

Tuần 08:

CRUD JAVA SWING MYSQL

Cho form có giao diện sau đây:

STT	Họ & Tên	Giới Tính	Tuổi	Email	Số ĐT
1	Giang Vinh Kiệt	Nam	24	vinhkiet@gmail....	0907079829

Hướng dẫn

Bước 01: Tạo class Student.java và code như sau:

```
public class Student {
    int id;
    String fullname, gender, email, phone;
    int age;

    public Student(int id, String fullname, String gender, String email, String
phone, int age) {
        this.id = id;
        this.fullname = fullname;
        this.gender = gender;
        this.email = email;
        this.phone = phone;
        this.age = age;
    }

    public Student() { }

    public Student(String fullname, String gender, String email, String phone, int
age) {
        this.fullname = fullname;
        this.gender = gender;
        this.email = email;
        this.phone = phone;
        this.age = age;
    }

    public int getId() {
        return id;
    }

    public void setId(int id) {
        this.id = id;
    }
}
```

```

public String getFullname() {
    return fullname;
}

public void setFullname(String fullname) {
    this.fullname = fullname;
}

public String getGender() {
    return gender;
}

public void setGender(String gender) {
    this.gender = gender;
}

public String getEmail() {
    return email;
}

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

public String getPhone() {
    return phone;
}

public void setPhone(String phone) {
    this.phone = phone;
}

public int getAge() {
    return age;
}

public void setAge(int age) {
    this.age = age;
}
}

```

Bước 02: Tạo class StudentModify.java và code như sau:

```

import java.sql.Connection;
import java.sql.DriverManager;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import java.sql.SQLException;
import java.util.ArrayList;
import java.util.List;

public class StudentModify {
    static final String HOST = "jdbc:mysql://localhost/qlsv";
    static final String USERNAME = "root";
    static final String PASSWORD = "";

    public static List<Student> getStudentList(String s) {
        List<Student> dataList = new ArrayList<>();

        Connection con = null;
        try {
            //Lay thông tin dữ liệu từ database
            //B1. Kết nối tới data
            con = DriverManager.getConnection(HOST, USERNAME, PASSWORD);

            //B2. Thực hiện lấy dữ liệu
            String sql = "select * from students";

```

```

        if(s != null && !s.isEmpty()) {
            sql += " where fullname like '"+s+"'";
        }
        PreparedStatement statement = con.prepareStatement(sql);

        ResultSet resultSet = statement.executeQuery();

        //Fetch data
        while(resultSet.next()) {
            Student std = new Student(
                resultSet.getInt("id"),
                resultSet.getString("fullname"),
                resultSet.getString("gender"),
                resultSet.getString("email"),
                resultSet.getString("phone"),
                resultSet.getInt("age"));
            dataList.add(std);
        }
    } catch (SQLException ex) {
        System.out.println("Lỗi");
    } finally {
        //B3. Dong ket noi toi database
        if(con != null) {
            try {
                con.close();
            } catch (SQLException ex) {
                System.out.println("Lỗi");
            }
        }
    }

    return dataList;
}

public static void update(Student std) {
    Connection con = null;
    try {
        //Lay thong tin du lieu tu database
        //B1. Ket noi toi data
        con = DriverManager.getConnection(HOST, USERNAME, PASSWORD);

        //B2. Thuc hien lay du lieu
        String sql = "update students set fullname = ?, email = ?,
gender = ?, age = ?, phone = ? where id = ?";
        PreparedStatement statement = con.prepareStatement(sql);

        statement.setString(1, std.getFullname());
        statement.setString(2, std.getEmail());
        statement.setString(3, std.getGender());
        statement.setInt(4, std.getAge());
        statement.setString(5, std.getPhone());
        statement.setInt(6, std.getId());

        statement.execute();
    } catch (SQLException ex) {
        System.out.println("Lỗi");
    } finally {
        //B3. Dong ket noi toi database
        if(con != null) {
            try {
                con.close();
            } catch (SQLException ex) {
                System.out.println("Lỗi");
            }
        }
    }
}

```

```

    }

    public static void delete(int id) {
        Connection con = null;
        try {
            //Lay thông tin du lieu tu database
            //B1. Ket noi toi data
            con = DriverManager.getConnection(HOST, USERNAME, PASSWORD);

            //B2. Thuc hien lay du lieu
            String sql = "delete from students where id = ?";
            PreparedStatement statement = con.prepareStatement(sql);

            statement.setInt(1, id);

            statement.execute();
        } catch (SQLException ex) {
            System.out.println("Lỗi");
        } finally {
            //B3. Dong ket noi toi database
            if(con != null) {
                try {
                    con.close();
                } catch (SQLException ex) {
                    System.out.println("Lỗi");
                }
            }
        }
    }

    public static void insert(Student std) {
        Connection con = null;
        try {
            //Lay thông tin du lieu tu database
            //B1. Ket noi toi data
            con = DriverManager.getConnection(HOST, USERNAME, PASSWORD);

            //B2. Thuc hien lay du lieu
            String sql = "insert into students (fullname, email, gender,
age, phone) values (?, ?, ?, ?, ?)";
            PreparedStatement statement = con.prepareStatement(sql);

            statement.setString(1, std.getFullname());
            statement.setString(2, std.getEmail());
            statement.setString(3, std.getGender());
            statement.setInt(4, std.getAge());
            statement.setString(5, std.getPhone());

            statement.execute();
        } catch (SQLException ex) {
            System.out.println("lỗi");
        } finally {
            //B3. Dong ket noi toi database
            if(con != null) {
                try {
                    con.close();
                } catch (SQLException ex) {
                    System.out.println("Lỗi");
                }
            }
        }
    }
}

```

Bước 03: tạo class StudentFrame.java và code như sau:

```
import java.awt.event.MouseEvent;
```

```

import java.awt.event.MouseListener;
import java.util.List;

import javax.swing.JFrame;
import javax.swing.JOptionPane;
import javax.swing.table.DefaultTableModel;

public class StudentFrame extends JFrame {
    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JTextField ageTxt;
    private javax.swing.JButton deleteBtn;
    private javax.swing.JTextField emailTxt;
    private javax.swing.JButton findBtn;
    private javax.swing.JTextField fullnameTxt;
    private javax.swing.JComboBox<String> genderCb;
    private javax.swing.JButton jButton3;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JLabel jLabel4;
    private javax.swing.JLabel jLabel5;
    private javax.swing.JPanel jPanel1;
    private javax.swing.JScrollPane jScrollPane1;
    private javax.swing.JTextField phoneTxt;
    private javax.swing.JButton resetBtn;
    private javax.swing.JButton saveBtn;
    private javax.swing.JTable studentTable;
    private static final long serialVersionUID = 1L;
    // End of variables declaration//GEN-END:variables
    DefaultTableModel tableModel;
    List<Student> studentList;
    int currentIndex = -1;

    public StudentFrame() {
        initComponents();

        tableModel = (DefaultTableModel) studentTable.getModel();
        studentList = StudentModify.getStudentList(null);
        showStudents();

        //Bat su kien khi click vao item trong table
        studentTable.addMouseListener(new MouseListener() {
            @Override
            public void mouseClicked(MouseEvent e) {
            }

            @Override
            public void mousePressed(MouseEvent e) {
                int position = studentTable.getSelectedRow();
                currentIndex = position;

                fullnameTxt.setText(studentList.get(position).getFullname());
                emailTxt.setText(studentList.get(position).getEmail());
                ageTxt.setText(studentList.get(position).getAge() + "");
                phoneTxt.setText(studentList.get(position).getPhone());
                genderCb.setSelectedItem(studentList.get(position).getFullnam
                    e());
            }

            @Override
            public void mouseReleased(MouseEvent e) { }
            @Override
            public void mouseEntered(MouseEvent e) { }
            @Override
            public void mouseExited(MouseEvent e) { }
        });
    }
}

```

```

private void showStudents() {
    tableModel.setRowCount(0);

    for (Student student : studentList) {
        tableModel.addRow(new Object[] {
            tableModel.getRowCount() + 1,
            student.getFullname(),
            student.getGender(),
            student.getAge(),
            student.getEmail(),
            student.getPhone(),
        });
    }
}

private void initComponents() {

    jButton3 = new javax.swing.JButton();
    jPanel1 = new javax.swing.JPanel();
    jLabel1 = new javax.swing.JLabel();
    fullnameTxt = new javax.swing.JTextField();
    jLabel2 = new javax.swing.JLabel();
    genderCb = new javax.swing.JComboBox<>();
    jLabel3 = new javax.swing.JLabel();
    ageTxt = new javax.swing.JTextField();
    jLabel4 = new javax.swing.JLabel();
    emailTxt = new javax.swing.JTextField();
    jLabel5 = new javax.swing.JLabel();
    phoneTxt = new javax.swing.JTextField();
    saveBtn = new javax.swing.JButton();
    resetBtn = new javax.swing.JButton();
    deleteBtn = new javax.swing.JButton();
    findBtn = new javax.swing.JButton();
    jScrollPane1 = new javax.swing.JScrollPane();
    studentTable = new javax.swing.JTable();

    jButton3.setText("Reset");

    setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
    setTitle("Quản Lý Sinh Viên");

    jPanel1.setBorder(javax.swing.BorderFactory.createTitledBorder("NHẬP THÔNG TIN SINH VIÊN"));
    jPanel1.setToolTipText("");
    jLabel1.setText("Họ Tên: ");
    jLabel2.setText("Giới Tính:");
    genderCb.setModel(new javax.swing.DefaultComboBoxModel<>(new String[] { "-- Chon --", "Nam", "Nữ", "Khác" }));
    jLabel3.setText("Tuổi:");
    jLabel4.setText("Email:");
    jLabel5.setText("Số ĐT:");
    saveBtn.setText("Lưu Trữ");
    saveBtn.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            saveBtnActionPerformed(evt);
        }
    });

    resetBtn.setText("Làm Mới");
    resetBtn.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            resetBtnActionPerformed(evt);
        }
    });

    deleteBtn.setText("Xóa");
    deleteBtn.addActionListener(new java.awt.event.ActionListener() {

```

```

        public void actionPerformed(java.awt.event.ActionEvent evt) {
            deleteBtnActionPerformed(evt);
        }
    });

    findBtn.setText("Tìm Kiếm");
    findBtn.addActionListener(new java.awt.event.ActionListener() {
        public void actionPerformed(java.awt.event.ActionEvent evt) {
            findBtnActionPerformed(evt);
        }
    });

    javax.swing.GroupLayout jPanel1Layout = new
    javax.swing.GroupLayout(jPanel1);
    jPanel1.setLayout(jPanel1Layout);
    jPanel1Layout.setHorizontalGroup(

    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(15, 15, 15)

            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(jLabel1)
                .addComponent(jLabel2)
                .addComponent(jLabel3)
                .addComponent(jLabel4)
                .addComponent(jLabel5))
            .addGap(53, 53, 53)

            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
                .addComponent(saveBtn)

                .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.RELATED)
                    .addComponent(resetBtn)

                    .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.RELATED)
                        .addComponent(deleteBtn)

                        .addComponent(findBtn))
                    .addComponent(phoneTxt, javax.swing.GroupLayout.PREFERRED_SIZE, 433,
                        javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(emailTxt, javax.swing.GroupLayout.PREFERRED_SIZE, 433,
                        javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(ageTxt, javax.swing.GroupLayout.PREFERRED_SIZE, 433,
                        javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(genderCb, javax.swing.GroupLayout.PREFERRED_SIZE,
                        javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE)
                    .addComponent(fullnameTxt, javax.swing.GroupLayout.PREFERRED_SIZE, 433,
                        javax.swing.GroupLayout.PREFERRED_SIZE))
                .addGap(23, Short.MAX_VALUE))
        );
    jPanel1Layout.setVerticalGroup(
    jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addGroup(jPanel1Layout.createSequentialGroup()
            .addGap(12, 12, 12)
            .addGroup(jPanel1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASELINE)
                .addComponent(jLabel1)
                .addComponent(fullnameTxt, javax.swing.GroupLayout.PREFERRED_SIZE,
                    javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
            .addGap(18, 18, 18)

```

```

.addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
.addComponent(jLabel2)
.addComponent(genderCb, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(18, 18, 18)
.addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
.addComponent(jLabel3)
.addComponent(ageTxt, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(18, 18, 18)
.addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
.addComponent(jLabel4)
.addComponent(emailTxt, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.UNRELATED)
.addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
.addComponent(jLabel5)
.addComponent(phoneTxt, javax.swing.GroupLayout.PREFERRED_SIZE,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.PREFERRED_SIZE))
.addGap(18, 18, 18)
.addGroup(jPanell1Layout.createParallelGroup(javax.swing.GroupLayout.Alignment.BASE
LINE)
.addComponent(saveBtn)
.addComponent(resetBtn)
.addComponent(deleteBtn)
.addComponent(findBtn))
.addContainerGap(23, Short.MAX_VALUE)
);

studentTable.setModel(new javax.swing.table.DefaultTableModel(
    new Object [][] {
    },
    new String [] {
        "STT", "Họ & Tên", "Giới Tính", "Tuổi", "Email", "Số ĐT"
    }
) {
    boolean[] canEdit = new boolean [] {
        false, false, false, false, false, false
    };

    public boolean isCellEditable(int rowIndex, int columnIndex) {
        return canEdit [columnIndex];
    }
});
jScrollPane.setViewportView(studentTable);
if (studentTable.getColumnModel().getColumnCount() > 0) {
    studentTable.getColumnModel().getColumn(0).setResizable(false);
    studentTable.getColumnModel().getColumn(1).setResizable(false);
    studentTable.getColumnModel().getColumn(2).setResizable(false);
    studentTable.getColumnModel().getColumn(3).setResizable(false);
    studentTable.getColumnModel().getColumn(4).setResizable(false);
    studentTable.getColumnModel().getColumn(5).setResizable(false);
}

javax.swing.GroupLayout layout = new
javax.swing.GroupLayout(getContentPane());
getContentPane().setLayout(layout);
layout.setHorizontalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()

```



```

        .addContainerGap()

    .addGroup(layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
        .addComponent(jPanell1,
javax.swing.GroupLayout.DEFAULT_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE)
        .addComponent(jScrollPane1))
        .addContainerGap()
    );
    layout.setVerticalGroup(

layout.createParallelGroup(javax.swing.GroupLayout.Alignment.LEADING)
    .addGroup(layout.createSequentialGroup()
        .addContainerGap()
        .addComponent(jPanell1,
javax.swing.GroupLayout.PREFERRED_SIZE, javax.swing.GroupLayout.DEFAULT_SIZE,
javax.swing.GroupLayout.PREFERRED_SIZE)

    .addPreferredGap(javax.swing.LayoutStyle.ComponentPlacement.RELATED)
        .addComponent(jScrollPane1,
javax.swing.GroupLayout.PREFERRED_SIZE, 181,
javax.swing.GroupLayout.PREFERRED_SIZE)
        .addContainerGap(javax.swing.GroupLayout.DEFAULT_SIZE,
Short.MAX_VALUE))
    );

    pack();
} // </editor-fold> // GEN-END: initComponents

    private void saveBtnActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_saveBtnActionPerformed
    // TODO add your handling code here:
    String fullname = fullnameTxt.getText();
    int age = Integer.parseInt(ageTxt.getText());
    String email = emailTxt.getText();
    String phone = phoneTxt.getText();
    String gender = genderCb.getSelectedItem().toString();

    if(currentIndex >= 0) {
        studentList.get(currentIndex).setFullname(fullname);
        studentList.get(currentIndex).setAge(age);
        studentList.get(currentIndex).setEmail(email);
        studentList.get(currentIndex).setPhone(phone);
        studentList.get(currentIndex).setGender(gender);

        StudentModify.update(studentList.get(currentIndex));

        currentIndex = -1;
    } else {
        Student std = new Student(fullname, gender, email, phone, age);
        StudentModify.insert(std);
        studentList = StudentModify.getStudentList(null);
    }

    resetBtnActionPerformed(null);

    showStudents();
} // GEN-LAST:event_saveBtnActionPerformed

    private void resetBtnActionPerformed(java.awt.event.ActionEvent evt)
{ // GEN-FIRST:event_resetBtnActionPerformed
    // TODO add your handling code here:
    fullnameTxt.setText("");
    ageTxt.setText("");
    emailTxt.setText("");
    phoneTxt.setText("");

```

```

        genderCb.setSelectedIndex(0);
    } //GEN-LAST:event_resetBtnActionPerformed

    private void deleteBtnActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_deleteBtnActionPerformed
        // TODO add your handling code here:
        if(currentIndex < 0) {
            JOptionPane.showMessageDialog(rootPane, "Chua co sinh vien nao
dc chon!!!");
            return;
        }
        int option = JOptionPane.showConfirmDialog(rootPane, "Ban chac chan
muon xoa sinh vien nay khong?");
        System.out.println("option: " + option);
        if(option != 0) {
            return;
        }

        StudentModify.delete(studentList.get(currentIndex).getId());
        studentList.remove(currentIndex);
        currentIndex = -1;

        showStudents();
    } //GEN-LAST:event_deleteBtnActionPerformed

    private void findBtnActionPerformed(java.awt.event.ActionEvent evt)
    { //GEN-FIRST:event_findBtnActionPerformed
        // TODO add your handling code here:
        String s = JOptionPane.showInputDialog(rootPane, "Nhap ten sinh vien
can tim kiem");
        studentList = StudentModify.getStudentList(s);

        showStudents();
    } //GEN-LAST:event_findBtnActionPerformed

    public static void main(String args[]) {
        /* Set the Nimbus look and feel */
        //<editor-fold defaultstate="collapsed" desc=" Look and feel setting
code (optional) ">
        /* If Nimbus (introduced in Java SE 6) is not available, stay with
the default look and feel.
         * For details see
http://download.oracle.com/javase/tutorial/uiswing/lookandfeel/plaf.html
         */
        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {
            java.util.logging.Logger.getLogger(StudentFrame.class.getName()).log(java.util.log
ging.Level.SEVERE, null, ex);
        } catch (InstantiationException ex) {
            java.util.logging.Logger.getLogger(StudentFrame.class.getName()).log(java.util.log
ging.Level.SEVERE, null, ex);
        } catch (IllegalAccessException ex) {
            java.util.logging.Logger.getLogger(StudentFrame.class.getName()).log(java.util.log
ging.Level.SEVERE, null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

```

```
java.util.logging.Logger.getLogger(StudentFrame.class.getName()).log(java.util.logging.Level.SEVERE, null, ex);
    }
    //</editor-fold>

    /* Create and display the form */
    java.awt.EventQueue.invokeLater(new Runnable() {
        public void run() {
            new StudentFrame().setVisible(true);
        }
    });
}
```