package total.store.dao.impl;  
  
import total.store.dao.BookRecoveryDao;  
import total.store.model.BookRecovery;  
import total.store.util.DbUtil;  
  
import java.sql.Connection;  
import java.sql.PreparedStatement;  
import java.sql.ResultSet;  
import java.util.ArrayList;  
import java.util.List;  
  
public class BookRecoveryDaoImpl implements BookRecoveryDao {  
 @Override  
 public Boolean bookre\_add\_audito(BookRecovery bookRecovery) {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con=null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("insert into 书籍回收表 values(?,?,?,?,?,?,?,?,?,?,?,?,'待审核',null)");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookRecovery.getBookid());  
 ps.setString(2,bookRecovery.getBookname());  
 ps.setString(3,bookRecovery.getBookautorname());  
 ps.setString(4,bookRecovery.getBookpublishinghouse());  
 ps.setInt(5,bookRecovery.getUser\_information());  
 ps.setString(6,bookRecovery.getImg\_1());  
 ps.setString(7,bookRecovery.getImg\_2());  
 ps.setString(8,bookRecovery.getImg\_3());  
 ps.setString(9,bookRecovery.getImg\_4());  
 ps.setString(10,bookRecovery.getImg\_5());  
 ps.setString(11,bookRecovery.getImg\_6());  
 ps.setString(12,bookRecovery.getImg\_7());  
 int num=ps.executeUpdate();  
 if(num==1){  
 return true;  
 }else {  
 return false;  
 }  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public List<BookRecovery> bookre\_list\_audito() {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con = null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("select \* from 书籍回收表 where 书籍状态='待审核'");  
 ps = con.prepareStatement(sb.toString());  
 rs=ps.executeQuery();  
 List<BookRecovery> bookRecoveryList=new ArrayList<>();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 return bookRecoveryList;  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public Boolean bookre\_mosta\_audito(BookRecovery bookRecovery) {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con=null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("update 书籍回收表 set 书籍状态='审核通过',积分=? where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setDouble(1,bookRecovery.getBookprice());  
 ps.setString(2,bookRecovery.getBookid());  
 int num=ps.executeUpdate();  
 if(num==1){  
 return true;  
 }else {  
 return false;  
 }  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public Boolean bookre\_monsta\_audito(String bookRecoveryid) {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con=null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("update 书籍回收表 set 书籍状态='审核未通过' where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookRecoveryid);  
 int num=ps.executeUpdate();  
 if(num==1){  
 return true;  
 }else {  
 return false;  
 }  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public List<BookRecovery> bookre\_hi\_list\_audito() {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con = null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("select \* from 书籍回收表 where 书籍状态!='待审核'");  
 ps = con.prepareStatement(sb.toString());  
 rs=ps.executeQuery();  
 List<BookRecovery> bookRecoveryList=new ArrayList<>();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecovery.setBookstatus(rs.getString("书籍状态"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 return bookRecoveryList;  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public List<BookRecovery> bookre\_list\_user(int userid) {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con = null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("select \* from 书籍回收表 where 用户信息=? and 书籍状态='审核通过'");  
 ps = con.prepareStatement(sb.toString());  
 ps.setInt(1,userid);  
 rs=ps.executeQuery();  
 List<BookRecovery> bookRecoveryList=new ArrayList<>();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecovery.setBookstatus(rs.getString("书籍状态"));  
 bookRecovery.setBookprice(rs.getDouble("积分"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 StringBuilder sb1 = new StringBuilder();  
 sb1.append("select \* from 书籍回收表 where 用户信息=? and 书籍状态='待审核'");  
 ps = con.prepareStatement(sb1.toString());  
 ps.setInt(1,userid);  
 rs=ps.executeQuery();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecovery.setBookstatus(rs.getString("书籍状态"));  
 bookRecovery.setBookprice(rs.getDouble("积分"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 StringBuilder sb2 = new StringBuilder();  
 sb2.append("select \* from 书籍回收表 where 用户信息=? and 书籍状态='已回收'");  
 ps = con.prepareStatement(sb2.toString());  
 ps.setInt(1,userid);  
 rs=ps.executeQuery();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecovery.setBookstatus(rs.getString("书籍状态"));  
 bookRecovery.setBookprice(rs.getDouble("积分"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 return bookRecoveryList;  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public int user\_infor\_search(String bookRecoveryid) {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con = null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return 0;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("select 用户信息 from 书籍回收表 where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookRecoveryid);  
 rs=ps.executeQuery();  
 int userid = 0;  
 while(rs.next()) {  
 userid = rs.getInt("用户信息");  
 }  
 return userid;  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return 0;  
 }  
  
 @Override  
 public double price\_infor(String bookRecoveryid) {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con = null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return 0;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("select 积分 from 书籍回收表 where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookRecoveryid);  
 rs=ps.executeQuery();  
 double num = 0;  
 while(rs.next()) {  
 num = rs.getDouble("积分");  
 }  
 return num;  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return 0;  
 }  
  
 @Override  
 public Boolean bookre\_monsta\_user(String bookRecoveryid) {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con=null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("update 书籍回收表 set 书籍状态='已回收' where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookRecoveryid);  
 int num=ps.executeUpdate();  
 if(num==1){  
 return true;  
 }else {  
 return false;  
 }  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public List<BookRecovery> bookre\_list\_history\_user(int userid) {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con = null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("select \* from 书籍回收表 where 用户信息=? and 书籍状态='审核未通过'");  
 ps = con.prepareStatement(sb.toString());  
 ps.setInt(1,userid);  
 rs=ps.executeQuery();  
 List<BookRecovery> bookRecoveryList=new ArrayList<>();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecovery.setBookstatus(rs.getString("书籍状态"));  
 bookRecovery.setBookprice(rs.getDouble("积分"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 StringBuilder sb1 = new StringBuilder();  
 sb1.append("select \* from 书籍回收表 where 用户信息=? and 书籍状态='已签收'");  
 ps = con.prepareStatement(sb1.toString());  
 ps.setInt(1,userid);  
 rs=ps.executeQuery();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecovery.setBookstatus(rs.getString("书籍状态"));  
 bookRecovery.setBookprice(rs.getDouble("积分"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 StringBuilder sb2 = new StringBuilder();  
 sb2.append("select \* from 书籍回收表 where 用户信息=? and 书籍状态='已取消回收'");  
 ps = con.prepareStatement(sb2.toString());  
 ps.setInt(1,userid);  
 rs=ps.executeQuery();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecovery.setBookstatus(rs.getString("书籍状态"));  
 bookRecovery.setBookprice(rs.getDouble("积分"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 StringBuilder sb3 = new StringBuilder();  
 sb3.append("select \* from 书籍回收表 where 用户信息=? and 书籍状态='已售卖'");  
 ps = con.prepareStatement(sb3.toString());  
 ps.setInt(1,userid);  
 rs=ps.executeQuery();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecovery.setBookstatus(rs.getString("书籍状态"));  
 bookRecovery.setBookprice(rs.getDouble("积分"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 return bookRecoveryList;  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public Boolean bookre\_densta\_user(String bookRecoveryid) {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con=null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("update 书籍回收表 set 书籍状态='已取消回收' where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookRecoveryid);  
 int num=ps.executeUpdate();  
 if(num==1){  
 return true;  
 }else {  
 return false;  
 }  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public List<BookRecovery> bookre\_list\_admin() {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con = null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("select \* from 书籍回收表 where 书籍状态='已回收'");  
 ps = con.prepareStatement(sb.toString());  
 rs=ps.executeQuery();  
 List<BookRecovery> bookRecoveryList=new ArrayList<>();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 return bookRecoveryList;  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public Boolean bookre\_mosta\_admin(String bookRecoveryid) {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con=null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("update 书籍回收表 set 书籍状态='已签收' where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookRecoveryid);  
 int num=ps.executeUpdate();  
 if(num==1){  
 return true;  
 }else {  
 return false;  
 }  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public List<BookRecovery> bookre\_list\_history\_admin() {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con = null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("select \* from 书籍回收表 where 书籍状态='已签收'");  
 ps = con.prepareStatement(sb.toString());  
 rs=ps.executeQuery();  
 List<BookRecovery> bookRecoveryList=new ArrayList<>();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecovery.setBookprice(rs.getDouble("积分"));  
 bookRecovery.setBookstatus(rs.getString("书籍状态"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 StringBuilder sb1 = new StringBuilder();  
 sb1.append("select \* from 书籍回收表 where 书籍状态='已售卖'");  
 ps = con.prepareStatement(sb1.toString());  
 rs=ps.executeQuery();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecovery.setBookprice(rs.getDouble("积分"));  
 bookRecovery.setBookstatus(rs.getString("书籍状态"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 return bookRecoveryList;  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public BookRecovery bookre\_one(String bookRecoveryid) {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con = null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("select \* from 书籍回收表 where 编号 = ?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookRecoveryid);  
 rs=ps.executeQuery();  
 BookRecovery bookRecovery = new BookRecovery();  
 while(rs.next()) {  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecovery.setBookprice(rs.getDouble("积分"));  
 }  
 return bookRecovery;  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public List<BookRecovery> book\_list\_buy\_history() {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con = null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("select \* from 书籍回收表 where 书籍状态='已签收'");  
 ps = con.prepareStatement(sb.toString());  
 rs=ps.executeQuery();  
 List<BookRecovery> bookRecoveryList=new ArrayList<>();  
 while(rs.next()) {  
 BookRecovery bookRecovery = new BookRecovery();  
 bookRecovery.setBookid(rs.getString("编号"));  
 bookRecovery.setBookname(rs.getString("书籍名"));  
 bookRecovery.setBookautorname(rs.getString("作者"));  
 bookRecovery.setBookpublishinghouse(rs.getString("出版社信息"));  
 bookRecovery.setImg\_1(rs.getString("书籍封面"));  
 bookRecovery.setImg\_2(rs.getString("内页一"));  
 bookRecovery.setImg\_3(rs.getString("内页二"));  
 bookRecovery.setImg\_4(rs.getString("内页三"));  
 bookRecovery.setImg\_5(rs.getString("内页四"));  
 bookRecovery.setImg\_6(rs.getString("内页五"));  
 bookRecovery.setImg\_7(rs.getString("书籍底面"));  
 bookRecovery.setBookprice(rs.getDouble("积分"));  
 bookRecoveryList.add(bookRecovery);  
 }  
 return bookRecoveryList;  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
  
 @Override  
 public Boolean bookre\_mosta\_user(String bookRecoveryid) {  
 DbUtil dbUtil =new DbUtil();  
 ResultSet rs = null;  
 PreparedStatement ps = null;  
 Connection con=null;  
 try {  
 con = dbUtil.getCon();  
 if (con == null) {  
 return null;  
 }  
 StringBuilder sb = new StringBuilder();  
 sb.append("update 书籍回收表 set 书籍状态='已售卖' where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookRecoveryid);  
 int num=ps.executeUpdate();  
 if(num==1){  
 return true;  
 }else {  
 return false;  
 }  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }finally {  
 try {  
 dbUtil.closeResultSet(rs);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closeCon(con);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 try {  
 dbUtil.closePs(ps);  
 } catch (Exception e) {  
 e.printStackTrace();  
 }  
 }  
 }  
 return null;  
 }  
}