**BookShop\src\java\com\sharanamvaishali\service**

**AuthorService.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.model.Author;

import com.sharanamvaishali.model.Book;

import java.util.List;

public interface AuthorService {

public void saveAuthor(Author author);

public List<Author> listAuthor();

public void removeAuthor(Integer authorNo);

public Author getAuthorById(Integer authorNo);

public List<Author> listOurAuthors();

public List<Book> getAllBooksByAuthor(Integer authorNo);

}

**AuthorServiceImpl.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.dao.AuthorDAO;

import com.sharanamvaishali.model.Author;

import com.sharanamvaishali.model.Book;

import java.util.List;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

@Service

public class AuthorServiceImpl implements AuthorService {

@Autowired

private AuthorDAO authorDAO;

@Transactional

@Override

public void saveAuthor(Author author) {

authorDAO.saveAuthor(author);

}

@Transactional

@Override

public List<Author> listAuthor() {

return authorDAO.listAuthor();

}

@Transactional

@Override

public void removeAuthor(Integer authorNo) {

authorDAO.removeAuthor(authorNo);

}

@Transactional

@Override

public Author getAuthorById(Integer authorNo) {

return authorDAO.getAuthorById(authorNo);

}

@Transactional

@Override

public List<Author> listOurAuthors() {

return authorDAO.listOurAuthors();

}

@Transactional

@Override

public List<Book> getAllBooksByAuthor(Integer authorNo) {

return authorDAO.getAllBooksByAuthor(authorNo);

}

}

**BookService.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.model.Book;

import java.util.List;

public interface BookService {

public void saveBook(Book book);

public List<Book> listBook();

public void removeBook(Integer bookNo);

public Book getBookById(Integer bookNo);

public List<Book> listNewReleases();

public List<Book> listUpdatedBooks();

public List<Book> listTopTitles();

public void updateHits(Integer bookNo);

public void notifyCustomersByMail(Book book);

}

**BookServiceImpl.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.dao.BookDAO;

import com.sharanamvaishali.model.Book;

import com.sharanamvaishali.model.User;

import java.util.List;

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

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

@Service

public class BookServiceImpl implements BookService {

@Value("${emailFrom}")

String emailFrom;

@Autowired

private UserService userService;

@Autowired

private MailService mailService;

@Autowired

private BookDAO bookDAO;

@Transactional

@Override

public void saveBook(Book book) {

bookDAO.saveBook(book);

}

@Transactional

@Override

public List<Book> listBook() {

return bookDAO.listBook();

}

@Transactional

@Override

public void removeBook(Integer bookNo) {

bookDAO.removeBook(bookNo);

}

@Transactional

@Override

public Book getBookById(Integer bookNo) {

return bookDAO.getBookById(bookNo);

}

@Transactional

@Override

public List<Book> listNewReleases() {

return bookDAO.listNewReleases();

}

@Transactional

@Override

public List<Book> listUpdatedBooks() {

return bookDAO.listUpdatedBooks();

}

@Transactional

@Override

public List<Book> listTopTitles() {

return bookDAO.listTopTitles();

}

@Transactional

@Override

public void updateHits(Integer bookNo) {

bookDAO.updateHits(bookNo);

}

@Override

public void notifyCustomersByMail(Book book) {

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

for (User user : users) {

String emailMessage = "<table width='100%' border='0' align='center' cellpadding='0' cellspacing='0' "

+ "style='font-family: Verdana, Arial, Helvetica, sans-serif; "

+ "font-size: 12pt; color:#5a5a5a;'><tr><td align='left'>"

+ "<p>Dear " + user.getFirstName() + ",</p></td></tr><tr>"

+ "<td align='left'>"

+ "<p>ISBN: " + book.getIsbn() + "<br /><br />"

+ "Edition: " + book.getEdition() + "<br /><br />"

+ "Synopsis: " + book.getSynopsis() + "<br /><br />"

+ "Topics covered: " + book.getTopicsCovered() + "<br /><br /></p>"

+ "<p>Thank you for using this site.<br /></p><br/><br/><p>"

+ "Regards,<br />Sharanam & Vaishali's Online Bookshop<br /></p><p><br />"

+ "<br />THIS IS AN AUTOMATED MESSAGE; PLEASE DO NOT REPLY. </p></td></tr></table>";

if(user.getAuthority().equals("CUSTOMER") && user.isNewRelease()) {

if(book.getBookNo() == null) {

String emailSubject = "Sharanam & Vaishali's Online Bookshop: " + book.getBookName() + " has been added.";

mailService.sendMail(emailFrom, user.getEmailAddress(), emailSubject, emailMessage);

}

}

if(user.getAuthority().equals("CUSTOMER") && user.isBookUpdates()) {

if(book.getBookNo()!=null) {

String emailSubject = "Sharanam & Vaishali's Online Bookshop: " + book.getBookName() + " has been updated.";

mailService.sendMail(emailFrom, user.getEmailAddress(), emailSubject, emailMessage);

}

}

}

}

}

**CartService.txt**

package com.sharanamvaishali.service;

import javax.servlet.http.HttpSession;

public interface CartService {

public void addBookToCart(Integer bookNo, String userName, HttpSession session);

public void saveCart(String userName, HttpSession session);

public void removeBookFromCart(Integer bookNo, String userName, HttpSession session);

}

**CartServiceImpl.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.dao.TransactionDAO;

import com.sharanamvaishali.model.Book;

import com.sharanamvaishali.model.CartItem;

import com.sharanamvaishali.model.Transaction;

import java.util.ArrayList;

import java.util.List;

import javax.servlet.http.HttpSession;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

@Service

public class CartServiceImpl implements CartService {

@Autowired

private BookService bookService;

@Autowired

private TransactionService transactionService;

@Autowired

private TransactionDAO transactionDAO;

@Transactional

@Override

public void addBookToCart(Integer bookNo, String userName, HttpSession session) {

List<CartItem> cartList;

if((List<CartItem>) session.getAttribute(userName)!=null) {

cartList = (List<CartItem>) session.getAttribute(userName);

}

else {

cartList = new ArrayList();

}

Book book = bookService.getBookById(bookNo);

boolean itemExists = false;

for (CartItem item : cartList) {

if (item.getBookNo().equals(bookNo)) {

item.setQty(item.getQty()+1);

item.setCost(item.getCost() \* item.getQty());

itemExists = true;

}

}

if (!itemExists) {

CartItem cartItem = new CartItem(userName, book.getBookNo(), book.getBookName(), book.getSynopsis(), book.getCost(), 1);

cartList.add(cartItem);

}

session.setAttribute(userName, cartList);

}

@Transactional

@Override

public void saveCart(String userName, HttpSession session) {

List<CartItem> cartList = (List<CartItem>) session.getAttribute(userName);

Integer nextTransactionNo = transactionDAO.getNextTransactionNo();

for (CartItem cartItem : cartList) {

if (cartItem.getUserName().equals(userName)) {

Transaction transaction = new Transaction();

transaction.setTransactionNo(nextTransactionNo);

transaction.setBookName(cartItem.getBookName());

transaction.setCost(cartItem.getCost());

transaction.setQty(cartItem.getQty());

transaction.setUserName(userName);

transactionService.saveTransaction(transaction);

}

}

}

@Override

public void removeBookFromCart(Integer bookNo, String userName, HttpSession session) {

List<CartItem> cartList;

if((List<CartItem>) session.getAttribute(userName)!=null) {

cartList = (List<CartItem>) session.getAttribute(userName);

}

else {

cartList = new ArrayList();

}

for (int i=0; i<cartList.size(); i++) {

if (cartList.get(i).getBookNo().equals(bookNo)) {

cartList.remove(i);

}

}

session.setAttribute(userName, cartList);

}

}

**CategoryService.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.model.Book;

import com.sharanamvaishali.model.Category;

import java.util.List;

public interface CategoryService {

public void saveCategory(Category category);

public List<Category> listCategory();

public void removeCategory(Integer categoryNo);

public Category getCategoryById(Integer categoryNo);

public List<Book> getAllBooksByCategory(Integer categoryNo);

}

**CategoryServiceImpl.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.dao.CategoryDAO;

import com.sharanamvaishali.model.Book;

import com.sharanamvaishali.model.Category;

import java.util.List;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

@Service

public class CategoryServiceImpl implements CategoryService {

@Autowired

private CategoryDAO categoryDAO;

@Transactional

@Override

public void saveCategory(Category category) {

categoryDAO.saveCategory(category);

}

@Transactional

@Override

public List<Category> listCategory() {

return categoryDAO.listCategory();

}

@Transactional

@Override

public void removeCategory(Integer categoryNo) {

categoryDAO.removeCategory(categoryNo);

}

@Transactional

@Override

public Category getCategoryById(Integer categoryNo) {

return categoryDAO.getCategoryById(categoryNo);

}

@Transactional

@Override

public List<Book> getAllBooksByCategory(Integer categoryNo) {

return categoryDAO.getAllBooksByCategory(categoryNo);

}

}

**CommonService.txt**

package com.sharanamvaishali.service;

import java.sql.Blob;

public interface CommonService {

public Blob getBlob(byte[] is);

}

**CommonServiceImpl.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.dao.CommonDAO;

import java.sql.Blob;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

@Service

public class CommonServiceImpl implements CommonService {

@Autowired

private CommonDAO commonDAO;

@Transactional

@Override

public Blob getBlob(byte[] is) {

return commonDAO.getBlob(is);

}

}

**CountryService.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.model.Country;

import java.util.List;

public interface CountryService {

public void saveCountry(Country country);

public List<Country> listCountry();

public void removeCountry(Integer countryNo);

public Country getCountryById(Integer countryNo);

}

**CountryServiceImpl.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.dao.CountryDAO;

import com.sharanamvaishali.model.Country;

import java.util.List;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

@Service

public class CountryServiceImpl implements CountryService {

@Autowired

private CountryDAO countryDAO;

@Transactional

@Override

public void saveCountry(Country country) {

countryDAO.saveCountry(country);

}

@Transactional

@Override

public List<Country> listCountry() {

return countryDAO.listCountry();

}

@Transactional

@Override

public void removeCountry(Integer countryNo) {

countryDAO.removeCountry(countryNo);

}

@Transactional

@Override

public Country getCountryById(Integer countryNo) {

return countryDAO.getCountryById(countryNo);

}

}

**MailService.txt**

package com.sharanamvaishali.service;

import java.net.MalformedURLException;

import java.util.logging.Level;

import java.util.logging.Logger;

import javax.mail.MessagingException;

import javax.mail.internet.MimeMessage;

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

import org.springframework.core.io.UrlResource;

import org.springframework.mail.MailParseException;

import org.springframework.mail.javamail.JavaMailSender;

import org.springframework.mail.javamail.MimeMessageHelper;

public class MailService {

@Value("${sendMail}")

String sendMail;

private JavaMailSender mailSender;

public void setMailSender(JavaMailSender mailSender) {

this.mailSender = mailSender;

}

public void sendMail(String from, String to, String subject, String msg) {

MimeMessage message = mailSender.createMimeMessage();

try{

MimeMessageHelper helper = new MimeMessageHelper(message, true);

helper.setFrom(from);

helper.setTo(to);

helper.setSubject(subject);

helper.setText(msg, true);

UrlResource file;

try {

file = new UrlResource("http://localhost:8080/BookShop/images/logo.png");

helper.addAttachment(file.getFilename(), file);

} catch (MalformedURLException ex) {

Logger.getLogger(MailService.class.getName()).log(Level.SEVERE, null, ex);

}

}catch (MessagingException e) {

throw new MailParseException(e);

}

if(sendMail.equals("true")) {

mailSender.send(message);

}

}

}

**PublisherService.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.model.Book;

import com.sharanamvaishali.model.Publisher;

import java.util.List;

public interface PublisherService {

public void savePublisher(Publisher publisher);

public List<Publisher> listPublisher();

public void removePublisher(Integer publisherNo);

public Publisher getPublisherById(Integer publisherNo);

public List<Publisher> listOurPublishers();

public List<Book> getAllBooksByPublisher(Integer publisherNo);

}

**PublisherServiceImpl.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.dao.PublisherDAO;

import com.sharanamvaishali.model.Book;

import com.sharanamvaishali.model.Publisher;

import java.util.List;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

@Service

public class PublisherServiceImpl implements PublisherService {

@Autowired

private PublisherDAO publisherDAO;

@Transactional

@Override

public void savePublisher(Publisher publisher) {

publisherDAO.savePublisher(publisher);

}

@Transactional

@Override

public List<Publisher> listPublisher() {

return publisherDAO.listPublisher();

}

@Transactional

@Override

public void removePublisher(Integer publisherNo) {

publisherDAO.removePublisher(publisherNo);

}

@Transactional

@Override

public Publisher getPublisherById(Integer publisherNo) {

return publisherDAO.getPublisherById(publisherNo);

}

@Transactional

@Override

public List<Publisher> listOurPublishers() {

return publisherDAO.listOurPublishers();

}

@Transactional

@Override

public List<Book> getAllBooksByPublisher(Integer publisherNo) {

return publisherDAO.getAllBooksByPublisher(publisherNo);

}

}

**SearchService.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.model.PopularSearch;

import java.util.List;

public interface SearchService {

public List searchResults(String searchCriteria);

public List searchAllResults();

public List listPopularSearches();

public void savePopularSearch(PopularSearch popularSeacrh);

}

**SearchServiceImpl.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.dao.SearchDAO;

import com.sharanamvaishali.model.PopularSearch;

import java.util.List;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

@Service

public class SearchServiceImpl implements SearchService {

@Autowired

private SearchDAO searchDAO;

@Transactional

@Override

public List searchResults(String searchCriteria) {

return searchDAO.searchResults(searchCriteria);

}

@Transactional

@Override

public List searchAllResults() {

return searchDAO.searchAllResults();

}

@Transactional

@Override

public List listPopularSearches() {

return searchDAO.listPopularSearches();

}

@Transactional

@Override

public void savePopularSearch(PopularSearch popularSearch) {

searchDAO.savePopularSearch(popularSearch);

searchDAO.deletePopularSearches();

}

}

**StateService.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.model.State;

import java.util.List;

public interface StateService {

public void saveState(State state);

public List<State> listState();

public void removeState(Integer stateNo);

public State getStateById(Integer stateNo);

}

**StateServiceImpl.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.model.State;

import java.util.List;

public interface StateService {

public void saveState(State state);

public List<State> listState();

public void removeState(Integer stateNo);

public State getStateById(Integer stateNo);

}

**TransactionService.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.model.Transaction;

import java.util.Date;

import java.util.List;

public interface TransactionService {

public void saveTransaction(Transaction transaction);

public List<Transaction> listTransactions();

public List<Transaction> listGroupedTransactions();

public List<Transaction> listTransactionsByUsername(String userName);

public List<Transaction> listGroupedTransactionsByUsername(String userName);

public List<Transaction> listTransactionsByDate(Date date);

public List<Transaction> listGroupedTransactionsByDate(Date date);

public void updateGoogleOrderNo(Integer transactionNo, String orderNo);

}

**TransactionServiceImpl.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.dao.TransactionDAO;

import com.sharanamvaishali.model.Transaction;

import java.util.Date;

import java.util.List;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

@Service

public class TransactionServiceImpl implements TransactionService {

@Autowired

private TransactionDAO transactionDAO;

@Transactional

@Override

public void saveTransaction(Transaction transaction) {

transactionDAO.saveTransaction(transaction);

}

@Transactional

@Override

public List<Transaction> listTransactions() {

return transactionDAO.listTransactions();

}

@Transactional

@Override

public List<Transaction> listGroupedTransactions() {

return transactionDAO.listGroupedTransactions();

}

@Transactional

@Override

public List<Transaction> listTransactionsByUsername(String userName) {

return transactionDAO.listTransactionsByUsername(userName);

}

@Transactional

@Override

public List<Transaction> listGroupedTransactionsByUsername(String userName) {

return transactionDAO.listGroupedTransactionsByUsername(userName);

}

@Transactional

@Override

public List<Transaction> listTransactionsByDate(Date date) {

return transactionDAO.listTransactionsByDate(date);

}

@Transactional

@Override

public List<Transaction> listGroupedTransactionsByDate(Date date) {

return transactionDAO.listGroupedTransactionsByDate(date);

}

@Transactional

@Override

public void updateGoogleOrderNo(Integer transactionNo, String orderNo) {

transactionDAO.updateGoogleOrderNo(transactionNo, orderNo);

}

}

**UserService.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.model.User;

import java.util.List;

public interface UserService {

public void saveUser(User User);

public List<User> listUser();

public void removeUser(Integer userNo);

public User getUserById(Integer userNo);

public User getUserByUsername(String userName);

}

**UserServiceImpl.txt**

package com.sharanamvaishali.service;

import com.sharanamvaishali.dao.UserDAO;

import com.sharanamvaishali.model.User;

import java.util.List;

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

import org.springframework.stereotype.Service;

import org.springframework.transaction.annotation.Transactional;

@Service

public class UserServiceImpl implements UserService {

@Autowired

private UserDAO userDAO;

@Transactional

@Override

public void saveUser(User User) {

userDAO.saveUser(User);

}

@Transactional

@Override

public List<User> listUser() {

return userDAO.listUser();

}

@Transactional

@Override

public void removeUser(Integer userNo) {

userDAO.removeUser(userNo);

}

@Transactional

@Override

public User getUserById(Integer userNo) {

return userDAO.getUserById(userNo);

}

@Transactional

@Override

public User getUserByUsername(String userName) {

return userDAO.getUserByUserName(userName);

}

}