//controller

package org.sporty\_shoes.controller;

import org.sporty\_shoes.model.Admin;

import org.sporty\_shoes.repository.AdminRepository;

import org.sporty\_shoes.repository.ProductRepository;

import org.sporty\_shoes.repository.PurchaseReportRepository;

import org.sporty\_shoes.repository.Repository;

import org.sporty\_shoes.repository.UserRepository;

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

import org.springframework.stereotype.Controller;

import org.springframework.ui.ModelMap;

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

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

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

@Controller

public class MyController {

@Autowired

AdminRepository adminRepository;

@Autowired

ProductRepository productRepository;

@Autowired

UserRepository userRepository;

@Autowired

PurchaseReportRepository purchaseReportRepository;

@Autowired

Repository repository;

@RequestMapping("/")

public String showHome() {

return "Home";

}

@RequestMapping(value="adminLogin",method =RequestMethod.GET)

public String adminLoginPage() {

return "AdminLogin";

}

@RequestMapping(value="forgotPassword",method =RequestMethod.GET)

public String forgotPassword() {

return "ForgotPassword";

}

@RequestMapping(value="adminForgotPassword",method =RequestMethod.POST)

public String adminForgotPasswordPage(@RequestParam("email")String email,

@RequestParam("password")String password,ModelMap map) {

if(repository.forgotPassword(email,password))

map.addAttribute("message","Password Updated");

else

map.addAttribute("message","Invalid Details");

return "ForgotPassword";

}

@RequestMapping(value="adminPage",method=RequestMethod.POST)

public String adminPage(@RequestParam(name="email")String email,

@RequestParam(name="password")String password,ModelMap map) {

if(adminRepository.verifyAdmin(new Admin(email,password)))

return "AdminPage";

else {

map.addAttribute("message", "Invalid Details");

return "AdminLogin";

}

}

@RequestMapping(value="product",method=RequestMethod.GET)

public String getAllProducts(ModelMap map) {

map.addAttribute("productList",productRepository.getAllProducts());

return "Product";

}

@RequestMapping(value="user",method=RequestMethod.GET)

public String getAllUsers(ModelMap map) {

map.addAttribute("userList",userRepository.getAllUsers());

return "User";

}

@RequestMapping(value="searchUser",method=RequestMethod.GET)

public String searchUser(@RequestParam("email")String email,ModelMap map) {

map.addAttribute("userList",userRepository.searchUser(email));

return "User";

}

@RequestMapping(value="purchaseReport",method=RequestMethod.GET)

public String getRport(ModelMap map) {

map.addAttribute("report",purchaseReportRepository.getReport());

return "PurchaseReport";

}

@RequestMapping(value="deleteProduct",method=RequestMethod.GET)

public String deleteProduct(@RequestParam("id")int id,ModelMap map) {

if(productRepository.deleteProduct(id))

map.addAttribute("message","Product Deleted Successfully");

else

map.addAttribute("message","Try after few minutes");

map.addAttribute("productList",productRepository.getAllProducts());

return "Product";

}

}

//model

package org.sporty\_shoes.model;

import java.io.Serializable;

import javax.persistence.Entity;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name = "admin")

public class Admin implements Serializable {

private static final long serialVersionUID = 1L;

@Id

private String email;

private String password;

public Admin() {

super();

}

public Admin(String email, String password) {

super();

this.email = email;

this.password = password;

}

public String getEmail() {

return email;

}

public void setEmail(String email) {

this.email = email;

}

public String getPassword() {

return password;

}

public void setPassword(String password) {

this.password = password;

}

@Override

public String toString() {

return String.format("Admin [email=%s, password=%s]", email, password);

}

}

package org.sporty\_shoes.model;

import java.io.Serializable;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name="product")

public class Product implements Serializable{

private static final long serialVersionUID = 1L;

@Id

@GeneratedValue(strategy=GenerationType.AUTO)

private int productId;

private String productName;

private int productPrice;

private String productDiscription;

public Product() {

super();

}

public Product(String productName, int productPrice, String productDiscription) {

super();

this.productName = productName;

this.productPrice = productPrice;

this.productDiscription = productDiscription;

}

public int getProductId() {

return productId;

}

public void setProductId(int productId) {

this.productId = productId;

}

public String getProductName() {

return productName;

}

public void setProductName(String productName) {

this.productName = productName;

}

public int getProductPrice() {

return productPrice;

}

public void setProductPrice(int productPrice) {

this.productPrice = productPrice;

}

public String getProductDiscription() {

return productDiscription;

}

public void setProductDiscription(String productDiscription) {

this.productDiscription = productDiscription;

}

@Override

public String toString() {

return String.format("Product [productId=%s, productName=%s, productprice=%s, productDiscription=%s]",

productId, productName, productPrice, productDiscription);

}

}

package org.sporty\_shoes.model;

import java.io.Serializable;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name="purchase\_report")

public class PurchaseReport implements Serializable {

private static final long serialVersionUID = 1L;

@Id

@GeneratedValue(strategy=GenerationType.AUTO)

private int reportId;

private String reportProductName;

private String reportUserEmail;

private String reportDate;

private int reportPrice;

public PurchaseReport() {

super();

}

public PurchaseReport(String reportProductName, String reportUserEmail, String reportDate, int reportPrice) {

super();

this.reportProductName = reportProductName;

this.reportUserEmail = reportUserEmail;

this.reportDate = reportDate;

this.reportPrice = reportPrice;

}

public int getReportId() {

return reportId;

}

public void setReportId(int reportId) {

this.reportId = reportId;

}

public String getReportProductName() {

return reportProductName;

}

public void setReportProductName(String reportProductName) {

this.reportProductName = reportProductName;

}

public String getReportUserEmail() {

return reportUserEmail;

}

public void setReportUserEmail(String reportUserEmail) {

this.reportUserEmail = reportUserEmail;

}

public String getReportDate() {

return reportDate;

}

public void setReportDate(String reportDate) {

this.reportDate = reportDate;

}

public int getReportPrice() {

return reportPrice;

}

public void setReportPrice(int reportPrice) {

this.reportPrice = reportPrice;

}

public static long getSerialversionuid() {

return serialVersionUID;

}

@Override

public String toString() {

return String.format(

"PurchaseReport [reportId=%s, reportProductName=%s, reportUserEmail=%s, reportDate=%s, reportPrice=%s]",

reportId, reportProductName, reportUserEmail, reportDate, reportPrice);

}

}

package org.sporty\_shoes.model;

import java.io.Serializable;

import javax.persistence.Entity;

import javax.persistence.GeneratedValue;

import javax.persistence.GenerationType;

import javax.persistence.Id;

import javax.persistence.Table;

@Entity

@Table(name="user")

public class User implements Serializable {

private static final long serialVersionUID = 1L;

@Id

@GeneratedValue(strategy=GenerationType.AUTO)

private int userId;

private String userName;

private String userEmail;

private int userAge;

private String userGender;

private String userAddress;

public User() {

super();

}

public User(String userName, String userEmail, int userAge, String userGender, String userAddress) {

super();

this.userName = userName;

this.userEmail = userEmail;

this.userAge = userAge;

this.userGender = userGender;

this.userAddress = userAddress;

}

public int getUserId() {

return userId;

}

public void setUserId(int userId) {

this.userId = userId;

}

public String getUserName() {

return userName;

}

public void setUserName(String userName) {

this.userName = userName;

}

public String getUserEmail() {

return userEmail;

}

public void setUserEmail(String userEmail) {

this.userEmail = userEmail;

}

public int getUserAge() {

return userAge;

}

public void setUserAge(int userAge) {

this.userAge = userAge;

}

public String getUserGender() {

return userGender;

}

public void setUserGender(String userGender) {

this.userGender = userGender;

}

public String getUserAddress() {

return userAddress;

}

public void setUserAddress(String userAddress) {

this.userAddress = userAddress;

}

public static long getSerialversionuid() {

return serialVersionUID;

}

@Override

public String toString() {

return String.format("User [userId=%s, userName=%s, userEmail=%s, userAge=%s, userGender=%s, userAddress=%s]",

userId, userName, userEmail, userAge, userGender, userAddress);

}

}

//repository

package org.sporty\_shoes.repository;

import java.util.List;

import org.hibernate.Query;

import org.hibernate.Session;

import org.hibernate.Transaction;

import org.sporty\_shoes.model.Admin;

import org.springframework.orm.hibernate5.HibernateTemplate;

@SuppressWarnings("deprecation")

public class AdminRepository {

private HibernateTemplate hibernateTemplate;

public HibernateTemplate getHibernateTemplate() {

return hibernateTemplate;

}

public void setHibernateTemplate(HibernateTemplate hibernateTemplate) {

this.hibernateTemplate = hibernateTemplate;

}

public boolean verifyAdmin(Admin a) {

Session session=hibernateTemplate.getSessionFactory().openSession();

Transaction transaction=session.beginTransaction();

String hql = "from Admin where email=:email and password=:password";

Query<Admin> query = session.createQuery(hql,Admin.class);

query.setParameter("email", a.getEmail());

query.setParameter("password", a.getPassword());

List<Admin> results = query.getResultList();

transaction.commit();

session.close();

return results.size()>0;

}

}

package org.sporty\_shoes.repository;

import java.util.List;

import org.hibernate.Session;

import org.hibernate.Transaction;

import org.sporty\_shoes.model.Product;

import org.springframework.orm.hibernate5.HibernateTemplate;

public class ProductRepository {

HibernateTemplate hibernateTemplate;

public ProductRepository() {

super();

}

public ProductRepository(HibernateTemplate hibernateTemplate) {

super();

this.hibernateTemplate = hibernateTemplate;

}

public HibernateTemplate getHibernateTemplate() {

return hibernateTemplate;

}

public void setHibernateTemplate(HibernateTemplate hibernateTemplate) {

this.hibernateTemplate = hibernateTemplate;

}

public List<Product> getAllProducts(){

List<Product> productList=hibernateTemplate.loadAll(Product.class);

return productList;

}

public Boolean deleteProduct(int id) {

try {

Session session=hibernateTemplate.getSessionFactory().openSession();

Transaction transaction=session.beginTransaction();

Product p=session.get(Product.class, id);

session.delete(p);

transaction.commit();

session.close();

}

catch(Exception e) {

return false;

}

return true;

}

}

package org.sporty\_shoes.repository;

import java.util.List;

import org.sporty\_shoes.model.PurchaseReport;

import org.springframework.orm.hibernate5.HibernateTemplate;

public class PurchaseReportRepository {

HibernateTemplate hibernateTemplate;

public HibernateTemplate getHibernateTemplate() {

return hibernateTemplate;

}

public void setHibernateTemplate(HibernateTemplate hibernateTemplate) {

this.hibernateTemplate = hibernateTemplate;

}

@SuppressWarnings("deprecation")

public List<PurchaseReport> getReport(){

@SuppressWarnings("unchecked")

List<PurchaseReport> report=(List<PurchaseReport>) hibernateTemplate.find("from PurchaseReport order by reportProductName,reportDate");

return report;

}

}

package org.sporty\_shoes.repository;

import org.hibernate.Query;

import org.hibernate.Session;

import org.hibernate.Transaction;

import org.sporty\_shoes.model.Admin;

import org.springframework.orm.hibernate5.HibernateTemplate;

@SuppressWarnings("deprecation")

public class Repository {

private HibernateTemplate hibernateTemplate;

public HibernateTemplate getHibernateTemplate() {

return hibernateTemplate;

}

public void setHibernateTemplate(HibernateTemplate hibernateTemplate) {

this.hibernateTemplate = hibernateTemplate;

}

public boolean forgotPassword(String email,String password) {

Session session=hibernateTemplate.getSessionFactory().openSession();

Transaction tx=session.beginTransaction();

@SuppressWarnings("unchecked")

Query<Admin> q=session.createQuery("update Admin set password=:p where email=:e");

q.setParameter("e",email);

q.setParameter("p",password);

int status=q.executeUpdate();

System.out.println(status);

tx.commit();

return status>0?true:false;

}

}

package org.sporty\_shoes.repository;

import java.util.List;

import org.hibernate.Query;

import org.hibernate.Session;

import org.hibernate.Transaction;

import org.sporty\_shoes.model.User;

import org.springframework.orm.hibernate5.HibernateTemplate;

@SuppressWarnings("deprecation")

public class UserRepository {

HibernateTemplate hibernateTemplate;

public UserRepository() {

super();

}

public UserRepository(HibernateTemplate hibernateTemplate) {

super();

this.hibernateTemplate = hibernateTemplate;

}

public HibernateTemplate getHibernateTemplate() {

return hibernateTemplate;

}

public void setHibernateTemplate(HibernateTemplate hibernateTemplate) {

this.hibernateTemplate = hibernateTemplate;

}

public List<User> getAllUsers(){

List<User> userList=hibernateTemplate.loadAll(User.class);

return userList;

}

public List<User> searchUser(String email){

Session session=hibernateTemplate.getSessionFactory().openSession();

Transaction transaction=session.beginTransaction();

String hql = "from User where userEmail=:email";

Query<User> query = session.createQuery(hql,User.class);

query.setParameter("email", email);

List<User> user = query.getResultList();

transaction.commit();

session.close();

return user;

}

}

//Home page

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

pageEncoding="ISO-8859-1"%>

<!DOCTYPE html>

<html>

<head>

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

<title>Sporty Shoes</title>

</head>

<body style="background-color:rgb(255,252,211)">

<h1 align="center">Sporty Shoes</h1><br><br>

<p align="center"><a href="adminLogin" style="font-size:25px;">Admin Login</a></p>

</body>

</html>

//dispatcher servlet

<?xml version="1.0" encoding="UTF-8"?>

<beans xmlns="http://www.springframework.org/schema/beans"

xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xmlns:mvc="http://www.springframework.org/schema/mvc"

xmlns:tx="http://www.springframework.org/schema/tx"

xmlns:context="http://www.springframework.org/schema/context"

xsi:schemaLocation="

http://www.springframework.org/schema/beans http://www.springframework.org/schema/beans/spring-beans.xsd

http://www.springframework.org/schema/context http://www.springframework.org/schema/context/spring-context.xsd

http://www.springframework.org/schema/mvc http://www.springframework.org/schema/mvc/spring-mvc.xsd

http://www.springframework.org/schema/tx http://www.springframework.org/schema/tx/spring-tx.xsd">

<context:component-scan base-package="org.sporty\_shoes.controller"/>

<mvc:annotation-driven/>

<bean class = "org.springframework.web.servlet.view.InternalResourceViewResolver">

<property name = "prefix" value = "/WEB-INF/view/" />

<property name = "suffix" value = ".jsp" />

</bean>

<bean id="dataSource" class="org.springframework.jdbc.datasource.DriverManagerDataSource">

<property name="driverClassName" value="com.mysql.cj.jdbc.Driver"></property>

<property name="url" value="jdbc:mysql://localhost:3306/sporty\_shoes"></property>

<property name="username" value="root"></property>

<property name="password" value="Vipul@4567"></property>

</bean>

<bean id="sessionFactory" class="org.springframework.orm.hibernate5.LocalSessionFactoryBean">

<property name="dataSource" ref="dataSource"></property>

<property name="packagesToScan" value="org.sporty\_shoes.model"></property>

<property name="hibernateProperties">

<props>

<prop key="hibernate.dialect">org.hibernate.dialect.MySQL8Dialect</prop>

<prop key="hibernate.hbm2ddl.auto">update</prop>

<prop key="hibernate.show\_sql">true</prop>

</props>

</property>

</bean>

<bean id="myTransactionManager" class="org.springframework.orm.hibernate5.HibernateTransactionManager">

<property name="sessionFactory" ref="sessionFactory"/>

</bean>

<tx:annotation-driven transaction-manager="myTransactionManager"/>

<bean id="hibernateTemplate" class="org.springframework.orm.hibernate5.HibernateTemplate">

<property name="sessionFactory" ref="sessionFactory"></property>

</bean>

<bean id="adminRepository" class="org.sporty\_shoes.repository.AdminRepository">

<property name="hibernateTemplate" ref="hibernateTemplate"></property>

</bean>

<bean id="productRepository" class="org.sporty\_shoes.repository.ProductRepository">

<property name="hibernateTemplate" ref="hibernateTemplate"></property>

</bean>

<bean id="userRepository" class="org.sporty\_shoes.repository.UserRepository">

<property name="hibernateTemplate" ref="hibernateTemplate"></property>

</bean>

<bean id="purchaseReportRepository" class="org.sporty\_shoes.repository.PurchaseReportRepository">

<property name="hibernateTemplate" ref="hibernateTemplate"></property>

</bean>

<bean id="repository" class="org.sporty\_shoes.repository.Repository">

<property name="hibernateTemplate" ref="hibernateTemplate"></property>

</bean>

</beans>

//web.xml

<?xml version="1.0" encoding="UTF-8"?>

<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xmlns="http://JAVA.sun.com/xml/ns/javaee" xsi:schemaLocation="http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_2\_5.xsd" version="2.5">

<display-name>sporty\_shoes</display-name>

<welcome-file-list>

<welcome-file>index.html</welcome-file>

<welcome-file>index.htm</welcome-file>

<welcome-file>index.jsp</welcome-file>

<welcome-file>default.html</welcome-file>

<welcome-file>default.htm</welcome-file>

<welcome-file>Home.jsp</welcome-file>

</welcome-file-list>

<servlet>

<servlet-name>DispatcherServlet</servlet-name>

<servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>

<load-on-startup>1</load-on-startup>

</servlet>

<servlet-mapping>

<servlet-name>DispatcherServlet</servlet-name>

<url-pattern>/</url-pattern>

</servlet-mapping>

</web-app>

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

pageEncoding="ISO-8859-1"%>

<!DOCTYPE html>

<html>

<head>

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

<title>Sporty Shoes</title>

</head>

<body style="background-color:rgb(255,252,222)">

<br>

<a href=Home.jsp style="color:black;text-decoration:none ;font-size:35px;font-weight:bold;">Sporty Shoes</a>

<br><br>

<h2>Admin Login</h2>

<div style="border:3px solid pink;width:25%;border-radius:20px;padding:20px">

<form action=adminPage method=post>

<label for=email>Email :-</label> <input type="email" name=email id=email /><br><br>

<label for=pass>Password :-</label> <input type="password" name=password id=pass /><br><br>

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

</form>

</div><br>

<a href=forgotPassword style="font-size:25;color:red;">Forgot Password</a>

<p style="color:green;">${message}</p>

</body>

</html>

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

pageEncoding="ISO-8859-1"%>

<!DOCTYPE html>

<html>

<head>

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

<title>Sporty Shoes</title>

<style>

div {

border:10px solid black;

color:blue;

font-size:20px;

height:100px;

width:450px;

}

a{

padding-left:50px;

}

</style>

</head>

<body style="background-color:rgb(255,252,211)">

<a href=Home.jsp style="color:black;text-decoration:none ;font-size:35px;font-weight:bold;">Sporty Shoes</a>

<br><br>

<div>

<br><br>

<a href="product" target="iframe">Products</a>

<a href="user" target="iframe">Users</a>

<a href="purchaseReport" target="iframe">Purchase Report</a>

</div><br><br>

<iframe src="product" name="iframe" style="height:400px;width:900px;"></iframe>

</body>

</html>

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

pageEncoding="ISO-8859-1"%>

<!DOCTYPE html>

<html>

<head>

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

<title>Sporty Shoes</title>

</head>

<body style="background-color:rgb(255,252,211)">

<br>

<a href=Home.jsp style="color:black;text-decoration:none ;font-size:35px;font-weight:bold;">Sporty Shoes</a>

<br><br>

<h2 style="color:blue;">Enter details to reset password</h2>

<div style="border:3px solid pink;width:25%;border-radius:20px;padding:20px">

<form action=adminForgotPassword method=post>

<label for=email>Email :-</label> <input type="email" name=email id=email /><br><br>

<label for=pass>New Password :-</label> <input type="password" name=password id=pass /><br><br>

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

</form>

</div><br><br>

<p style="color:green;">${message}</p>

</body>

</html>

<%@ 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">

<title>Sporty Shoes</title>

<style>

table,th,td {

border:1px solid black;

text-align:center;

}

</style>

</head>

<body style="background-color:yellow;">

<h2>All Available Products</h2>

<br>

<table style="width:100%">

<tr>

<th>Product Name</th>

<th>Product Discription</th>

<th>Product Price</th>

</tr>

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

<tr>

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

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

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

<td><a href="deleteProduct?id=${product.productId}">Delete</a></td>

</tr>

</c:forEach>

</table>

<br><br>

<p style="color:green;">${message}</p>

</body>

</html>

<%@ 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">

<title>Sporty Shoes</title>

<style>

table,th,td {

border:1px solid black;

text-align:center;

}

</style>

</head>

<body style="background-color:yellow;">

<h2>Purchase Report</h2>

<br>

<table style="width:100%">

<tr>

<th>Product name</th>

<th>User Email</th>

<th>Date</th>

<th>Price</th>

</tr>

<c:forEach var="report" items="${report}">

<tr>

<td> ${report.reportProductName} </td>

<td> ${report.reportUserEmail} </td>

<td> ${report.reportDate} </td>

<td> ${report.reportPrice} </td>

</tr>

</c:forEach>

</table>

</body>

</html>

<%@ 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">

<title>Sporty Shoes</title>

<style>

table,th,td {

border:1px solid black;

text-align:center;

}

</style>

</head>

<body style="background-color:yellow;">

<h2>Users Details</h2>

<br>

<form action=searchUser>

<label for="search">Search User By Email :-</label> <input type="email" name="email" id="search"/>

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

</form><br><br>

<table style="width:100%">

<tr>

<th>Name</th>

<th>Email</th>

<th>Age</th>

<th>Gender</th>

<th>Address</th>

</tr>

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

<tr>

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

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

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

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

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

</tr>

</c:forEach>

</table>

</body>

</html>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<groupId>org.teksystems.sportyshoes</groupId>

<artifactId>sporty\_shoes</artifactId>

<version>1.0-SNAPSHOT</version>

<packaging>war</packaging>

<name>sporty\_shoes</name>

<description>Sporty Shoes is a company that manufactures and sells sports shoes.</description>

<properties>

<maven.compiler.source>1.8</maven.compiler.source>

<maven.compiler.target>1.8</maven.compiler.target>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>5.3.16</version>

</dependency>

<dependency>

<groupId>commons-logging</groupId>

<artifactId>commons-logging</artifactId>

<version>1.2</version>

</dependency>

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>javax.servlet-api</artifactId>

<version>4.0.1</version>

<scope>provided</scope>

</dependency>

<dependency>

<groupId>org.hibernate</groupId>

<artifactId>hibernate-core</artifactId>

<version>5.4.27.Final</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-orm</artifactId>

<version>5.3.16</version>

</dependency>

<dependency>

<groupId>mysql</groupId>

<artifactId>mysql-connector-java</artifactId>

<version>8.0.28</version>

</dependency>

<dependency>

<groupId>org.glassfish.web</groupId>

<artifactId>javax.servlet.jsp.jstl</artifactId>

<version>1.2.1</version>

</dependency>

<dependency>

<groupId>javax.servlet.jsp.jstl</groupId>

<artifactId>javax.servlet.jsp.jstl-api</artifactId>

<version>1.2.1</version>

</dependency>

</dependencies>

</project>