Practical-8

Aim:-Implement an application using Netbeans and Mongodb.

- Add jar files of mongo driver to this project.
- Design a form like given below.

NewFrame.java

Code

```
import com.mongodb.BasicDBObject;

import com.mongodb.BasicDBObjectBuilder;

import com.mongodb.DB;

import com.mongodb.DBCollection;

import com.mongodb.DBCursor;

import com.mongodb.DBObject;

import com.mongodb.MongoClient;



import com.mongodb.client.MongoDatabase;

import com.mongodb.WriteResult;

import com.mongodb.client.MongoCollection;

import javax.swing.JOptionPane;

import javax.swing.table.DefaultTableModel;
```

```
public class NewJFrame extends javax.swing.JFrame {

    /** Creates new form NewJFrame */
    private static DB database;  //verified

    MongoClient dbcon;        //verified
    DBCollection table;       //verified
  public NewJFrame() {
        initComponents();
dbcon=dbclass.getConnection();
        database=dbcon.getDB("jtest");
        table=database.getCollection("user123");
}
private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:
        //Insert Button
        User u1=new User();
        u1.setFname(fname.getText());
        u1.setLname(lname.getText());
        u1.setEmail(email.getText());
        u1.setPhone(Long.parseLong(phone.getText()));
        DBObject doc=createDBObject (u1);
        WriteResult r1=table.insert(doc);
         LoadData();
        fname.setText(" ");
        lname.setText("");
        email.setText("");
        phone.setText("");

 }
private void jButton3ActionPerformed(java.awt.event.ActionEvent evt) {
```

```
        // TODO add your handling code here:
        //DELETE Button
        DefaultTableModel tb=(DefaultTableModel)jUser.getModel();
        int sindex=jUser.getSelectedRow();
        tb.removeRow(sindex);
         fname.setText(" ");
          lname.setText("");
         email.setText("");
         phone.setText("");

    }
private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {
        // TODO add your handling code here:

        //UPDATE BUTTON
         int i=jUser.getSelectedRow();
          DefaultTableModel tb=(DefaultTableModel)jUser.getModel();
          if(i>=0)
          { tb.setValueAt(fname, i, 0);
             tb.setValueAt(lname, i, 1);
             tb.setValueAt(email, i, 2);
             tb.setValueAt(phone, i, 3);
             BasicDBObject newDocument = new BasicDBObject();
              newDocument.append("$set", new BasicDBObject().append("lname",lname.getText()));
              BasicDBObject searchQuery = new BasicDBObject().append("fname",fname.getText());
              table.update(searchQuery, newDocument);
              LoadData();
              fname.setText(" ");
             lname.setText("");
             email.setText("");
             phone.setText("");
```

```
        }
    }
//Mouse move event of table so when we choose any row from table it is binded to the respective textbox.
private void jUserMouseClicked(java.awt.event.MouseEvent evt) {
        // TODO add your handling code here:
        DefaultTableModel tb=(DefaultTableModel)jUser.getModel();
        int sindex=jUser.getSelectedRow();
        fname.setText(tb.getValueAt(sindex, 0).toString());
         lname.setText(tb.getValueAt(sindex, 1).toString());
          email.setText(tb.getValueAt(sindex, 2).toString());
           phone.setText(tb.getValueAt(sindex, 3).toString());
    }
private void jButton4ActionPerformed(java.awt.event.ActionEvent evt) {

        // TODO add your handling code here:
        //Show Button
        LoadData();
    }
private static DBObject createDBObject(User u1)
{
    BasicDBObjectBuilder dbuilder=BasicDBObjectBuilder.start();
    dbuilder.append("fname", u1.getFname());
    dbuilder.append("lname", u1.getLname());
    dbuilder.append("email", u1.getEmail());
    dbuilder.append("phone", u1.getPhone());
    return dbuilder.get();
}
private void LoadData()
    {
```

```
        DefaultTableModel tb=(DefaultTableModel)jUser.getModel();   //verified
        tb.setRowCount(0);      //verified
        BasicDBObject query=new BasicDBObject();    //verified
        BasicDBObject field=new BasicDBObject();    //verified
        DBCursor cr = table.find(query,field);      //verified
        while(cr.hasNext())
        {
            BasicDBObject obj=(BasicDBObject)cr.next();
           // Object[]
ROW={obj.getString("fname"),obj.getStirng("lname"),obj.getString("email"),obj.getLong("phone")};
            Object[]
ROW={obj.getString("fname"),obj.getString("lname"),obj.getString("email"),obj.getString("phone")};
            tb.addRow(ROW);
        }


}
```

Add one class file

User.java

```
public class User {
    private String fname;
    private String lname;
    private String email;
    private long phone;

    private String extra;

    public String getExtra() {
        return extra;
    }

    public void setExtra(String extra) {
        this.extra = extra;
```

```
}

public String getEmail() {
   return email;
}

public void setEmail(String email) {
   this.email = email;
}

public String getFname() {
   return fname;
}

public void setFname(String fname) {
   this.fname = fname;
}

public String getLname() {
   return lname;
}

public void setLname(String lname) {
   this.lname = lname;
}

public long getPhone() {
   return phone;
}

public void setPhone(long phone) {
```

```
        this.phone = phone;
    }
}
```

Add one class file dbclass.java

```
package finaljavamango;
import com.mongodb.MongoClient;
public class dbclass {
    public static MongoClient getConnection()
    {   MongoClient mg=null;

        try
        {
            mg=new MongoClient("localhost",27017);

        }
        catch(Exception e)
        {e.printStackTrace();
        }
        return mg;
    }

}
```

- Create one table in mongodb.
- Table must have four fields. Fname,lastname,email,phoneno.
- When you are running your application in backend mongodb server must run. Otherwise we cannot display data in our application.