

1) TẠO DATABASE

Tạo CSDL đặt tên **DB_QuanLyNhanVien**

Tạo table có tên **NhanVien**

Với **id** là thuộc tính khóa **tự động tăng**

Column Name	Data Type	Allow Nulls
id	int	<input type="checkbox"/>
name	nvarchar(50)	<input checked="" type="checkbox"/>
age	int	<input checked="" type="checkbox"/>
phone	nchar(10)	<input checked="" type="checkbox"/>
address	nvarchar(50)	<input checked="" type="checkbox"/>
		<input type="checkbox"/>

Thêm vài dòng dữ liệu vào table

id	name	age	phone	address
1	Trần Văn A	25	0919159669	Cần Thơ
2	Nguyễn Thanh B	32	0908123901	Cà Mau
3	Lê Minh C	29	0903903666	An Giang
* NULL	NULL	NULL	NULL	NULL

2) TẠO PROJECT GUIQuanLyNhanVien

Thực hiện tạo các package để lưu các class cần thao tác, bao gồm các package: database, model, test, view

2.1. Tạo package "database"

Tại đây tạo file class dùng để thực hiện kết nối CSDL, cụ thể file class được đặt tên là: **DatabaseHelper.java**

```

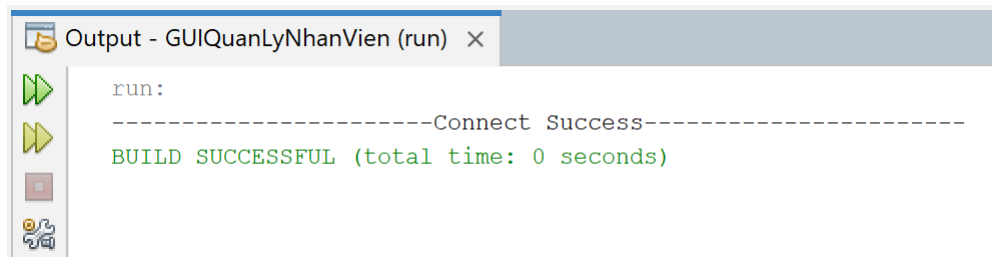
1. /*
2.  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt
to change this license
3.  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit
this template
4.  */
5. package database;
6. import java.sql.Connection;
7. import java.sql.DriverManager;
8. import java.sql.ResultSet;
9. import java.sql.SQLException;
10. import java.sql.Statement;
11. /**
12.  *
13.  * @author trinh

```

```

14.  */
15. public class DatabaseHelper {
16.     public static final String connectionUrl = "jdbc:sqlserver://localhost;"
17.         + "databaseName=DB_QLNhanVien;user=sa;password=2012";
18.     public static Connection getDBConnect() {
19.         Connection conn = null;
20.         try {
21.             Class.forName("com.microsoft.sqlserver.jdbc.SQLServerDriver");
22.             System.out.println("-----Connect Success-----");
23.         } catch (Exception e) {
24.             System.out.println("Chưa có Driver!" + e.toString());
25.         }
26.         try {
27.             conn = DriverManager.getConnection(connectionUrl);
28.             return conn;
29.         } catch (Exception e) {
30.             System.out.println("Lỗi connect" + e.toString());
31.         }
32.         return null;
33.     }
34.     public static void main(String[] args) {
35.         getDBConnect();
36.     }
37. }
38.

```



2.2. Tạo package “model”

2.2.1. NhanVien.java

```

1.  /*
2.  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt
  to change this license
3.  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit
  this template
4.  */
5.  package models;
6.
7.  /**
8.   *
9.   * @author trinh
10.  */
11. public class NhanVien {
12.     private int id;
13.     private String name;
14.     private int age;
15.     private String phone;
16.     private String address;

```

```

17.
18.     public NhanVien() {
19.     }
20.
21.     public NhanVien(int id, String name, int age, String phone, String
address) {
22.         this.id = id;
23.         this.name = name;
24.         this.age = age;
25.         this.phone = phone;
26.         this.address = address;
27.     }
28.
29.     public int getId() {
30.         return id;
31.     }
32.
33.     public void setId(int id) {
34.         this.id = id;
35.     }
36.
37.     public String getName() {
38.         return name;
39.     }
40.
41.     public void setName(String name) {
42.         this.name = name;
43.     }
44.
45.     public int getAge() {
46.         return age;
47.     }
48.
49.     public void setAge(int age) {
50.         this.age = age;
51.     }
52.
53.     public String getPhone() {
54.         return phone;
55.     }
56.
57.     public void setPhone(String phone) {
58.         this.phone = phone;
59.     }
60.
61.     public String getAddress() {
62.         return address;
63.     }
64.
65.     public void setAddress(String address) {
66.         this.address = address;
67.     }
68.
69.     @Override
70.     public String toString() {
71.         return "NhanVien{" + "id=" + id + ", name=" + name + ", age=" + age +
", phone=" + phone + ", address=" + address + '}';
72.     }
73.

```

74. }
75.

2.2.2. NhanVienDAO.java

```

5  package models;
6  import database.DatabaseHelper;
7  import java.sql.*;
8  import java.util.ArrayList;
9  import java.util.List;
10 /**
11  *
12  * @author trinh
13  */
14 public class NhanVienDAO {
15     Connection conn = null;
16     PreparedStatement sttm = null;
17
18     //THEM MOI DU LIEU VAO tbNhanVien
19     public int add(NhanVien nv){...28 lines }
20
21     //SỬA DU LIEU tbNhanVien
22     public int update(NhanVien nv){...28 lines }
23
24     //XÓA DỮ LIỆU
25     public int delete(int id){...24 lines }
26
27     //HIỂN THỊ TOÀN BỘ DỮ LIỆU TRONG TABLE NhanVien
28     public List<NhanVien> getAll() {...45 lines }
29
30     //TÌM KIẾM NHÂN VIÊN THEO MÃ
31     public NhanVien getNhanVienByID(int id){...44 lines }
32
33     //
34 }

```

a) Đoạn mã code dùng để **thêm mới 1 nhân viên vào CSDL**

```

1. //THEM MOI DU LIEU VAO tbNhanVien
2. public int add(NhanVien nv){
3.     Connection conn = null;
4.     PreparedStatement sttm = null;
5.     try{
6.         String sSQL = "INSERT NhanVien(name, age, phone, address)\n" +
7.         "VALUES(?,?,?,?); ";
8.         conn = DatabaseHelper.getDBConnect();
9.         sttm = conn.prepareStatement(sSQL);
10.
11.         sttm.setString(1, nv.getName());
12.         sttm.setInt(2, nv.getAge());
13.         sttm.setString(3, nv.getPhone());
14.         sttm.setString(4, nv.getAddress());
15.
16.         if (sttm.executeUpdate() > 0) {
17.             System.out.println("THÊM DỮ LIỆU THÀNH CÔNG");
18.             return 1;
19.         }
20.     }catch(Exception e){
21.         System.out.println("Error: " +e.toString());
22.     }
23.     finally{
24.         try{
25.             conn.close();sttm.close();
26.         }catch(Exception e){}
27.     }
28.     return -1; //Them khong thanh cong

```

```

29.     }
30.    //
31.

```

Test đoạn mã ở hàm main quan sát kết quả thực thi

```

1. public class MainTestNhanVien {
2.     public static void main(String[] args) {
3.         Connection conn = DatabaseHelper.getDBConnect();
4.         NhanVienDAO nvDAO = new NhanVienDAO();
5.
6.         if (conn != null) {
7.             System.out.println("KẾT NỐI tbNhanVien THÀNH CÔNG.....");
8.             //THEM DU LIEU VAO tbNhanVien
9.             NhanVien nv = new NhanVien(0,"Liêu Minh Ánh", 33, "0919123901",
"Sóc Trăng");
10.            nvDAO.add(nv);
11.        }
12.    } else {
13.        System.out.println("LỖI KẾT NỐI");
14.    }
15. }
16. }
17.

```

Kết quả thực thi:

```

run:
-----Connect Success-----
KẾT NỐI tbNhanVien THÀNH CÔNG.....
-----Connect Success-----
THÊM DỮ LIỆU THÀNH CÔNG
-----Connect Success-----
-----HIỂN THỊ CSDL TABLE NhanVien -----
ID      NAME      AGE      PHONE      ADDRESS
1013    Trần Văn A      33      0932902103  Cần Thơ
1014    Liêu Minh Ánh   33      0919123901  Sóc Trăng
BUILD SUCCESSFUL (total time: 0 seconds)

```

b) Tại class thiết kế đoạn mã code dùng để SỬA dữ liệu

```

1. public int update(NhanVien nv){
2.     Connection conn = null;
3.     PreparedStatement sttm = null;
4.     try{
5.         String sSQL = "UPDATE NhanVien SET name=?, age=?, phone=?,
address=? WHERE id=?";
6.         conn = DatabaseHelper.getDBConnect();
7.         sttm = conn.prepareStatement(sSQL);
8.
9.         sttm.setString(1, nv.getName());
10.        sttm.setInt(2, nv.getAge());
11.        sttm.setString(3, nv.getPhone());
12.        sttm.setString(4, nv.getAddress());
13.        sttm.setInt(5, nv.getId());
14.

```

```

15.         if (sttm.executeUpdate() > 0) {
16.             System.out.println("CẬP NHẬT DỮ LIỆU THÀNH CÔNG");
17.             return 1;
18.         }
19.     }catch(Exception e){
20.         System.out.println("Error: " +e.toString());
21.     }
22.     finally{
23.         try{
24.             conn.close();sttm.close();
25.         }catch(Exception e){}
26.     }
27.     return -1; //Update không thành công
28. }
29.

```

c) Tại class thiết kế đoạn mã code dùng để XÓA dữ liệu

```

1.     public int delete(int id){
2.         Connection conn = null;
3.         PreparedStatement sttm = null;
4.         try{
5.             String sSQL = "DELETE NhanVien WHERE id = ?";
6.             conn = DatabaseHelper.getDBConnect();
7.             sttm = conn.prepareStatement(sSQL);
8.
9.             sttm.setInt(1, id);
10.
11.             if (sttm.executeUpdate() > 0) {
12.                 System.out.println("XÓA DỮ LIỆU THÀNH CÔNG");
13.                 return 1;
14.             }
15.         }catch(Exception e){
16.             System.out.println("Error: " +e.toString());
17.         }
18.         finally{
19.             try{
20.                 conn.close();sttm.close();
21.             }catch(Exception e){}
22.         }
23.         return -1; //Xóa không thành công
24.     }
25.

```

d) Tại class thiết kế đoạn mã code dùng để HIỂN THỊ TOÀN BỘ dữ liệu

```

//HIEN THI TOAN BO DU LIEU TRONG TABLE NhanVien
104.
105.     public List<NhanVien> getAll() {
106.
107.         List<NhanVien> ls = new ArrayList<>();
108.
109.         Connection conn = null;
110.         ResultSet rs = null;
111.         Statement sttm = null;
112.
113.         try {
114.             String sSQL = "SELECT * FROM NhanVien";
115.             conn = DatabaseHelper.getDBConnect();

```

```

116.         sttm = conn.createStatement();
117.         rs = sttm.executeQuery(sSQL);
118.
119.         System.out.println("-----HIỂN THỊ CSDL TABLE NhanVien -----
");
120.         System.out.printf("%-10s %-30s %-10s %-10s %-10s \n",
121.             "ID", "NAME", "AGE", "PHONE", "ADDRESS");
122.
123.         while (rs.next()) {
124.             NhanVien nv = new NhanVien();
125.             nv.setId(rs.getInt(1));
126.             nv.setName(rs.getString(2));
127.             nv.setAge(rs.getInt(3));
128.             nv.setPhone(rs.getString(4));
129.             nv.setAddress(rs.getString(5));
130.
131.
132.             System.out.printf("%-10s %-30s %-10s %-10s %-10s \n",
133.                 rs.getInt(1), rs.getString(2),
134.                 rs.getInt(3), rs.getString(4), rs.getString(5));
135.
136.             ls.add(nv);
137.         }
138.     } catch (Exception e) {
139.         System.out.println("Error: " + e.toString());
140.     } finally {
141.         try {
142.             rs.close();
143.             sttm.close();
144.             conn.close();
145.         } catch (Exception e) {
146.             }
147.     }
148.     return ls;
149. }

```

e) Tại class thiết kế đoạn mã code dùng để TÌM KIẾM NhanVien THEO MÃ

```

//TÌM KIẾM NHÂN VIÊN MÃ NHÂN VIÊN
152.     public NhanVien getNhanVienByID(int id){
153.         ResultSet rs = null;
154.         Statement sttm = null;
155.         NhanVien nv = new NhanVien();
156.
157.         try{
158.             String sSQL = "SELECT id, name, age, phone, address FROM NhanVien
WHERE id='"+id+"'";
159.             conn = DatabaseHelper.getDBConnect();
160.             //sttm = conn.prepareStatement(sSQL);
161.             sttm = conn.createStatement();
162.             //sttm.setInt(1, id);
163.             rs = sttm.executeQuery(sSQL);
164.             /*
165.             System.out.println("-----HIỂN THỊ CSDL TABLE NhanVien -----
");
166.             System.out.printf("%-10s %-30s %-10s %-10s %-10s \n",
167.                 "ID", "NAME", "AGE", "PHONE", "ADDRESS");
168.             */
169.             while(rs.next()){

```

```

170.         //NhanVien nv = new NhanVien();
171.
172.         nv.setId(rs.getInt(1));
173.         nv.setName(rs.getString(2));
174.         nv.setAge(rs.getInt(3));
175.         nv.setPhone(rs.getString(4));
176.         nv.setAddress(rs.getString(5));
177.         /*
178.         System.out.printf("%-10s %-30s %-10s %-10s %-10s \n",
179.             rs.getInt(1), rs.getString(2),
180.             rs.getInt(3), rs.getString(4),rs.getString(5));
181.         */
182.         return nv;
183.     }
184. }catch (Exception e) {
185.     System.out.println("Error: " + e.toString());
186. } finally {
187.     try {
188.         rs.close();
189.         sttm.close();
190.         conn.close();
191.     } catch (Exception e) {
192.     }
193. }
194. return null;
195. }

```

f

Toàn bộ file class NhanVienDAO.java

```

1.  /*
2.  * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-
default.txt to change this license
3.  * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit
this template
4.  */
5.  package models;
6.  import database.DatabaseHelper;
7.  import java.sql.*;
8.  import java.util.ArrayList;
9.  import java.util.List;
10. /**
11.  *
12.  * @author trinh
13.  */
14. public class NhanVienDAO {
15.     Connection conn = null;
16.     PreparedStatement sttm = null;
17.
18.     //THEM MOI DU LIEU VAO tbNhanVien
19.     public int add(NhanVien nv){
20.         Connection conn = null;
21.         PreparedStatement sttm = null;
22.         try{
23.             String sSQL = "INSERT NhanVien(name, age, phone, address)\n" +
24. "VALUES(?,?,?,?); ";
25.             conn = DatabaseHelper.getDBConnect();
26.             sttm = conn.prepareStatement(sSQL);

```



```

27.
28.         sttm.setString(1, nv.getName());
29.         sttm.setInt(2, nv.getAge());
30.         sttm.setString(3, nv.getPhone());
31.         sttm.setString(4, nv.getAddress());
32.
33.         if (sttm.executeUpdate() > 0) {
34.             System.out.println("THÊM DỮ LIỆU THÀNH CÔNG");
35.             return 1;
36.         }
37.     }catch(Exception e){
38.         System.out.println("Error: " +e.toString());
39.     }
40.     finally{
41.         try{
42.             conn.close();sttm.close();
43.         }catch(Exception e){}
44.     }
45.     return -1; //Them khong thanh cong
46. }
47. //
48. //THEM MOI DU LIEU VAO tbNhanVien
49.     public int update(NhanVien nv){
50.         Connection conn = null;
51.         PreparedStatement sttm = null;
52.         try{
53.             String sSQL = "UPDATE NhanVien SET name=?, age=?, phone=?,
address=? WHERE id=?";
54.             conn = DatabaseHelper.getDBConnect();
55.             sttm = conn.prepareStatement(sSQL);
56.
57.             sttm.setString(1, nv.getName());
58.             sttm.setInt(2, nv.getAge());
59.             sttm.setString(3, nv.getPhone());
60.             sttm.setString(4, nv.getAddress());
61.             sttm.setInt(5, nv.getId());
62.
63.             if (sttm.executeUpdate() > 0) {
64.                 System.out.println("CẬP NHẬT DỮ LIỆU THÀNH CÔNG");
65.                 return 1;
66.             }
67.         }catch(Exception e){
68.             System.out.println("Error: " +e.toString());
69.         }
70.         finally{
71.             try{
72.                 conn.close();sttm.close();
73.             }catch(Exception e){}
74.         }
75.         return -1; //Update khong thanh cong
76.     }
77. //
78.     public int delete(int id){
79.         Connection conn = null;
80.         PreparedStatement sttm = null;
81.         try{
82.             String sSQL = "DELETE NhanVien WHERE id = ?";
83.             conn = DatabaseHelper.getDBConnect();
84.             sttm = conn.prepareStatement(sSQL);

```

```

85.
86.         sttm.setInt(1, id);
87.
88.         if (sttm.executeUpdate() > 0) {
89.             System.out.println("XÓA DỮ LIỆU THÀNH CÔNG");
90.             return 1;
91.         }
92.     }catch(Exception e){
93.         System.out.println("Error: " +e.toString());
94.     }
95.     finally{
96.         try{
97.             conn.close();sttm.close();
98.         }catch(Exception e){}
99.     }
100.     return -1; //Xóa không thành công
101. }
102.
103. //HIỂN THI TOÀN BỘ DỮ LIỆU TRONG TABLE NhanVien
104.
105.     public List<NhanVien> getAll() {
106.
107.         List<NhanVien> ls = new ArrayList<>();
108.
109.         Connection conn = null;
110.         ResultSet rs = null;
111.         Statement sttm = null;
112.
113.         try {
114.             String sSQL = "SELECT * FROM NhanVien";
115.             conn = DatabaseHelper.getDBConnect();
116.             sttm = conn.createStatement();
117.             rs = sttm.executeQuery(sSQL);
118.
119.             System.out.println("-----HIỂN THI CSDL TABLE NhanVien -----
120. ");
121.             System.out.printf("%-10s %-30s %-10s %-10s %-10s \n",
122.                 "ID", "NAME", "AGE", "PHONE", "ADDRESS");
123.
124.             while (rs.next()) {
125.                 NhanVien nv = new NhanVien();
126.                 nv.setId(rs.getInt(1));
127.                 nv.setName(rs.getString(2));
128.                 nv.setAge(rs.getInt(3));
129.                 nv.setPhone(rs.getString(4));
130.                 nv.setAddress(rs.getString(5));
131.
132.                 System.out.printf("%-10s %-30s %-10s %-10s %-10s \n",
133.                     rs.getInt(1), rs.getString(2),
134.                     rs.getInt(3), rs.getString(4),rs.getString(5));
135.
136.                 ls.add(nv);
137.             }
138.         } catch (Exception e) {
139.             System.out.println("Error: " + e.toString());
140.         } finally {
141.             try {
142.                 rs.close();

```

```

143.         sttm.close();
144.         conn.close();
145.     } catch (Exception e) {
146.     }
147. }
148.     return ls;
149. }
150. //
151. //TÌM KIẾM NHÂN VIÊN MÃ NHÂN VIÊN
152.     public NhanVien getNhanVienByID(int id){
153.         ResultSet rs = null;
154.         Statement sttm = null;
155.         NhanVien nv = new NhanVien();
156.
157.         try{
158.             String sSQL = "SELECT id, name, age, phone, address FROM NhanVien
WHERE id='"+id+"'";
159.             conn = DatabaseHelper.getDBConnect();
160.             //sttm = conn.prepareStatement(sSQL);
161.             sttm = conn.createStatement();
162.             //sttm.setInt(1, id);
163.             rs = sttm.executeQuery(sSQL);
164.             /*
165.             System.out.println("-----HIỂN THỊ CSDL TABLE NhanVien -----
");
166.             System.out.printf("%-10s %-30s %-10s %-10s %-10s \n",
167.                 "ID", "NAME", "AGE", "PHONE", "ADDRESS");
168.             */
169.             while(rs.next()){
170.                 //NhanVien nv = new NhanVien();
171.
172.                 nv.setId(rs.getInt(1));
173.                 nv.setName(rs.getString(2));
174.                 nv.setAge(rs.getInt(3));
175.                 nv.setPhone(rs.getString(4));
176.                 nv.setAddress(rs.getString(5));
177.                 /*
178.                 System.out.printf("%-10s %-30s %-10s %-10s %-10s \n",
179.                     rs.getInt(1), rs.getString(2),
180.                     rs.getInt(3), rs.getString(4),rs.getString(5));
181.                 */
182.                 return nv;
183.             }
184.         } catch (Exception e) {
185.             System.out.println("Error: " + e.toString());
186.         } finally {
187.             try {
188.                 rs.close();
189.                 sttm.close();
190.                 conn.close();
191.             } catch (Exception e) {
192.             }
193.         }
194.         return null;
195.     }
196. //
197. }
198.

```

2.3. Tạo package “test”

2.3.1. MainNhanVien.java

```

1.  /*
2.   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt
  to change this license
3.   * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Class.java to edit
  this template
4.   */
5.  package test;
6.
7.  import database.DatabaseHelper;
8.  import java.sql.Connection;
9.  import models.NhanVien;
10. import models.NhanVienDAO;
11.
12. /**
13.  *
14.  * @author trinh
15.  */
16. public class MainTestNhanVien {
17.     public static void main(String[] args) {
18.         Connection conn = DatabaseHelper.getDBConnect();
19.         NhanVienDAO nvDAO = new NhanVienDAO();
20.
21.         if (conn != null) {
22.             System.out.println("KẾT NỐI tbNhanVien THÀNH CÔNG.....");
23.             //THEM DU LIEU VAO tbNhanVien
24.             // NhanVien nv = new NhanVien(0, "Liêu Minh Ánh", 33, "0919123901",
  "Sóc Trăng");
25.             // nvDAO.add(nv);
26.             //SUA
27.             //NhanVien nv = new NhanVien(5, "Liêu Thị Minh Anh", 33,
  "0919123901", "Sóc Trăng");
28.             //nvDAO.update(nv);
29.             //XOA
30.             //nvDAO.delete(5);
31.             //HIEN THI TOAN BO DU LIEU
32.             nvDAO.getAll();
33.             //TIM KIEM NhanVien co id = 3;
34.             NhanVien find = nvDAO.getNhanVienByID(3);
35.             //System.out.println(">" + nvDAO.getNhanVienByID(3));
36.             if (find != null) {
37.                 System.out.println("TÌM THẤY");
38.                 System.out.println(find.toString());
39.             } else {
40.                 System.out.println("KHÔNG TÌM THẤY MÃ NHÂN VIÊN NÀY");
41.             }
42.
43.         } else {
44.             System.out.println("LỖI KẾT NỐI");
45.         }
46.     }
47. }
48.

```

Hiển thị toàn bộ dữ liệu hiện có

```
1. //HIEN THI TOAN BO DU LIEU
2. nvDAO.getAll();
3.
```

```
Output - GUIQuanLyNhanVien (run) X
run:
-----Connect Success-----
KẾT NỐI tbNhanVien THÀNH CÔNG.....
-----Connect Success-----
-----HIỂN THỊ CSDL TABLE NhanVien -----
ID          NAME          AGE          PHONE          ADDRESS
1           Trần Văn A          25           0919159669     Cần Thơ
2           Nguyễn Thanh B      32           0908123901     Cà Mau
3           Lê Minh C           29           0903903666     An Giang
BUILD SUCCESSFUL (total time: 0 seconds)
```

3) Tạo giao diện JFRQuanLyNhanVien

 btdTrinh-demoQuanLyNhanVien

—

□

✕

QUẢN LÝ NHÂN VIÊN

ID

FIND

NAME

NEW

AGE

SAVE

PHONE

EDIT

ADDRESS

DEL

ID	NAME	AGE	PHONE	ADDRESS
1013	Trần Văn A	33	0932902103	Cần Thơ
1014	Liêu Min...	33	0919123901	Sóc Trăng

```

5 package view;
6 import javax.swing.JOptionPane;
7 import javax.swing.table.DefaultTableModel;
8 import models.NhanVien;
9 import models.NhanVienDAO;
10 /**
11  *
12  * @author trinh
13  */
14 public class JFrameQuanLyNhanVien extends javax.swing.JFrame {
15     NhanVienDAO nvDAO = new NhanVienDAO();
16     String id NV;
17     /** Creates new form JFrameQuanLyNhanVien ...3 lines */
18     public JFrameQuanLyNhanVien() {
19         initComponents();
20         setLocationRelativeTo( c: null);
21         fillDataTable();
22     }
23
24     public void fillDataTable(){...13 lines }
25
26     /** This method is called from within the constructor to initialize the form ...5 lines */
27     @SuppressWarnings("unchecked")
28     Generated Code
29     private void btnEditActionPerformed(java.awt.event.ActionEvent evt) {...17 lines }
30
31     private void resetForm(){...7 lines }
32
33     private void btnNewActionPerformed(java.awt.event.ActionEvent evt) {...4 lines }
34
35     public boolean validateform(){...6 lines }
36
37     public NhanVien getModel(){...9 lines }
38
39     private void btnSaveActionPerformed(java.awt.event.ActionEvent evt) {...16 lines }
40
41     public void setModel(NhanVien nv){...7 lines }
42
43     private void tbNhanVienMouseClicked(java.awt.event.MouseEvent evt) {...7 lines }
44
45     private void btnDeleteActionPerformed(java.awt.event.ActionEvent evt) {...17 lines }
46
47     private void btnFindActionPerformed(java.awt.event.ActionEvent evt) {...10 lines }
48
49
50     /**...3 lines */
51     public static void main(String args[]) {...31 lines }

```

a) Đoạn mã code cho phương thức **fillDataTable()**

```

25 public void fillDataTable(){
26     DefaultTableModel model = (DefaultTableModel) tbNhanVien.getModel();
27     model.setRowCount(0);
28     for (NhanVien nv : nvDAO.getAll()) {
29         Object dataRow[] = new Object[5];
30         dataRow[0] = nv.getId();
31         dataRow[1] = nv.getName();
32         dataRow[2] = nv.getAge();
33         dataRow[3] = nv.getPhone();
34         dataRow[4] = nv.getAddress();
35         model.addRow( rowData: dataRow);
36     }
37 }

```

b) Đoạn mã code cho phương thức **validateform()**

```

261 public boolean validateform(){
262     if (txtID.getText().isEmpty() || txtName.getText().isEmpty()) {
263         return false;
264     }
265     return true;
266 }

```

c) Đoạn mã code cho phương thức **resetForm()**

```

250 private void resetForm(){
251     txtAddress.setText( t: "");
252     txtID.setText( t: "");
253     txtName.setText( t: "");
254     txtAge.setText( t: "");
255     txtPhone.setText( t: "");
256 }

```

d) Đoạn mã code cho phương thức **getModel()**

```

267 public NhanVien getModel() {
268     NhanVien nv = new NhanVien();
269     nv.setId(id: Integer.parseInt(s: txtID.getText()));
270     nv.setName(name: txtName.getText());
271     nv.setAge(age: Integer.parseInt(s: txtAge.getText()));
272     nv.setPhone(phone: txtPhone.getText());
273     nv.setAddress(address: txtAddress.getText());
274     return nv;
275 }

```

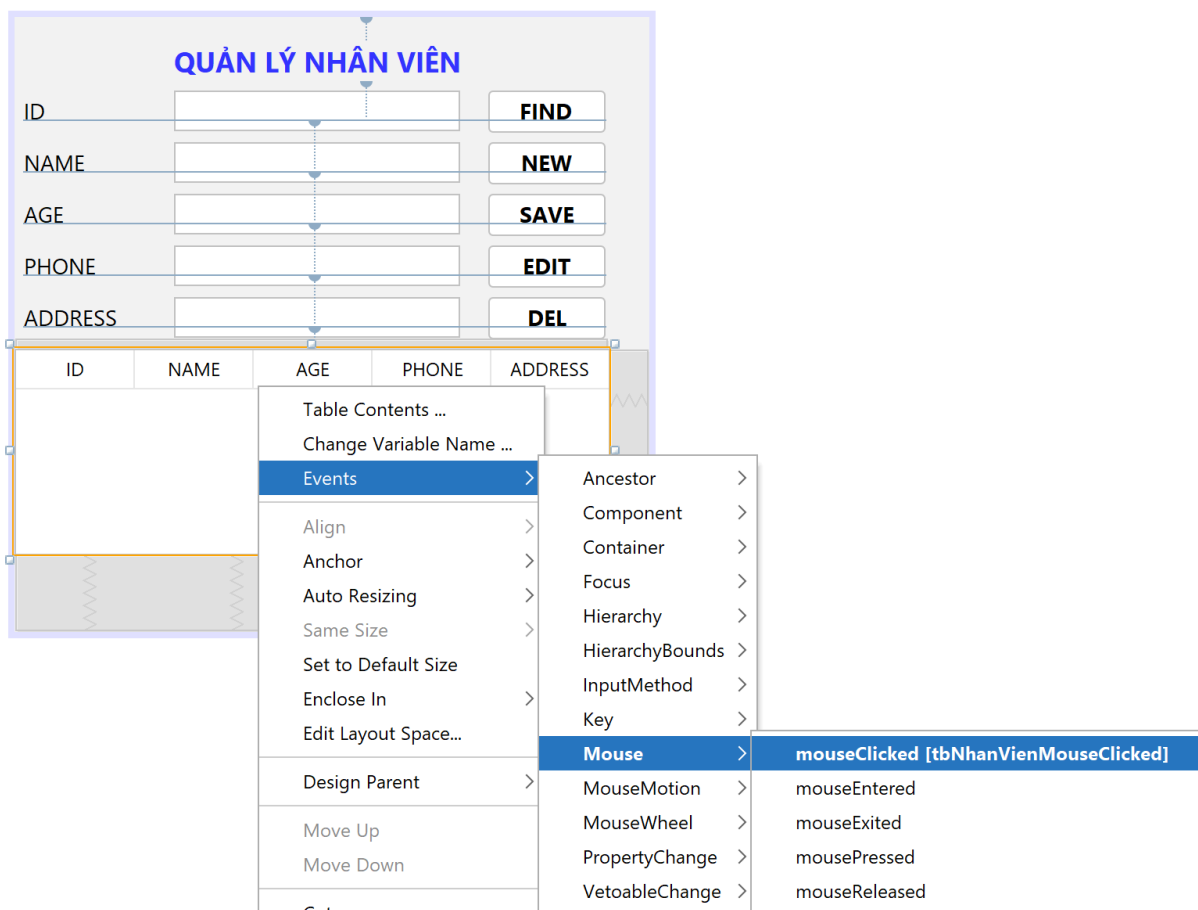
e) Đoạn mã code cho phương thức **setModel()**

```

292 public void setModel(NhanVien nv) {
293     txtID.setText(t: String.valueOf(i: nv.getId()));
294     txtName.setText(t: nv.getName());
295     txtAge.setText(t: String.valueOf(i: nv.getAge()));
296     txtPhone.setText(t: nv.getPhone());
297     txtAddress.setText(t: nv.getAddress());
298 }

```

f) Đoạn mã code cho hành vi người dùng chọn 1 dòng tại tbNhanVien thì dữ liệu sẽ đổ ngược lên form



```

299 private void tbNhanVienMouseClicked(java.awt.event.MouseEvent evt) {
300     // TODO add your handling code here:
301     int id = tbNhanVien.rowAtPoint( point:evt.getPoint());
302     id_NV = tbNhanVien.getValueAt( row:id, column:0).toString();
303     NhanVien nv = nvDAO.getNhanVienByID( id:Integer.parseInt( s:id_NV));
304     setModel(nv);
305 }

```

g) Đoạn mã code các button: NEW, EDIT, SAVE, DELETE, FIND

Click double chuột tại button *btnNew* và thiết kế đoạn mã sau:

```

258 private void btnNewActionPerformed(java.awt.event.ActionEvent evt) {
259     // TODO add your handling code here:
260     resetForm();
261 }

```

Click double chuột tại button *btnEdit* và thiết kế đoạn mã sau:

```

234 private void btnEditActionPerformed(java.awt.event.ActionEvent evt) {
235     // TODO add your handling code here:
236     if (validateform()) {
237         try {
238             NhanVien nvNew = getModel();
239             if (nvDAO.update( nv:nvNew) > 0) {
240                 JOptionPane.showMessageDialog( parentComponent: this, message: "Cập nhật thành công");
241                 fillDataTable();
242             }
243             catch (Exception e) {
244                 JOptionPane.showMessageDialog( parentComponent: this, "Lỗi: " + e.toString());
245             }
246         } else {
247             JOptionPane.showMessageDialog( parentComponent: this, message: "Bạn chưa nhập thông tin");
248         }
249     }
250 }

```

Click double chuột tại button *btnSave* và thiết kế đoạn mã sau:

```

277 private void btnSaveActionPerformed(java.awt.event.ActionEvent evt) {
278     // TODO add your handling code here:
279     if (validateform()) {
280         NhanVien nv = getModel();
281         try{
282             if (nvDAO.add(nv) > 0) {
283                 JOptionPane.showMessageDialog( parentComponent: this, message: "Lưu Thành Công...");
284                 fillDataTable();
285             }
286             catch (Exception e){
287                 JOptionPane.showMessageDialog( parentComponent: this, "Lỗi: " + e.toString());
288             }
289         } else{
290             JOptionPane.showMessageDialog( parentComponent: this, message: "Chưa nhập đủ thông tin");
291         }
292     }
293 }

```

Click double chuột tại button *btnDelete* và thiết kế đoạn mã sau:


```

307 private void btnDeleteActionPerformed(java.awt.event.ActionEvent evt) {
308     // TODO add your handling code here:
309     if (txtID.getText().isEmpty() || id_NV.length()==0) {
310         JOptionPane.showMessageDialog( parentComponent: this, message: "Chưa nhập ID để xóa");
311     }else{
312         try{
313             if (id_NV.length() > 0) {
314                 nvDAO.delete( id: Integer.parseInt( s:id_NV));
315             }else{
316                 nvDAO.delete( id: Integer.parseInt( s:txtID.getText()));
317             }
318             fillDataTable();
319         }catch (Exception e) {
320             JOptionPane.showMessageDialog( parentComponent: this, "Lỗi: " +e.toString());
321         }
322     }
323 }

```

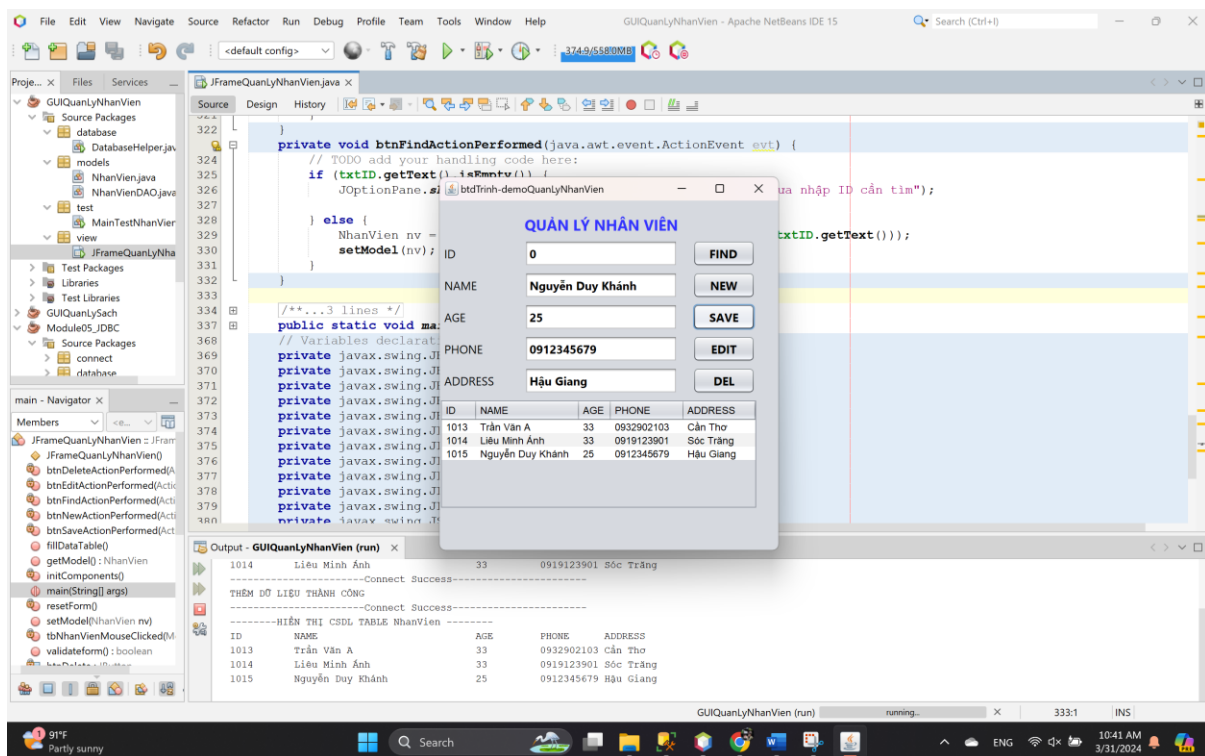
Click double chuột tại button **btnFind** và thiết kế đoạn mã sau:

```

324 private void btnFindActionPerformed(java.awt.event.ActionEvent evt) {
325     // TODO add your handling code here:
326     if (txtID.getText().isEmpty()) {
327         JOptionPane.showMessageDialog( parentComponent: this, message: "Chưa nhập ID cần tìm");
328     } else {
329         NhanVien nv = nvDAO.getNhanVienByID( id: Integer.parseInt( s:txtID.getText()));
330         setModel( nv);
331     }
332 }

```

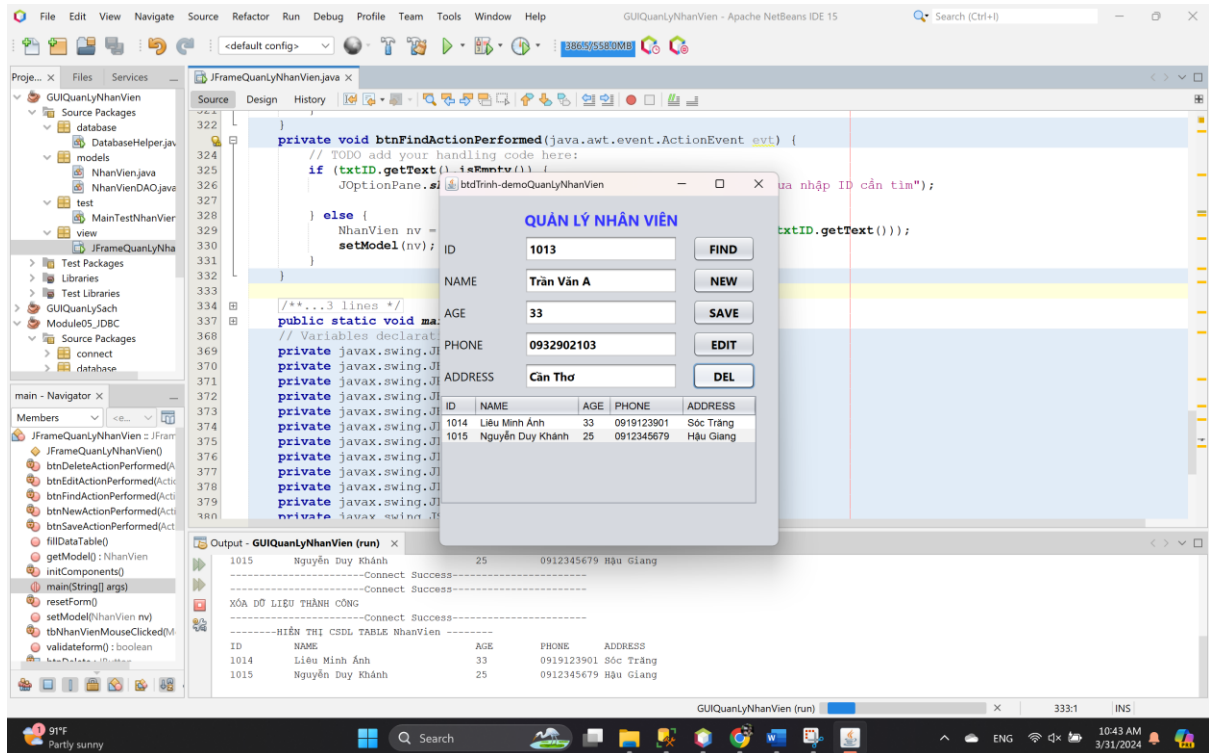
KHI NGƯỜI DÙNG BẤM NÚT “SAVE” ĐỂ THÊM MỚI 1 NHÂN VIÊN



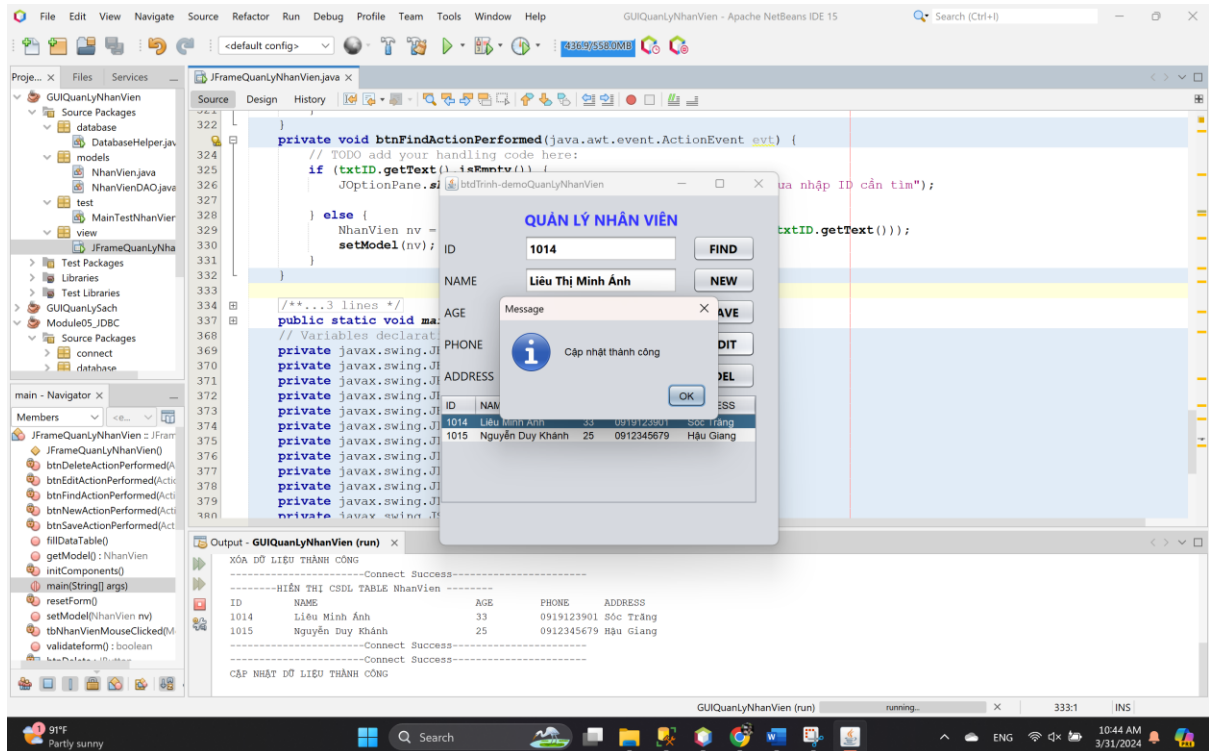
Khi bấm vào nút “NEW” thì dữ liệu ở các ô sẽ bị xóa trắng do có lời gọi đến phương thức resetForm()

KHI NGƯỜI DÙNG BẤM NÚT “DEL” ĐỂ XÓA 1 NHÂN VIÊN

GUIQuanLyNhanVien



KHI NGƯỜI DÙNG BẤM NÚT “EDIT” ĐỂ CHỈNH SỬA DỮ LIỆU 1 NHÂN VIÊN



Dữ liệu sau khi đã cập nhật

GUIQuanLyNhanVien

