DipIT08:-Introduction to programming II

Topic :-Tower of Hanoi and GUI

Student Name:Samir Singh Chhetri

Student Id: NP03S190055

Submitted date: 09 Sep

Submitted to: Raj p.d Shrestha

Title page:-

This project is about the tower of hanoi which is a mathematical problem and programming and GUI to solve the tower oh hanoie problem I have design program and mention the technique of recurssion.

Acknowledgement:-

I would like to thanks Mr Raj p.d Shrestha and Mr Rizzu Rohit Bhandari to help us in completing this timebox project and I would like to thank my friends for providing help.

Contents

6.1. Recursion

Recursion is a process of calling itself by a method during its execution. The corresponding function is called recursive function and method that incorporate recursion are called recursive methods.

6.2. Solving techniques using recursion

To solve a particular problem using recursion following requirements should be met:

1. A recursive program must call itself in its own body.
2. There should be a base case for termination of the program.
3. The program should proceed towards the termination of it.

6.3. Tower of Hanoi

Tower of Hanoi is a mathematical problem where three rods and n disks are observed. The purpose of this problem is to move the entire stack of discs to another rod, obeying the following simple rules:

1) One disc can be moved at a time.
2) Disc cannot be placed in the descending order of size. 3) The uppermost disc can only be moved to another rod.

(GeeksForGeeks, 2019)

6.4. Tower of Hanoi algorithm

```
Start
mt(n, source,  aux , dest)
if n==1 THEN
Move n from Source to dest
ELSE
mt(n-1,  source,  dest, aux)  move n
from source to dest  mt(n-1,  aux,
source, dest)
END IF
STOP
```

let n be the number of discs, source be the source tower, aux be the auxiliary tower, dest be the destination tower.

If we have 2 disks −

- First, we move the smaller (top) disk to aux tower.
- Then, we move the larger (bottom) disk to dest tower.
- And finally, we move the smaller disk from aux to dest tower.

So, if the no of disc is n the first n-1 disc should be moved to aux and the final biggest disc should be moved to destination tower and finally the n-1 disc should be moved from aux to dest tower.

6.5. Tower of Hanoi source code

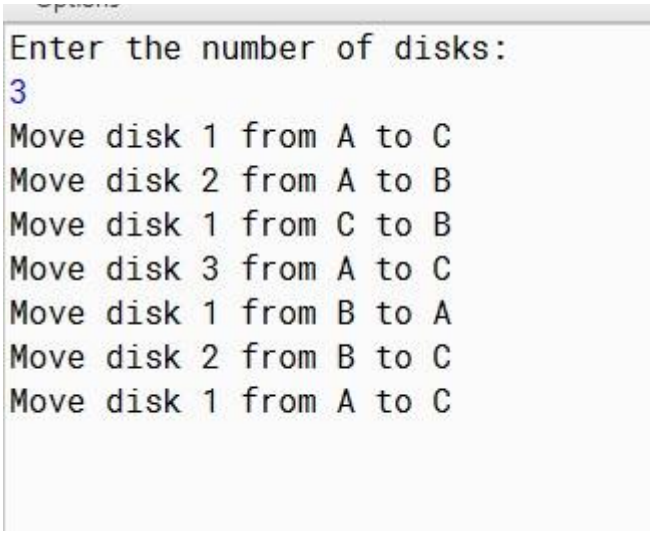
TowerOfHanoi

Compile | Undo | Cut | Copy | Paste | Find... | Close

```
import java.util.Scanner;
public class TowerOfHanoi
{
    public static void tower(int disk, String from,String aux,String to){
        if(disk==1)

        {
            System.out.println("Move disk "+disk+ " from "+from+" to "+to);
        }
        else{
            tower(disk-1,from,to,aux);
            System.out.println("Move disk "+disk+ " from "+from+ " to "+to);
            tower(disk-1,aux,from,to);
        }
    }
```

```
    public static void main(){
        Scanner s = new Scanner(System.in);
        System.out.println("Enter the number of disks: ");
        try{
        int n =s.nextInt();
        if(n<=0 || n>8){
          throw new Exception();
          }
        tower(n,"A","B","C");
    }
        catch(Exception e){
          System.out.println("Please choose valid disk");
          }
    }
}
```

6.6. Testing of Tower of Hanoi with a screenshot



About Event- GUI Driven Programming

The graphical user interface is a type of user interface that enables users to communicate with electronic devices through graphic icons and visual indicators such as secondary notation rather than text user interfaces, command labels or text navigation.

When a user clicks, key presses, mouse movements, drag an drops, etc. on a graphical component user generate an event. In event-driven programming the program responds to these events.

A.About used Java Swing GUI components

Java Swing is a lightweight GUI toolkit that stores a collection of widgets. It contains a package that allows users to create GUI components for Java applications and is independent of the platform.

The Swing library is created on top of the Java Abstract Widget Toolkit (AWT), an older GUI toolkit that depends on the platform. User can use the library's Java GUI components such as button, textbox, label etc. and they don't need to create the parts from scratch.

GUI Components

JFrame: - JFrame is a class of javax.swing package extended by java.awt.frame, it adds support for JFC/SWING component architecture. It is the top-level window, with a title bar and border. JFrame class contains many methods which can be used to modify it.

JButton: - JButton class is used to construct a labeled button that has platform independent implementation. The application result in some action when the button is clicked it inherits Abstract Button class.

JLabel: - JLabel is used to display a short string or an image icon. JLabel can display both text and image. JLabel is only a display of text or image and it cannot get focus. JLabel is passive to input events such as keyboard or mouse focus.

JRadioButton: - Builds a radio button that has the specified image and text, and which is unselected in initial state.

JPanel: - JPanel is a Swing's lightweight container which is applied to group a set of components together. JPanel is a quite simple components which, normally, does not have a GUI (except when it is being set an opaque background or has a visual border)

JTable: - The JTable class is a component of Java Swing Package which is generally used to display or edit two-dimensional information that is having both rows and columns. It is similar to a spreadsheet. It arranges data in a tabular form.

JScrollPane: - JScrollPane is used to make scrollable perspective on a segment. At the point when screen size is limited, we utilize a scroll pane to show a huge segment or a segment whose size can change progressively.

JMenuItem: - The object of JMenu class is a pull-down menu component which is displayed from the menu bar. It inherits the JMenuItem class. The object of JMenuItem class adds a simple labeled menu item. The items used in a menu must belong to the JMenuItem or any of its subclass.

JMenuBar: - JMenuBar class is used to display menu bar on the window or frame. It may have many menus. The object of JMenu class is a pull-down menu component which is displayed from the menu bar. It inherits the JMenuItem class.

JCheckBox: - JCheckBox is a component of Java Swing package. JCheckBox can be selected or deselected. It displays it state to the user. JCheckBox is an implementation to checkbox.

JTextField: - JTextField is a lightweight component which lets users to edit a line of text. JFormattedTextField, JPasswordField are derived classes of JTextField.

GUI Code

QN.1

```
import javax.swing.*;
  import java.awt.*;
  import javax.swing.border.*;
  import java.awt.event.*;
  import javax.swing.table.*;
  import javax.swing.table.DefaultTableModel;
  import java.io.*;
  import java.util.*;
  import javax.swing.UIManager.LookAndFeelInfo;
    public class GUI1 implements ActionListener
{
```

```
JFrame frame;
JTextField tf1,tf2,tf3,tf4;
JPanel pnl1,pnl2,pnl3,pnl4,pnl5;
JButton btn1,btn2,btn3,btn4;
JLabel lbl1,lbl2,lbl3,lbl4;
JTable tbl1;
JScrollPane spl;
JMenuItem clear,search,add,remove,neww,save,save_as,open,tools,about,exit;
JMenu file,edit,help;
JMenuBar menubar;
JCheckBox cb;
JRadioButton rbtn1,rbtn2;
JSeparator sep;
DefaultTableModel dtm;
ButtonGroup bg;


GUI1(){
        frame =new JFrame("Phone Book");
         frame.setSize(600,500);
         frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
         frame.setLayout(new GridLayout(1,2));

         pnl1=new JPanel();
         pnl1.setLayout(new GridLayout(1,1));
         pnl1.setBorder(new TitledBorder("Name"));
         tbl1 = new JTable(0,4);
```

```
  tbl1.getColumnModel().getColumn(0).setHeaderValue("First Name");
  tbl1.getColumnModel().getColumn(1).setHeaderValue("Last Name");
  tbl1.getColumnModel().getColumn(2).setHeaderValue("Ph-number");
  tbl1.getColumnModel().getColumn(3).setHeaderValue("Private");


spl=new JScrollPane(tbl1);
pnl1.add(spl);
frame.add(pnl1);




pnl2=new JPanel();
pnl2.setBorder(new TitledBorder("Info"));
pnl2.setLayout(new GridLayout(3,1));

pnl3=new JPanel();
pnl3.setBorder(new TitledBorder("Name"));
pnl3.setLayout(new GridLayout(4,1));

lbl1=new JLabel();
lbl1.setText("First Name");
pnl3.add(lbl1);
tf1=new JTextField(25);
pnl3.add(tf1);
```

```
lbl2=new JLabel();
lbl2.setText("Second Name");
pnl3.add(lbl2);
tf2=new JTextField(25);
pnl3.add(tf2);

lbl3=new JLabel();
lbl3.setText("Phone");

pnl3.add(lbl3);
tf3=new JTextField(25);
pnl3.add(tf3);
cb= new JCheckBox("Private");
tf1.setToolTipText("Enter your First name");
tf2.setToolTipText("Enter your Last name ");
tf3.setToolTipText("Enter your phone number");
cb.setToolTipText("Select if this is private number");
pnl3.add(cb);
lbl4 = new JLabel();
pnl2.add(pnl3);

pnl4=new JPanel();
pnl4.setBorder(new TitledBorder("File As"));
pnl4.setLayout(new GridLayout(2,1));
rbtn1 =new JRadioButton("Name,Surname");
rbtn1.setToolTipText("Select to display Name first");
rbtn2 =new JRadioButton("Surname,Name");
rbtn2.setToolTipText("Select to display Surname first");
```

```
ButtonGroup bg=new ButtonGroup();
bg.add(rbtn1);
bg.add(rbtn2);
pnl4.add(rbtn1);
pnl4.add(rbtn2);
pnl2.add(pnl4);


pnl5=new JPanel();
pnl5.setLayout(new GridLayout(2,2));
btn1=new JButton("Clear");
btn1.setToolTipText("Press to clear data in text field");
btn1.setMnemonic(KeyEvent.VK_C);
btn1.addActionListener(this);
btn2=new JButton("Search");

btn2.setToolTipText("Press to Search data from table");
btn2.setMnemonic(KeyEvent.VK_Q);
btn2.addActionListener(this);
btn3=new JButton("Add");
btn3.setToolTipText("Press to add data in table");
btn3.setMnemonic(KeyEvent.VK_A);
btn3.addActionListener(this);
btn4=new JButton("Remove");
btn4.setToolTipText("Press to remove data from table");
btn4.setMnemonic(KeyEvent.VK_R);
btn4.addActionListener(this);
pnl5.add(btn1);
```

```
pnl5.add(btn2);
pnl5.add(btn3);
pnl5.add(btn4);

pnl2.add(pnl5);
frame.add(pnl2);



neww = new JMenuItem("New");
neww.setToolTipText("Create New file");
open = new JMenuItem("Open");
open.setToolTipText("Open file");
clear = new JMenuItem("Close");
clear.setToolTipText("Press to clear text");
add = new JMenuItem("Add");
add.setToolTipText("Press add to add data");
add.addActionListener(this);
search = new JMenuItem("Search");
search.setToolTipText("Press search to search data");
search.addActionListener(this);
remove = new JMenuItem("Remove");
remove.setToolTipText("Press remove to delete data");
remove.addActionListener(this);
save = new JMenuItem("Save");
remove.setToolTipText("Press to save data");
save.setMnemonic(KeyEvent.VK_S);
save.addActionListener(this);
about = new JMenuItem("About");
```

```
about.setToolTipText("Press to see detail of the program");
about.addActionListener(this);
exit=new JMenuItem("Exit");
exit.setMnemonic(KeyEvent.VK_X);
exit.setToolTipText("Press  to close program");
exit.addActionListener(this);
file = new JMenu("File");
file.setToolTipText("Press to find more features");
file.setMnemonic(KeyEvent.VK_F);
file.add(neww);
file.add(open);
file.add(save);


file.add(exit);
sep=new JSeparator();
edit = new JMenu("Edit");
edit.setToolTipText("Press to edit details");
edit.setMnemonic(KeyEvent.VK_E);
edit.add(clear);
edit.add(search);
edit.add(sep);
edit.add(add);
edit.add(remove);

help = new JMenu("Help");
help.setToolTipText("Press if you need some help from us");
help.setMnemonic(KeyEvent.VK_H);
```

```
        help.add(about);

        menubar=new JMenuBar();
        menubar.add(file);
        menubar.add(edit);
        menubar.add(help);
        frame.setJMenuBar(menubar);

        frame.setVisible(true);


    }
public void actionPerformed(ActionEvent e){

        String tx;
        String tx2;
        String tx3;
        String pri;

        dtm=(DefaultTableModel)tbl1.getModel();

                tx=tf1.getText();
                tx2=tf2.getText();
                tx3=tf3.getText();
                pri="";

      Object[] row=new Object[4];
      if(e.getSource()==btn1){ //Clear data
```

```
            tf1.setText("");
            tf2.setText("");
            tf3.setText("");
            cb.setSelected(false);
          }
        if(e.getSource()==exit){
             System.exit(0);
          }
        if(e.getSource()==btn2 ||e.getSource()==search){ //Search data
           JOptionPane.showMessageDialog(frame,"Search button will  support in the Professional version");
          }
        if(e.getSource()==btn3 || e.getSource()==add){

         try{
            if(tx.isEmpty() ||tx2.isEmpty() || tx3.isEmpty()){
                   JOptionPane.showMessageDialog(frame,"Please Fill all required data.");
               }
               else{

            try{
            for(int i=0;i<tx.length();i++){
               //String check1=data1.substring(i,i+1);
               char ch1 = tx.charAt(i);
               if(((ch1>='a')&&(ch1<='z'))||((ch1>='A')&&(ch1<='Z'))||(ch1==' ')){
                  //System.out.print(ch1);
               }else{
                  //System.out.println(ch1);
                  throw new NumberFormatException();
```

```
    }
  }
    try{
       for(int i=0;i<tx2.length();i++){
    //String check1=data1.substring(i,i+1);
    char ch1 = tx2.charAt(i);
    if(((ch1>='a')&&(ch1<='z'))||((ch1>='A')&&(ch1<='Z'))||(ch1==' ')){
       //System.out.print(ch1);
    }else{
       //System.out.println(ch1);
       throw new NumberFormatException();
    }
  }

 try
       {
          // checking valid integer and double using parseDouble() method
          double d =Double.parseDouble(tx2);
          JOptionPane.showMessageDialog(frame,"Please enter your Last Name ");

       }
       catch(Exception i){
         try{
                    long f=Long.parseLong(tx3);

                  if(cb.isSelected()==true){
                  pri="Private";
                }
```

```
    else{
        pri="Not-Private";
    }
  if(rbtn1.isSelected()){
        rbtn2.setEnabled(false);

    tbl1.getColumnModel().getColumn(0).setHeaderValue("First Name");
    tbl1.getColumnModel().getColumn(1).setHeaderValue("Last Name");
    tbl1.getColumnModel().getColumn(2).setHeaderValue("Ph-number");
    tbl1.getColumnModel().getColumn(3).setHeaderValue("Private");
    frame.repaint();
    row[0]=tx;
    row[1]=tx2;
    row[2]=Long.toString(f);
    row[3]=pri;
    dtm.addRow(row);
}
    else if(rbtn2.isSelected()) {
        rbtn1.setEnabled(false);
        tbl1.getColumnModel().getColumn(0).setHeaderValue("Last Name");
        tbl1.getColumnModel().getColumn(1).setHeaderValue("First Name");
        tbl1.getColumnModel().getColumn(2).setHeaderValue("Ph-number");
        tbl1.getColumnModel().getColumn(3).setHeaderValue("Private");
        frame.repaint();
        row[0]=tx2;
        row[1]=tx;
        row[2]=Long.toString(f);
        row[3]=pri;
```

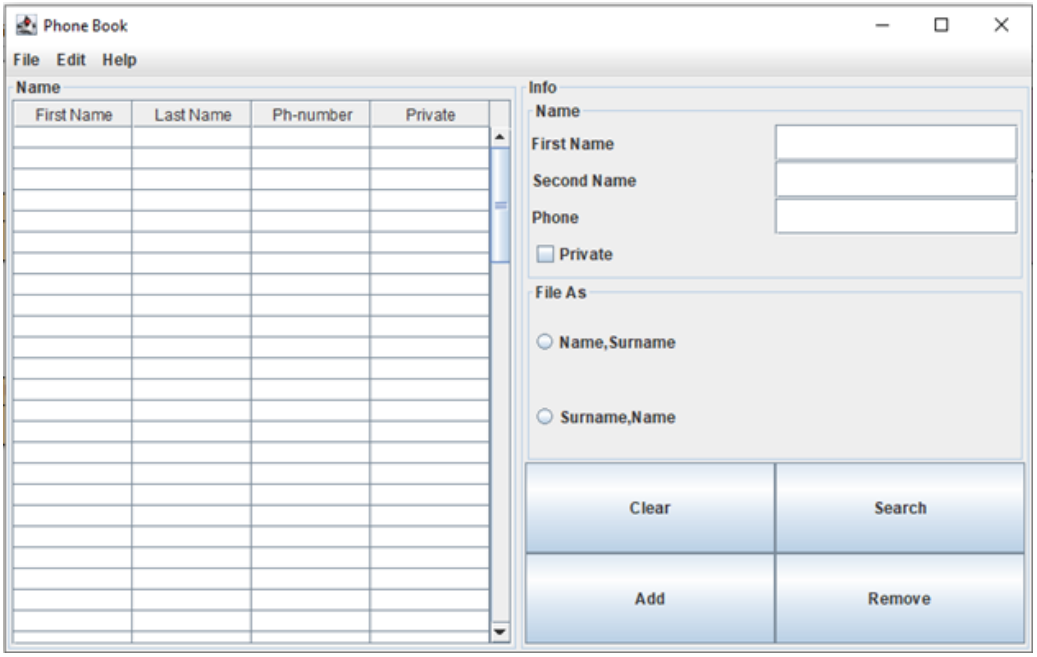
```
                    dtm.addRow(row);
                }

                    else{
                        JOptionPane.showMessageDialog(frame,"Please select File as field ");
                    }
                }
                  catch(Exception j){
                    JOptionPane.showMessageDialog(frame,"Please enter valid Phone number ");
                  }
                }

              }
                catch(Exception h){
                  JOptionPane.showMessageDialog(frame,"Enter your valid Last name");
                }

            }
            catch(Exception g){
            JOptionPane.showMessageDialog(frame,"Enter your valid First name");
            }



            }
           }
           catch(Exception f){
            System.out.println(f);
            }
```

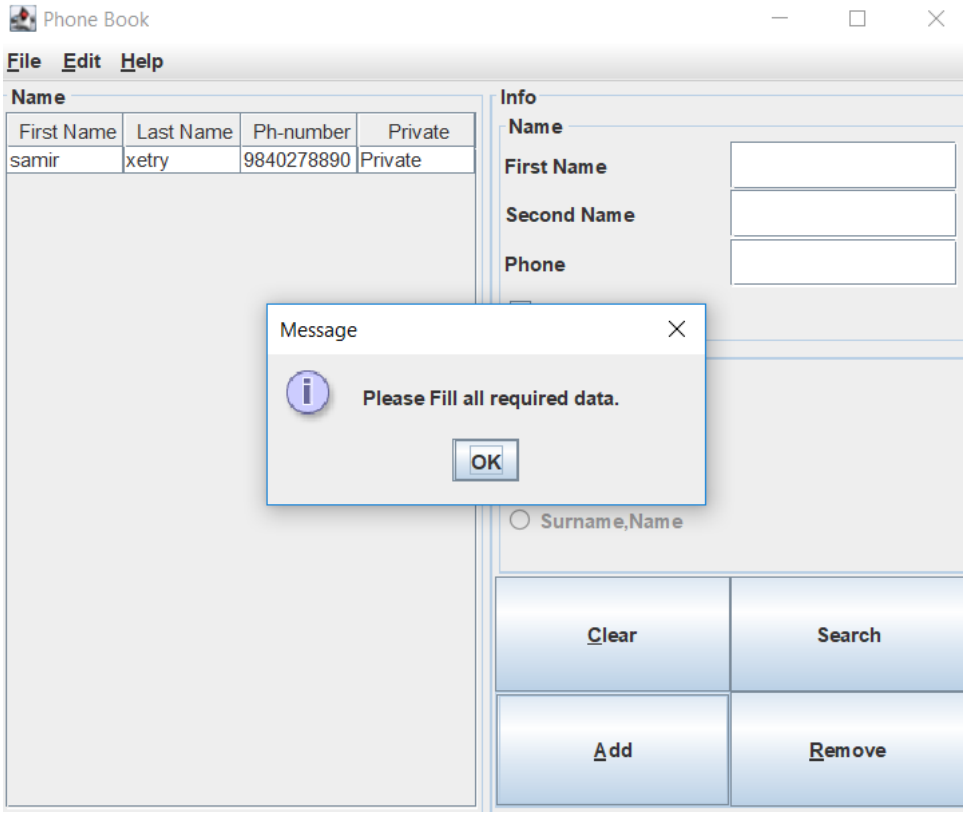
```
}

  if(e.getSource()==save || e.getSource()==save_as){
     try{
        File file=new File("Phonebook.txt");
         if(file.exists()){

            JOptionPane.showMessageDialog(frame," file is already created. ");
         }
         else{
            file.createNewFile();

         }
         FileWriter fw=new FileWriter(file,true);
         String str=tx;
         String str1=tx2;
         String str2=tx3;
         String str4=pri;
         fw.write(str+"  ");
         fw.write(str1+" ");
         fw.write(str2+" ");
         fw.write(pri+" ");
         fw.close();



      }
         catch(Exception y){
```

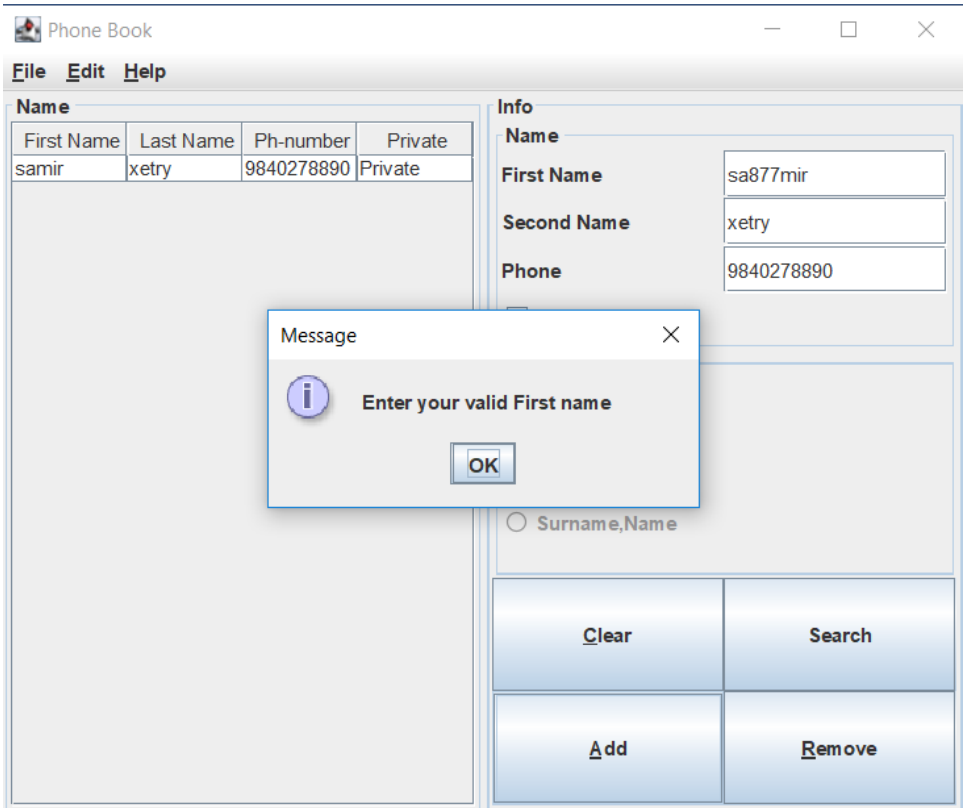
```
            System.out.println(y);
        }

    }



  if(e.getSource()==btn4 ||e.getSource()==remove){ //Remove data
     dtm=(DefaultTableModel)tbl1.getModel();
     int d_row=tbl1.getSelectedRow();
   if(d_row>=0)
   {
     dtm.removeRow(d_row);
   }
   else
   {
      JOptionPane.showMessageDialog(frame," Select Row to remove data");
   }
}

    if(e.getSource()==about){
        try{
          JOptionPane.showMessageDialog(frame,"It is a trial version.");
      }
      catch(Exception h){

      }
   }
```

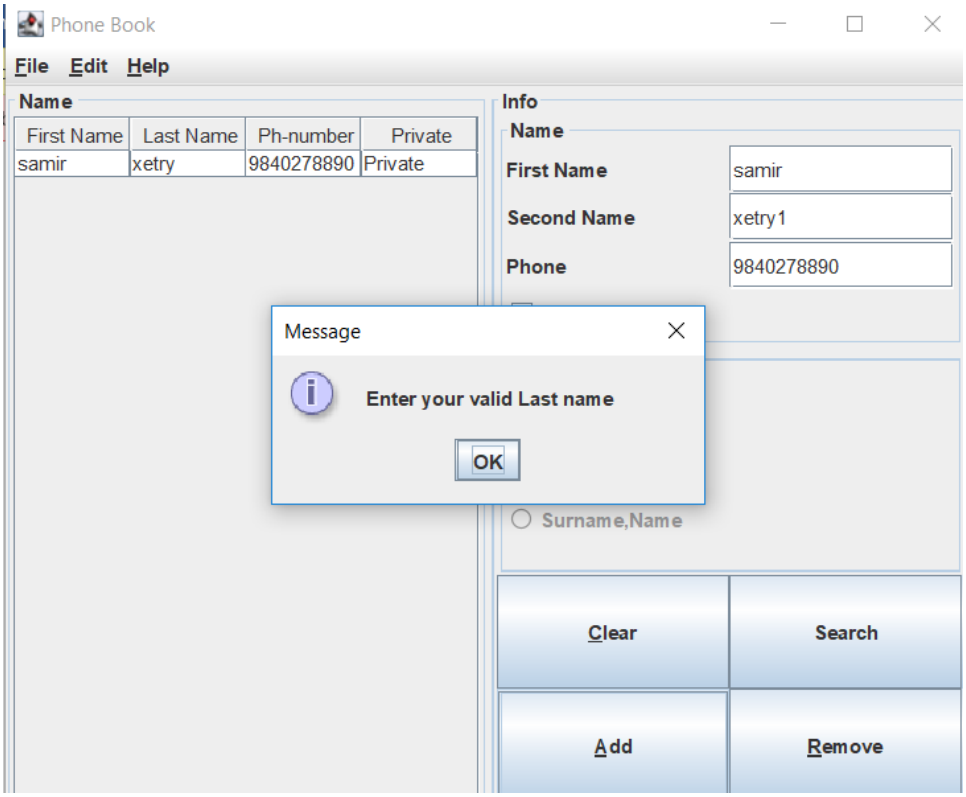
```
    }
    public static void main(String []args){
        new GUI1();
      }
}
```

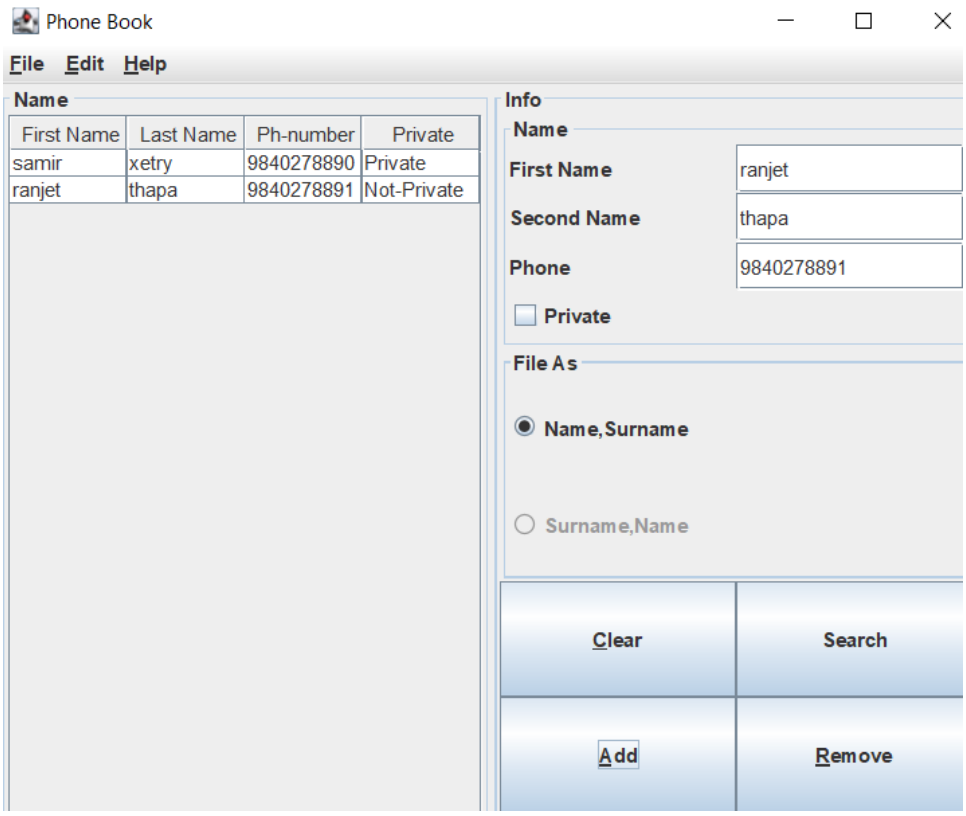
ScreenShot

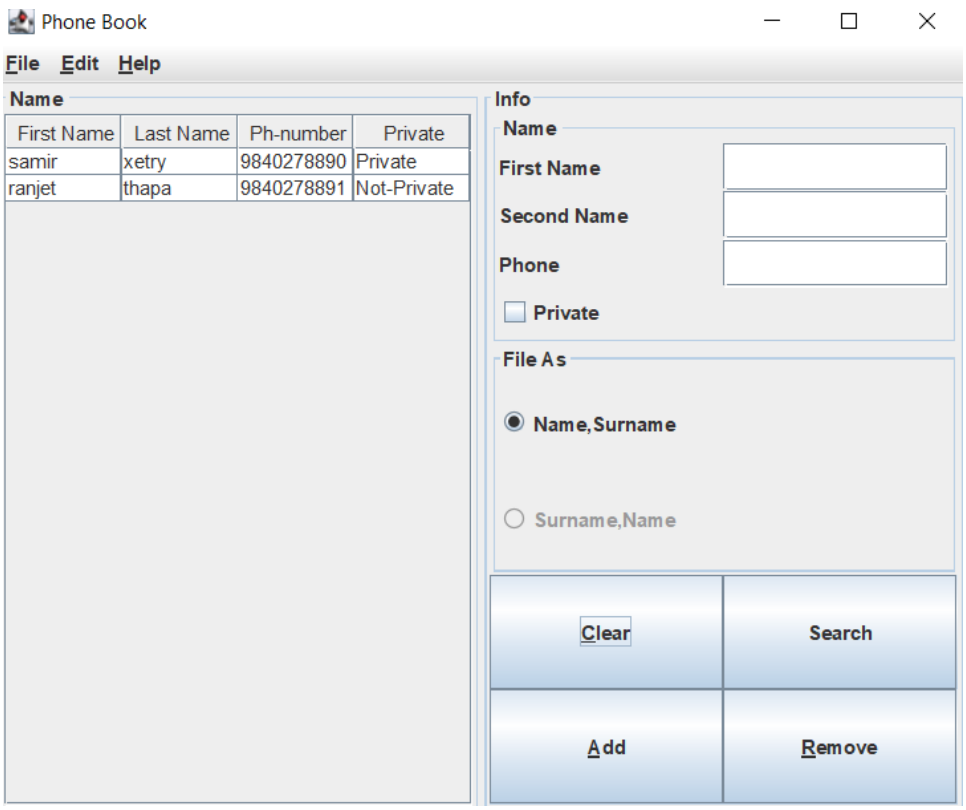


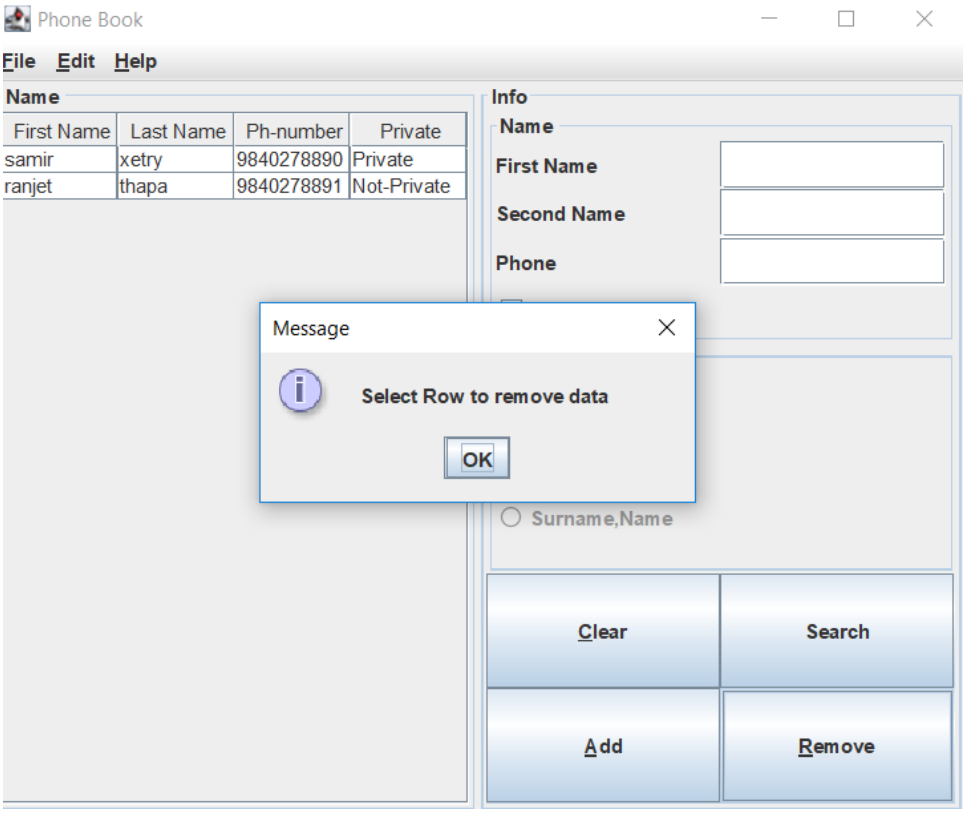

Phone Book
File Edit Help
Name
First Name
Last Name
Ph-number
Private
samir
xetry
9840278890
Private
Info
Name
First Name
Second Name
Phone
Message
Please Fill all required data.
OK
Surname,Name
Clear
Search
Add
Remove

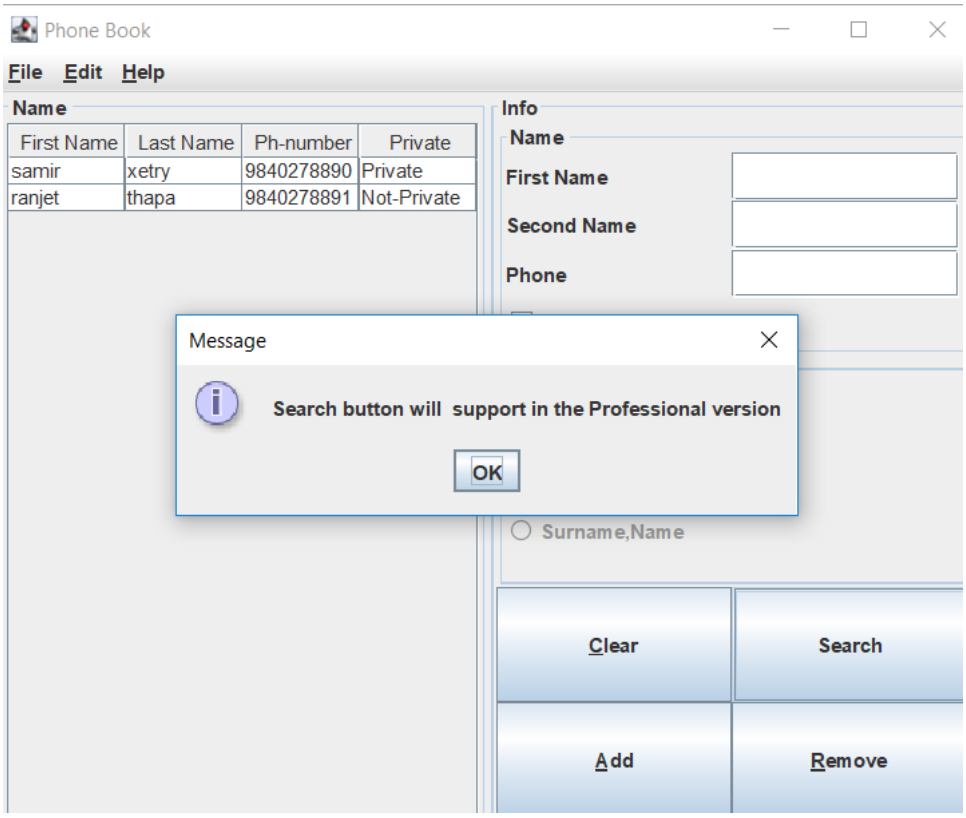

Phone Book
File Edit Help
Name
First Name Last Name Ph-number Private
samir xetry 9840278890 Private
Info
Name
First Name
sa877mir
Second Name
xetry
Phone
9840278890
Message
Enter your valid First name
OK
Surname,Name
Clear
Search
Add
Remove


Phone Book
File Edit Help
Name
First Name Last Name Ph-number Private
samir xetry 9840278890 Private
Info
Name
First Name samir
Second Name xetry1
Phone 9840278890
Message
Enter your valid Last name
OK
Surname,Name
Clear
Search
Add
Remove


Phone Book
File Edit Help
Name
First Name Last Name Ph-number Private
samir xetry 9840278890 Private
ranjet thapa 9840278891 Not-Private
Info
Name
First Name ranjet
Second Name thapa
Phone 9840278891
Private
File As
Name,Surname
Surname,Name
Clear
Search
Add
Remove


Phone Book
File Edit Help
Name
First Name Last Name Ph-number Private
samir xetry 9840278890 Private
ranjet thapa 9840278891 Not-Private
Info
Name
First Name
Second Name
Phone
Private
File As
Name,Surname
Surname,Name
Clear
Search
Add
Remove


Phone Book
File Edit Help
Name
First Name Last Name Ph-number Private
samir xetry 9840278890 Private
ranjet thapa 9840278891 Not-Private
Info
Name
First Name
Second Name
Phone
Message
Select Row to remove data
OK
Surname,Name
Clear
Search
Add
Remove


Phone Book
File Edit Help
Name
First Name Last Name Ph-number Private
samir xetry 9840278890 Private
ranjet thapa 9840278891 Not-Private
Info
Name
First Name
Second Name
Phone
Message
Search button will support in the Professional version
OK
Surname,Name
Clear
Search
Add
Remove