

Source Code

```
/* Project - Library Record System */
```

```
import java.io.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;
import java.sql.*;
import java.awt.Graphics;
import javax.swing.JPopupMenu;
import com.sun.java.swing.plaf.nimbus.NimbusLookAndFeel;
import javax.swing.UIManager.*;
import javax.swing.table.DefaultTableModel;
import java.awt.Desktop;
import java.util.Vector;
import java.text.DateFormat;
import java.text.Format;
import javax.swing.JFormattedTextField;
public class ProjectLibrary extends JFrame implements ActionListener
{
    JTextField id,phno,name,add,city,padd,eadd,class1,Year;
    JTextField progress,usert,rn,date,userA,passA;
    JPasswordField pass;
    JTextField bookTF,authorTF,priceTF,bcodeTF,issuer1TF;
    JTextField cdcodeTF,cdtitleTF,cdisbnTF;
    JTextField cdeditionTF,cdpubTF,issuer2TF;
    JTextField bco,rlo,drl,dvdt,rnb;
    JFrame c,d,bbox,dbox;
    JButton next1,next11,previous11,previous22;
    JButton search,save,delete,exit,update,login,signup;
    JButton searBook,saveNow,update1,bexit,retb;
    JButton cdsearch,cdsave,cddelete,cdexit,cdupdate,retc;
    JButton iss,issd,recc,rec,createA;
    JLabel cdlogo,cdcode,cdtitle,cdisbn,cdedit,cdpub;
    JFrame b,aboutus,sbox,book,cd;
    JMenuBar menuBar,menuBar1;
```

```
JMenu menu1,menu2,menu3,menu4;  
JMenuItem menuItemN,menuItemB,menuItemC;  
JMenuItem menuItemH,menuItemA,menuItemE,menuItemI;  
JMenuItem menuItemST,menuItemBA,menuItemCD;  
JMenuItem menuItemBook,menuItemDVD;
```

```
public ProjectLibrary()  
{
```

```
    b=new JFrame("Library Record System");  
    b.setSize(800,450);  
    b.getContentPane().setLayout(null);  
    b.getContentPane().setBackground(new Color(27,35,58));  
    b.getContentPane().setForeground(Color.white);  
    b.setBounds(0,0,800,450);  
    b.setResizable(false);
```

```
    ImageIcon king = new ImageIcon("images/mainpage.jpg");  
    JLabel upp= new JLabel(king);
```

```
    JLabel user = new JLabel("User Name");  
    user.setForeground(Color.white);  
    JLabel password = new JLabel("Password");  
    password.setForeground(Color.white);
```

```
    usert = new JTextField();  
    usert.setForeground(Color.black);  
    usert.setBackground(Color.white);
```

```
    pass= new JPasswordField();  
    pass.setForeground(Color.black);  
    pass.setBackground(Color.white);
```

```
user.setBounds(240,120,670,150);  
password.setBounds(240,225,150,20);
```

```
usert.setBounds(320,185,152,26);  
pass.setBounds(320,225,152,26);  
upp.setBounds(0,0,800,450);
```

```
login=new JButton("Login",new ImageIcon("images/login_icon.png"));  
login.setBounds(320,280,110,30);
```

```
login.setToolTipText("Press to Log Into Account");  
login.addActionListener(this);
```

```
b.getContentPane().add(user);  
b.getContentPane().add(password);  
b.getContentPane().add(usert);  
b.getContentPane().add(pass);  
b.getContentPane().add(login);  
b.add(upp);
```

```
b.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);  
b.setVisible(true);
```

```
}//End of Constructor
```

```
public void AccountCreator()  
{
```

```
    d=new JFrame("Create New Login Account");  
    d.setSize(748,410);  
    d.getContentPane().setLayout(null);  
    d.getContentPane().setBackground(new Color(27,35,58));  
    d.getContentPane().setForeground(Color.white);
```

```
d.setBounds(0,0,748,410);  
d.setResizable(false);
```

```
ImageIcon kimk = new ImageIcon("images/mainpageCreator.jpg");  
JLabel page= new JLabel(kimk);
```

```
JLabel userac= new JLabel("Choose User Name");  
userac.setForeground(Color.white);  
JLabel passwordac= new JLabel("Choose Password");  
passwordac.setForeground(Color.white);
```

```
userA = new JTextField();  
userA.setForeground(Color.black);  
userA.setBackground(Color.white);
```

```
passA= new JTextField();  
passA.setForeground(Color.black);  
passA.setBackground(Color.white);
```

```
userac.setBounds(220,120,670,150);  
passwordac.setBounds(220,225,150,20);
```

```
userA.setBounds(340,185,152,26);  
passA.setBounds(340,225,152,26);  
page.setBounds(0,0,748,410);
```

```
createA=new JButton("Create Account",new  
ImageIcon("images/login_icon.png"));  
createA.setBounds(260,280,180,30);
```

```
createA.setToolTipText("Press to Create New Account");  
createA.addActionListener(this);
```

```
d.getContentPane().add(userac);  
d.getContentPane().add(passwordac);
```

```
d.getContentPane().add(userA);  
d.getContentPane().add(passA);  
d.getContentPane().add(createA);  
d.add(page);
```

```
d.setDefaultCloseOperation(JFrame.DISPOSE_ON_CLOSE);  
d.setVisible(true);
```

```
}
```

```
public void Library()  
{
```

```
    c=new JFrame("Student Record");
```

```
    c.setSize(765,600);
```

```
    c.getContentPane().setLayout(null);
```

```
    c.getContentPane().setBackground(new Color(27,35,58));
```

```
    c.getContentPane().setForeground(Color.white);
```

```
    c.setBounds(0,0,765,600);
```

```
    c.getContentPane().setFont(new Font("Calibri",3,14));
```

```
    c.setResizable(false);
```

```
    menuBar = new JMenuBar();
```

```
    menu1 = new JMenu("Library");
```

```
    menu1.setBackground(Color.white);
```

```
    menu1.setMnemonic('L');
```

```
    menu3 = new JMenu("View");
```

```
    menu3.setMnemonic('V');
```

```
    menu3.setBackground(Color.white);
```

```
    menu2 = new JMenu("Help");
```

```
menu2.setMnemonic('H');  
menu2.setBackground(Color.white);
```

```
menu4 = new JMenu("Issue/Return");  
menu4.setMnemonic('I');  
menu4.setBackground(Color.white);
```

```
menuItemBook= new JMenuItem("Issue/Return Book",new  
ImageIcon("images/book.png"));  
menuItemBook.setBackground(Color.white);  
menuItemBook.setMnemonic('B');
```

```
menuItemDVD= new JMenuItem("Issue/Return CD/DVD",new  
ImageIcon("images/cd.png"));  
menuItemDVD.setBackground(Color.white);  
menuItemDVD.setMnemonic('C');
```

```
menuItemST= new JMenuItem("All Students",new  
ImageIcon("images/user.png"));  
menuItemST.setBackground(Color.white);  
menuItemST.setMnemonic('S');
```

```
menuItemBA= new JMenuItem("All Books",new  
ImageIcon("images/book.png"));  
menuItemBA.setBackground(Color.white);  
menuItemBA.setMnemonic('B');
```

```
menuItemCD= new JMenuItem("All CDs/DVDs",new  
ImageIcon("images/cd.png"));  
menuItemCD.setBackground(Color.white);  
menuItemCD.setMnemonic('C');
```

```
menuItemN = new JMenuItem("New",new  
ImageIcon("images/new.png"));  
menuItemN.setBackground(Color.white);  
menuItemN.setMnemonic('N');
```

```
        JMenuItemI = new JMenuItem("Issued",new  
ImageIcon("images/issued.png"));  
        JMenuItemI.setBackground(Color.white);  
        JMenuItemI.setMnemonic('I');
```

```
        JMenuItemB = new JMenuItem("Books",new  
ImageIcon("images/book.png"));  
        JMenuItemB.setBackground(Color.white);  
        JMenuItemB.setMnemonic('B');
```

```
        JMenuItemC = new JMenuItem("CD's",new  
ImageIcon("images/cd.png"));  
        JMenuItemC.setBackground(Color.white);  
        JMenuItemC.setMnemonic('C');
```

```
        JMenuItemE = new JMenuItem("Exit",new  
ImageIcon("images/exit.png"));  
        JMenuItemE.setBackground(Color.white);  
        JMenuItemE.setMnemonic('E');
```

```
        JMenuItemH = new JMenuItem("Help Library",new  
ImageIcon("images/help.gif"));  
        JMenuItemH.setBackground(Color.white);  
        JMenuItemH.setMnemonic('H');
```

```
        JMenuItemA = new JMenuItem("About Developer",new  
ImageIcon("images/lrs.png"));  
        JMenuItemA.setBackground(Color.white);  
        JMenuItemA.setMnemonic('A');
```

```
c.setJMenuBar(menuBar);
```

```
JLabel messg= new JLabel("Enter Roll No. and Click Search to  
View Student Record");  
messg.setForeground(Color.white);
```

```
JLabel lDate = new JLabel("Date of Birth");
```

```
lDate.setForeground(Color.white);
```

```
JLabel ph = new JLabel("Phone Number");  
ph.setForeground(Color.white);
```

```
JLabel lname = new JLabel("Student's Name");  
lname.setForeground(Color.white);
```

```
JLabel address = new JLabel("Current Address");  
address.setForeground(Color.white);
```

```
JLabel lid = new JLabel("Roll No.");  
lid.setForeground(Color.white);
```

```
JLabel lcity = new JLabel("City");  
lcity.setForeground(Color.white);
```

```
JLabel pad = new JLabel("Permanent Address");  
pad.setForeground(Color.white);
```

```
JLabel leadd = new JLabel("Email Address");  
leadd.setForeground(Color.white);
```

```
JLabel lclass = new JLabel("Course");  
lclass.setForeground(Color.white);
```

```
JLabel lYear = new JLabel("Year");  
lYear.setForeground(Color.white);
```

```
JLabel lprogress = new JLabel("Progress");  
lprogress.setForeground(Color.white);
```

```
ImageIcon background = new ImageIcon("backg.jpg");  
JLabel img = new JLabel(background);
```



```
ImageIcon college = new ImageIcon("images/library.jpg");  
JLabel colLogo = new JLabel(college);
```

```
ImageIcon LineStraight = new  
ImageIcon("images/sideline.png");  
JLabel Line = new JLabel(LineStraight);
```

```
date = new JTextField();  
date.setForeground(Color.black);  
date.setBackground(Color.white);
```

```
phno = new JTextField();  
phno.setForeground(Color.black);  
phno.setBackground(Color.white);
```

```
name = new JTextField();  
name.setForeground(Color.black);  
name.setBackground(Color.white);
```

```
add = new JTextField();  
add.setForeground(Color.black);  
add.setBackground(Color.white);
```

```
eadd = new JTextField();  
eadd.setForeground(Color.black);  
eadd.setBackground(Color.white);
```

```
city = new JTextField();  
city.setForeground(Color.black);  
city.setBackground(Color.white);
```

```
padd = new JTextField();  
padd.setForeground(Color.black);  
padd.setBackground(Color.white);
```

```

id = new JTextField();
id.setForeground(Color.black);
id.setBackground(Color.white);

class1 = new JTextField();
class1.setForeground(Color.black);
class1.setBackground(Color.white);

Year = new JTextField();
Year.setForeground(Color.black);
Year.setBackground(Color.white);

progress = new JTextField();
progress.setForeground(Color.black);
progress.setBackground(Color.white);

next1 = new JButton("New",new
Imagelcon("images/new.png"));
search = new JButton("Search",new
Imagelcon("images/search.png"));
save = new JButton("Save", new
Imagelcon("images/save.png"));
delete = new JButton("Delete", new
Imagelcon("images/remove.png"));
exit = new JButton("Exit", new Imagelcon("images/exit.png"));
update = new JButton("Update", new
Imagelcon("images/update.gif"));

next1.setToolTipText("Clears All Field To Enter New
Record");

search.setToolTipText("Search Student Record");
save.setToolTipText("Save Student Record");
delete.setToolTipText("Delete Student Record");
exit.setToolTipText("Exit From The Library");
update.setToolTipText("Update Student Record");

messg.setBounds(240,150,600,20);

```

```
lid.setBounds(80,135,670,150);  
lname.setBounds(80,240,150,20);  
lDate.setBounds(80,280,150,20);  
lclass.setBounds(80,320,150,20);  
lYear.setBounds(80,360,150,20);  
lprogress.setBounds(80,400,150,20);
```

```
ph.setBounds(400,200,150,20);  
address.setBounds(400,240,150,20);  
lcity.setBounds(400,280,150,20);  
pad.setBounds(400,320,150,20);  
leadd.setBounds(400,360,150,20);
```

```
img.setBounds(90,100,20,40);  
colLogo.setBounds(0,0,759,150);  
Line.setBounds(0,90,48,800);
```

```
id.setBounds(220,200,152,26);  
name.setBounds(220,240,152,26);  
date.setBounds(220,280,152,26);
```

```
class1.setBounds(220,320,152,26);  
Year.setBounds(220,360,152,26);  
progress.setBounds(220,400,152,26);
```

```
phno.setBounds(550,200,152,26);  
add.setBounds(550,240,152,26);  
city.setBounds(550,280,152,26);  
padd.setBounds(550,320,152,26);  
eadd.setBounds(550,360,180,26);
```

```
next1.setBounds(70,490,110,30);  
search.setBounds(182,490,110,30);
```

```
save.setBounds(294,490,110,30);
delete.setBounds(406,490,110,30);
update.setBounds(518,490,110,30);
exit.setBounds(630,490,110,30);
```

```
menu1.add(menuItemN);
```

```
menu1.add(menuItemB);
menu1.add(menuItemC);
menu1.add(menuItemE);
menuBar.add(menu1);
```

```
menu3.add(menuItemST);
menu3.add(menuItemBA);
menu3.add(menuItemCD);
menuBar.add(menu3);
```

```
menu4.add(menuItemBook);
menu4.add(menuItemDVD);
menu4.add(menuItemI);
menuBar.add(menu4);
```

```
menu2.add(menuItemH);
menu2.add(menuItemA);
menuBar.add(menu2);
```

```
next1.addActionListener(this);
save.addActionListener(this);
search.addActionListener(this);
delete.addActionListener(this);
update.addActionListener(this);
exit.addActionListener(this);
menuItemN.addActionListener(this);
menuItemI.addActionListener(this);
menuItemB.addActionListener(this);
menuItemC.addActionListener(this);
```

```
menuItemH.addActionListener(this);
menuItemA.addActionListener(this);
menuItemE.addActionListener(this);
menuItemST.addActionListener(this);
menuItemBA.addActionListener(this);
menuItemCD.addActionListener(this);
menuItemBook.addActionListener(this);
menuItemDVD.addActionListener(this);
```

```
c.getContentPane().add(next1);
c.getContentPane().add(search);
c.getContentPane().add(save);
c.getContentPane().add(delete);
c.getContentPane().add(exit);
c.getContentPane().add(update);
```

```
c.getContentPane().add(messg);
c.getContentPane().add(lDate);
c.getContentPane().add(ph);
c.getContentPane().add(lname);
c.getContentPane().add(address);
c.getContentPane().add(lid);
c.getContentPane().add(lcity);
c.getContentPane().add(pad);
c.getContentPane().add(leadd);
c.getContentPane().add(lclass);
c.getContentPane().add(lYear);
c.getContentPane().add(lprogress);
```

```
c.getContentPane().add(img);
c.getContentPane().add(colLogo);
c.getContentPane().add(date);
c.getContentPane().add(phno);
c.getContentPane().add(name);
c.getContentPane().add(add);
c.getContentPane().add(city);
c.getContentPane().add(padd);
```

```
        c.getContentPane().add(id);
        c.getContentPane().add(eadd);
        c.getContentPane().add(class1);
        c.getContentPane().add(progress);
        c.getContentPane().add(Year);
        c.getContentPane().add(Line);
```

```
c.setDefaultCloseOperation(DO_NOTHING_ON_CLOSE);
c.setVisible(true);
} //End of Library Method
```

```
public static DefaultTableModel buildTableModel(ResultSet rs) throws
SQLException
{
```

```
    ResultSetMetaData metaData = rs.getMetaData();
```

```
    Vector<String> db = new Vector<String>();
```

```
    int columnCount = metaData.getColumnCount();
    for (int column = 1; column <= columnCount; column++)
    {
        db.addElement(metaData.getColumnName(column));
    }
```

```
    Vector<Vector<Object>> data= new Vector<Vector<Object>>();
```

```
    while(rs.next())
    {
        Vector<Object> vector = new Vector<Object>();
```

```

        for (int columnIndex = 1; columnIndex <= columnCount;
columnIndex++)
        {
            vector.addElement(rs.getObject(columnIndex));

        }
        data.addElement(vector);

    }

    return new DefaultTableModel(data,db);

} //End of DefaultTable Model

```

```

public void actionPerformed(ActionEvent ae)
{

    String str1=(String)ae.getActionCommand();
    Object source = ae.getSource();

    if(source==login)
    {
        String value1=usert.getText();
        String value2=pass.getText();
        try{
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            Connection con =
DriverManager.getConnection("jdbc:odbc:Student");
            Statement st=con.createStatement();
            ResultSet rs=st.executeQuery("select * from login where
username='"+value1+"' and password='"+value2+"'");
            String uname="",passt="";
            if(rs.next())
            {

```

```

        uname=rs.getString("username");
        passt=rs.getString("password");
    }

    if(value1.equals("") && value2.equals(""))
    {
        JOptionPane.showMessageDialog(null,"Enter UserName or
        Password","Error",JOptionPane.ERROR_MESSAGE);
    }

    else if(value1.equals(uname) && value2.equals(passt))
    {

        if(value1.equals("Admin"))
        {
            AccountCreator();
        }
        else
        {
            b.dispose();
            Library();
        }

    }

    else if (!value1.equals(uname) && !value2.equals(pass))
    {
        JOptionPane.showMessageDialog(null,"Wrong User Name or
        Password","Error",JOptionPane.ERROR_MESSAGE);
    }

    con.close();
    st.close();
}
catch(Exception ex)
{

```



```

        JOptionPane.showMessageDialog(this,ex,"EXCEPTION",JOptionPane.
ERROR_MESSAGE);
    }

} //End of Login

if(source==createA)
{

    try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
        Statement st = c.createStatement();

        PreparedStatement ps=c.prepareStatement("Insert into login
values(?,?)");
        ps.setString(1,userA.getText());
        ps.setString(2,passA.getText());

        ps.executeUpdate();

        JOptionPane.showMessageDialog(this,"User Account
Created","SUCCESS",JOptionPane.INFORMATION_MESSAGE);

        c.close();
        st.close();
        d.dispose();
    }
    catch(ClassNotFoundException cnf)
    {

        JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.
ERROR_MESSAGE);
    }
}

```

```

        catch(SQLException sql)
        {
            JOptionPane.showMessageDialog(this,"User Already
Exists","EXCEPTION",JOptionPane.ERROR_MESSAGE);
        }
    }
}

```

```

if(source==menuItemA)
{
    aboutus = new JFrame("About Developer");
    aboutus.setSize(700,525);
    aboutus.getContentPane().setLayout(null);
    ImageIcon design = new ImageIcon("images/about.png");
    JLabel cover = new JLabel(design);
    aboutus.getContentPane().add(cover);
    aboutus.setResizable(false);
    cover.setBounds(0,0,700,525);
    aboutus.setVisible(true);
}

```

}//End of about us

```

if(source==menuItemN || source==next1)
{
    id.setEditable(true);
    phno.setEditable(true);
    name.setEditable(true);
    add.setEditable(true);
    city.setEditable(true);
    padd.setEditable(true);
    eadd.setEditable(true);
    class1.setEditable(true);
    Year.setEditable(true);
    progress.setEditable(true);
    date.setEditable(true);

    id.setText(null);
}

```

```
phno.setText(null);
name.setText(null);
add.setText(null);
city.setText(null);
padd.setText(null);
eadd.setText(null);
class1.setText(null);
Year.setText(null);
progress.setText(null);
date.setText(null);
```

```
}//End of setting student record null
```

```
if(source==menuItem1)
{
```

```
    sbbox=new JFrame("Issue Details");
    sbbox.setSize(800,400);
    sbbox.getContentPane().setLayout(null);
    sbbox.getContentPane().setBackground(new Color(27,35,58));
    sbbox.getContentPane().setForeground(Color.white);
    sbbox.setResizable(false);
```

```
        ImageIcon kim= new
        ImageIcon("images/mainpageissued.png");
        JLabel sb= new JLabel(kim);
```

```
        JLabel bookinfo= new JLabel("Enter Roll No. and Click Display
Record to View Results");
        bookinfo.setForeground(Color.white);
```

```
        JLabel bookde= new JLabel("Book Issue Details");
        bookde.setForeground(Color.white);
```

```
        JLabel cdde= new JLabel("CD/DVD Issue Details");
```

```
cdde.setForeground(Color.white);
```

```
JLabel rno= new JLabel("Roll No.");  
rno.setForeground(Color.white);
```

```
JLabel rncd= new JLabel("Roll No.");  
rncd.setForeground(Color.white);
```

```
rn= new JTextField();  
rn.setForeground(Color.black);  
rn.setBackground(Color.white);
```

```
rnb= new JTextField();  
rnb.setForeground(Color.black);  
rnb.setBackground(Color.white);
```

```
bookinfo.setBounds(250,100,600,30);  
bookde.setBounds(185,155,150,20);  
rno.setBounds(140,185,150,20);  
rn.setBounds(200,185,152,26);
```

```
cdde.setBounds(485,155,150,20);  
rncd.setBounds(450,185,150,20);  
rnb.setBounds(510,185,150,26);
```

```
sb.setBounds(0,0,800,400);
```

```
rec=new JButton("Display Record",new  
ImageIcon("images/disp.png"));  
rec.setToolTipText("Display Book Issue Records");  
rec.setBounds(170,250,180,30);
```

```
recc=new JButton("Display Record",new  
ImageIcon("images/disp.png"));  
recc.setToolTipText("Display CD/DVD Issue Records");  
recc.setBounds(470,250,180,30);
```

```
rec.addActionListener(this);
recc.addActionListener(this);
sbox.getContentPane().add(bookinfo);
sbox.getContentPane().add(bookde);
sbox.getContentPane().add(cdde);
sbox.getContentPane().add(rno);
sbox.getContentPane().add(rncd);
sbox.getContentPane().add(rnb);
sbox.getContentPane().add(rn);
sbox.getContentPane().add(rec);
sbox.getContentPane().add(recc);
sbox.add(sb);
sbox.setVisible(true);
```

```
//End of sbox
```

```
if(source==rec)
{
```

```
    try
    {
```

```
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
        Statement st =
c.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.CONCUR_
READ_ONLY);
        ResultSet rs = st.executeQuery("SELECT BookCode as 'Book
Code',BookName as 'Title of Book' FROM Book where Rollb="+rn.getText());
```

```
        if (!rs.isBeforeFirst())
        {
```

```
        JOptionPane.showMessageDialog(this,"Either No Books Issued  
to this Roll No or Roll No does not Exist","Information  
Message",JOptionPane.ERROR_MESSAGE);
```

```
    }  
    else  
    {
```

```
        JTable table = new JTable(buildTableModel(rs));  
        JOptionPane.showMessageDialog(this,new JScrollPane(table),"Books Issued  
To Roll No."+rn.getText(),JOptionPane.PLAIN_MESSAGE);
```

```
        rn.setText(null);  
        c.close();  
        st.close();  
    }
```

```
    }  
    catch(ClassNotFoundException cnf)  
    {
```

```
        JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.ERRO  
R_MESSAGE);  
    }
```

```
    catch(SQLException sql)  
    {
```

```
        JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.  
ERROR_MESSAGE);  
    }
```

```
    }//End of rec
```

```
    if(source==recc)  
    {
```

```

try
{

    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
    Statement st =
c.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.CONCUR_
READ_ONLY);
    ResultSet rs = st.executeQuery("SELECT CDCode as 'CD DVD
Code',Title as 'Title of CD DVD' FROM CD where Rollc="+rnb.getText());

    if (!rs.isBeforeFirst())
    {
        JOptionPane.showMessageDialog(this,"Either No CD/DVD is
Issued to this Roll No or Roll No does not Exist","Information
Message",JOptionPane.ERROR_MESSAGE);
    }
    else
    {

        JTable table = new JTable(buildTableModel(rs));
        JOptionPane.showMessageDialog(this,new JScrollPane(table),"CD/DVD
Issued To Roll No."+rn.getText(),JOptionPane.PLAIN_MESSAGE);
        rnb.setText(null);
        c.close();
        st.close();
    }
}
catch(ClassNotFoundException cnf)
{

JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.ERRO
R_MESSAGE);

```

```

    }

    catch(SQLException sql)
    {

        JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.
        ERROR_MESSAGE);
    }

} //End of rec

```

```

if(source==menuItemB)
{
    book=new JFrame("Book Record");
    book.setSize(650,580);
    book.getContentPane().setLayout(null);
    book.getContentPane().setBackground(new Color(27,35,58));
    book.getContentPane().setForeground(Color.white);
    book.setResizable(false);

    JLabel messg2= new JLabel("Enter Book Code and Click Search
to View Book Record");
    messg2.setForeground(Color.white);
    book.getContentPane().add(messg2);

    JLabel BookName = new JLabel("Book Name");
    BookName.setForeground(Color.white);
    book.getContentPane().add(BookName);
    JLabel AuthorName = new JLabel("Author Name");
    AuthorName.setForeground(Color.white);
    book.getContentPane().add(AuthorName);
    JLabel Pri = new JLabel("Price");
    Pri.setForeground(Color.white);
    book.getContentPane().add(Pri);
}

```



```
JLabel Bcode = new JLabel("Book Code");  
Bcode.setForeground(Color.white);  
book.getContentPane().add(Bcode);  
JLabel issuer1 = new JLabel("Issued To Roll No.");  
issuer1.setForeground(Color.white);  
book.getContentPane().add(issuer1);
```

```
bookTF = new JTextField();  
book.getContentPane().add(bookTF);  
authorTF = new JTextField();  
book.getContentPane().add(authorTF);  
priceTF = new JTextField();  
book.getContentPane().add(priceTF);  
bcodeTF = new JTextField();  
book.getContentPane().add(bcodeTF);  
issuer1TF = new JTextField();  
book.getContentPane().add(issuer1TF);
```

```
ImageIcon logobook = new ImageIcon("images/lib.jpg");  
JLabel logoBook = new JLabel(logobook);  
book.getContentPane().add(logoBook);
```

```
next11=new JButton("New", new  
ImageIcon("images/new.png"));  
next11.setToolTipText("Clears All Fields to Enter New Book  
Record");  
book.getContentPane().add(next11);
```

```
previous11=new JButton("Return", new  
ImageIcon("images/return.png"));  
previous11.setToolTipText("Returns Book to Library");  
book.getContentPane().add(previous11);
```

```
        searBook = new JButton("Search", new
ImageIcon("images/search.png"));
        searBook.setToolTipText("Display Book Details");
        book.getContentPane().add(searBook);

        saveNow = new JButton("Save", new
ImageIcon("images/save.png"));
        saveNow.setToolTipText("Save New Book Records to Library
Database");
        book.getContentPane().add(saveNow);

        update1 = new JButton("Issue", new
ImageIcon("images/update.gif"));
        update1.setToolTipText("Issues Book to Student");
        book.getContentPane().add(update1);

        bexit= new JButton("Exit", new
ImageIcon("images/exit.png"));
        bexit.setToolTipText("Exits Application");
        book.getContentPane().add(bexit);

        logoBook.setBounds(0,0,650,140);
        messg2.setBounds(100,150,600,20);
        Bcode.setBounds(33,200,112,30);
        BookName.setBounds(33,250,112,30);
        AuthorName.setBounds(30,310,112,30);
        Pri.setBounds(31,370,112,30);
        issuer1.setBounds(30,430,112,30);

        bcodeTF.setBounds(123,200,150,26);
        bookTF.setBounds(123,255,150,26);
        authorTF.setBounds(123,315,150,26);
        priceTF.setBounds(123,375,150,26);
        issuer1TF.setBounds(140,435,150,26);
```

```
next11.setBounds(30,500,92,30);
searBook.setBounds(130,500,95,30);
saveNow.setBounds(230,500,92,30);
update1.setBounds(330,500,95,30);
previous11.setBounds(430,500,95,30);
bexit.setBounds(530,500,92,30);
```

```
saveNow.addActionListener(this);
update1.addActionListener(this);
searBook.addActionListener(this);
bexit.addActionListener(this);
next11.addActionListener(this);
previous11.addActionListener(this);
book.setVisible(true);
```

```
}//End of Book Frame
```

```
if(source==next11)
{
bcodeTF.setEditable(true);
bookTF.setEditable(true);
authorTF.setEditable(true);
priceTF.setEditable(true);
issuer1TF.setEditable(true);
```

```
bcodeTF.setText(null);
bookTF.setText(null);
authorTF.setText(null);
priceTF.setText(null);
issuer1TF.setText(null);
```

```
}
```

```

if(source==previous11)
{

try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
        Statement st = c.createStatement();

        PreparedStatement ps=c.prepareStatement("Update Book set
Rollb=? where BookCode="+bcodeTF.getText());

        ps.setString(1,"0");
        ps.executeUpdate();

        JOptionPane.showMessageDialog(this,"Book Returned To
Library Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);

        c.close();
        st.close();
    }
    catch(ClassNotFoundException cnf)
    {

JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.ERRO
R_MESSAGE);
    }
    catch(SQLException sql)
    {

JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.ERRO
R_MESSAGE);
    }

}
}

```

```
if(source==menuItemE || source==cdexit || source==bexit || source==exit)
{
```

```
    int reply = JOptionPane.showConfirmDialog(null," Do You Really  
Want to Quit ?", "Quit", JOptionPane.YES_NO_OPTION);  
    if (reply == JOptionPane.YES_OPTION)  
        System.exit(0);
```

```
}//End of Exit from Menu
```

```
    if(source==saveNow)
    {
        try
        {
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");  
            Connection c =  
DriverManager.getConnection("jdbc:odbc:Student");  
            Statement st = c.createStatement();  
  
            PreparedStatement ps=c.prepareStatement("Insert into Book  
values(?,?,?,?)");  
            ps.setString(1,bcodeTF.getText());  
            ps.setString(2,bookTF.getText());  
            ps.setString(3,authorTF.getText());  
            ps.setString(4,priceTF.getText());  
            ps.setString(5,"0");  
            ps.executeUpdate();  
  
            JOptionPane.showMessageDialog(this,"Record Inserted  
Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);  
  
            c.close();  
            st.close();  
        }  
        catch(ClassNotFoundException cnf)
```

```

    {

        JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.
        ERROR_MESSAGE);
    }
    catch(SQLException sql)
    {
        JOptionPane.showMessageDialog(this,"Record Already
        Exists","EXCEPTION",JOptionPane.ERROR_MESSAGE);
    }
} //End of Saving New Book

if(source==update1)
{
    try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection c =
        DriverManager.getConnection("jdbc:odbc:Student");
        Statement st = c.createStatement();

        PreparedStatement ps=c.prepareStatement("Update Book set
        BookName=?,AuthorName=? , Price=?, Rollb=? where
        BookCode="+bcodeTF.getText());

        ps.setString(1,bookTF.getText());
        ps.setString(2,authorTF.getText());
        ps.setString(3,priceTF.getText());
        ps.setString(4,issuer1TF.getText());
        ps.executeUpdate();

        JOptionPane.showMessageDialog(this,"Record Updated
        Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);

        c.close();
        st.close();
    }
}

```

```

        catch(ClassNotFoundException cnf)
        {

JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.ERROR_
MESSAGE);
        }
        catch(SQLException sql)
        {

JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.ERROR_
MESSAGE);
        }

    } //End of Update Book

```

```

if(source==menuItemC)
{
    cd=new JFrame("CD/DVDs Record");
    cd.setSize(620,510);
    cd.getContentPane().setLayout(null);
    cd.getContentPane().setBackground(new Color(27,35,58));

    cd.setResizable(false);

    ImageIcon backg = new ImageIcon("images/cdlabel.png");
    JLabel cdlogo = new JLabel(backg);
    cd.getContentPane().add(cdlogo);

    JLabel cdinfo=new JLabel("Enter CD/DVD Code and Click
Search to View CD/DVD Record");cdinfo.setForeground(Color.white);
    cdcode = new JLabel("CD/DVD Code");
    cdcode.setForeground(Color.white);
    cdtitle = new JLabel("Title");
    cdtitle.setForeground(Color.white);
    cdisbn = new JLabel("ISBN NO"); cdisbn
.setForeground(Color.white);

```

```
        cdedit = new JLabel("Edition"); cdedit
.setForeground(Color.white);
        cdpub = new JLabel("Publication");cdpub
.setForeground(Color.white);
        JLabel issuer2 = new JLabel("Issued To Roll No.");
        issuer2.setForeground(Color.white);
```

```
        cdcodeTF = new JTextField();
        cdtitleTF = new JTextField();
        cdisbnTF = new JTextField();
        cdeditionTF = new JTextField();
        cdpubTF = new JTextField();
        issuer2TF = new JTextField();
```

```
        previous22 = new JButton("New" , new
ImageIcon("images/new.png"));
        previous22.setToolTipText("Clears All Fields to Enter New
CD/DVD Record");
        cdsearch = new JButton("Search" , new
ImageIcon("images/search.png"));
        cdsearch.setToolTipText("Display CD/DVD Details");
        cdsave = new JButton("Save", new
ImageIcon("images/save.png"));
        cdsave.setToolTipText("Saves New CD/DVD Record to Library
Database");
        cddelete = new JButton("Return", new
ImageIcon("images/return.png"));
        cddelete.setToolTipText("Returns CD/DVD to Library");
        cdupdate = new JButton("Issue", new
ImageIcon("images/update.gif"));
        cdupdate.setToolTipText("Issues CD/DVD to Student");
        cdexit = new JButton("Exit", new
ImageIcon("images/exit.png"));
        cdexit.setToolTipText("Exits the Application");
        cd.getContentPane().add(cdinfo);
```



```
cd.getContentPane().add(cdcode);  
cd.getContentPane().add(cdttitle);  
cd.getContentPane().add(cdisbn);  
cd.getContentPane().add(cdedit);  
cd.getContentPane().add(cdpub);  
cd.getContentPane().add(issuer2);
```

```
cd.getContentPane().add(cdcodeTF);  
cd.getContentPane().add(cdttitleTF);  
cd.getContentPane().add(cdisbnTF);  
cd.getContentPane().add(cdeditionTF);  
cd.getContentPane().add(cdpubTF);  
cd.getContentPane().add(issuer2TF);
```

```
cd.getContentPane().add(previous22);  
cd.getContentPane().add(cdsearch);  
cd.getContentPane().add(cdsave);  
cd.getContentPane().add(cddelete);  
cd.getContentPane().add(cdupdate);  
cd.getContentPane().add(cdexit);
```

```
cdlogo.setBounds(0,0,650,140);  
cdinfo.setBounds(100,145,600,30);  
cdcode.setBounds(60,180,112,30);  
cdtitle.setBounds(60,220,112,30);  
cdisbn.setBounds(60,260,112,30);  
cdedit.setBounds(60,300,112,30);  
cdpub.setBounds(60,340,112,30);  
issuer2.setBounds(60,380,112,30);
```

```
cdcodeTF.setBounds(175,180,190,28);  
cdtitleTF.setBounds(175,220,190,28);  
cdisbnTF.setBounds(175,260,190,28);  
cdeditionTF.setBounds(175,300,190,28);  
cdpubTF.setBounds(175,340,190,28);
```

```
        issuer2TF.setBounds(175,380,190,28);

        previous22.setBounds(10,430,98,30);
        cdsearch.setBounds(110,430,98,30);
        cdsave.setBounds(210,430,98,30);
        cddelete.setBounds(310,430,98,30);
        cdupdate.setBounds(410,430,98,30);
        cdexit.setBounds(510,430,98,30);

        previous22.addActionListener(this);
        cdsearch.addActionListener(this);
        cdsave.addActionListener(this);
        cddelete.addActionListener(this);
        cdexit.addActionListener(this);
        cdupdate.addActionListener(this);
        cd.setVisible(true);
    } //End of CD Frame
```

```
    if(source==previous22)
    {

        cdcodeTF.setEditable(true);
        cdtitleTF.setEditable(true);
        cdisbnTF.setEditable(true);
        cdeditionTF.setEditable(true);
        cdpubTF.setEditable(true);
        issuer2TF.setEditable(true);
```

```
        cdcodeTF.setText(null);
        cdtitleTF.setText(null);
        cdisbnTF.setText(null);
        cdeditionTF.setText(null);
        cdpubTF.setText(null);
        issuer2TF.setText(null);
```

```
}
```

```
if(source==cdsearch)
{
    try
    {

        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
        Statement st =
c.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.CONCUR_
READ_ONLY);
        ResultSet rs = st.executeQuery("select * from CD where
CdCode="+cdcodeTF.getText());

        if(!rs.isBeforeFirst())
        {
            JOptionPane.showMessageDialog(this,"Please Check the
CD/DVD Code You Entered","Information
Message",JOptionPane.ERROR_MESSAGE);
        }

        else
        {
            while(rs.next())
            {

                cdtitleTF.setText(rs.getString("Title"));
                cdisbnTF.setText(rs.getString("IsbnNO"));
                cdeditionTF.setText(rs.getString("Edition"));
                cdpubTF.setText(rs.getString("Publication"));
                issuer2TF.setText(rs.getString("Rollc"));
            }
        }
    }
}
```

```

        }
        }
        c.close();
        st.close();

    }
    catch(ClassNotFoundException cnf)
    {

OptionPane.showMessageDialog(this,cnf,"EXCEPTION",OptionPane.ERRO
R_MESSAGE);
    }

    catch(SQLException sql)
    {

OptionPane.showMessageDialog(this,sql,"EXCEPTION",OptionPane.
ERROR_MESSAGE);
    }

} //End of CD Search

if(source==cdsave)
{
    try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
        Statement st = c.createStatement();
        PreparedStatement ps = c.prepareStatement("Insert into CD
values(?,?,?, ?, ?, ?)");

        ps.setString(1,cdcodeTF.getText());
        ps.setString(2,cdtitleTF.getText());
        ps.setString(3,cdisbnTF.getText());
        ps.setString(4,cdeditionTF.getText());
    }
    catch (Exception e)
    {
        JOptionPane.showMessageDialog(this,e,"EXCEPTION",JOptionPane.ERROR_MESSAGE);
    }
}
}

```

```
ps.setString(5,cdpubTF.getText());
ps.setString(6,issuer2TF.getText());
ps.executeUpdate();
```

```
JOptionPane.showMessageDialog(this,"Record Inserted
Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);
```

```
        c.close();
        st.close();
    }
    catch(ClassNotFoundException cnf)
    {
```

```
        JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.
ERROR_MESSAGE);
    }
```

```
    catch(SQLException sql)
    {
        JOptionPane.showMessageDialog(this,"Record Already
Exists","EXCEPTION",JOptionPane.ERROR_MESSAGE);
    }
```

```
    }//End of CD Save
```

```
if(source==cddelete)
{
```

```
    try
    {
```

```
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection c =
```

```
DriverManager.getConnection("jdbc:odbc:Student");
        Statement st = c.createStatement();
```

```
        PreparedStatement ps=c.prepareStatement("Update CD set
Rollc=? where CDCode="+cdcodeTF.getText());
```

```
        ps.setString(1,"0");
```

```

        ps.executeUpdate();

        JOptionPane.showMessageDialog(this,"CD/DVD Returned To
Library Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);

        c.close();
        st.close();
    }
    catch(ClassNotFoundException cnf)
    {

JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.ERRO
R_MESSAGE);
    }
    catch(SQLException sql)
    {

JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.ERRO
R_MESSAGE);
    }

    }

    if(source==cdupdate)
    {
        try
        {

            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
            Statement st = c.createStatement();
            PreparedStatement ps=c.prepareStatement ("Update CD set
Title=?,IsbnNO=?,Edition=?,Publication=?,Rollc=? where
CdCode="+cdcodeTF.getText());

```

```

        ps.setString(1,cdtitleTF.getText());
        ps.setString(2,cdisbnTF.getText());
        ps.setString(3,cdeditionTF.getText());
        ps.setString(4,cdpubTF.getText());
        ps.setString(5,issuer2TF.getText());

        ps.executeUpdate();
        JOptionPane.showMessageDialog(this,"Record Updated
        Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);

        c.close();
        st.close();
    }
    catch(ClassNotFoundException cnf)
    {
        System.out.println("Cnf Exception");
    }
    catch(SQLException sql)
    {

        JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.
        ERROR_MESSAGE);
    }

}

} //End of CD Update


if(source==menuItemH)
{
    try
    {
        Desktop.getDesktop().open(new
        File("help/index.html"));
    }
    catch(Exception e)
    {

```

```
        JOptionPane.showMessageDialog(this,e,"EXCEPTION",JOptionPane.E  
RROR_MESSAGE);  
    }
```

```
    }//End of Help File
```

```
if(source==menuItemST)  
{
```

```
    try
```

```
    {
```

```
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
```

```
        Connection c =
```

```
DriverManager.getConnection("jdbc:odbc:Student");
```

```
        Statement st = c.createStatement();
```

```
        ResultSet rs = st.executeQuery("select Roll as 'Roll No',Name as  
'Student Name' from Directory order by roll asc");
```

```
        JTable table = new JTable(buildTableModel(rs));
```

```
JOptionPane.showMessageDialog(this,new JScrollPane(table),"Student  
Details",JOptionPane.PLAIN_MESSAGE);
```

```
        c.close();
```

```
        st.close();
```

```
    }
```

```
    catch(ClassNotFoundException cnf)
```

```
    {
```

```
JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.ERRO  
R_MESSAGE);
```

```
    }
```

```
    catch(SQLException sql)
```

```
    {
```



```
        JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.  
ERROR_MESSAGE);  
    }
```

```
}
```

```
if(source==menuItemBA)  
{
```

```
    try
```

```
    {
```

```
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
```

```
        Connection c =
```

```
DriverManager.getConnection("jdbc:odbc:Student");
```

```
        Statement st = c.createStatement();
```

```
        ResultSet rs = st.executeQuery("select BookCode as 'Book  
Code',BookName as 'Book Title' from Book");
```

```
        JTable table = new JTable(buildTableModel(rs));
```

```
JOptionPane.showMessageDialog(this,new JScrollPane(table),"Book  
Details",JOptionPane.PLAIN_MESSAGE);
```

```
        c.close();
```

```
        st.close();
```

```
    }
```

```
    catch(ClassNotFoundException cnf)
```

```
    {
```

```
JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.ERRO  
R_MESSAGE);
```

```
    }
```

```
    catch(SQLException sql)
```

```

        {

            JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.
ERROR_MESSAGE);
        }

    }

    if(source==menuItemCD)
    {

        try
        {

            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
            Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
            Statement st = c.createStatement();
            ResultSet rs = st.executeQuery("select CdCode as 'CD/DVD
Code',Title as 'CD/DVD Title' from CD");

            JTable table = new JTable(buildTableModel(rs));
            JOptionPane.showMessageDialog(this,new JScrollPane(table),"CDs/DVDs in
Library",JOptionPane.PLAIN_MESSAGE);

            c.close();
            st.close();

        }
        catch(ClassNotFoundException cnf)
        {

            JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.ERRO
R_MESSAGE);
        }
    }

```

```

        catch(SQLException sql)
        {

            JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.
            ERROR_MESSAGE);
        }

    }

    if(source==menuItemBook)
    {
        bbox=new JFrame("Issue/Return Book");
        bbox.setSize(800,450);
        bbox.getContentPane().setLayout(null);
        bbox.getContentPane().setBackground(new Color(27,35,58));
        bbox.getContentPane().setForeground(Color.white);
        bbox.setResizable(false);

        ImageIcon kiml= new ImageIcon("images/mainpagebook.jpg");
        JLabel bb= new JLabel(kiml);

        JLabel IssRetb= new JLabel("Enter Book Code of Book to be
        Issued and the Roll No. of Student to be Issued to");
        IssRetb.setForeground(Color.white);

        JLabel bc= new JLabel("Enter Book Code");
        bc.setForeground(Color.white);

        JLabel rl= new JLabel("Enter Roll No.");
        rl.setForeground(Color.white);

        bco= new JTextField();
        bco.setForeground(Color.black);
    }

```

```

        bco.setBackground(Color.white);

        rlo= new JTextField();
        rlo.setForeground(Color.black);
        rlo.setBackground(Color.white);

        IssRetb.setBounds(160,100,700,30);
        bc.setBounds(250,185,150,20);
        bco.setBounds(350,185,152,26);
        rl.setBounds(250,220,150,20);
        rlo.setBounds(350,220,152,26);

        bb.setBounds(0,0,800,450);

        iss=new JButton("Issue",new ImageIcon("images/update.gif"));
        iss.setBounds(320,260,150,30);
        iss.setToolTipText("Issues Book to Student");

        retb=new JButton("Return",new ImageIcon("images/return.png"));
        retb.setBounds(320,295,150,30);
        retb.setToolTipText("Return Book to Library");
        retb.addActionListener(this);
        iss.addActionListener(this);
        bbox.getContentPane().add(IssRetb);
        bbox.getContentPane().add(bc);
        bbox.getContentPane().add(bco);
        bbox.getContentPane().add(rl);
        bbox.getContentPane().add(rlo);
        bbox.getContentPane().add(iss);
        bbox.getContentPane().add(retb);
        bbox.add(bb);
        bbox.setVisible(true);

    }

    if(source==iss)

```

```

{
try
{
Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
Connection c = DriverManager.getConnection("jdbc:odbc:Student");
Statement st = c.createStatement();
PreparedStatement pb=c.prepareStatement ("Update book set Rollb=?
where BookCode="+bco.getText());
pb.setString(1,rlo.getText());
pb.executeUpdate();

JOptionPane.showMessageDialog(this,"Book Issued From Library
Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);

        c.close();
        st.close();
    }
    catch(ClassNotFoundException cnf)
    {

        JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.
ERROR_MESSAGE);
    }
    catch(SQLException sql)
    {

        JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.
ERROR_MESSAGE);
    }

}

if(source==retb)
{

```

```

try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
        Statement st = c.createStatement();

        PreparedStatement ps=c.prepareStatement("Update Book set
Rollb=? where BookCode="+bco.getText());

        ps.setString(1,"0");
        ps.executeUpdate();

        JOptionPane.showMessageDialog(this,"Book Returned To
Library Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);

        c.close();
        st.close();
    }
    catch(ClassNotFoundException cnf)
    {

JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.ERRO
R_MESSAGE);
    }
    catch(SQLException sql)
    {

JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.ERRO
R_MESSAGE);
    }

}

```

```
if(source==menuItemDVD)
{
    dbox=new JFrame("Issue/Return CD/DVD");
    dbox.setSize(800,450);
    dbox.getContentPane().setLayout(null);
    dbox.getContentPane().setBackground(new Color(27,35,58));
    dbox.getContentPane().setForeground(Color.white);
    dbox.setResizable(false);

    ImageIcon kimj= new ImageIcon("images/mainpagecd.jpg");
    JLabel dbb= new JLabel(kimj);

    JLabel dvdinfo= new JLabel("Enter CD/DVD Code of CD/DVD to
be Issued and the Roll No. of Student to be Issued to");
    dvdinfo.setForeground(Color.white);

    JLabel dvd= new JLabel("Enter CD/DVD Code");
    dvd.setForeground(Color.white);

    JLabel rld= new JLabel("Enter Roll No.");
    rld.setForeground(Color.white);

    dvdt= new JTextField();
    dvdt.setForeground(Color.black);
    dvdt.setBackground(Color.white);

    drl= new JTextField();
    drl.setForeground(Color.black);
    drl.setBackground(Color.white);

    dvdinfo.setBounds(160,100,700,30);
    dvd.setBounds(240,185,150,20);
    dvdt.setBounds(350,185,152,26);
```

```

        rld.setBounds(260,220,150,20);
        drl.setBounds(350,220,152,26);

        dbb.setBounds(0,0,800,450);

        issd=new JButton("Issue",new ImageIcon("images/update.gif"));
        issd.setBounds(320,260,150,30);
        issd.setToolTipText("Issues CD/DVD to Student");
        retc=new JButton("Return",new ImageIcon("images/return.png"));
        retc.setBounds(320,295,150,30);
        retc.setToolTipText("Returns CD/DVD to Library");
        issd.addActionListener(this);
        retc.addActionListener(this);
        dbox.getContentPane().add(dvd);
        dbox.getContentPane().add(dvdinfo);
        dbox.getContentPane().add(dvdt);
        dbox.getContentPane().add(rld);
        dbox.getContentPane().add(drl);
        dbox.getContentPane().add(issd);
        dbox.getContentPane().add(retc);
        dbox.add(dbb);
        dbox.setVisible(true);

```

```

}

```

```

if(source==issd)
{
    try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection c = DriverManager.getConnection("jdbc:odbc:Student");
        Statement st = c.createStatement();
        PreparedStatement pb=c.prepareStatement ("Update cd set Rollc=? where
        CdCode="+drl.getText());
    }
}

```



```
pb.setString(1,drl.getText());
pb.executeUpdate();
```

```
JOptionPane.showMessageDialog(this,"CD/DVD issued
Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);
```

```
        c.close();
        st.close();
    }
    catch(ClassNotFoundException cnf)
    {
```

```
        JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.
ERROR_MESSAGE);
    }
    catch(SQLException sql)
    {
```

```
        JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.
ERROR_MESSAGE);
    }
```

```
}
```

```
if(source==retc)
{
```

```
try
{
    Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
    Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
    Statement st = c.createStatement();
```

```
        PreparedStatement ps=c.prepareStatement("Update CD set  
Rollc=? where CDCode="+drl.getText());
```

```
        ps.setString(1,"0");  
        ps.executeUpdate();
```

```
        JOptionPane.showMessageDialog(this,"CD/DVD Returned To  
Library Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);
```

```
        c.close();  
        st.close();  
    }  
    catch(ClassNotFoundException cnf)  
    {
```

```
        JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.ERROR_  
MESSAGE);
```

```
    }  
    catch(SQLException sql)  
    {
```

```
        JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.ERROR_  
MESSAGE);
```

```
    }
```

```
}
```

```
if(source==searBook)
```

```
{
```

```
    try  
    {
```

```
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
```

```
        Connection c =
```

```
DriverManager.getConnection("jdbc:odbc:Student");
```

```
        Statement st =
```

```
c.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.CONCUR_  
READ_ONLY);
```

```
        ResultSet rs =st.executeQuery("select  
BookCode,BookName,AuthorName,Price,Rollb from book where  
BookCode="+bcodeTF.getText());
```

```
        if(!rs.isBeforeFirst())  
        {  
            JOptionPane.showMessageDialog(this,"Please Check the  
Book Code You Entered","Information  
Message",JOptionPane.ERROR_MESSAGE);  
        }
```

```
        else  
        {  
            while(rs.next())  
            {  
                bcodeTF.setText(rs.getString("BookCode"));  
                bookTF.setText(rs.getString("BookName"));  
                authorTF.setText(rs.getString("AuthorName"));  
                priceTF.setText(rs.getString("Price"));  
                issuer1TF.setText(rs.getString("Rollb"));  
            }  
        }  
        c.close();  
        st.close();  
    }  
    catch(ClassNotFoundException cnf)  
    {
```

```
JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.ERRO  
R_MESSAGE);
```

```
    }  
    catch(SQLException sql)  
    {
```

```
JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.ERRO  
R_MESSAGE);
```

```

    }

} //End of Book Search

if(source==search)
{
    try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
        Statement st =
c.createStatement(ResultSet.TYPE_SCROLL_SENSITIVE,ResultSet.CONCUR_
READ_ONLY);
        ResultSet rs = st.executeQuery("select * from Directory where
Roll="+id.getText());

        if(!rs.isBeforeFirst())
        {
            JOptionPane.showMessageDialog(this,"Please Check the
Roll No You Entered","Information
Message",JOptionPane.ERROR_MESSAGE);
        }
        else
        {
            while(rs.next())
            {

                date.setText(rs.getString("DateToday"));
                name.setText(rs.getString("Name"));
                add.setText(rs.getString("Address"));
                class1.setText(rs.getString("Class"));
                progress.setText(rs.getString("Progress"));
                phno.setText(rs.getString("PhoneNo"));
                Year.setText(rs.getString("Year"));
                city.setText(rs.getString("City"));
                padd.setText(rs.getString("Permanent_Address"));
            }
        }
    }
}

```

```

        eadd.setText(rs.getString("eaddress"));
    }

    }
    c.close();
    st.close();
}
catch(ClassNotFoundException cnf)
{

    JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.
ERROR_MESSAGE);
}
catch(SQLException sql)
{

    JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.
ERROR_MESSAGE);
}

} //End of Student Record Search

if(source==save)
{
    try
    {
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
        Statement st = c.createStatement();
        PreparedStatement ps=c.prepareStatement("Insert into Directory
values(?,?,?,?,?,?,?,?,?)");

        ps.setString(1,date.getText());
        ps.setString(2,id.getText());
        ps.setString(3,name.getText());
        ps.setString(4,class1.getText());

```

```
ps.setString(5,Year.getText());
ps.setString(6,progress.getText());
ps.setString(7,add.getText());
ps.setString(8,city.getText());
ps.setString(9,padd.getText());
ps.setString(10,phno.getText());
ps.setString(11,eadd.getText());
ps.executeUpdate();
```

```
JOptionPane.showMessageDialog(this,"Record Inserted
Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);
```

```
        c.close();
        st.close();
    }
    catch(ClassNotFoundException cnf)
    {
```

```
        JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.
ERROR_MESSAGE);
    }
```

```
    catch(SQLException sql)
    {
```

```
        JOptionPane.showMessageDialog(this,"Record Already
Exists","EXCEPTION",JOptionPane.ERROR_MESSAGE);
    }
```

```
    }//End of Saving Student Record
```

```
if(source==delete)
{
```

```
    try
    {
```

```
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
```

```
        Connection c =
```

```
DriverManager.getConnection("jdbc:odbc:Student");
```

```
        Statement st = c.createStatement();
```

```
PreparedStatement ps=c.prepareStatement("Delete from Directory where  
Roll="+id.getText());
```

```
PreparedStatement pb=c.prepareStatement ("Update book set Rollb=?  
where Rollb="+id.getText());
```

```
PreparedStatement pc=c.prepareStatement ("Update cd set Rollc=? where  
Rollc="+id.getText());
```

```
ps.executeUpdate();
```

```
pb.setString(1,"0");
```

```
pb.executeUpdate();
```

```
pc.setString(1,"0");
```

```
pc.executeUpdate();
```

```
JOptionPane.showMessageDialog(this,"Record Deleted  
Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);
```

```
c.close();
```

```
st.close();
```

```
}
```

```
catch(ClassNotFoundException cnf)
```

```
{
```

```
JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.  
ERROR_MESSAGE);
```

```
}
```

```
catch(SQLException sql)
```

```
{
```

```
JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.  
ERROR_MESSAGE);
```

```
}
```

```
}//End of Delete Student Record
```

```
if(source==update)
```

```
{
```

```
try
```

```
{
```

```
        Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");
        Connection c =
DriverManager.getConnection("jdbc:odbc:Student");
        Statement st = c.createStatement();
        PreparedStatement ps=c.prepareStatement ("Update Directory set
        Name=?, Class=?,Year=?, Progress=?, Address=?, City=?,
        Permanent_Address=?, PhoneNo=?, eaddress=?, DateToday=? where
        Roll="+id.getText());
```

```
        ps.setString(1,name.getText());
        ps.setString(2,class1.getText());
        ps.setString(3,Year.getText());
        ps.setString(4,progress.getText());
        ps.setString(5,add.getText());
        ps.setString(6,city.getText());
        ps.setString(7,padd.getText());
        ps.setString(8,phno.getText());
        ps.setString(9,eadd.getText());
        ps.setString(10,date.getText());
```

```
        ps.executeUpdate();
```

```
JOptionPane.showMessageDialog(this,"Record Updated
Successfully","SUCCESS",JOptionPane.INFORMATION_MESSAGE);
```

```
        c.close();
        st.close();
    }
    catch(ClassNotFoundException cnf)
    {
```

```
        JOptionPane.showMessageDialog(this,cnf,"EXCEPTION",JOptionPane.
        ERROR_MESSAGE);
    }
    catch(SQLException sql)
    {
```



```
        JOptionPane.showMessageDialog(this,sql,"EXCEPTION",JOptionPane.  
ERROR_MESSAGE);  
    }  
    }//End of Update Student Record
```

```
}//End of ActionPerformed
```

```
public static void main(String args[])  
{  
  
    try  
    {  
        for (LookAndFeelInfo info : UIManager.getInstalledLookAndFeels())  
        {  
            if ("Nimbus".equals(info.getName()))  
            {  
                UIManager.setLookAndFeel(info.getClassName());  
                break;  
            }  
        }  
    }  
    catch (Exception e)  
    {  
        System.out.println("Exception :"+e);  
    }  
}
```

```
        InputMap im =  
(InputMap)UIManager.get("Button.focusInputMap");  
        im.put( KeyStroke.getKeyStroke( "ENTER" ), "pressed" );  
        im.put( KeyStroke.getKeyStroke( "released ENTER" ), "released" );  
        ProjectLibrary lib=new ProjectLibrary();
```

```
}//End of Main
```

```
}//End of Class
```