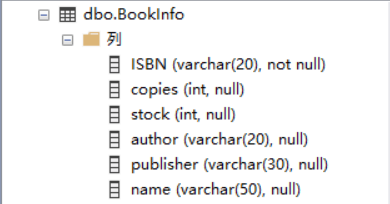
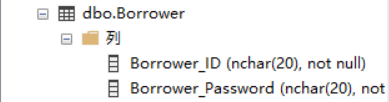
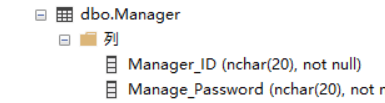
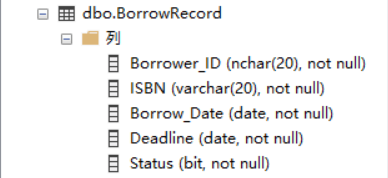
1. SQL Sever实现

1、建表









2、触发器：

1）进行借书操作时改变书本库存信息

USE [BookManage]

GO

/\*\*\*\*\*\* Object: Trigger [dbo].[tbr\_insert] Script Date: 2018/6/27 16:03:45 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER TRIGGER [dbo].[tbr\_insert] ON [dbo].[BorrowRecord]

FOR INSERT AS

IF EXISTS(SELECT \* FROM inserted a WHERE a.Borrower\_ID NOT IN

(SELECT b.Borrower\_ID FROM Borrower b)

OR a.ISBN NOT IN (SELECT c.ISBN FROM BookInfo c)

OR EXISTS(SELECT \* FROM BookInfo WHERE stock<=0 AND a.ISBN= BookInfo.ISBN AND a.Status='0') )

BEGIN

RAISERROR ('违背数据的一致性.', 16, 1)

ROLLBACK TRANSACTION

END

ELSE IF EXISTS(select \* from inserted where inserted.Status='0')

BEGIN

UPDATE BookInfo SET stock=stock-1 WHERE BookInfo.ISBN IN(SELECT ISBN FROM inserted)

END

2）进行还书操作时改变书本库存信息

USE [BookManage]

GO

/\*\*\*\*\*\* Object: Trigger [dbo].[tbr\_update] Script Date: 2018/6/27 16:23:55 \*\*\*\*\*\*/

SET ANSI\_NULLS ON

GO

SET QUOTED\_IDENTIFIER ON

GO

ALTER TRIGGER [dbo].[tbr\_update] ON [dbo].[BorrowRecord]

FOR UPDATE AS

IF EXISTS(SELECT \* FROM inserted a WHERE a.Borrower\_ID NOT IN

(SELECT b.Borrower\_ID FROM Borrower b)

OR a.ISBN NOT IN (SELECT c.ISBN FROM BookInfo c)

OR EXISTS(SELECT \* FROM BookInfo WHERE stock>=copies AND a.ISBN= BookInfo.ISBN AND a.Status=1) )

BEGIN

RAISERROR ('违背数据的一致性.', 16, 1)

ROLLBACK TRANSACTION

END

ELSE IF EXISTS(select \* from inserted where inserted.Status=1)

BEGIN

UPDATE BookInfo SET stock=stock+1 WHERE BookInfo.ISBN IN(SELECT inserted.ISBN FROM inserted)

END

3、存储过程：通过书号获取书本信息，信息放入结果集

use BookManage

go

create procedure bkinfo

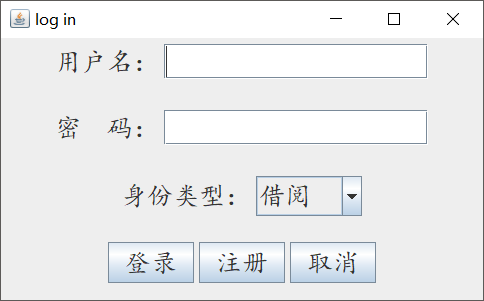
@isbn varchar(20)

as

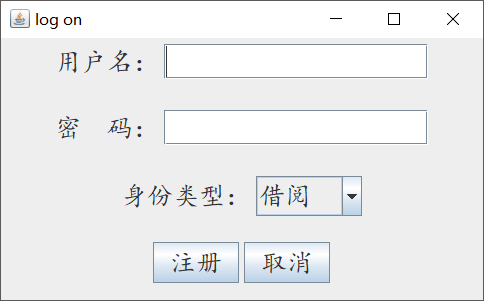
select \* from BookInfo where ISBN=@isbn

go

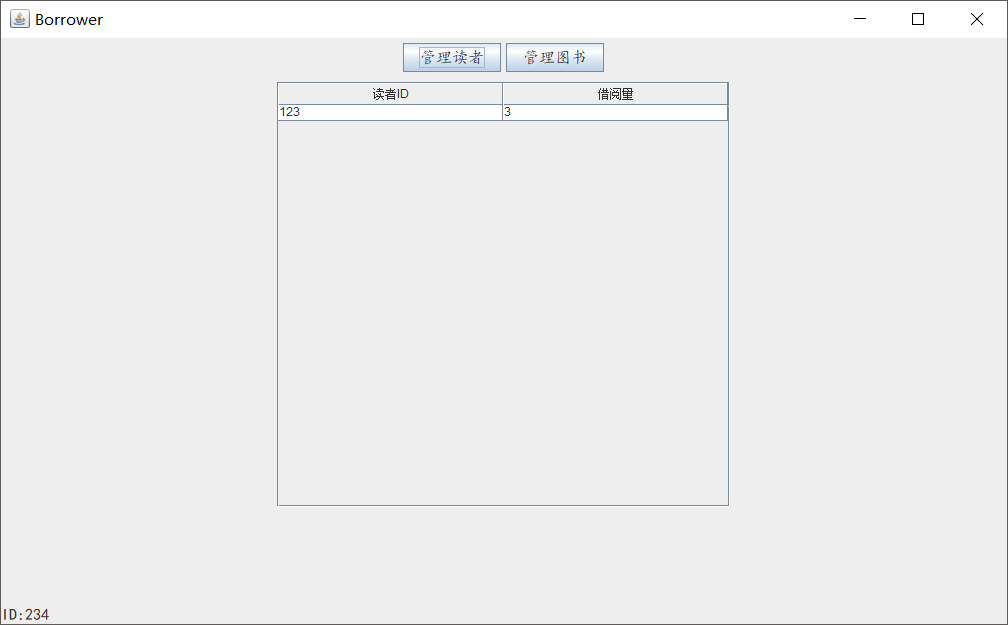
1. java实现
2. 登录界面



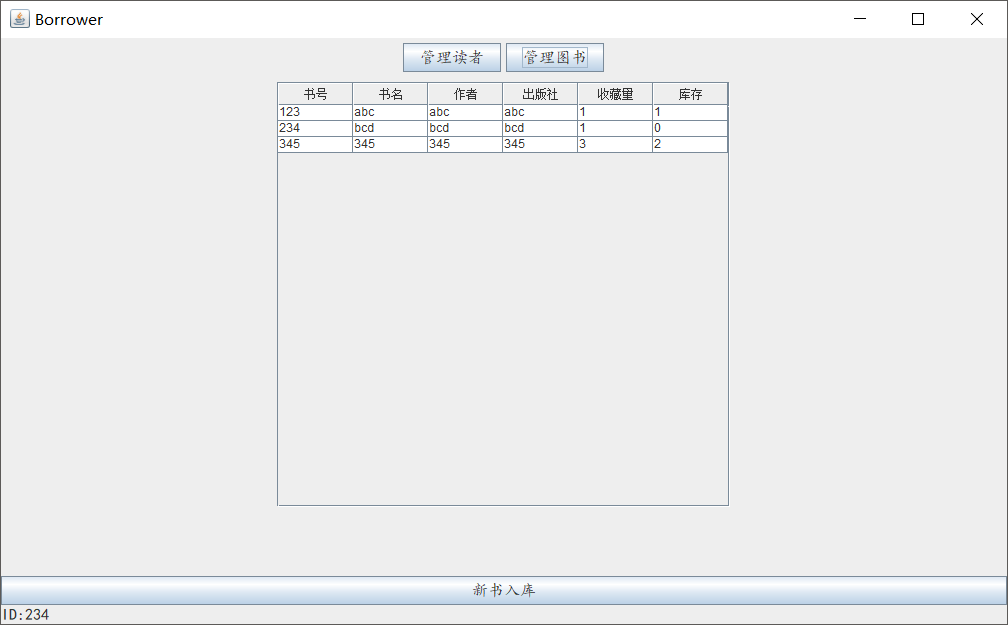
1. 注册界面



1. 管理员界面
2. 管理读者



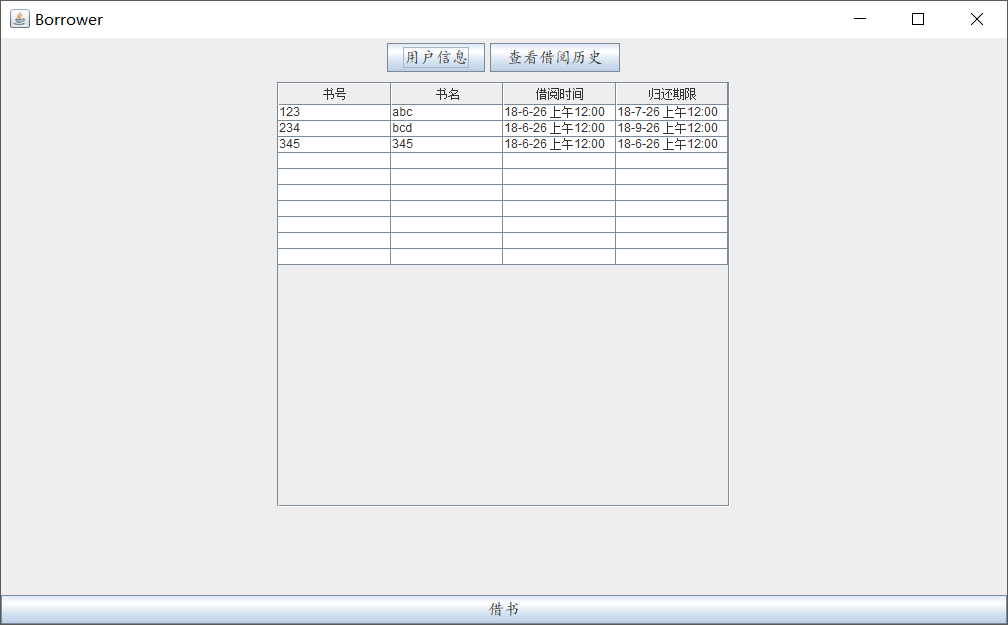
1. 管理图书



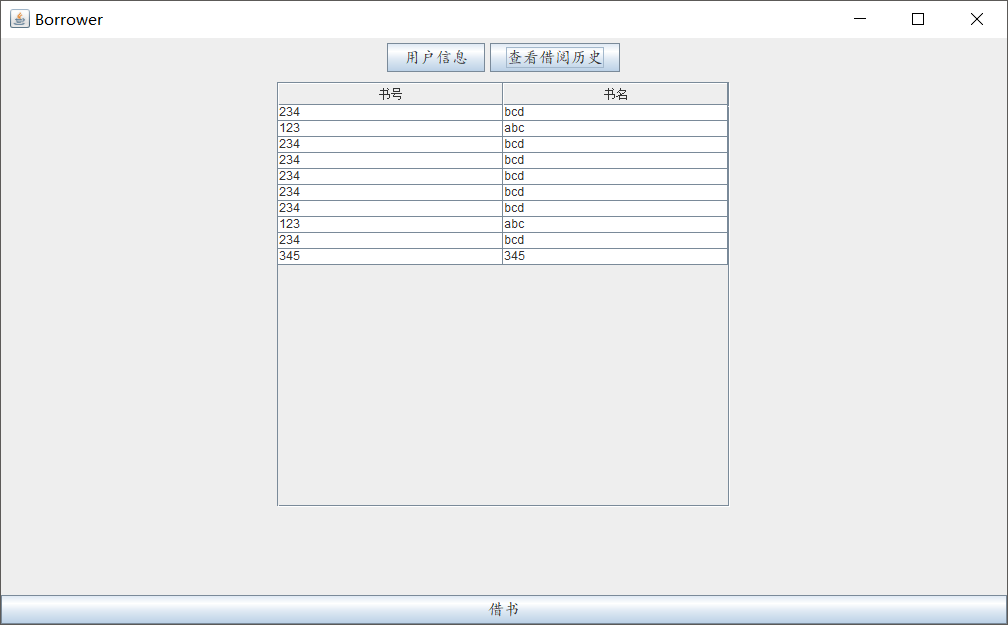
新书入库



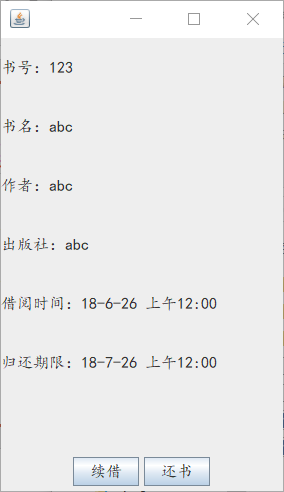
1. 借阅者界面
2. 查看未还书目



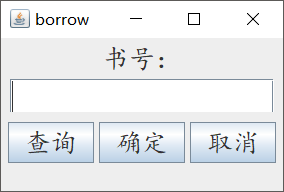
1. 查看借阅历史



1. 还书/续借（双击已借书目进入）



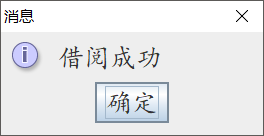
1. 借书



查询



借阅成功提示



1. 实现代码

1-> package LibInfo;

import java.sql.\*;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class Login extends JFrame

implements ActionListener{

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

public static String URL="jdbc:sqlserver://localhost:1433;DatabaseName=BookManage";

public static String USERNAME="olly";

public static String PASSWORD="199811";

Statement ps;

ResultSet rs;

Font f=new Font("楷体",Font.PLAIN,26);

{

UIManager.put("Button.font", f);

UIManager.put("Label.font", f);

UIManager.put("TextField.font", f);

UIManager.put("PasswordField.font", f);

UIManager.put("ComboBox.font", f);

}

JPanel jp1,jp2,jp3,jp4;

JPanel logPanel=new JPanel();

JLabel jl1,jl2,jl3;

JButton jb1,jb2,jb3;

JComboBox<String> jcb=new JComboBox<>();

{

jcb.addItem("借阅");

jcb.addItem("管理员");

}

JTextField jtf1,jtf2;

public Login(){

super("log in");

setLayout(new GridLayout(4,1,5,5));

jp1=new JPanel();

jl1=new JLabel("用户名：");

jtf1=new JTextField(20);

jp1.add(jl1);

jp1.add(jtf1);

jp2=new JPanel();

jl2=new JLabel("密 码：");

jtf2=new JPasswordField(20);

jp2.add(jl2);jp2.add(jtf2);

jp3=new JPanel();

jl3=new JLabel("身份类型：");

jp3.add(jl3);jp3.add(jcb);

jp4=new JPanel();

jb1=new JButton("登录");

jb2=new JButton("注册");

jb3=new JButton("取消");

jp4.add(jb1);jp4.add(jb2);jp4.add(jb3);

this.add(jp1);

this.add(jp2);

this.add(jp3);

this.add(jp4);

this.setSize(500,309);

setLocationRelativeTo(null);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setVisible(true);

jb1.addActionListener(new ActionListener(){

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

if((!jtf1.getText().equals(""))&&(!jtf2.getText().equals("")))

{

try{

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Connection con=DriverManager.getConnection(URL, USERNAME,PASSWORD);

ps=con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_READ\_ONLY);

}catch (ClassNotFoundException|SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

switch(jcb.getSelectedItem().toString())

{

case "管理员":

try {

rs=ps.executeQuery("select \* from Manager where Manager\_ID='"+jtf1.getText().trim()+"'");

if(!rs.next())

JOptionPane.showMessageDialog(null, "没有此管理员");

else if(!(rs.getString("Manage\_Password").trim().equals(jtf2.getText())))

JOptionPane.showMessageDialog(null, "密码错误");

else

{

new Manager\_win(jtf1.getText());

dispose();

}

} catch (SQLException e1) {

// TODO Auto-generated catch block

String error=e1.getMessage();

JOptionPane.showMessageDialog(null, error);

e1.printStackTrace();

}

break;

case "借阅":

try{

rs=ps.executeQuery("select \* from Borrower where Borrower\_ID='"+jtf1.getText()+"'");

if(!rs.next())

JOptionPane.showMessageDialog(null, "该用户不存在");

else if(!(rs.getString("Borrower\_Password").trim().equals(jtf2.getText())))

JOptionPane.showMessageDialog(null, "密码错误");

else

{

new Borrower\_win(jtf1.getText());

dispose();

}

}catch(SQLException e1){

String error=e1.getMessage();

JOptionPane.showMessageDialog(null, error);

e1.printStackTrace();

}

break;

}

}

else if(jtf1.getText().equals(""))

JOptionPane.showMessageDialog(null, "用户名不能为空");

else

JOptionPane.showMessageDialog(null, "密码不能为空");

}

});

jb2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

new Logon();

dispose();

}

});

jb3.addActionListener(new ActionListener(){

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

System.exit(0);

}

});

}

public static void main(String[] args)

throws Exception{

new Login();

}

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

}

}

2-> package LibInfo;

import java.sql.\*;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class Logon extends JFrame

implements ActionListener{

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

public static String URL="jdbc:sqlserver://localhost:1433;DatabaseName=BookManage";

public static String USERNAME="olly";

public static String PASSWORD="199811";

Statement ps;

ResultSet rs;

Font f=new Font("楷体",Font.PLAIN,26);

{

UIManager.put("Button.font", f);

UIManager.put("Label.font", f);

UIManager.put("TextField.font", f);

UIManager.put("PasswordField.font", f);

UIManager.put("ComboBox.font", f);

}

JPanel jp1,jp2,jp3,jp4;

JPanel logPanel=new JPanel();

JLabel jl1,jl2,jl3;

JButton jb2,jb3;

JComboBox<String> jcb=new JComboBox<>();

{

jcb.addItem("借阅");

jcb.addItem("管理员");

}

JTextField jtf1,jtf2;

public Logon(){

super("log on");

setLayout(new GridLayout(4,1,5,5));

jp1=new JPanel();

jl1=new JLabel("用户名：");

jtf1=new JTextField(20);

jp1.add(jl1);

jp1.add(jtf1);

jp2=new JPanel();

jl2=new JLabel("密 码：");

jtf2=new JPasswordField(20);

jp2.add(jl2);jp2.add(jtf2);

jp3=new JPanel();

jl3=new JLabel("身份类型：");

jp3.add(jl3);jp3.add(jcb);

jp4=new JPanel();

jb2=new JButton("注册");

jb3=new JButton("取消");

jp4.add(jb2);jp4.add(jb3);

this.add(jp1);

this.add(jp2);

this.add(jp3);

this.add(jp4);

this.setSize(500,309);

setLocationRelativeTo(null);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setVisible(true);

jb2.addActionListener(new ActionListener(){

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

if((!jtf1.getText().equals(""))&&(!jtf2.getText().equals("")))

{

try{

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Connection con=DriverManager.getConnection(URL, USERNAME,PASSWORD);

ps=con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_READ\_ONLY);

}catch (ClassNotFoundException|SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

switch(jcb.getSelectedItem().toString())

{

case "管理员":

try {

rs=ps.executeQuery("select \* from Manager where Manager\_ID='"+jtf1.getText().trim()+"'");

if(rs.next())

JOptionPane.showMessageDialog(null, "此账户已存在");

else

{

ps.executeUpdate("insert into Manager values('"+jtf1.getText()+"','"+jtf2.getText()+"')");

JOptionPane.showMessageDialog(null, "注册成功");

}

} catch (SQLException e1) {

// TODO Auto-generated catch block

String error=e1.getMessage();

JOptionPane.showMessageDialog(null, error);

e1.printStackTrace();

}

break;

case "借阅":

try{

rs=ps.executeQuery("select \* from Borrower where Borrower\_ID='"+jtf1.getText()+"'");

if(rs.next())

JOptionPane.showMessageDialog(null, "该用户已存在");

else

{

ps.executeUpdate("insert into Borrower values('"+jtf1.getText()+"','"+jtf2.getText()+"')");

JOptionPane.showMessageDialog(null, "注册成功");

}

}catch(SQLException e1){

String error=e1.getMessage();

JOptionPane.showMessageDialog(null, error);

e1.printStackTrace();

}

break;

}

}

else if(jtf1.getText().equals(""))

JOptionPane.showMessageDialog(null, "用户名不能为空");

else

JOptionPane.showMessageDialog(null, "密码不能为空");

}

});

jb3.addActionListener(new ActionListener(){

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

System.exit(0);

}

});

}

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

}

}

3-> package LibInfo;

import java.sql.\*;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import javax.swing.table.\*;

public class Manager\_win extends JFrame

implements ActionListener{

private static final long serialVersionUID = 1L;

public static String URL="jdbc:sqlserver://localhost:1433;DatabaseName=BookManage";

public static String USERNAME="olly";

public static String PASSWORD="199811";

Statement ps;

ResultSet rs;

Font f=new Font("楷体",Font.PLAIN,16);

{

UIManager.put("Button.font", f);

UIManager.put("Label.font", f);

UIManager.put("TextField.font", f);

UIManager.put("PasswordField.font", f);

UIManager.put("ComboBox.font", f);

}

CardLayout card=new CardLayout();

Manager\_win(String ID){

super("Borrower");

setLayout(new BorderLayout());

JButton jb1=new JButton("管理读者");

JButton jb2=new JButton("管理图书");

JPanel bjp=new JPanel();

bjp.add(jb1);bjp.add(jb2);

add(bjp, BorderLayout.NORTH);

JPanel jp=new JPanel(card);

JLabel jl=new JLabel("ID:"+ID);

add(jl,BorderLayout.SOUTH);

JPanel jp1=InitReaderTable();

jp.add("1",jp1);

JPanel jp2=new JPanel(new BorderLayout());

JPanel jpt=InitBookTable();

jp2.add(jpt,BorderLayout.CENTER);

JButton jb=new JButton("新书入库");

jb.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

new AddBook(ID);

dispose();

}

});

jp2.add(jb,BorderLayout.SOUTH);

jp.add("2",jp2);

jb1.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

card.show(jp, "1");

}

});

jb2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

card.show(jp, "2");

}

});

add(jp,BorderLayout.CENTER);

this.setSize(1024,633);

setLocationRelativeTo(null);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setVisible(true);

}

public JPanel InitReaderTable(){

JPanel jp=new JPanel();

JTable table=new JTable(new ReaderTable());

/\*table.addMouseListener(new MouseAdapter() {

public void mouseClicked(MouseEvent e){

if(e.getClickCount()==2)

{

int row=((JTable)e.getSource()).rowAtPoint(e.getPoint());

new ManageBorrower(ID,getInfo(ID),row);

dispose();

}

}

});\*/

JScrollPane scroll=new JScrollPane(table);

jp.add(scroll);

return jp;

}

private class ReaderTable extends AbstractTableModel{

private static final long serialVersionUID = 1L;

String[] columnNames={"读者ID","借阅量"};

String[][] obj;

public ReaderTable(){

obj=getReaderInfo();

}

@Override

public int getRowCount() {

// TODO Auto-generated method stub

return obj.length;

}

@Override

public int getColumnCount() {

// TODO Auto-generated method stub

return columnNames.length;

}

@Override

public String getColumnName(int column) {

return columnNames[column];

}

@Override

public Object getValueAt(int rowIndex, int columnIndex) {

// TODO Auto-generated method stub

return obj[rowIndex][columnIndex];

}

}

public String[][] getReaderInfo(){

String[][] obj = null;

try{

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Connection con=DriverManager.getConnection(URL, USERNAME,PASSWORD);

ps=con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_READ\_ONLY);

rs=ps.executeQuery("select \* from Borrower");

rs.last();

int rowcnt=rs.getRow();

obj=new String[rowcnt][2];

int i=0;

rs=ps.executeQuery("select \* from Borrower");

while(rs.next())

{

String id=rs.getString("Borrower\_ID");

obj[i][0]=id;

Statement st=con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_READ\_ONLY);

ResultSet rs1=st.executeQuery("select count(\*)num from BorrowRecord where Borrower\_ID='"+id+"' and Status=0");

int bnum=0;

if(rs1.next())

bnum=rs1.getInt("num");

obj[i][1]=String.valueOf(bnum);

i++;

}

}catch (ClassNotFoundException|SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

return obj;

}

public JPanel InitBookTable(){

JPanel jp=new JPanel();

JTable table=new JTable(new BookTable());

/\*table.addMouseListener(new MouseAdapter() {

public void mouseClicked(MouseEvent e){

if(e.getClickCount()==2)

{

int row=((JTable)e.getSource()).rowAtPoint(e.getPoint());

new ManageBook(ID,getInfo(ID),row);

dispose();

}

}

});\*/

JScrollPane scroll=new JScrollPane(table);

jp.add(scroll);

return jp;

}

private class BookTable extends AbstractTableModel{

private static final long serialVersionUID = 1L;

String[] columnNames={"书号","书名","作者","出版社","收藏量","库存"};

String[][] obj;

public BookTable(){

obj=getBookInfo();

}

@Override

public int getRowCount() {

// TODO Auto-generated method stub

return obj.length;

}

@Override

public int getColumnCount() {

// TODO Auto-generated method stub

return columnNames.length;

}

@Override

public String getColumnName(int column) {

return columnNames[column];

}

@Override

public Object getValueAt(int rowIndex, int columnIndex) {

// TODO Auto-generated method stub

return obj[rowIndex][columnIndex];

}

}

public String[][] getBookInfo(){

String[][] obj = null;

try{

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Connection con=DriverManager.getConnection(URL, USERNAME,PASSWORD);

ps=con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_READ\_ONLY);

rs=ps.executeQuery("select \* from BookInfo");

rs.last();

int rowcnt=rs.getRow();

obj=new String[rowcnt][6];

int i=0;

rs=ps.executeQuery("select \* from BookInfo");

while(rs.next())

{

String isbn=rs.getString("ISBN");

String name=rs.getString("name");

String author=rs.getString("author");

String publisher=rs.getString("publisher");

int copies=rs.getInt("copies");

int stock=rs.getInt("stock");

obj[i][0]=isbn;obj[i][1]=name;

obj[i][2]=author;obj[i][3]=publisher;

obj[i][4]=String.valueOf(copies);obj[i][5]=String.valueOf(stock);

i++;

}

}catch (ClassNotFoundException|SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

return obj;

}

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

}

}

package LibInfo;

import java.sql.\*;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

public class AddBook extends JFrame

implements ActionListener{

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

public static String URL="jdbc:sqlserver://localhost:1433;DatabaseName=BookManage";

public static String USERNAME="olly";

public static String PASSWORD="199811";

Statement ps;

ResultSet rs;

Font f=new Font("楷体",Font.PLAIN,26);

{

UIManager.put("Button.font", f);

UIManager.put("Label.font", f);

UIManager.put("TextField.font", f);

UIManager.put("PasswordField.font", f);

UIManager.put("ComboBox.font", f);

}

JPanel jp1,jp2,jp3,jp4,jp5,jp6;

JPanel logPanel=new JPanel();

JLabel jl1,jl2,jl3,jl4,jl5;

JButton jb1,jb2;

JTextField jtf1,jtf2,jtf3,jtf4,jtf5;

AddBook(String ID){

super("add book");

setLayout(new GridLayout(6,1,5,5));

jp1=new JPanel();

jl1=new JLabel("书号：");

jtf1=new JTextField(20);

jp1.add(jl1);

jp1.add(jtf1);

jp2=new JPanel();

jl2=new JLabel("书名：");

jtf2=new JTextField(20);

jp2.add(jl2);jp2.add(jtf2);

jp3=new JPanel();

jl3=new JLabel("作者：");

jtf3=new JTextField(20);

jp3.add(jl3);jp3.add(jtf3);

jp4=new JPanel();

jl4=new JLabel("出版社：");

jtf4=new JTextField(20);

jp4.add(jl4);jp4.add(jtf4);

jp5=new JPanel();

jl5=new JLabel("数量：");

jtf5=new JTextField(20);

jp5.add(jl5);jp5.add(jtf5);

jp6=new JPanel();

jb1=new JButton("确定");

jb2=new JButton("取消");

jp6.add(jb1);jp6.add(jb2);

this.add(jp1);

this.add(jp2);

this.add(jp3);

this.add(jp4);

this.add(jp5);

this.add(jp6);

this.setSize(633,1024);

setLocationRelativeTo(null);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setVisible(true);

jb1.addActionListener(new ActionListener(){

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

if(!jtf1.getText().equals(""))

{

try{

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Connection con=DriverManager.getConnection(URL, USERNAME,PASSWORD);

ps=con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_READ\_ONLY);

rs=ps.executeQuery("select \* from BookInfo where ISBN='"+jtf1.getText().trim()+"'");

if(rs.next())

JOptionPane.showMessageDialog(null, "此书已存在");

else

{

ps.executeUpdate("insert into BookInfo values('"+jtf1.getText()+"','"+Integer.valueOf(jtf5.getText())+"','"+Integer.valueOf(jtf5.getText())+"','"+jtf3.getText()+"','"+jtf4.getText()+"','"+jtf2.getText()+"')");

JOptionPane.showMessageDialog(null, "入库成功");

new Manager\_win(ID);

dispose();

}

}catch (ClassNotFoundException|SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

}

else

JOptionPane.showMessageDialog(null, "书号不能为空");

}

});

jb2.addActionListener(new ActionListener(){

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

new Manager\_win(ID);

dispose();

}

});

}

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

}

}

4-> package LibInfo;

import java.sql.\*;

import java.text.\*;

import java.awt.\*;

import java.awt.event.\*;

import javax.swing.\*;

import javax.swing.table.\*;

public class Borrower\_win extends JFrame

implements ActionListener{

/\*\*

\*

\*/

private static final long serialVersionUID = 1L;

public static String URL="jdbc:sqlserver://localhost:1433;DatabaseName=BookManage";

public static String USERNAME="olly";

public static String PASSWORD="199811";

Statement ps;

ResultSet rs;

Font f=new Font("楷体",Font.PLAIN,16);

{

UIManager.put("Button.font", f);

UIManager.put("Label.font", f);

UIManager.put("TextField.font", f);

UIManager.put("PasswordField.font", f);

UIManager.put("ComboBox.font", f);

}

CardLayout card=new CardLayout();

Borrower\_win(String ID)

{

super("Borrower");

setLayout(new BorderLayout());

JButton jb1=new JButton("用户信息");

JButton jb2=new JButton("查看借阅历史");

JPanel bjp=new JPanel();

bjp.add(jb1);bjp.add(jb2);

add(bjp, BorderLayout.NORTH);

JPanel jp=new JPanel(card);

JLabel jl=new JLabel("ID:"+ID);

add(jl,BorderLayout.SOUTH);

JPanel jp1=InitInfoTable(ID);

jp.add("1",jp1);

JPanel jp2=InitHisTable(ID);

jp.add("2",jp2);

jb1.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

card.show(jp, "1");

}

});

jb2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

card.show(jp, "2");

}

});

JButton jb=new JButton("借书");

jb.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

try {

new Borrow(ID);

} catch (Exception e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

}

});

add(jp,BorderLayout.CENTER);

add(jb,BorderLayout.SOUTH);

this.setSize(1024,633);

setLocationRelativeTo(null);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setVisible(true);

}

public JPanel InitInfoTable(String ID){

JPanel jp=new JPanel();

JTable table=new JTable(new InfoTable(ID));

table.addMouseListener(new MouseAdapter() {

public void mouseClicked(MouseEvent e){

if(e.getClickCount()==2)

{

int row=((JTable)e.getSource()).rowAtPoint(e.getPoint());

new BorrowerBook(ID,getInfo(ID),row);

dispose();

}

}

});

JScrollPane scroll=new JScrollPane(table);

jp.add(scroll);

return jp;

}

private class InfoTable extends AbstractTableModel{

private static final long serialVersionUID = 1L;

String[] columnNames={"书号","书名","借阅时间","归还期限"};

String[][] obj;

public InfoTable(String ID){

obj=getInfo(ID);

}

@Override

public int getRowCount() {

// TODO Auto-generated method stub

return obj.length;

}

@Override

public int getColumnCount() {

// TODO Auto-generated method stub

return columnNames.length;

}

@Override

public String getColumnName(int column) {

return columnNames[column];

}

@Override

public Object getValueAt(int rowIndex, int columnIndex) {

// TODO Auto-generated method stub

return obj[rowIndex][columnIndex];

}

}

public String[][] getInfo(String ID){

String[][] obj = null;

try{

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Connection con=DriverManager.getConnection(URL, USERNAME,PASSWORD);

ps=con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_READ\_ONLY);

rs=ps.executeQuery("select \* from BorrowRecord");

rs.last();

int rowcnt=rs.getRow();

obj=new String[rowcnt][4];

rs=ps.executeQuery("select \* from BorrowRecord where Borrower\_ID='"+ID+"' and Status=0");

int i=0;

while(rs.next())

{

String isbn=rs.getString("ISBN");

obj[i][0]=isbn;

Statement st=con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_READ\_ONLY);

ResultSet rs1=st.executeQuery("select \* from BookInfo where ISBN='"+isbn+"'");

if(rs1.next())

{

String bkname=rs1.getString("name");

DateFormat df=new SimpleDateFormat();

String borrowdate=df.format(rs.getTimestamp("Borrow\_Date").getTime());

String deadline=df.format(rs.getTimestamp("Deadline").getTime());

obj[i][1]=bkname;

obj[i][2]=borrowdate;

obj[i][3]=deadline;

}

i++;

}

}catch (ClassNotFoundException|SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

return obj;

}

public JPanel InitHisTable(String ID){

JPanel jp=new JPanel();

JTable table=new JTable(new HisTable(ID));

/\*table.addMouseListener(new MouseAdapter() {

public void mouseClicked(MouseEvent e){

if(e.getClickCount()==2)

{

int row=((JTable)e.getSource()).rowAtPoint(e.getPoint());

new BorrowerBook(ID,getInfo(ID),row);

dispose();

}

}

});\*/

JScrollPane scroll=new JScrollPane(table);

jp.add(scroll);

return jp;

}

private class HisTable extends AbstractTableModel{

private static final long serialVersionUID = 1L;

String[] columnNames={"书号","书名"};

String[][] obj;

public HisTable(String ID){

obj=HisInfo(ID);

}

@Override

public int getRowCount() {

// TODO Auto-generated method stub

return obj.length;

}

@Override

public int getColumnCount() {

// TODO Auto-generated method stub

return columnNames.length;

}

@Override

public String getColumnName(int column) {

return columnNames[column];

}

@Override

public Object getValueAt(int rowIndex, int columnIndex) {

// TODO Auto-generated method stub

return obj[rowIndex][columnIndex];

}

}

public String[][] HisInfo(String ID){

String[][] obj = null;

try{

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Connection con=DriverManager.getConnection(URL, USERNAME,PASSWORD);

ps=con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_READ\_ONLY);

rs=ps.executeQuery("select \* from BorrowRecord");

rs.last();

int rowcnt=rs.getRow();

obj=new String[rowcnt][2];

rs=ps.executeQuery("select \* from BorrowRecord where Borrower\_ID='"+ID+"'");

int i=0;

while(rs.next())

{

String isbn=rs.getString("ISBN");

obj[i][0]=isbn;

Statement st=con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_READ\_ONLY);

ResultSet rs1=st.executeQuery("select \* from BookInfo where ISBN='"+isbn+"'");

if(rs1.next())

{

String bkname=rs1.getString("name");

obj[i][1]=bkname;

}

i++;

}

}catch (ClassNotFoundException|SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

return obj;

}

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

}

}

package LibInfo;

import java.awt.\*;

import java.awt.event.\*;

import java.sql.\*;

import javax.swing.\*;

public class BorrowerBook extends JFrame

implements ActionListener{

private static final long serialVersionUID = 1L;

public static String URL="jdbc:sqlserver://localhost:1433;DatabaseName=BookManage";

public static String USERNAME="olly";

public static String PASSWORD="199811";

Statement ps;

ResultSet rs;

Font f=new Font("楷体",Font.PLAIN,16);

{

UIManager.put("Button.font", f);

UIManager.put("Label.font", f);

UIManager.put("TextField.font", f);

UIManager.put("PasswordField.font", f);

UIManager.put("ComboBox.font", f);

}

BorrowerBook(String ID,String[][] obj,int row){

setLayout(new BorderLayout());

JPanel jp1=new JPanel(new GridLayout(7,1));

String[] bkinfo=getBookInfo(obj[row][0]);

JLabel jl1=new JLabel("书号："+bkinfo[0]);

JLabel jl2=new JLabel("书名："+bkinfo[1]);

JLabel jl3=new JLabel("作者："+bkinfo[2]);

JLabel jl4=new JLabel("出版社："+bkinfo[3]);

JLabel jl5=new JLabel("借阅时间："+obj[row][2]);

JLabel jl6=new JLabel("归还期限："+obj[row][3]);

jp1.add(jl1);jp1.add(jl2);jp1.add(jl3);jp1.add(jl4);jp1.add(jl5);jp1.add(jl6);

JButton jb1=new JButton("续借");

jb1.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

try{

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Connection con=DriverManager.getConnection(URL, USERNAME,PASSWORD);

PreparedStatement pres=con.prepareStatement("update BorrowRecord set Deadline=DATEADD(mm,1,Deadline) where Borrower\_ID='"+ID+"' and ISBN='"+bkinfo[0]+"'");

pres.execute();

new Borrower\_win(ID);

dispose();

}catch (ClassNotFoundException|SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

}

});

JButton jb2=new JButton("还书");

jb2.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

try{

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Connection con=DriverManager.getConnection(URL, USERNAME,PASSWORD);

PreparedStatement pres=con.prepareStatement("update BorrowRecord set Status=Status+1 where Borrower\_ID='"+ID+"' and ISBN='"+bkinfo[0]+"'");

pres.execute();

new Borrower\_win(ID);

dispose();

}catch (ClassNotFoundException|SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

}

});

JPanel jp2=new JPanel();

jp2.add(jb1);jp2.add(jb2);

add(jp1,BorderLayout.CENTER);add(jp2,BorderLayout.SOUTH);

this.setSize(300,500);

setLocationRelativeTo(null);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setVisible(true);

}

public String[] getBookInfo(String ISBN){

String[] bkinfo = new String[4];

try{

Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Connection con=DriverManager.getConnection(URL, USERNAME,PASSWORD);

ps=con.createStatement(ResultSet.TYPE\_SCROLL\_INSENSITIVE, ResultSet.CONCUR\_READ\_ONLY);

rs=ps.executeQuery("select \* from BookInfo where ISBN='"+ISBN+"'");

while(rs.next())

{

String isbn=rs.getString("ISBN");

bkinfo[0]=isbn;

String name=rs.getString("name");

bkinfo[1]=name;

String author=rs.getString("author");

bkinfo[2]=author;

String publisher=rs.getString("publisher");

bkinfo[3]=publisher;

}

}catch (ClassNotFoundException|SQLException e1) {

// TODO Auto-generated catch block

e1.printStackTrace();

}

return bkinfo;

}

@Override

public void actionPerformed(ActionEvent e) {

// TODO Auto-generated method stub

}

}

**package** LibInfo;

**import** java.sql.\*;

**import** java.awt.\*;

**import** java.awt.event.\*;

**import** javax.swing.\*;

**public** **class** Book\_win

**implements** ActionListener{

**public** **static** String *URL*="jdbc:sqlserver://localhost:1433;DatabaseName=BookManage";

**public** **static** String *USERNAME*="olly";

**public** **static** String *PASSWORD*="199811";

Statement ps;

ResultSet rs;

Font f=**new** Font("楷体",Font.***PLAIN***,26);

{

UIManager.*put*("Button.font", f);

UIManager.*put*("Label.font", f);

UIManager.*put*("TextField.font", f);

UIManager.*put*("PasswordField.font", f);

UIManager.*put*("ComboBox.font", f);

}

Book\_win(String isbn){

**try**{

Class.*forName*("com.microsoft.sqlserver.jdbc.SQLServerDriver");

Connection con=DriverManager.*getConnection*(*URL*, *USERNAME*,*PASSWORD*);

ps=con.createStatement(ResultSet.***TYPE\_SCROLL\_INSENSITIVE***, ResultSet.***CONCUR\_READ\_ONLY***);

rs=ps.executeQuery("select \* from BookInfo where ISBN='"+isbn.trim()+"'");

**if**(rs.next())

{

rs=ps.executeQuery("exec bkinfo '"+isbn+"'");

/\*CallableStatement cst=con.prepareCall("call bkinfo(?,?,?,?,?,?)");

System.out.println(isbn);

cst.setString(1, isbn);

cst.registerOutParameter(2, Types.VARCHAR);

cst.registerOutParameter(3, Types.VARCHAR);

cst.registerOutParameter(4, Types.VARCHAR);

cst.registerOutParameter(5, Types.INTEGER);

cst.registerOutParameter(6, Types.INTEGER);

cst.execute();\*/

**if**(rs.next())

{

String bname=rs.getString("name");

String writer=rs.getString("author");

String pub=rs.getString("publisher");

**int** cop=rs.getInt("copies");

**int** sto=rs.getInt("stock");

JPanel jp1=**new** JPanel(**new** GridLayout(7,1));

JLabel jl1=**new** JLabel("书号："+isbn);

JLabel jl2=**new** JLabel("书名："+bname);

JLabel jl3=**new** JLabel("作者："+writer);

JLabel jl4=**new** JLabel("出版社："+pub);

JLabel jl5=**new** JLabel("收藏量："+String.*valueOf*(cop));

JLabel jl6=**new** JLabel("库中剩余量："+String.*valueOf*(sto));

JButton jb=**new** JButton("确定");

jp1.add(jl1);jp1.add(jl2);jp1.add(jl3);jp1.add(jl4);jp1.add(jl5);jp1.add(jl6);jp1.add(jb);

JFrame jf=**new** JFrame();

jf.add(jp1);

jb.addActionListener(**new** ActionListener() {

@Override

**public** **void** actionPerformed(ActionEvent e) {

// **TODO** Auto-generated method stub

jf.dispose();

}

});

jf.setSize(300,500);

jf.setLocationRelativeTo(**null**);

jf.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

jf.setVisible(**true**);

}

}

**else**

{

JOptionPane.*showMessageDialog*(**null**, "本馆无此书");

}

}**catch** (ClassNotFoundException|SQLException e1) {

// **TODO** Auto-generated catch block

e1.printStackTrace();

}

}

@Override

**public** **void** actionPerformed(ActionEvent e) {

// **TODO** Auto-generated method stub

}

}