package com.mjava.service;

import java.sql.Date;

import java.util.List;

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

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

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

import org.springframework.stereotype.Service;

import com.mjava.dao.OrdersDao;

import com.mjava.dao.ShoesDao;

import com.mjava.dao.UsersDao;

import com.mjava.model.OrderedShoeModel;

import com.mjava.model.ShoesDataModel;

import com.mjava.model.UserInfoModel;

@Service

public class ShoesService {

@Autowired

private ShoesDao shoesDao;

@Autowired

private OrdersDao orderDao;

@Autowired

private UsersDao userDao;

public List<ShoesDataModel> getMensShoeData() {

List<ShoesDataModel> mensData= (List)shoesDao.findByCategory(1);

System.out.println(mensData);

return mensData;

}

public List<ShoesDataModel> getWomensShoeData() {

List<ShoesDataModel> womensData= (List)shoesDao.findByCategory(2);

System.out.println(womensData);

return womensData;

}

public List<ShoesDataModel> getKidsShoeData() {

List<ShoesDataModel> kidsData= (List)shoesDao.findByCategory(3);

return kidsData;

}

public List<ShoesDataModel> getShoeDatabyCategeory(int categeoryid) {

List<ShoesDataModel> shoesDatabyCategeory= (List)shoesDao.findByCategory(categeoryid);

return shoesDatabyCategeory;

}

public ShoesDataModel getshoesDataByIdService(int seletedShoeId)

{

ShoesDataModel selectedShoesData= shoesDao.getshoesDataById(seletedShoeId);

System.out.println(selectedShoesData);

return selectedShoesData;

}

public boolean insertNewProductService(int categeory,double price,String name,String imagelink)

{

boolean isInsertedNewProduct = false;

try {

ShoesDataModel s=new ShoesDataModel(categeory,price, name, imagelink);

ShoesDataModel count= shoesDao.save(s);

isInsertedNewProduct = true;

return isInsertedNewProduct;

} catch (Exception e) {

System.out.println("Exception at insertNewProductService(int categeory,double price,String name,String imagelink) " + e.getMessage());

return false;

}

}

public boolean deleteProductwithIdService(int id) {

int count=0;

boolean isdeleted=false;

count= shoesDao.deleteProductwithId(id);

if(count>0)

{

isdeleted=true;

}

return isdeleted;

}

public boolean updateShoeProductService( int categeory, double price,

String name, String imagelink,

int id) {

int count=0;

boolean isUpdated=false;

count= shoesDao.updateShoeProduct(categeory, price, name, imagelink, id);

if(count>0)

{

isUpdated=true;

}

return isUpdated;

}

}