

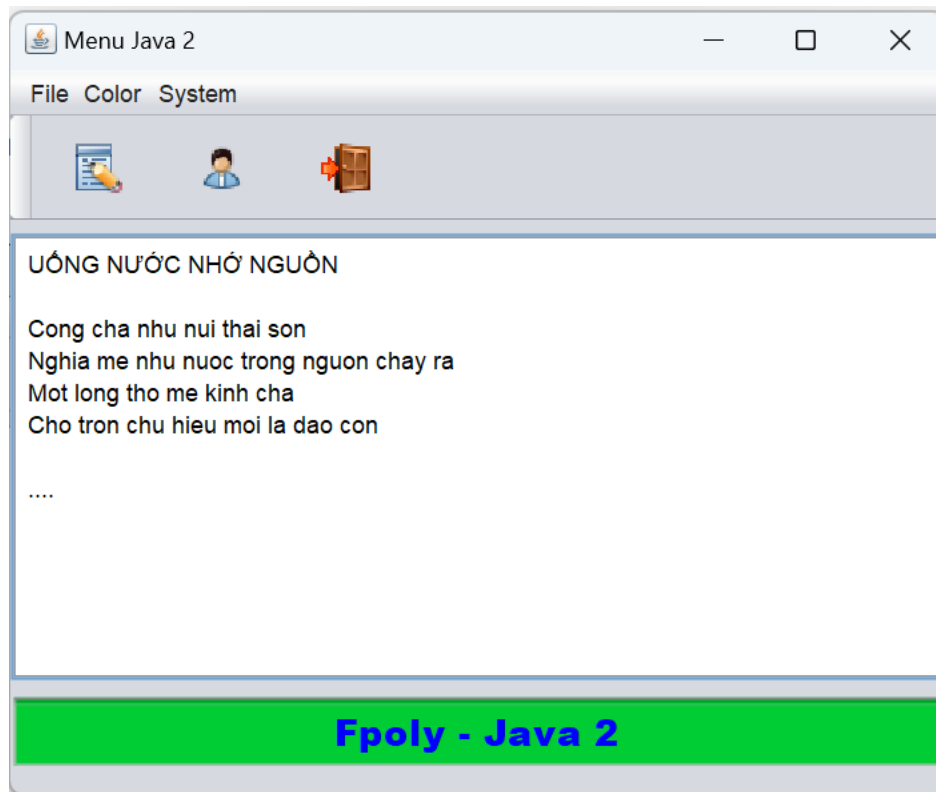
MỤC TIÊU:

Kết thúc bài thực hành này bạn có khả năng:

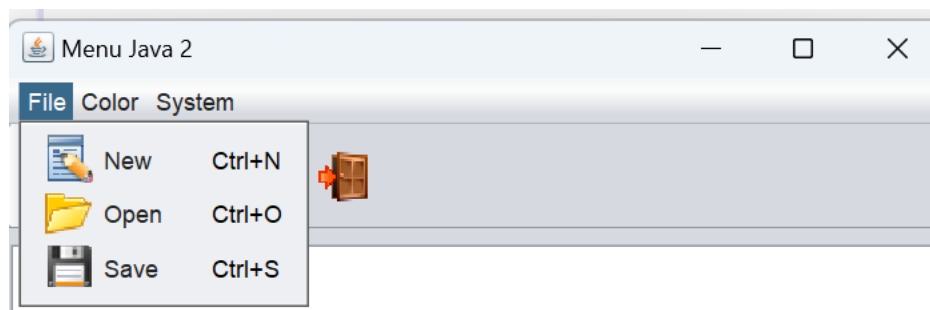
- ✓ Hiểu được và ứng dụng menu vào bài làm
- ✓ Cách dùng JFileChooser, cách dùng JcolorChooser
- ✓ Cách lưu file, mở file

BÀI 1 (5 ĐIỂM)

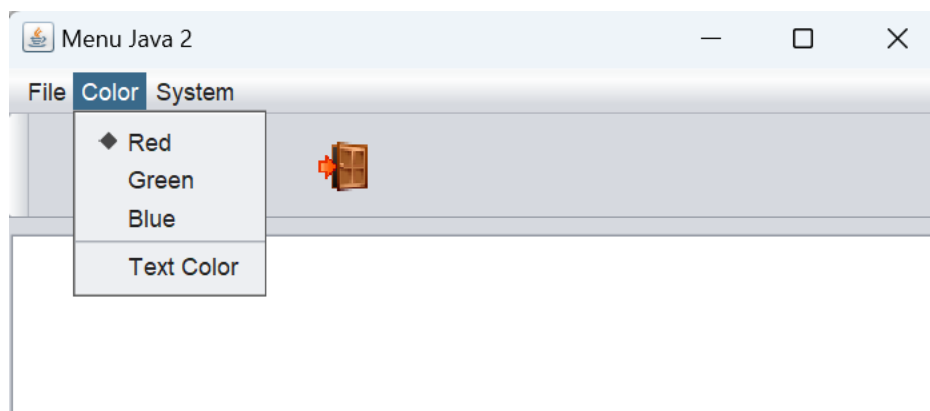
Thiết kế ứng dụng như hình và thực hiện các chức năng theo yêu cầu sau:



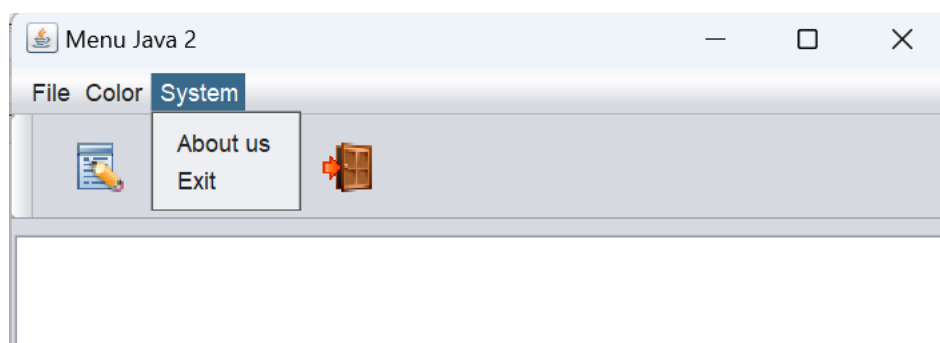
File menu



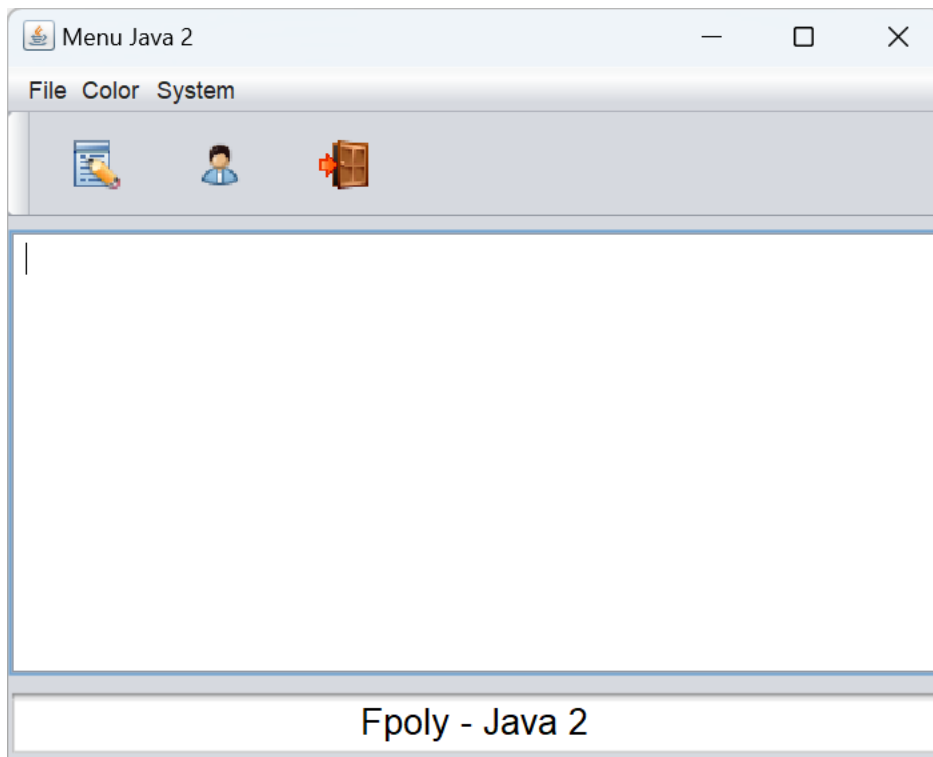
Color menu



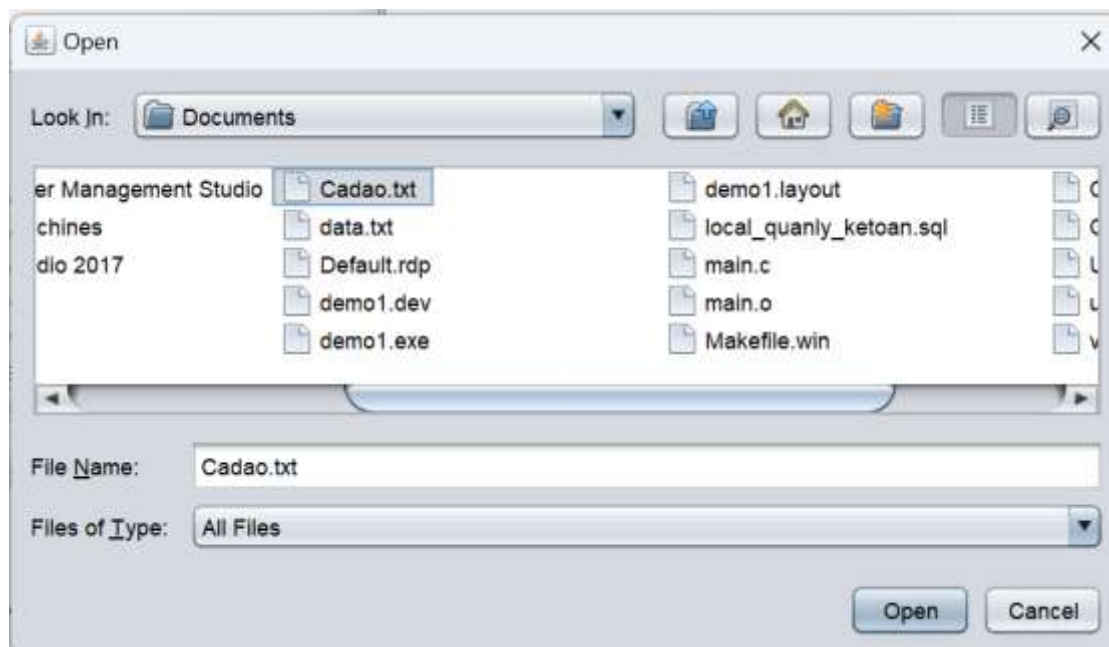
System menu



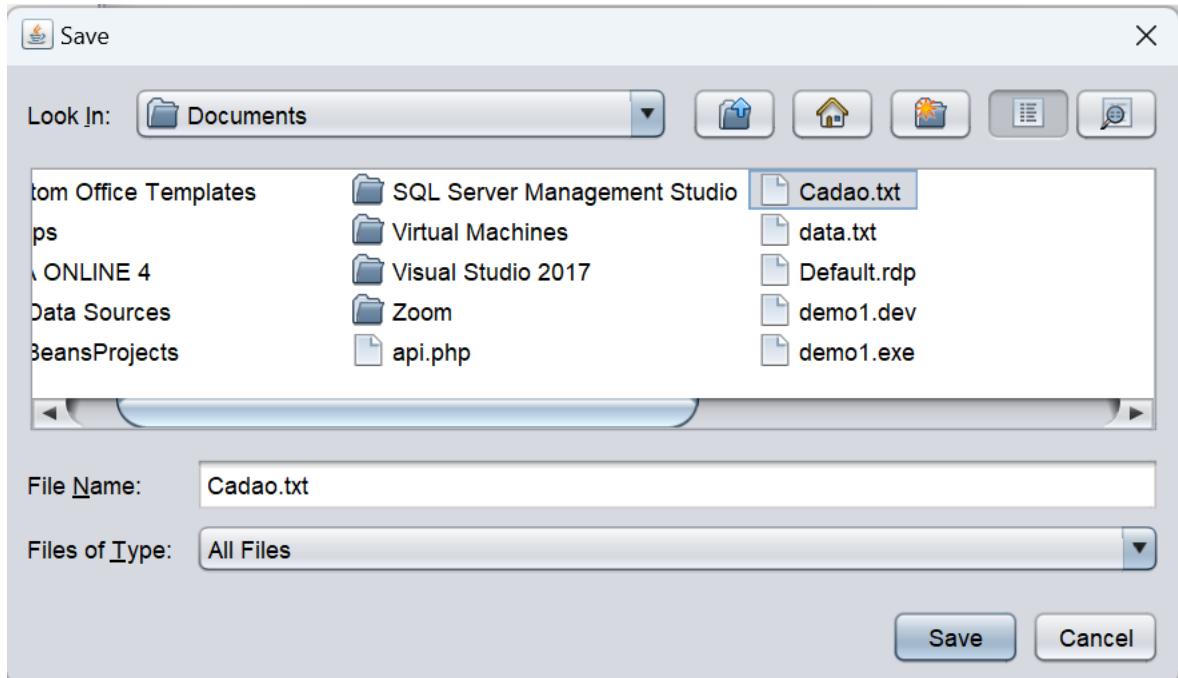
1. Khi người dùng chọn New menu, xóa dữ liệu text hiện tại cho người dùng text mới



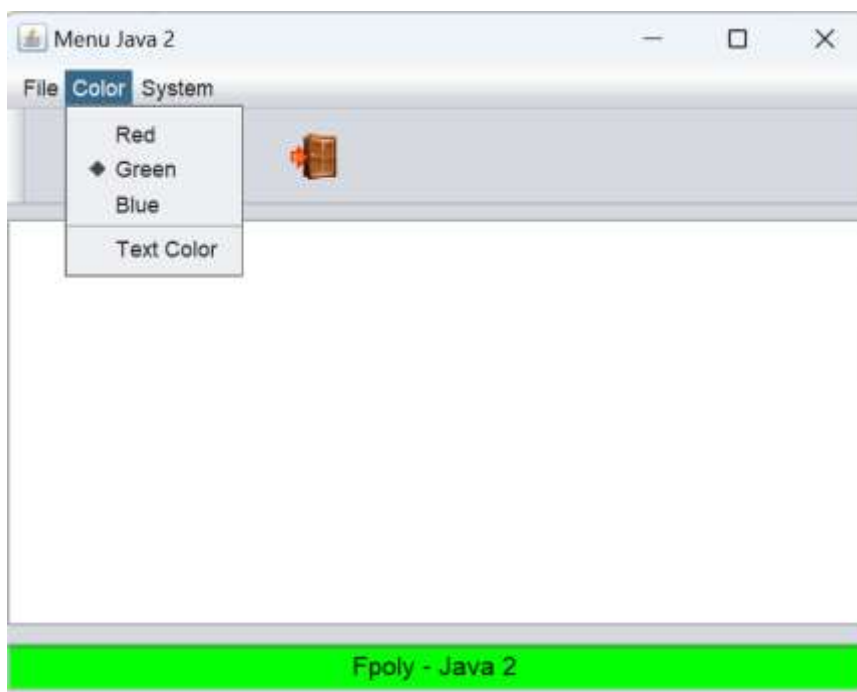
2. Khi người dùng chọn Open menu, hiển thị hộp thoại cho phép người dùng chọn một tập tin và đọc nội dung tập tin đó hiển thị ra màn hình



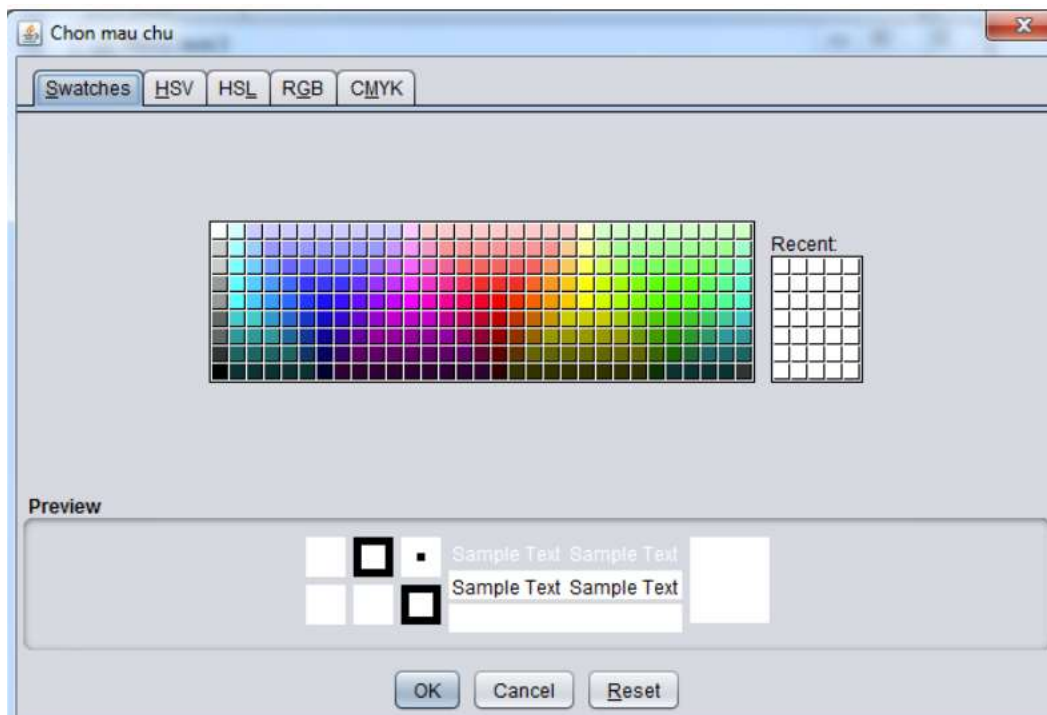
3. Khi người dùng chọn Save menu, hiển thị hộp thoại cho phép người dùng lưu các nội dung hiện tại vào file



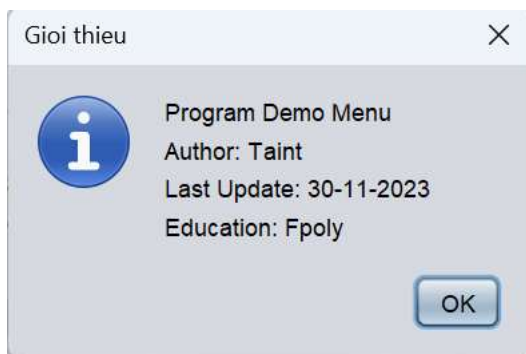
4. Khi người dùng chọn “Color menu”, ứng dụng sẽ thiết lập màu nền tương ứng cho textbox “Floly – Java 2”



5. Khi người dùng chọn “Text menu”, ứng dụng sẽ hiển thị hộp thoại cho người dùng chọn màu, lấy giá trị màu đã chọn set cho text “Fpoly – Java 2”

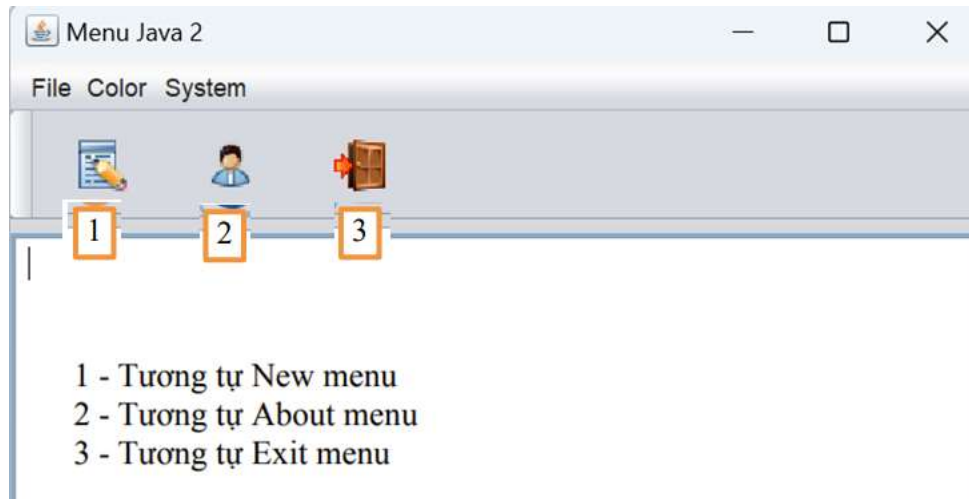


6. Khi người dùng chọn About menu, ứng dụng hiển thị thông tin giới thiệu



7. Khi người dùng chọn Exit menu, ứng dụng sẽ bị đóng

Xử lý tool bar



Tham khảo code

```
private void mnFileOpenActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    JFileChooser fileChooser = new JFileChooser();
    int result = fileChooser.showOpenDialog(parent: null);
    if(result == JFileChooser.APPROVE_OPTION) {
        try {
            edPage.setPage("file:/// " + fileChooser.getSelectedFile().getPath());
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
private void mnFileSaveActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    JFileChooser fileChooser = new JFileChooser();
    int result = fileChooser.showSaveDialog(parent: null);
    if(result == JFileChooser.APPROVE_OPTION) {
        File fname = fileChooser.getSelectedFile();
        try {
            PrintWriter pw = new PrintWriter(file: fname);
            pw.println(x: edPage.getText());
            pw.close();
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

```
private void mnAboutActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    SimpleDateFormat df = new SimpleDateFormat("dd-MM-yyyy");
    Date now = new Date();

    String str = "Program Demo Menu";
    str += "\nAuthor: Taint";
    str += "\nLast Update: " + df.format(now);
    str += "\nEducation: Fpoly";
    JOptionPane.showMessageDialog(null, str, title: "Giới thiệu", messageType: JOptionPane.INFORMATION_MESSAGE);
}

private void mnColorTextColorActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    Color c = JColorChooser.showDialog(component: null, title: "Chọn màu chữ", initialColor: Color.white);
    txtFpoly.setForeground(c);
}

private void mnColorRedActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    txtFpoly.setBackground(bg: Color.red);
}
```

BÀI 2 (2 ĐIỂM)

Xây dựng một thư viện tiện ích XFile gồm các hàm readObject() và writeObject() cho phép đọc và ghi đối tượng:

HƯỚNG DẪN:

✓ Tạo lớp Xfile

```
package poly.io;

import java.io.FileInputStream;
import java.io.FileOutputStream;

public class XFile {
    /**
     * Đọc file đối tượng
     * @param path là đường dẫn file cần đọc
     * @return đối tượng đọc được
     * @throws đọc file có lỗi
     */
    public static Object readObject(String path) {...}
    /**
     * Ghi file đối tượng
     * @param path là đường dẫn file cần ghi
     * @param object đối tượng cần ghi vào file
     * @throws đọc file có lỗi
     */
    public static void writeObject(String path, Object object) {...}
}
```

✓ Viết mã cho hàm readObject

```
try {
    ObjectInputStream ois =
        new ObjectInputStream(new FileInputStream(path));
    Object object = ois.readObject();
    ois.close();
    return object;
}
catch (Exception e) {
    throw new RuntimeException(e);
}
```


- ✓ Viết mã cho hàm writeObject

```
try {  
    ObjectOutputStream oos =  
        new ObjectOutputStream(new FileOutputStream(path));  
    oos.writeObject(object);  
    oos.close();  
}  
catch (Exception e) {  
    throw new RuntimeException(e);  
}
```

BÀI 3 (2 ĐIỂM)

Viết chương trình quản lý nhân viên có giao diện như sau. Sử dụng thư viện XFile để đọc ghi List<Staff>

HỌ VÀ TÊN	LƯƠNG
Tuấn	5000000.0
Cường	9000000.0
Phương	6000000.0

1. Thiết kế giao diện như trên
2. Xây dựng lớp Staff

```
public class Staff implements Serializable{
    public String fullname;
    public double salary;
}
```

3. Bổ sung mã vào JFrame

```
// Nắm giữ danh sách nhân viên nhập từ người dùng
List<Staff> list = new ArrayList<>();

// Tạo một nhân viên mới và bổ sung vào List<Staff>
private void addStaff() {...}

// Hiển thị List<Staff> lên bảng
private void fillToTable() {...}
```

4. Viết mã cho nút [THÊM]

✓ Xử lý sự kiện click

```
private void btnThemActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    this.addStaff();
    this.fillToTable();
}
```

✓ Mã phương thức addStaff()

```
/*
 * Tạo nhân viên với thông tin nhập từ form
 */
Staff nv = new Staff();
nv.fullname = txtHoTen.getText();
nv.salary = Double.parseDouble(txtLuong.getText());
// Bổ sung nhân viên vào List<Staff>
list.add(nv);
```

✓ Mã phương thức fillToTable()

```

/*
 * Lấy mô hình dữ liệu của bảng và xóa sách các hàng
 */
DefaultTableModel model =
    (DefaultTableModel) tblStaffs.getModel();
model.setRowCount(0);
/*
 * Duyệt List<Staff> và bổ sung các nhân viên vào bảng
 */
for(Staff nv : list){
    Object[] row = new Object[]{nv.fullname, nv.salary};
    model.addRow(row);
}

```

5. Viết mã cho nút [LƯU]

```

private void btnLuuActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    XFile.writeObject("c:/temp/staffs.dat", list); // lưu list vào file
}

```

6. Viết mã cho nút [ĐỌC]

```

private void btnDocActionPerformed(java.awt.event.ActionEvent evt) {
    // TODO add your handling code here:
    list = (List<Staff>) XFile.readObject("c:/temp/staffs.dat"); // đọc list từ file
    this.fillToTable();
}

```

BÀI 4: Giảng viên cho thêm (1 ĐIỂM)

*** YÊU CẦU NỘP BÀI:

Sv nén file bao gồm các yêu cầu đã thực hiện trên, nộp lms đúng thời gian quy định của giảng viên. Không nộp bài coi như không có điểm.

--- Hết ---

