3. Explain the delegation event model. Design a Java front end for a login page using

swing components and on clicking button check user is valid or not (username and

password = syit or SYIT then user is valid give the message as per validation and

allow user to enter user name and password 3 times if it is wrong. )

\*Explain delegation event model in pdf document\*

**PROGRAM**

package login;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPanel;

import javax.swing.JPasswordField;

import javax.swing.JTextField;

public class code implements ActionListener

{

private static JFrame frame;

private static JPanel panel;

private static JLabel userlabel;

private static JTextField usertext;

private static JLabel passlabel;

private static JPasswordField passwordtext;

private static JButton button;

private static JLabel success;

public static void main(String[]args)

{

frame = new JFrame();

panel = new JPanel();

frame.setSize(100,100);

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setVisible(true);

frame.add(panel);

panel.setLayout(null);

userlabel = new JLabel("Username");

userlabel.setBounds(10,20,80,25);

panel.add(userlabel);

usertext = new JTextField();

usertext.setBounds(100,20,165,25);

panel.add(usertext);

passlabel = new JLabel("Password");

passlabel.setBounds(10,50,80,25);

panel.add(passlabel);

passwordtext = new JPasswordField();

passwordtext.setBounds(100,50,165,25);

panel.add(passwordtext);

button = new JButton("Login");

button.setBounds(10,80,80,25);

button.addActionListener(new code());

panel.add(button);

success = new JLabel("");

success.setBounds(10,110,300,25);

panel.add(success);

success.setText("");

frame.setVisible(true);

}

public void actionPerformed(ActionEvent e)

{

String user = usertext.getText();

String password = passwordtext.getText();

if(user.equals("SYIT") && password.contentEquals("SYIT"))

{

success.setText("Login successful!");

}

else

{

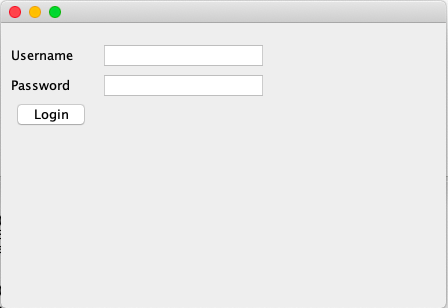
success.setText("Wrong username and password");

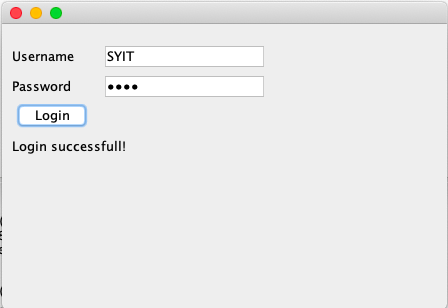
}

}

}

**OUTPUT**





6. Create a RMI application for checking whether a given number is prime or not.

**PROGRAM**

import java.rmi.\*;

public interface Prime extends Remote

{

public int checkprime(int n) throws RemoteException;

}

import java.rmi.\*;

import java.rmi.server.\*;

public class PrimeImply extends UnicastRemoteObject implements Prime

{

public PrimeImply() throws Exception

{

}

public int checkprime(int n) throws RemoteException

{

int i,count=0;

for(i=2;i<=n/2;i++)

if(n%i==0)

{

count++;

break;

}

return (count);

}

}

import java.rmi.\*;

import java.net.\*;

public class Primeserver

{

public static void main(String args[])

{

try

{

PrimeImply primpli=new PrimeImply();

Naming.rebind("RmiPrime",primpli);

}

catch(Exception ex){}

}

}

import java.io.\*;

import java.rmi.\*;

import java.net.\*;

public class Primeclient

{

public static void main(String args[])

{

try

{

String url="rmi://127.0.0.1/RmiPrime";

Prime intf=(Prime)Naming.lookup(url);

BufferedReader br=new BufferedReader(new InputStreamReader(System.in));

System.out.println("Enter a number : ");

int n=Integer.parseInt(br.readLine());

int x=intf.checkprime(n);

if(x==0)

System.out.println(n+" is prime number");

else

System.out.println(n+" is not prime number");

}

catch(Exception ex){}

}

}