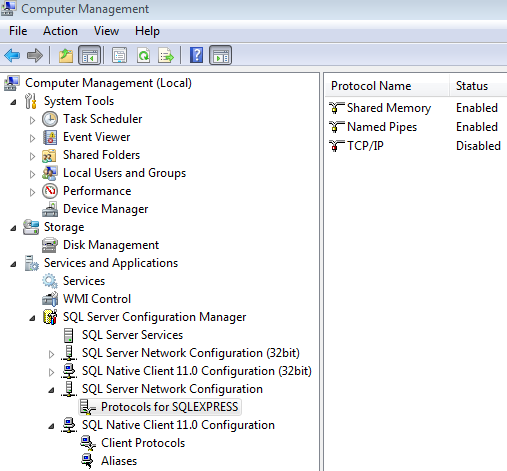
**BÀI THƯC HÀNH LẬP TRÌNH JAVA - BUỔI 6**

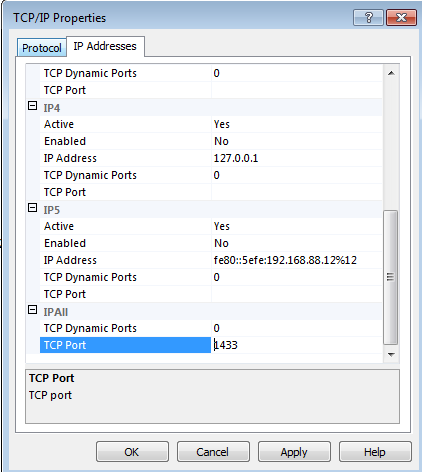
Hướng dẫn lại một số bước chuẩn bị cho bài thực hành (đã hướng dẫn trong buổi 5):

1. **Bật TCP/IP SQL Server với port 1433**

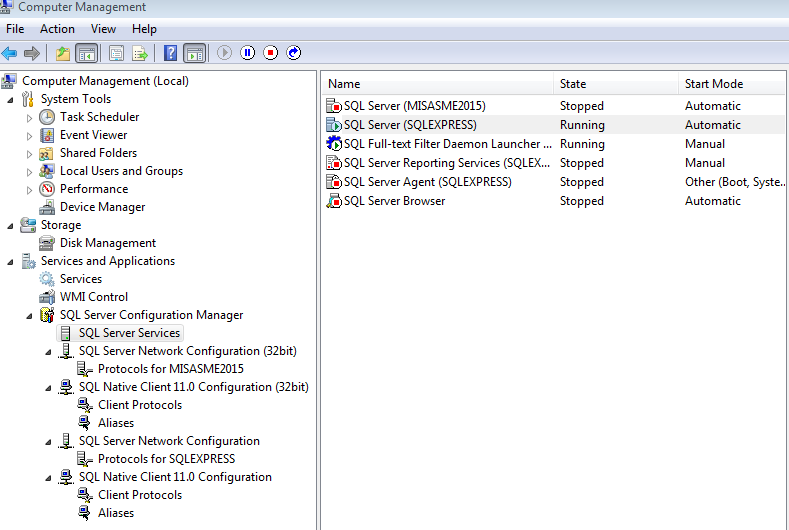
Vào Search Windows → Computer Management, tìm đến PCI/IP như sau



Chuột phải TCP/IP → Properties → qua thẻ IP Addresses, điền port 1433 như sau:



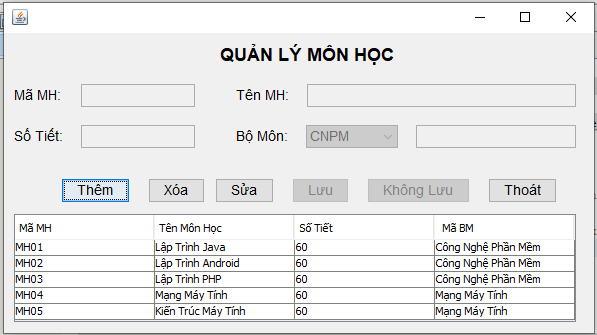
Sau khi OK ta chuột phải TCP/IP → Enabled sau đó ta vào restart lại SQL Server như sau để những thay đổi được cập nhật



Tìm đến SQL Server (SQLEXPRESS) chuột phải chọn Restart.

- Tiến hành đăng nhập SQLServer với chế độ SQL Server Authenticate với tài khoản sa (username: sa, password: 123456) và thực hiện việc tạo CSDL với SQLServer trong các bài tập bên dưới.

**Đề Bài**: Xây dựng ứng dụng quản lý Môn Học theo mô hình 3 lớp

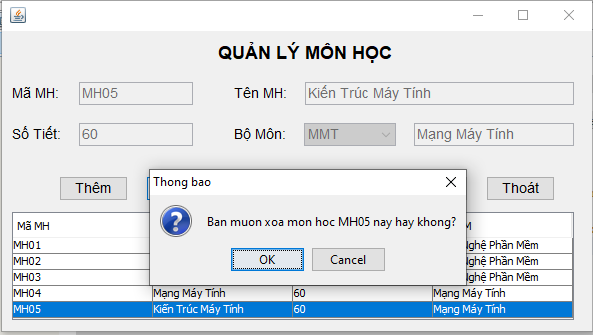


**Yêu cầu**:

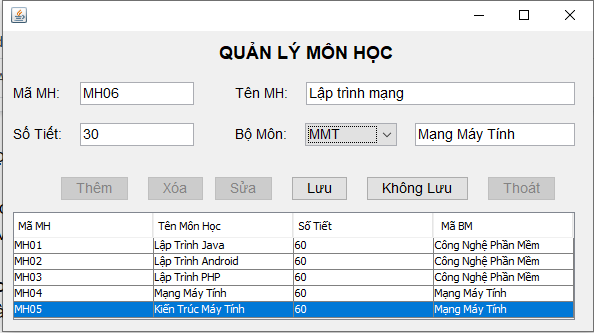
1. Kết nối cơ sở dữ liệu load thông tin môn học, database được hướng dẫn bên dưới.
2. Khi load chương trình:
   1. Tất cả textbox, combobox: bị vô hiệu hóa
   2. Các nút được bật/vô hiệu hóa như hình
   3. Danh sách các loại sản phẩm bằng JTable được load từ cơ sở dữ liệu (SQL Server)

c. Khi chọn bất kì dòng nào trong Table, dữ liệu dòng đó hiển thi tương ứng trên các textbox.

* Khi click nút “Xóa”, chương trình xóa data đó ra khỏi cơ sở dữ liệu, trước đó hiển thi hộp thoại nhắc:



d. Khi click nút “Thêm”, người dùng có thể nhập Mã MH, Tên MH, Số Tiết và chọn Bộ Môn tương ứng từ combo box như sau



Sau đó click Lưu, chương trình lưu xuống cơ sở dữ liệu thông tin vừa mới thêm vào table.

e. Tương tự, khi click nút “Sửa”, ta cho người dùng thay đổi thông tin như sau và sau đó lưu vào cơ sở dữ liệu

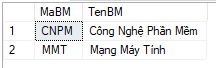
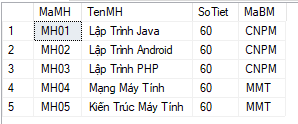
f. Nút “Không Lưu” thì sẽ reset lại form như trước và không lưu bất kì thông tin nào người dùng chỉnh sửa.

g. Nút “Thoát”, thoát khỏi chương trình

1. **Thiết Kế & Cài đặt CSDL**: **QLMonHoc**, có 02 Table:

* BoMon(MaBM char(4), TenBM nvarchar(20))
* MonHoc(MaMH char(4), TenMH nvarchar(20), SoTiet BigInt, MaBM char(4))

*Nhập liệu như sau*:

**Hướng dẫn**: Tạo CSDL trên trong SQL Server như sau:

Các bạn có thê tạo database bằng giao diện SQL Server hoặc bằng câu lệnh sql như sau:

Đăng nhập SQL Server với tài khoản sa, tạo CSDL và nhập liệu mẫu như sau

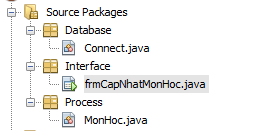
|  |
| --- |
| *--Tạo CSDL: QLMonHoc* **Create** **DataBase** QLMonHoc; **use** QLMonHoc;  *---Tao Table BoMon* **Create** **Table** BoMon ( MaBM char(4) primary **key**, TenBM **nvarchar**(20) ); **Insert** **into** BoMon **values**(**'CNPM'**,N**'Công Nghệ Phần Mềm'**); **Insert** **into** BoMon **values**(**'MMT'**,N**'Mạng Máy Tính'**);  *----Tạo table MonHoc* **Create** **Table** MonHoc ( MaMH char(4) primary **key**, TenMH **nvarchar**(20), SoTiet bigint, MaBM char(4) foreign **key** **references** BoMon(MaBM) **on** **delete** **cascade** ); **Insert** **into** MonHoc **values**(**'MH01'**,N**'Lập Trình Java'**,60,**'CNPM'**); **Insert** **into** MonHoc **values**(**'MH02'**,N**'Lập Trình Android'**,60,**'CNPM'**); **Insert** **into** MonHoc **values**(**'MH03'**,N**'Lập Trình PHP'**,60,**'CNPM'**); **Insert** **into** MonHoc **values**(**'MH04'**,N**'Mạng Máy Tính'**,60,**'MMT'**); **Insert** **into** MonHoc **values**(**'MH05'**,N**'Kiến Trúc Máy Tính'**,60,**'MMT'**); |

2. **Tạo Project: QLMonHoc**

Tạo project Java Application:

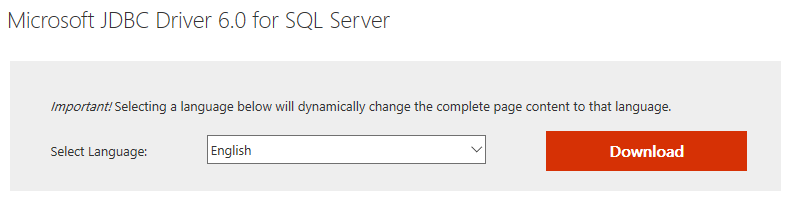
* Tên Project: **QLMonHoc**
* Vị trí lưu Project: Tùy ý
* Không tạo MainClass: Bỏ dấu Check Create Main Class

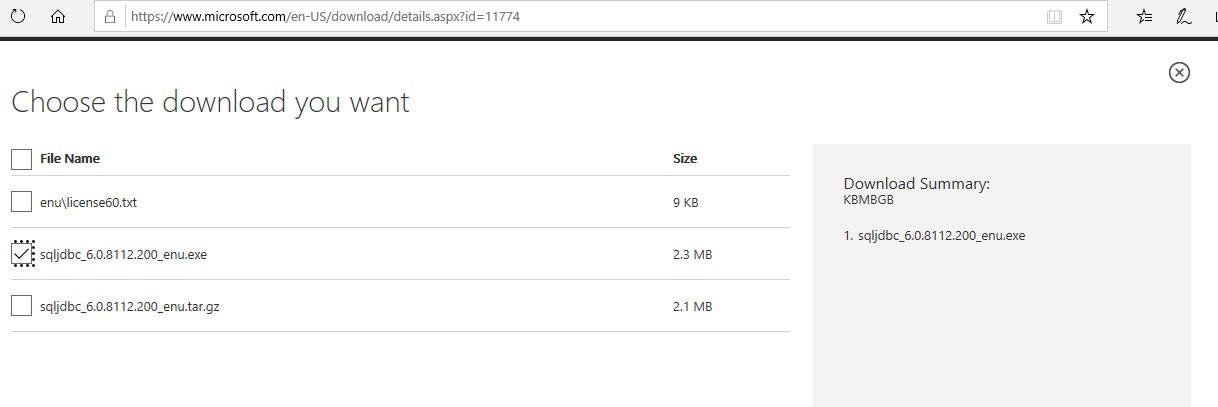
**3. Tạo 3 gói – Java Package tương ứng 03 lớp: Database, Interface, Process**

****

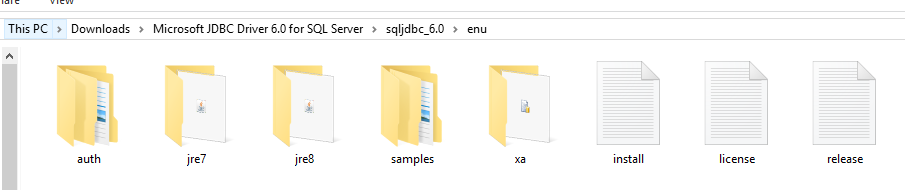
**4. Cài đặt thư viện từ Driver cho SQLServer:**

* Tải MS SQL Server JDBC Driver và giải nén: <https://www.microsoft.com/en-US/download/details.aspx?id=11774>





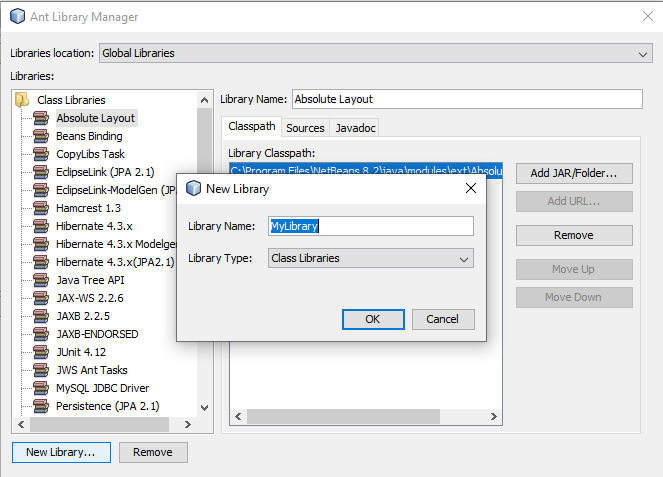
* Giải nén file vừa download về



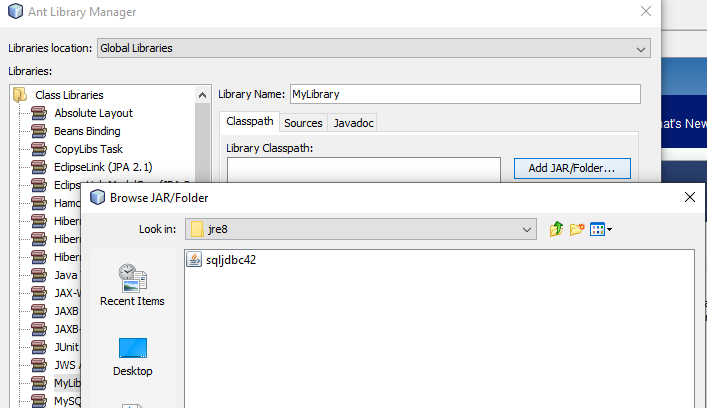
- Cài đặt Microsoft SQL Server JDBC Driver cho NetBeans:

1. Chọn menu tool -> library -> New Library

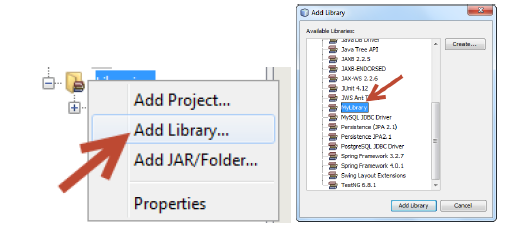
* Ở Textfield Library Name: chọn tên thư viện mà bạn muốn đặt , ví dụ: MyLibrary
* Còn ở Library Type thì cứ để mặc định → OK.



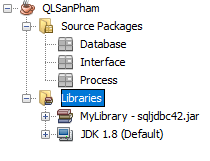
* Tiếp tục Chọn Nút Add JAR/Folder: Chỉ đường dẫn đến file JAR mới vừa down về trong thư mục giải nén trên ….\jre8\sqljdbc42



* Bổ sung thư viện MyLibrary cho Project: (hình ảnh minh họa từ bài Lab ngày 5)



* Sau khi add ta được cấu trúc như sau:



5. Thiết kế Form như yêu cầu đề bài

6. **Hướng dẫn phần source code tham khảo:**

* Code cho lớp Connect trong Package Database

|  |
| --- |
| **package** Database;  **import** java.sql.\*; **import** javax.swing.\*;  **public** **class** Connect {   **public** Connection conn = **null**;   **public** **void** connectSQL() **throws** SQLException {  **try** {  String userName = **"sa"**;  String password = **"123456"**;  String url = **"jdbc:sqlserver://localhost:1433;databaseName=QLMonHoc;"**;  Class.forName(**"com.microsoft.sqlserver.jdbc.SQLServerDriver"**);  conn = java.sql.DriverManager.getConnection(url, userName, password);  } **catch** (ClassNotFoundException e) {  JOptionPane.showMessageDialog(**null**, **"Ket noi CSDL that bai"**, **"Thong bao"**, 1);  }  }   **public** ResultSet LoadData(String sql) {  ResultSet result = **null**;  **try** {  Statement statement = conn.createStatement();  **return** statement.executeQuery(sql);  } **catch** (SQLException e) {  e.printStackTrace();  **return** **null**;  }  }   **public** **void** UpdateData(String sql) {  **try** {  Statement statement = conn.createStatement();  statement.executeUpdate(sql);  } **catch** (SQLException e) {  e.printStackTrace();  }  } } |

- Code cho lớp MonHoc trong Package Proccess:

|  |
| --- |
| **package** Process;  **import** Database.Connect; **import** java.sql.\*;  **public** **class** MonHoc {   **public** Connect cn = **new** Connect();    **public** ResultSet LoadBoMon() **throws** SQLException {  cn.connectSQL();  String sql = **"SELECT \* FROM BoMon"**;  **return** cn.LoadData(sql);  }    **public** ResultSet LoadBoMon(String maBM) **throws** SQLException {  String sql = **"SELECT \* FROM BoMon where maBM='"** + maBM + **"'"**;  **return** cn.LoadData(sql);  }    **public** ResultSet LoadMonHoc(String maMH) {  String sql = **"SELECT MaMH, TenMH, SoTiet, BM.MaBM, TenBM "**  + **"FROM MonHoc MH, BoMon BM WHERE BM.MaBM = MH.MaBM AND MH.MaMH='"** + maMH + **"'"**;  **return** cn.LoadData(sql);  }    **public** ResultSet LoadMonHoc() **throws** SQLException {  cn.connectSQL();  String sql = **"SELECT MaMH, TenMH, SoTiet, MH.MaBM, TenBM "**  + **"FROM MonHoc MH, BoMon BM WHERE BM.MaBM = MH.MaBM"**;  **return** cn.LoadData(sql);  }   **public** ResultSet ShowSPTheoma(String maMH) **throws** SQLException {  String sql = **"SELECT MaMH, TenMH, SoTiet, MH.MaBM, TenBM "**  + **"SELECT MaMH, TenMH, SoTiet, MH.MaBM, TenBM and MaMH='"** + maMH + **"'"**;  **return** cn.LoadData(sql);  }    **public** **void** InsertMonHoc(String maMH, String tenMH, **int** soTiet, String maBM)  **throws** SQLException {  String sql = **"INSERT INTO MonHoc values('"** + maMH + **"',N'"** + tenMH + **"',"** + soTiet + **",'"** + maBM + **"')"**;  cn.UpdateData(sql);  }   **public** **void** EditMonHoc(String maMH, String tenMH, **int** soTiet, String maBM) **throws** SQLException {  String sql = **"Update MonHoc Set TenMH = N'"** + tenMH + **"',SoTiet="** + soTiet + **",MaBM='"** + maBM + **"' "**  + **"Where MaMH='"** + maMH + **"'"**;  cn.UpdateData(sql);  }    **public** **void** DeleteMonHoc(String maMH) **throws** SQLException {  String sql = **"Delete from MonHoc where MaMH='"** + maMH + **"'"**;  cn.UpdateData(sql);  } } |

9. Cập nhật Code cho form: frmCapnhatMonHoc.Java. Ngoài các code tự sinh khi thiết kế giao diện dùng hình thức Design. Cần lần lượt sổ sung các code sau:

- Import các thư viện cần thiết:

|  |
| --- |
| **package** Interface;  **import** Process.MonHoc; **import** java.sql.ResultSet; **import** java.sql.SQLException; **import** java.util.logging.Level; **import** java.util.logging.Logger; **import** javax.swing.JOptionPane; **import** javax.swing.table.DefaultTableModel; |

- Xử lý hiện thị dữ liệu khi Form được mở:

|  |
| --- |
| **public** **class** frmCapNhatMonHoc **extends** javax.swing.JFrame {   **private** **final** MonHoc monHoc = **new** MonHoc();  **private** **final** DefaultTableModel tableModel = **new** DefaultTableModel();  **private** **boolean** isNew = **true**;    **public** **final** **void** LoadDataCombo() {  ResultSet result = **null**;  **try** {  result = monHoc.LoadBoMon();  **while** (result.next()) {  cboBoMon.addItem(result.getString(**"MaBM"**));  }  } **catch** (SQLException e) {  }  }   **public** **final** **void** LoadDataTable() **throws** SQLException {  ResultSet result = **null**;  result = monHoc.LoadMonHoc();  **try** {  ClearDataTable();  **while** (result.next()) {  String rows[] = **new** String[4];  rows[0] = result.getString(1);  rows[1] = result.getString(2);  rows[2] = result.getString(3);  rows[3] = result.getString(5);  tableModel.addRow(rows);  }  } **catch** (SQLException e) {  }  }   **private** **void** setEmptyTextBox() {  txtMaMH.setText(**null**);  txtMaMH.requestFocus();  txtTenMH.setText(**null**);  txtSoTiet.setText(**null**);  txtTenBM.setText(**null**);  }    **private** **void** lockTextBox(**boolean** e) {  txtMaMH.setEnabled(!e);  txtTenMH.setEnabled(!e);  txtSoTiet.setEnabled(!e);  cboBoMon.setEnabled(!e);  txtTenBM.setEnabled(!e);  }   **private** **void** enableButton(**boolean** e) {  btnThem.setEnabled(e);  btnXoa.setEnabled(e);  btnSua.setEnabled(e);    btnLuu.setEnabled(!e);  btnKhongLuu.setEnabled(!e);  btnThoat.setEnabled(e);  }    **public** **void** ClearDataTable() **throws** SQLException {  **int** n = tableModel.getRowCount() - 1;  **for** (**int** i = n; i >= 0; i--) {  tableModel.removeRow(i);  }  } |

- Xử lý load form bằng các phương thức trên

|  |
| --- |
| **public** frmCapNhatMonHoc() **throws** SQLException{  initComponents();  *// Define table columns*  String[] colsName = {**"Mã MH"**, **"Tên Môn Học"**, **"Số Tiết"**, **" Mã BM"**};  tableModel.setColumnIdentifiers(colsName);  jTableMonHoc.setModel(tableModel);    *// Load data table*  LoadDataTable();   *// Load combobox BoMon*  LoadDataCombo();   *// set empty all textboxes*  setEmptyTextBox();    *// Dsiable all textboxes*  lockTextBox(**true**);    *// Enable some buttons and disable others*  enableButton(**true**);  } |

Chú ý, hàm **main** nên chỉnh lại như sau để throws SQLException, bỏ phần phát sinh của chương trình

|  |
| --- |
| java.awt.EventQueue.invokeLater(**new** Runnable() {  **public** **void** run() {  **new** frmCapNhatMonHoc().setVisible(**true**);  }  }); |

Và thay bằng

|  |
| --- |
| **public** **static** **void** main(String args[]) **throws** SQLException {  frmCapNhatMonHoc f = **new** frmCapNhatMonHoc();  f.setVisible(**true**); } |

Bổ sung code cho sự kiên click của Table

|  |
| --- |
| **private** **void** jTableMonHocMouseClicked(java.awt.event.MouseEvent evt) {   **try** {  **int** row = jTableMonHoc.getSelectedRow();  String maMH = (jTableMonHoc.getModel().getValueAt(row, 0)).toString();  ResultSet rs = monHoc.LoadMonHoc(maMH);  **if** (rs.next())  {  txtMaMH.setText(rs.getString(**"MaMH"**));  txtTenMH.setText(rs.getString(**"TenMH"**));  txtSoTiet.setText(rs.getString(**"SoTiet"**));  cboBoMon.setSelectedItem(rs.getString(**"MaBM"**));  txtTenBM.setText(rs.getString(**"TenBM"**));  }  } **catch** (SQLException e) {  }  } |

* Bổ sung sự kiện khi combo box thay đổi giá trị:

|  |
| --- |
| **private** **void** cboBoMonItemStateChanged(java.awt.event.ItemEvent evt) {   **try** {  String maBM = cboBoMon.getSelectedItem().toString();  ResultSet result = monHoc.LoadBoMon(maBM);  **if** (result.next()) {  txtTenBM.setText(result.getString(**"TenBM"**));  }  } **catch** (SQLException ex) {  Logger.getLogger(frmCapNhatMonHoc.class.getName()).log(Level.SEVERE, **null**, ex);  } |

- Bổ sung sự kiện khi Click vào các jbutton:

* Nút “Thoát”

|  |
| --- |
| **private** **void** btnThoatActionPerformed(java.awt.event.ActionEvent evt) {   **this**.dispose();  } |

+ Nút “Xóa”

|  |
| --- |
| **private** **void** btnXoaActionPerformed(java.awt.event.ActionEvent evt) {   String maMH = txtMaMH.getText();  **try** {  **if**(maMH.length() == 0) {  JOptionPane.showMessageDialog(**null**, **"Can chon 1 mon hoc de xoa"**, **"Thong bao"**, 1);  } **else** {  **if** (JOptionPane.showConfirmDialog(**null**, **"Ban muon xoa mon hoc "** + maMH  + **" nay hay khong?"**, **"Thong bao"**, 2) == 0) {  monHoc.DeleteMonHoc(maMH);  ClearDataTable();  LoadDataTable();  setEmptyTextBox();  }  }  } **catch** (SQLException ex) {  Logger.getLogger(frmCapNhatMonHoc.class.getName()).log(Level.SEVERE, **null**, ex);  }  } |

- Xử lý nút “Thêm”

|  |
| --- |
| **private** **void** btnThemActionPerformed(java.awt.event.ActionEvent evt) {   setEmptyTextBox();  lockTextBox(**false**);  enableButton(**false**);  isNew = **true**;  } |

- Xử lý nút “Sửa”

|  |
| --- |
| **private** **void** btnSuaActionPerformed(java.awt.event.ActionEvent evt) {   String maMH = txtMaMH.getText();  **if** (maMH.length() == 0)  {  JOptionPane.showMessageDialog(**null**, **"Vui long chon loi can sua"**,  **"Thong Bao"**, 1);  } **else** {  lockTextBox(**false**);  enableButton(**false**);  isNew = **false**;  txtMaMH.setEnabled(**false**);  }  } |

- Xử lý nút “Lưu” và nút “Không Lưu”: tự làm