package total.store.dao.impl;  
  
import total.store.dao.BookDonationDao;  
import total.store.model.BookDonation;  
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 BookDonationDaoImpl implements BookDonationDao {  
 @Override  
 public Boolean bookdo\_add(BookDonation bookDonation) {  
 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 书籍捐赠表 values(?,?,?,?,?,?,?,?,?,?,?,?,'待审核',null)");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookDonation.getDonationbookid());  
 ps.setString(2,bookDonation.getDonationbookname());  
 ps.setString(3,bookDonation.getDonationbookautorname());  
 ps.setString(4,bookDonation.getDonationbookpublishinghouse());  
 ps.setInt(5,bookDonation.getUser\_information());  
 ps.setString(6,bookDonation.getImg\_1());  
 ps.setString(7,bookDonation.getImg\_2());  
 ps.setString(8,bookDonation.getImg\_3());  
 ps.setString(9,bookDonation.getImg\_4());  
 ps.setString(10,bookDonation.getImg\_5());  
 ps.setString(11,bookDonation.getImg\_6());  
 ps.setString(12,bookDonation.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<BookDonation> bookdore\_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 书籍捐赠表 where 书籍状态='待审核'");  
 ps = con.prepareStatement(sb.toString());  
 rs=ps.executeQuery();  
 List<BookDonation> bookDonationList=new ArrayList<>();  
 while(rs.next()) {  
 BookDonation bookDonation = new BookDonation();  
 bookDonation.setDonationbookid(rs.getString("编号"));  
 bookDonation.setDonationbookname(rs.getString("书籍名"));  
 bookDonation.setDonationbookautorname(rs.getString("作者"));  
 bookDonation.setDonationbookpublishinghouse(rs.getString("出版社信息"));  
 bookDonation.setImg\_1(rs.getString("书籍封面"));  
 bookDonation.setImg\_2(rs.getString("内页一"));  
 bookDonation.setImg\_3(rs.getString("内页二"));  
 bookDonation.setImg\_4(rs.getString("内页三"));  
 bookDonation.setImg\_5(rs.getString("内页四"));  
 bookDonation.setImg\_6(rs.getString("内页五"));  
 bookDonation.setImg\_7(rs.getString("书籍底面"));  
 bookDonationList.add(bookDonation);  
 }  
 return bookDonationList;  
 } 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 bookdo\_mo(String bookDonation, double award) {  
 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 书籍捐赠表 set 书籍状态='审核通过',爱心积分=? where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setDouble(1,award);  
 ps.setString(2,bookDonation);  
 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 bookdo\_mo\_not(String bookDonation) {  
 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 书籍捐赠表 set 书籍状态='审核不通过' where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookDonation);  
 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<BookDonation> bookdore\_list\_his\_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 书籍捐赠表 where 书籍状态!='待审核'");  
 ps = con.prepareStatement(sb.toString());  
 rs=ps.executeQuery();  
 List<BookDonation> bookDonationList=new ArrayList<>();  
 while(rs.next()) {  
 BookDonation bookDonation = new BookDonation();  
 bookDonation.setDonationbookid(rs.getString("编号"));  
 bookDonation.setDonationbookname(rs.getString("书籍名"));  
 bookDonation.setDonationbookautorname(rs.getString("作者"));  
 bookDonation.setDonationbookpublishinghouse(rs.getString("出版社信息"));  
 bookDonation.setImg\_1(rs.getString("书籍封面"));  
 bookDonation.setImg\_2(rs.getString("内页一"));  
 bookDonation.setImg\_3(rs.getString("内页二"));  
 bookDonation.setImg\_4(rs.getString("内页三"));  
 bookDonation.setImg\_5(rs.getString("内页四"));  
 bookDonation.setImg\_6(rs.getString("内页五"));  
 bookDonation.setImg\_7(rs.getString("书籍底面"));  
 bookDonation.setDonationbookstatus(rs.getString("书籍状态"));  
 bookDonationList.add(bookDonation);  
 }  
 return bookDonationList;  
 } 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<BookDonation> bookdore\_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 书籍捐赠表 where 书籍状态='审核通过' and 用户信息=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setInt(1,userid);  
 rs=ps.executeQuery();  
 List<BookDonation> bookDonationList=new ArrayList<>();  
 while(rs.next()) {  
 BookDonation bookDonation = new BookDonation();  
 bookDonation.setDonationbookid(rs.getString("编号"));  
 bookDonation.setDonationbookname(rs.getString("书籍名"));  
 bookDonation.setDonationbookautorname(rs.getString("作者"));  
 bookDonation.setDonationbookpublishinghouse(rs.getString("出版社信息"));  
 bookDonation.setImg\_1(rs.getString("书籍封面"));  
 bookDonation.setImg\_2(rs.getString("内页一"));  
 bookDonation.setImg\_3(rs.getString("内页二"));  
 bookDonation.setImg\_4(rs.getString("内页三"));  
 bookDonation.setImg\_5(rs.getString("内页四"));  
 bookDonation.setImg\_6(rs.getString("内页五"));  
 bookDonation.setImg\_7(rs.getString("书籍底面"));  
 bookDonation.setDonationbookstatus(rs.getString("书籍状态"));  
 bookDonation.setAward(rs.getDouble("爱心积分"));  
 bookDonationList.add(bookDonation);  
 }  
 return bookDonationList;  
 } 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<BookDonation> bookdore\_list\_his\_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 书籍捐赠表 where 书籍状态!='审核通过' and 用户信息=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setInt(1,userid);  
 rs=ps.executeQuery();  
 List<BookDonation> bookDonationList=new ArrayList<>();  
 while(rs.next()) {  
 BookDonation bookDonation = new BookDonation();  
 bookDonation.setDonationbookid(rs.getString("编号"));  
 bookDonation.setDonationbookname(rs.getString("书籍名"));  
 bookDonation.setDonationbookautorname(rs.getString("作者"));  
 bookDonation.setDonationbookpublishinghouse(rs.getString("出版社信息"));  
 bookDonation.setImg\_1(rs.getString("书籍封面"));  
 bookDonation.setImg\_2(rs.getString("内页一"));  
 bookDonation.setImg\_3(rs.getString("内页二"));  
 bookDonation.setImg\_4(rs.getString("内页三"));  
 bookDonation.setImg\_5(rs.getString("内页四"));  
 bookDonation.setImg\_6(rs.getString("内页五"));  
 bookDonation.setImg\_7(rs.getString("书籍底面"));  
 bookDonation.setDonationbookstatus(rs.getString("书籍状态"));  
 bookDonationList.add(bookDonation);  
 }  
 return bookDonationList;  
 } 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(String bookDonationid) {  
 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 书籍捐赠表 where 编号 = ?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookDonationid);  
 rs=ps.executeQuery();  
 int num = 0;  
 while(rs.next()) {  
 num=rs.getInt("用户信息");  
 }  
 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 double donationaware(String bookDonationid) {  
 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 爱心积分 from 书籍捐赠表 where 编号 = ?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookDonationid);  
 rs=ps.executeQuery();  
 double num = 0;  
 while(rs.next()) {  
 num=rs.getDouble("爱心积分");  
 }  
 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 bookdo\_modi(String bookDonationid) {  
 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 书籍捐赠表 set 书籍状态='已捐献' where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookDonationid);  
 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 bookdo\_delete(String bookDonationid) {  
 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("delete from 书籍捐赠表 where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookDonationid);  
 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<BookDonation> bookdore\_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 书籍捐赠表 where 书籍状态='已捐献'");  
 ps = con.prepareStatement(sb.toString());  
 rs=ps.executeQuery();  
 List<BookDonation> bookDonationList=new ArrayList<>();  
 while(rs.next()) {  
 BookDonation bookDonation = new BookDonation();  
 bookDonation.setDonationbookid(rs.getString("编号"));  
 bookDonation.setDonationbookname(rs.getString("书籍名"));  
 bookDonation.setDonationbookautorname(rs.getString("作者"));  
 bookDonation.setDonationbookpublishinghouse(rs.getString("出版社信息"));  
 bookDonation.setImg\_1(rs.getString("书籍封面"));  
 bookDonation.setImg\_2(rs.getString("内页一"));  
 bookDonation.setImg\_3(rs.getString("内页二"));  
 bookDonation.setImg\_4(rs.getString("内页三"));  
 bookDonation.setImg\_5(rs.getString("内页四"));  
 bookDonation.setImg\_6(rs.getString("内页五"));  
 bookDonation.setImg\_7(rs.getString("书籍底面"));  
 bookDonationList.add(bookDonation);  
 }  
 return bookDonationList;  
 } 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 bookdo\_modi\_admin(String bookDonationid) {  
 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 书籍捐赠表 set 书籍状态='已签收' where 编号=?");  
 ps = con.prepareStatement(sb.toString());  
 ps.setString(1,bookDonationid);  
 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<BookDonation> bookdore\_hi\_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 书籍捐赠表 where 书籍状态='已签收'");  
 ps = con.prepareStatement(sb.toString());  
 rs=ps.executeQuery();  
 List<BookDonation> bookDonationList=new ArrayList<>();  
 while(rs.next()) {  
 BookDonation bookDonation = new BookDonation();  
 bookDonation.setDonationbookid(rs.getString("编号"));  
 bookDonation.setDonationbookname(rs.getString("书籍名"));  
 bookDonation.setDonationbookautorname(rs.getString("作者"));  
 bookDonation.setDonationbookpublishinghouse(rs.getString("出版社信息"));  
 bookDonation.setImg\_1(rs.getString("书籍封面"));  
 bookDonation.setImg\_2(rs.getString("内页一"));  
 bookDonation.setImg\_3(rs.getString("内页二"));  
 bookDonation.setImg\_4(rs.getString("内页三"));  
 bookDonation.setImg\_5(rs.getString("内页四"));  
 bookDonation.setImg\_6(rs.getString("内页五"));  
 bookDonation.setImg\_7(rs.getString("书籍底面"));  
 bookDonation.setDonationbookstatus(rs.getString("书籍状态"));  
 bookDonationList.add(bookDonation);  
 }  
 return bookDonationList;  
 } 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;  
 }  
}