**Encrypt Decrypt**

package sliit**.**ead**.**encryptdecrypt**.**ejb**;**

**import** java**.**security**.**Security**;**

**import** javax**.**crypto**.**Cipher**;**

**import** javax**.**crypto**.**spec**.**IvParameterSpec**;**

**import** javax**.**crypto**.**spec**.**SecretKeySpec**;**

**import** javax**.**swing**.**JOptionPane**;**

public class EncryptDecrypt **extends** javax**.**swing**.**JFrame **{**

byte**[]** input**;**

byte**[]** keyBytes **=** "12345678"**.**getBytes**();** // our own key

byte**[]** ivBytes **=** "input123"**.**getBytes**();**

SecretKeySpec key **=** **new** SecretKeySpec**(**keyBytes**,** "DES"**);**

IvParameterSpec ivSpec **=** **new** IvParameterSpec**(**ivBytes**);**

Cipher cipher**;**

byte**[]** chiperText**;**

int ctLenght**;**

/\*\*

\* Creates new form EncryptDecrypt

\*/

public EncryptDecrypt**()** **{**

initComponents**();**

**}**

private void btn\_encryptActionPerformed**(**java**.**awt**.**event**.**ActionEvent evt**)** **{**

**try** **{**

Security**.**addProvider**(new** org**.**bouncycastle**.**jce**.**provider**.**BouncyCastleProvider**());**

//type of security

input **=** text**.**getText**().**getBytes**();**

SecretKeySpec key **=** **new** SecretKeySpec**(**keyBytes**,** "DES"**);**

IvParameterSpec ivSpec **=** **new** IvParameterSpec**(**ivBytes**);**

cipher **=** Cipher**.**getInstance**(**"DES/CTR/NoPadding"**,** "BC"**);** //CTR of encryption

cipher**.**init**(**Cipher**.**ENCRYPT\_MODE**,** key**,** ivSpec**);**

chiperText **=** **new** byte**[**cipher**.**getOutputSize**(**input**.**length**)];**

ctLenght **=** cipher**.**update**(**input**,** 0**,** input**.**length**,** chiperText**,** 0**);**

ctLenght **+=** cipher**.**doFinal**(**chiperText**,** ctLenght**);**

encrypt**.**setText**(new** String**(**chiperText**));**

**}** **catch** **(**Exception e**)** **{**

JOptionPane**.**showMessageDialog**(null,** e**);**

**}**

**}**

private void btn\_decryptActionPerformed**(**java**.**awt**.**event**.**ActionEvent evt**)** **{**

**try** **{**

cipher**.**init**(**Cipher**.**DECRYPT\_MODE**,** key**,**ivSpec**);**

byte**[]** plainText **=** **new** byte**[**cipher**.**getOutputSize**(**ctLenght**)];**

int ptLength **=** cipher**.**update**(**chiperText**,** 0**,**ctLenght**,**plainText**,**0**);**

ptLength **+=** cipher**.**doFinal**(**plainText**,** ptLength**);**

decrypt**.**setText**(new** String**(**plainText**));**

**}** **catch** **(**Exception e**)** **{**

JOptionPane**.**showMessageDialog**(null,** e**);**

**}**

**}**

public static void main**(**String args**[])** **{ //main method**

/\* Create and display the form \*/

java**.**awt**.**EventQueue**.**invokeLater**(new** Runnable**()** **{**

public void run**()** **{**

**new** EncryptDecrypt**().**setVisible**(true);**

**}**

**});**

**}**

**}**

**Mongo DB**

Start the server

In the bin inside the extracted folder shift + right click -> open command window here

Create Data folder inside extracted folder

In the cmd -> mongod.exe –dbpath “path of the Data folder”

Mongo db is a document oriented db. Only documents are inserted. They must json docs

If there are no collections, when inserting db will automatically create a collection (table) and insert the values to the created collection

Create collection -> Insert/Import documents

DBManager.java

public class DB\_Manager **{**

private static DB database**;**

public static DB getDatabase**()** **{**

**if(**database **==** **null)** **{**

MongoClient mongo**;**

**try** **{**

mongo **=** **new** MongoClient**(**"localhost"**,** 27017**);**

database **=** mongo**.**getDB**(**"Student"**); //Collection name**

**}** **catch** **(**UnknownHostException e**)** **{**

// TODO Auto-generated catch block

e**.**printStackTrace**();**

**}**

**}**

**return** database**;**

**}**

**}**

Entity class

Create class, add variables, getters and setters



private void jButton1ActionPerformed**(**java**.**awt**.**event**.**ActionEvent evt**)** **{**

// TODO add your handling code here:

StudentAddFrame add **=** **new** StudentAddFrame**();**

add**.**setVisible**(true);**

**}**

private void jButton4ActionPerformed**(**java**.**awt**.**event**.**ActionEvent evt**)** **{**

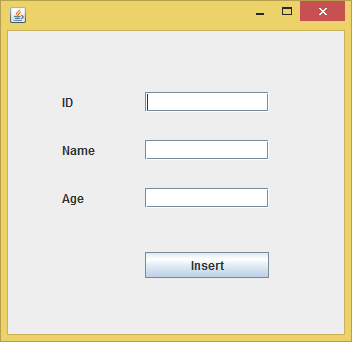
// TODO add your handling code here:

StudentViewFrame view **=** **new** StudentViewFrame**();**

view**.**setVisible**(true);**

**}**

Insert Student



private void btn\_insertActionPerformed**(**java**.**awt**.**event**.**ActionEvent evt**)** **{**

// TODO add your handling code here:

StudentClass stu **=** **new** StudentClass**();**

stu**.**setID**(**Integer**.**parseInt**(**text\_id**.**getText**()));**

stu**.**setName**(**text\_name**.**getText**());**

stu**.**setAge**(**Integer**.**parseInt**(**text\_age**.**getText**()));**

DBObject obj **=** createDBObject**(**stu**);**

DB userdb **=** DB\_Manager**.**getDatabase**();**

DBCollection col **=** userdb**.**getCollection**(**"Student"**);**

WriteResult result **=** col**.**insert**(**obj**);**

JOptionPane**.**showMessageDialog**(**rootPane**,** "Student Inserted**);**

text\_id**.**setText**(**""**);**

text\_name**.**setText**(**""**);**

text\_age**.**setText**(**""**);**

**}**

private static DBObject createDBObject**(**StudentClass student**)**

**{**

BasicDBObjectBuilder db **=** BasicDBObjectBuilder**.**start**();**

db**.**append**(**"ID"**,** student**.**getID**());**

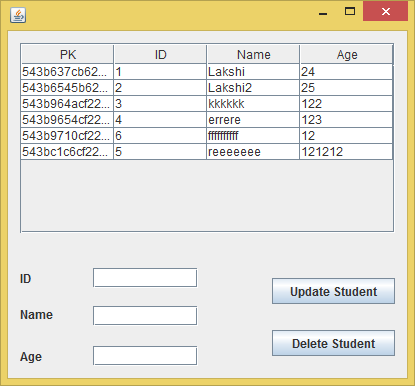
db**.**append**(**"name"**,**student**.**getName**());**

db**.**append**(**"age"**,**student**.**getAge**());**

**return** db**.**get**();**

**}**

View All Students



public StudentViewFrame**()** **{**

initComponents**();**

loadStudents**();**

setDefaultCloseOperation**(**JFrame**.**DISPOSE\_ON\_CLOSE**); // Not to close parent**

**}**

public void loadStudents**()**

**{**

DefaultTableModel model **=** **new** DefaultTableModel**();**

model**.**addColumn**(**"ID"**); // Column names to display in the table**

model**.**addColumn**(**"Name"**);**

model**.**addColumn**(**"Age"**);**

DB userDB **=** DB\_Manager**.**getDatabase**();**

DBCollection col **=** userDB**.**getCollection**(**"Student"**);**

DBCursor cur **=** col**.**find**();**

**while(**cur**.**hasNext**())**

**{**

DBObject user **=** **(**DBObject**)** cur**.**next**();**

model**.**addRow**(new** Object**[]** **{**user**.**get**(**"ID"**),** user**.**get**(**"name"**),** user**.**get**(**"age"**)});**

**}**

tbl\_students**.**setModel**(**model**);**

**}**

Update Student

Mouse click event on table

private void tbl\_studentsMouseClicked**(**java**.**awt**.**event**.**MouseEvent evt**)** **{**

// TODO add your handling code here:

int s **=** tbl\_students**.**getSelectedRow**();**

//JOptionPane.showMessageDialog(rootPane, "Student Inserted"+s);

up\_pk**.**setText**(**tbl\_students**.**getValueAt**(**s**,** 0**).**toString**());**

up\_id**.**setText**(**tbl\_students**.**getValueAt**(**s**,** 1**).**toString**());**

up\_name**.**setText**(**tbl\_students**.**getValueAt**(**s**,** 2**).**toString**());**

up\_age**.**setText**(**tbl\_students**.**getValueAt**(**s**,** 3**).**toString**());**

**}**

Update button click event

private void btn\_updateActionPerformed**(**java**.**awt**.**event**.**ActionEvent evt**)** **{**

// TODO add your handling code here:

DB userDB **=** DB\_Manager**.**getDatabase**();**

DBCollection col **=** userDB**.**getCollection**(**"Student"**);**

BasicDBObject query **=** **new** BasicDBObject**(**"ID"**,** **new** BasicDBObject**(**"$eq"**,** Integer**.**parseInt**(**up\_id**.**getText**())));**

BasicDBObject newdoc **=** **new** BasicDBObject**();**

newdoc**.**put**(**"ID"**,** up\_id**.**getText**());**

newdoc**.**put**(**"name"**,** up\_name**.**getText**());**

newdoc**.**put**(**"age"**,** up\_age**.**getText**());**

BasicDBObject updateObj **=** **new** BasicDBObject**();**

updateObj**.**put**(**"$set"**,** newdoc**);**

col**.**update**(**query**,** updateObj**,** **false,** **true);**

JOptionPane**.**showMessageDialog**(**rootPane**,** "Student Updated"**);**

loadStudents**();**

**}**

Delete Student

private void btn\_deleteActionPerformed**(**java**.**awt**.**event**.**ActionEvent evt**)** **{**

DB userDB **=** DB\_Manager**.**getDatabase**();**

DBCollection col **=** userDB**.**getCollection**(**"Student"**);**

BasicDBObject query **=** **new** BasicDBObject**(**"ID"**,** **new** BasicDBObject**(**"$eq"**,** Integer**.**parseInt**(**up\_id**.**getText**())));**

col**.**remove**(**query**);**

JOptionPane**.**showMessageDialog**(**rootPane**,** "Student Deleted"**);**

loadStudents**();**

**}**

**Web Service in Java**

New project -> Java Web -> Web Application

Server -> Glass Fish Server Java EE Version -> Java EE 6 Web

Right click on project -> New -> Other -> Web Services -> Web Service

Create package -> Tick the checkbox Stateless session bean -> Finish

Delete default @WebMethod

Go to Design -> Add operations -> Name: methodName -> Add parameters -> Ok

Implement the method according to the requirement

Right click on project -> Deploy

Web Services -> Right click on created web service -> Test web service

Click on WSDL File -> Copy the link and add as a service reference

@WebService**(**serviceName **=** "NewWebService"**)**

@Stateless**()**

public class NewWebService **{**

// Web service operation

@WebMethod**(**operationName **=** "addOperation"**)**

public int addOperation**(**@WebParam**(**name **=** "num1"**)** int num1**,** @WebParam**(**name **=** "num2"**)** int num2**)** **{**

//TODO write your implementation code here:

**return** num1**+**num2**;**

**}**

**}**

.NET Button click event

private void button1\_Click(object sender, EventArgs e)

{

if (string.IsNullOrEmpty(textTemp.Text))

{

MessageBox.Show("Enter a tempurature");

}

else

{

try

{

//conTemp.Text = (new ServiceReferenceTempConvert.ServiceSoapClient().CelciusToFahrenheit (float.Parse(textTemp.Text.ToString()))).ToString();

ServiceReferenceTempConvert.ServiceSoapClient obj1 = new ServiceReferenceTempConvert.ServiceSoapClient();

conTemp.Text = obj1.CelciusToFahrenheit(float.Parse(textTemp.Text.ToString())).ToString();

}

catch (Exception)

{

MessageBox.Show("Enter a valid tempurature");

}

}

}

WebService in .NET - New -> Website -> WebService (Select Frmework 3.5)