I. MỤC TIÊU:

- Tìm hiểu các thành phần giao diện cơ bản trong java:
 - JTextField.
 - **JButton**
 - **JCheckbox**
 - JRadioButton ButtonGroup
 - **JComboBox**
 - JList DefaultListModel
 - JTable DefaultTableModel
 - **JOptionPane**

II. TÓM TẮT LÝ THUYẾT:

1. JTextField: ITextField

- setText(String s): gán giá trị cho textbox
- getText(): lấy giá trị từ textbox

2. JButton: JButton

Sự kiện ActionListener: xảy ra khi nút lệnh được nhấn public void actionPerform(ActionEvent e) { //Code xử lý sự kiện }

3. JCheckBox: ✓ JCheckBox

- setSelected(boolean b): check vào checkbox.
- getSelected(): lấy trang thái của checkbox : true/false

4. JRadioButton - ButtonGroup: • JRadioButton

- setSelected(boolean b): check vào radiobutton.
- getSelected(): lấy trạng thái của radiobutton: true/false
- Các radiobutton phải được nhóm lại bằng đối tượng ButtonGroup 🔭 buttonGroup thì mới có thể chon được.
 - add (AbstractButton button);

5. JComboBox: SComboBox

- Hiển thi một danh sách các giá tri, chỉ được chon một
- getSelectedIndex(): trả về vị trí được chọn, 0 là vị trí đầu tiên, -1 là không có phần tử nào được chọn.
- setSeletectedIndex(int index): thiết lập vị trí được chọn trong danh sách.

6. JList – DefaultListModel: List

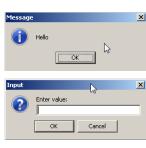
- Hiển thị một danh sách các giá trị, có thể chọn một hoặc nhiều
- Hiện thực theo cơ chế MVC: View→JList, Model→DefaultListModel
- JList:
 - setModel(ListModel model): gắn kết list với model.
 - getSelectedIndex(): trả về vị trí được chọn, 0 là vị trí đầu tiên, -1 nếu
 không có phần tử nào được chọn
 - setSelectedIndex(): đặt vị trí được chọn.
 - getSelectedItem(): trả về đối tượng được chọn
- DefaultListModel:
 - add(Object o): thêm phần tử vào danh sách
 - remove(int index): xóa phần tử tại vị trí index

7. JTable - DefaultTableModel:

- Hiển thị một danh sách dưới dạng bảng, có thể chọn một hoặc nhiều
- Hiện thực theo cơ chế MVC: View→JTable, Model→DefaultTableModel
- JTable:
 - setModel(TableModel model): gắn kết table với model.
 - getSelectedRow(): trả về dòng được chọn, 0 là vị trí đầu tiên, -1 nếu không có dòng nào nào được chọn
- DefaultTableModel:
 - addColoumn(Object o): thêm cột vào table.
 - addRow(Object[] rowData): thêm dòng dữ liệu vào table
 - removeRow(int index): xóa dòng tại vị trí index.

8. JOptionPane:

- showMessageDialog():
- showInputDialog():
- showConfirmDialog():

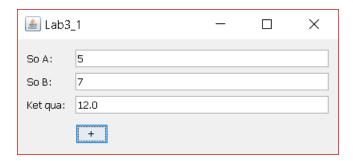




III. NỘI DUNG THỰC HÀNH:

Sinh viên import file lab3.zip vào netbeans, thực hiện các bài lab sau:

1. Mở Lab3_1.java:

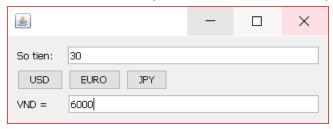


Yêu cầu:

- Chạy chương trình, quan sát kết quả.
- Hãy thiết kế thêm 3 button -, x, /. Viết code xử lý sự kiện cho các nút đó và xuất kết quả trong textbox.

Bài tập 1:

Thiết kế Frame như hình, viết code xử lý cho các nút trên frame. Biết rằng 1USD = 22800 VND, 1 EURO = 27800 VND, 1 JPY = 200 VND



2. Mở Lab3_2.java:



Yêu cầu:

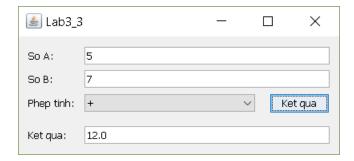
- Chạy chương trình, quan sát kết quả.
- Hãy viết code xử lý cho tùy chọn -, *, /.

Bài tập 2:

Thiết kế Frame như hình, viết code xử lý cho nút Đổi tiền.



3. Mở Lab3_3.java:

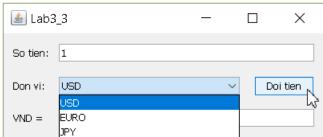


Yêu cầu:

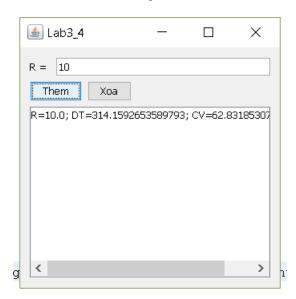
- Chạy chương trình, quan sát kết quả
- Hãy thêm vào các phép tính -, *, / vào combobox, viết code xử lý cho các phép tính trên.

Bài tập 3:

Thiết kế Frame như hình, viết code xử lý cho nút Đổi tiền.



4. Mở Lab3_4.java:



Yêu cầu:

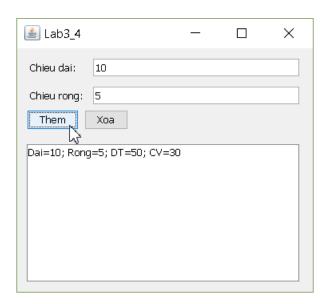
- Chạy chương trình, quan sát kết quả
- Hãy viết lệnh cho nút Xoa: xóa item đã chọn trong list.

<u>Hướng dẫn:</u>

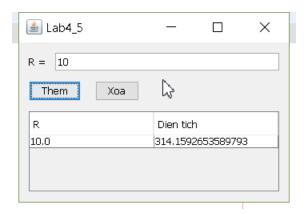
- Dùng phương thức **getSelectedIndex**() của Jlist lấy vị trí item được chọn.
- Dùng phương thức remove() của model xóa item tại vị trí được chọn.

Bài tập 4:

- Tạo lớp **HinhChuNhat** có 2 thuộc tính, chiều dài, chiều rộng, có các phương thức getDientich(), getChuvi().
- Thiết kế Frame như hình, viết code xử lý cho nút Them: thêm đối tượng HinhChuNhat vào JList và Xoa: xóa đối tượng đã chọn.



5. Mở Lab3_5.java:



Yêu cầu:

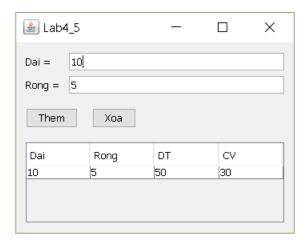
- Chạy chương trình, quan sát kết quả
- Hãy thêm cột Chu vi hiển thị chu vi đường tròn, viết lệnh cho nút Xoa: xóa item đã chọn trong list.

Hướng dẫn:

- Dùng phương thức **getSelectedRow**() của Jlist lấy vị trí item được chọn.
- Dùng phương thức removeRow() của model xóa item tại vị trí được chọn.
- Dùng phương thức **remove**() của list để xóa dữ liệu trong List.

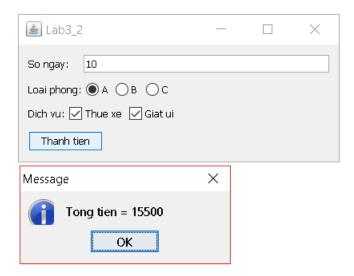
Bài tập 5:

- Sử dụng lại lớp HinhChuNhat ở bài tập 4.
- Thiết kế Frame như hình, viết code xử lý cho nút **Them**: thêm đối tượng HinhChuNhat vào JTable, nút Xoa: xóa đối tượng đã chọn.



IV. <u>BÀI TẬP LÀM THÊM:</u>

1. Thiết kế frame như hình:



Yêu cầu:

- Cách tính tiền phòng như sau:

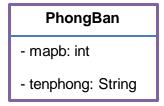
Loại phòng: Loại A: 1500; Loại B: 1200; Loại C: 1000

Dịch vụ: 300 thuê xe, 200 giặt ủi

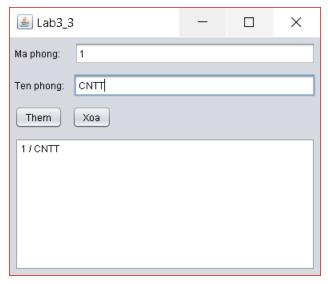
Thành tiền = Giáloại phòng x số ngày + các dịch vụ đi kèm

LAB 3: LẬP TRÌNH GIAO DIỆN

2. Tạo lớp PhongBan như sau:



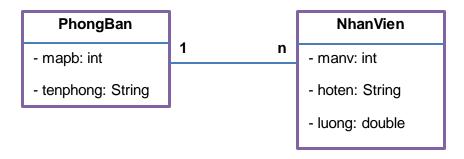
- Thiết kế Frame như hình:



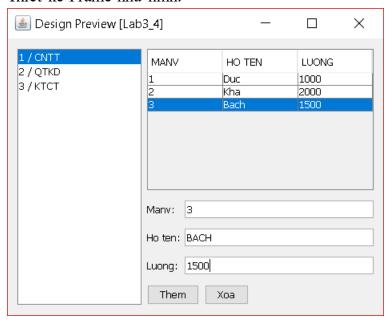
Yêu cầu:

- Khi kích vào nút Add: tạo đối tượng PhongBan, thêm vào JList
- Khi chọn đối tượng PhongBan trong Jlist, kích nút Xoa: xoa đối tượng đã chọn.

3. Tạo sơ đồ lớp như sau:

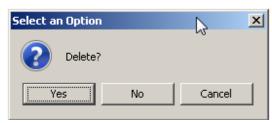


- Thiết kế Frame như hình:



Yêu cầu:

- Tạo danh sách PhongBan nạp vào Jlist
- Khi chọn đối tượng PhongBan, nạp danh sách NhanVien vào Jtable
- Khi kích nút **Them**: tạo đối tượng NhanVien, thêm vào PhongBan đã chọn.
- Khi kích nút **Xoa**:
 - Xuất hiện cảnh báo trước khi xóa



- Nếu chọn Yes: xóa NhanVien đã chọn.