetGroup}

</option>

</c:when>

<c:otherwise>

<option value=*"*${category.id}*"*>${category.brandName} | ${category.style} | ${category.targetGroup}</option>

</c:otherwise>

</c:choose>

</c:forEach>

</select>

<label for=*"image\_Url"*>Image </label>

<%-- <form:input path="image\_Url"/> --%>

<img class=*""* height=*"150px"* width=*"150px"* src=*'/productImages/*${product.image\_Url}*'* alt=*"*${product.productName}*"*>

<!-- select from folder -->

<input class=*"btn, btn-success btn-lg .btn-block"* type=*"submit"*

name=*"Edit"* value=*"Edit Product"* >

<div style="color:*red*">${errMessageAddCategory}</div>

</form:form>

</div>

</body>

<style>

/\* Style all input fields \*/

**label,** *.action-label* {

font-size: *x-large*;

color: *black*;

}

**input** {

width: *100%*;

padding: *12px*;

border: *3px solid #ccc*;

border-radius: *4px*;

box-sizing: *border-box*;

margin-top: *6px*;

margin-bottom: *16px*;

}

/\* Style the container for inputs \*/

*.container* {

/\* width: 50%;

margin-left: 25%;

margin-top: 8% \*/

position: *relative*;

z-index: *2*;

width: *500px*;

margin: *50px auto*;

padding: *20px*;

background: *white*;

opacity: *90%*;

}

*.btn* {

background-color: *#04AA6D*;

color: *#0099CC*;

border: *2px solid #0099CC*;

border-radius: *6px*;

text-transform: *uppercase*;

}

</style>

</html>

**editProductCategory.jsp**

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

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

<%@ taglib prefix=*"form"* uri=*"http://www.springframework.org/tags/form"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*

>

<link rel=*"stylesheet"* href=*"https://use.fontawesome.com/releases/v5.7.0/css/all.css"*

>

<title>Edit Product Category </title>

</head>

<body>

<div class=*"container"*>

<form:form action=*"/category/update/save"* modelAttribute=*"productCategory"* method=*"post"*>

<form:hidden path=*"id"*/>

<label for=*"brandName"*>Brand Name </label>

<form:input path=*"brandName"*/>

<label for=*"style"*>Style</label>

<form:input path=*"style"*/>

<label for=*"targetGroup"*>Target Group </label>

<form:input path=*"targetGroup"*/>

<input class=*"btn, btn-success btn-lg .btn-block"* type=*"submit"*

name=*"Edit"* value=*"Edit Category"* >

<div style="color:*red*">${errMessageAddCategory}</div>

</form:form>

</div>

</body>

<style>

/\* Style all input fields \*/

**label,** *.action-label* {

font-size: *x-large*;

color: *black*;

}

**input** {

width: *100%*;

padding: *12px*;

border: *3px solid #ccc*;

border-radius: *4px*;

box-sizing: *border-box*;

margin-top: *6px*;

margin-bottom: *16px*;

}

/\* Style the container for inputs \*/

*.container* {

/\* width: 50%;

margin-left: 25%;

margin-top: 8% \*/

position: *relative*;

z-index: *2*;

width: *500px*;

margin: *50px auto*;

padding: *20px*;

background: *white*;

opacity: *90%*;

}

*.btn* {

background-color: *#04AA6D*;

color: *#0099CC*;

border: *2px solid #0099CC*;

border-radius: *6px*;

text-transform: *uppercase*;

}

</style>

</html>

**Footer.jsp**

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

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

<!DOCTYPE html>

<html>

<head>

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

<style type=*"text/css"*>

*#bg* {

position: *fixed*;

top: *-50%*;

left: *-50%*;

width: *200%*;

height: *200%*;

}

*#bg* **img** {

position: *absolute*;

top: *0*;

left: *0*;

right: *0*;

bottom: *0*;

margin: *auto*;

min-width: *50%*;

min-height: *50%*;

}

</style>

</head>

<div id=*"bg"*>

<img src=*"/images/background.jpg"* alt=*""*>

</div>

<body>

</body>

</html>

**Header.jsp**

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

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

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<!DOCTYPE html>

<html>

<meta name=*"viewport"* content=*"width=device-width, initial-scale=1"*>

<script src=*"https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"* type=*"text/javascript"*></script>

<script src=*"https://ajax.aspnetcdn.com/ajax/jquery.ui/1.12.1/jquery-ui.min.js"* type=*"text/javascript"*></script>

<link href=*"https://code.jquery.com/ui/1.12.1/themes/smoothness/jquery-ui.css"* rel=*"stylesheet"* type=*"text/css"*>

<link rel=*"stylesheet"*

href=*"https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/css/bootstrap.min.css"*>

<script

src=*"https://maxcdn.bootstrapcdn.com/bootstrap/4.0.0/js/bootstrap.min.js"*></script>

<style>

*.navbar* {

z-index: *2*;

}

**h3**{

margin-right:*600px*;

color: *rgb(37, 150, 190)*;

}

</style>

<nav class=*"navbar navbar-dark bg-dark navbar-expand-sm "* >

<div class=*"container-fluid"*>

<!-- <nav class="navbar navbar-dark bg-dark navbar-expand-sm"> -->

<img src=*"/images/shoeslogo.png"* width=*"50"* height=*"50"* class=*"d-inline-block align-top"* alt=*"Sporty Shoes"*>

<h3>Sporty Shoes</h3>

<ul class=*"nav nav-pills"* role=*"tablist"*>

<li class=*"nav-item"*><a class=*"nav-link"* href=*"/homePage"*>Home</a></li>

<li class=*"nav-item"*><a class=*"nav-link"* href=*"/shop"*>Shop</a></li>

<c:choose>

<c:when test=*"*${**empty** userEmailId }*"*>

<li class=*"nav-item"*><a class=*"nav-link"* href=*"/login"*>Login</a></li>

</c:when>

<c:when test = *"*${**not empty** userEmailId }*"*>

<li><span class=*"badge badge-success"* style = "font-size: *x-large*;"> ${userName}</span></li>

<li class=*"nav-item"*><a class=*"nav-link"* href=*"/logout"*>

<img src=*"/images/logout.png"* width=*"30"* height=*"30"* class=*"d-inline-block align-top"* alt=*""*>

Logout</a>

</li>

</c:when>

</c:choose>

<li class=*"nav-item"*>

<a class=*"nav-link"* href=*"/cart"*>

<img src=*"/images/shopping-cart.gif"* width=*"30"* height=*"30"* class=*"d-inline-block align-top"* alt=*""*>

Cart

<span class=*"badge badge-primary badge-pill"* style="vertical-align: *top*;" > ${cartCount}</span>

</a>

</li>

</ul>

</div>

</nav>

</html>

**Login.jsp**

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

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

<%@ taglib prefix=*"form"* uri=*"http://www.springframework.org/tags/form"*%>

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*

>

<title>Login</title>

<script type=*"text/javascript"*>

**function** getEmailId() {

**var** emailIdVar = document.getElementById('emailId').value;

}

</script>

</head>

<jsp:include page=*"header.jsp"* />

<div class=*"container"*>

<h1 style="text-align:*center*">User Login</h1>

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

<label for=*"usrname"*>Email Address</label>

<input type=*"text"*

id=*"emailId"* name=*"emailId"* required >

<label for=*"psw"*>Password</label>

<input type=*"password"* id=*"password"* name=*"password"*

required>

<input class=*"btn, btn-success btn-lg .btn-block"* type=*"submit"*

name=*"Login"* value=*"Login"* >

<a class=*"action-label btn, .btn-link"*

href=*"/login/changePassword/*${emailIdVar}*"* >Change Password </a>

&ensp;&ensp;&ensp;

<a class=*"action-label btn, .btn-link"* href=*"/register"*>User Sign Up</a>

<div style="color:*red*">${errorMessage}</div>

</form>

</div>

</body>

<style>

/\* Style all input fields \*/

**label,** *.action-label* {

font-size: *x-large*;

color: *black*;

}

**input** {

width: *100%*;

padding: *12px*;

border: *3px solid #ccc*;

border-radius: *4px*;

box-sizing: *border-box*;

margin-top: *6px*;

margin-bottom: *16px*;

}

/\* Style the container for inputs \*/

*.container* {

/\* width: 50%;

margin-left: 25%;

margin-top: 8% \*/

position: *relative*;

z-index: *2*;

width: *500px*;

margin: *50px auto*;

padding: *20px*;

background: *white*;

opacity: *90%*;

}

*.btn* {

background-color: *#04AA6D*;

color: *#0099CC*;

border: *2px solid #0099CC*;

border-radius: *6px*;

text-transform: *uppercase*;

}

</style>

</html>

**productCategoryList.jsp**

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

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

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*

>

<link rel=*"stylesheet"* href=*"https://use.fontawesome.com/releases/v5.7.0/css/all.css"*

>

<title>Product Category List</title>

<style>

**td**{

text-align:*center*;

}

</style>

</head>

<body>

<jsp:include page=*"adminHeader.jsp"* />

<div class=*"container"*>

<a href=*"/admin/categories/add"* style="margin: *20px 0*" class=*"btn btn-primary"*>Add Category</a>

<table class =*"table table-striped"* border=*"1"* width=*"100%"* >

<thead class=*"thead-dark"*>

<tr>

<th>Brand Name<th>Wear Type</th><th>Target group</th><th>Edit</th><th>Delete</th>

</tr>

</thead>

<c:forEach var=*"prodCategory"* items=*"*${categories}*"*>

<c:url var=*"editLink"* value=*"/admin/category/Edit"*>

<c:param name =*"categoryId"* value=*"*${prodCategory.id}*"*/>

</c:url>

<c:url var=*"deleteLink"* value=*"/admin/category/Delete"*>

<c:param name =*"categoryId"* value=*"*${prodCategory.id}*"*/>

</c:url>

<tr>

<td>${prodCategory.brandName}</td>

<td>${prodCategory.style}</td>

<td>${prodCategory.targetGroup}</td>

<td><a class=*"btn btn-primary"* style="padding-left:*25px*; padding-right:*25px*; color:*white*"; href=*"*${editLink}*"*>Edit</a></td>

<td><a class=*"btn btn-danger"* href=*"*${deleteLink}*"* onclick="if (!('are you sure you want to delete this todo?'))) return false">Delete</a></td>

</tr>

</c:forEach>

</table>

</div>

</body>

</html>

**productList.jsp**

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

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

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*>

<link rel=*"stylesheet"* href=*"https://use.fontawesome.com/releases/v5.7.0/css/all.css"*>

<title>Product Management Page</title>

<style>

**td**{

text-align:*center*;

}

</style>

</head>

<body>

<jsp:include page=*"adminHeader.jsp"* />

<div class=*"container"*>

<a href=*"/admin/product/add"* style="margin: *20px 0*" class=*"btn btn-primary"*>Add Product</a>

<table class =*"table table-striped"* border=*"1"* width=*"100%"* >

<thead class=*"thead-dark"*>

<tr>

<th>Code<th>Name</th><th>Brand</th><th>Preview Image</th><th>Edit</th><th>Delete</th>

</tr>

</thead>

<c:forEach var=*"product"* items=*"*${products}*"*>

<c:url var=*"editLink"* value=*"/admin/product/Edit"*>

<c:param name =*"productId"* value=*"*${product.productCode}*"*/>

</c:url>

<c:url var=*"deleteLink"* value=*"/admin/product/Delete"*>

<c:param name =*"productId"* value=*"*${product.productCode}*"*/>

</c:url>

<tr>

<td>${product.productCode}</td>

<td>${product.productName}</td>

<td>${product.getProductCategory().getBrandName()}</td>

<%-- <td>${product.image\_Url} </td> --%>

<td>

<%-- <img class="mt-5" height="300px" width="300px" src="@{${'/productImages/' + product.image\_Url+'.jpg'}}"

alt=""> --%>

<img height=*"100px"* width=*"100px"* src=*'/productImages/*${product.image\_Url}*'*

alt=*""*>

</td>

<td><a class=*"btn btn-primary"* style="padding-left:*25px*; padding-right:*25px*; margin-top:*30px*" href=*"*${editLink}*"*>Edit</a></td>

<td><a class=*"btn btn-danger"* style="margin-top:*30px*" href=*"*${deleteLink}*"* onclick="if (!('are you sure you want to delete this todo?'))) return false">Delete</a></td>

</tr>

</c:forEach>

</table>

</div>

</body>

</html>

**purchaseReport.jsp**

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

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

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*>

<link rel=*"stylesheet"* href=*"https://use.fontawesome.com/releases/v5.7.0/css/all.css"*>

<script src=*"https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"*></script>

<script src=*"https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"*></script>

<title> Purchase Report</title>

</head>

<body>

<jsp:include page=*"adminHeader.jsp"* />

<div class=*"container"*>

<h3>Purchase Report</h3>

<input class=*"form-control"* id=*"myInput"* type=*"text"* placeholder=*"Search.."*>

<table class =*"table table-striped"* border=*"1"* width=*"100%"* >

<thead class=*"thead-dark"*>

<tr>

<th>Order No.<th>User Name</th><th>User Email Id</th><th>Product Name</th>

<th>Product Category</th><th>Order Date</th><th>Price</th>

</tr>

</thead>

<tbody id=*"myTable"*>

<c:forEach items=*"*${purchaseItemList}*"* var=*"p"*>

<tr><td>${p.purchaseOrder.ID}</td>

<td>${p.user.firstName}</td>

<td>${p.user.emailId}</td>

<td>${p.poduct.productName}</td>

<td>${p.poduct.productCategory.brandName}</td>

<td>${p.purchaseOrder.date}</td>

<td>${p.price}</td>

</c:forEach>

</tbody>

</table>

</div>

<script>

$(document).ready(**function**(){

$("#myInput").on("keyup", **function**() {

**var** value = $(**this**).val().toLowerCase();

$("#myTable tr").filter(**function**() {

$(**this**).toggle($(**this**).text().toLowerCase().indexOf(value) > -1)

});

});

});

</script>

</body>

</html>

**Register.jsp**

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

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

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*>

<title>New Customer Register</title>

</head>

<body>

<jsp:include page=*"header.jsp"* />

<div class=*"container"*>

<h2>User Signup</h2>

<form action=*"/register/save"* method=*"post"* >

<label for=*"usrname"*>Email Id </label>

<input type=*"text"* id=*"emailId"* name=*"emailId"* required placeholder=*"Email Id"*>

<label for=*"psw"*>Password</label>

<input type=*"password"* id=*"psw"* name=*"password"*

pattern=*"(?=.\*\d)(?=.\*[a-z])(?=.\*[A-Z]).{8,}"*

title=*"Must contain at least one number and one uppercase and lowercase letter, and at least 8 or more characters"*

required placeholder=*"Password"*>

<label for=*"firstName"*>First Name </label>

<input type=*"text"* id=*"firstName"* name=*"firstName"* required placeholder=*"First Name"*>

<label for=*"lastName"*>Last Name</label>

<input type=*"text"* id=*"lastName"* name=*"lastName"* required placeholder=*"Last Name"*>

<label for=*"country"*>Country </label>

<input type=*"text"* id=*"country"* name=*"country"* required placeholder=*"Country"*>

<label for=*"dateOfBirth"* class=*"text-dark"*>Date Of Birth</label>

<input name=*"dateOfBirth"* type=*"text"*

id=*"dateOfBirth"* required=*"required"* placeholder=*"Date of Birth YYYY/MM/DD"*/>

<div>

<label for=*"gender"* class=*"text-dark "*>Gender</label>

<div class=*"form-check-inline"*>

<label class=*"radio-inline form-check-label"*><input class=*"form-check-input"* type=*"radio"* value=*"M"* name=*"gender"* checked=*"checked"* />Male</label>

</div>

<div class=*"form-check-inline"*>

<label class=*"radio-inline form-check-label"*><input class=*"form-check-input"* type=*"radio"* value=*"F"* name=*"gender"* />Female</label>

</div>

</div>

<input class=*"btn, btn-success btn-lg .btn-block"* type=*"submit"*

name=*"SignUp"* value=*"Create Account"* >

<div style="color:*red*">${errMessageSignIn}</div>

</form>

</div>

<script>

$(**function**() {

$("#dateOfBirth").datepicker({

dateFormat: 'yy/mm/dd',

changeMonth: **true**,

changeYear: **true**,

inline: **true**,

minDate: 0

});

});

</script>

</body>

<style>

/\* Style all input fields \*/

**label,** *.action-label* {

font-size: *x-large*;

color: *black*;

}

**input** {

width: *100%*;

padding: *12px*;

border: *3px solid #ccc*;

border-radius: *4px*;

box-sizing: *border-box*;

margin-top: *6px*;

margin-bottom: *16px*;

}

/\* Style the container for inputs \*/

*.container* {

/\* width: 50%;

margin-left: 25%;

margin-top: 8% \*/

position: *relative*;

z-index: *2*;

width: *500px*;

margin: *50px auto*;

padding: *20px*;

background: *white*;

opacity: *90%*;

}

*.btn* {

background-color: *#04AA6D*;

color: *#0099CC*;

border: *2px solid #0099CC*;

border-radius: *6px*;

text-transform: *uppercase*;

}

</style>

</html>

**registeredUserList.jsp**

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

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

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<!DOCTYPE html>

<html>

<head>

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

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*>

<link rel=*"stylesheet"* href=*"https://use.fontawesome.com/releases/v5.7.0/css/all.css"*>

<script src=*"https://ajax.googleapis.com/ajax/libs/jquery/3.6.0/jquery.min.js"*></script>

<script src=*"https://maxcdn.bootstrapcdn.com/bootstrap/3.4.1/js/bootstrap.min.js"*></script>

<title>Registered Users List</title>

</head>

<jsp:include page=*"adminHeader.jsp"* />

<body>

<!-- <div>

<form action = "/admin/users/search" method ="post">

<h2> Registered User</h2>

<label for="SearchCriteria">Search</label>

<input type="text" name="searchId" placeholder="Enter any value">

<button> Search </button>

</form>

</div>

-->

<div class=*"container"*>

<h3>Registered User List </h3>

<input class=*"form-control"* id=*"myInput"* type=*"text"* placeholder=*"Search any value.."*>

<table class =*"table table-striped"* border=*"1"* width=*"100%"* >

<thead class=*"thead-dark"*>

<tr>

<th>Email Id<th>First Name</th><th>Last Name</th><th>Country</th>

<th>Gender</th><th>Date Of Birth</th>

</tr>

</thead>

<tbody id=*"myTable"*>

<c:forEach items=*"*${userList}*"* var=*"user"*>

<tr><td>${user.emailId}</td>

<td>${user.firstName}</td>

<td>${user.lastName}</td>

<td>${user.country}</td>

<td>${user.gender}</td>

<td>${user.dateOfBirth}</td>

</c:forEach>

</tbody>

</table>

</div>

<script>

$(document).ready(**function**(){

$("#myInput").on("keyup", **function**() {

**var** value = $(**this**).val().toLowerCase();

$("#myTable tr").filter(**function**() {

$(**this**).toggle($(**this**).text().toLowerCase().indexOf(value) > -1)

});

});

});

</script>

</body>

</html>

**Shop.jsp**

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

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

<%@ taglib uri=*"http://java.sun.com/jsp/jstl/core"* prefix=*"c"*%>

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*

>

<link rel=*"stylesheet"* href=*"https://use.fontawesome.com/releases/v5.7.0/css/all.css"*

>

<title>Main Shop Page</title>

</head>

<jsp:include page=*"header.jsp"* />

<body>

<div class=*"container"* style="margin-top: *30px*">

<div class=*"row"*>

<div class=*"col-sm-3"*>

<div class=*"col-sm-10 sidenav"*>

<h6 class=*"font-weight-bold"*>Category</h6>

<div class=*"list-group list-group-flush list-group-no-border list-group-sm"*>

<a href=*"/shop"* class=*"list-group-item list-group-item-action"*>All Products</a>

<a href=*"/shop/Gender/Men"* class=*"list-group-item list-group-item-action"*>Men</a>

<a href=*"/shop/Gender/Women"* class=*"list-group-item list-group-item-action"*>Women</a>

<a href=*""* class=*"list-group-item list-group-item-action"*></a>

</div>

</div>

<div class=*"col-sm-10 sidenav"*>

<h6 class=*"font-weight-bold"*>Brand</h6>

<div class=*"list-group list-group-flush list-group-no-border list-group-sm"*>

<a href=*"/shop/brand/Nike"* class=*"list-group-item list-group-item-action"*>Nike</a>

<a href=*"/shop/brand/WoodLand"* class=*"list-group-item list-group-item-action"*>WoodLand</a>

<a href=*"/shop/brand/Adidas"* class=*"list-group-item list-group-item-action"*>Adidas</a>

</div>

</div>

<div class=*"col-sm-10 sidenav"*>

<h6 class=*"font-weight-bold"*>Style</h6>

<div class=*"list-group list-group-flush list-group-no-border list-group-sm"*>

<a href=*"/shop/style/Running"* class=*"list-group-item list-group-item-action"*>Running</a>

<a href=*"/shop/style/Basketball"* class=*"list-group-item list-group-item-action"*>Basketball</a>

<a href=*"/shop/style/Sneakers"* class=*"list-group-item list-group-item-action"*>Sneakers</a>

</div>

</div>

</div>

<div class=*"col-sm-9"*>

<div class=*"card"* style="margin-top: *20px*" >

<c:forEach var=*"product"* items=*"*${products}*"*>

<div class=*"row no-gutters"* style="margin-top: *20px*">

<div class=*"col-sm-5 d-flex justify-content-center"*>

<img class=*""* height=*"150px"* width=*"150px"* src=*'/productImages/*${product.image\_Url}*'* alt=*"*${product.productName}*"*>

</div>

<div class=*"col-sm-7 d-flex justify-content-center"*>

<div class=*"card-body"*>

<h4 class=*"card-title"* >${product.productName}</h4>

<h5> <span>$${product.price}</span></h5>

<h5>Size:

<span> ${product.size}</span>

</h5>

<a href=*"/shop/viewProduct/*${product.productCode}*"* class=*"btn btn-primary"*>View Product</a>

</div>

</div>

</div>

</c:forEach>

</div>

</div>

</div>

</div>

</body>

</html>

**viewProduct.jsp**

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

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

<!DOCTYPE html>

<html>

<head>

<meta charset=*"UTF-8"*>

<meta name=*"viewport"*

content=*"width=device-width, user-scalable=no, initial-scale=1.0, maximum-scale=1.0, minimum-scale=1.0"*>

<meta http-equiv=*"X-UA-Compatible"* content=*"ie=edge"*>

<link rel=*"stylesheet"* href=*"https://stackpath.bootstrapcdn.com/bootstrap/4.4.1/css/bootstrap.min.css"*>

<link rel=*"stylesheet"* href=*"https://use.fontawesome.com/releases/v5.7.0/css/all.css"*>

<title>View Product</title>

</head>

<body>

<jsp:include page=*"header.jsp"* />

<div class=*"container mt-5"*>

<div class=*"row"*>

<div class=*"col-sm-5 d-flex justify-content-center"*>

<img class=*"mt-5"* height=*"300px"* width=*"300px"* src=*'/productImages/*${product.image\_Url}*'*

alt=*""*>

</div>

<div class=*"col-sm-7 mt-5"*>

<h2 >${product.productName}</h2>

<h2><span> $${product.price}</span></h2>

<h5>${product.productCategory.brandName}</h5>

<h5>${product.productCategory.targetGroup} Shoes</h5>

<h5>Style : ${product.productCategory.style} </h5>

<h5>Size:

<span> ${product.size}</span>

</h5>

<a href=*"/addToCart/*${product.productCode}*"* class=*"btn btn-primary"*>Add to cart</a>

</div>

</div>

</div>

</body>

</html>

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*END\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*