I. MUC 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: TextField



- 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 trạng 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 lai bằng đối tương ButtonGroup 🔭 buttonGroup thì mới có thể chon được.
 - add (AbstractButton button);

5. JComboBox: GComboBox

- Hiển thị một danh sách các giá trị, chỉ được chọn 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: JList

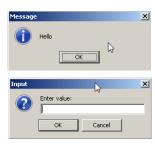
- Hiển thị một danh sách các giá trị, có thể chọn một hoặc nhiều
- Hiện thực theo cơ chế MVC: View→JList, Model→DefaultListModel
- JList:
 - setModel(ListModel model): gắn kết list với model.
 - getSelectedIndex(): trả về vị trí được chọn, 0 là vị trí đầu tiên, -1 nếu không có phần tử nào được chọn
 - 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: Table

- 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():





9. Mouse listener và mouse move event

- Để thêm event và move event trong swing ta sử dụng hàm addMouseListener và addMouseMotionListener và viết đè các hàm của lớp cha

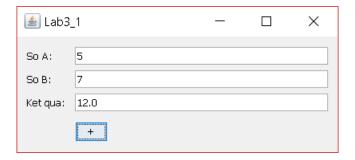
```
public void mouseReleased(MouseEvent arg0) {
               // TODO Auto-generated method stub
              textPane.setText("mouse release");
          }
         public void mousePressed(MouseEvent arg0) {
              // TODO Auto-generated method stub
              textPane.setText("mouse pressed");
          }
         public void mouseExited(MouseEvent arg0) {
              // TODO Auto-generated method stub
              textPane.setText("mouse exited");
          }
         public void mouseEntered(MouseEvent arg0) {
              // TODO Auto-generated method stub
              textPane.setText("mouse entered");
          }
         public void mouseClicked(MouseEvent arg0) {
              // TODO Auto-generated method stub
              textPane.setText("mouse clicked");
          }
public void mouseMoved(MouseEvent arg0) {
 // TODO Auto-generated method stub
  textPane.setText(String.format("Move mouse x: %d, y: %d", arg0.getX(), arg0.getY()));
}
public void mouseDragged(MouseEvent arg0) {
  // TODO Auto-generated method stub
  textPane.setText(String.format("dragged mouse x: %d, y: %d", arg0.getX(), arg0.getY()));
```

III. NỘI DUNG THỰC HÀNH:

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

Bài tập 1:

LAB 3: LẬP TRÌNH GIAO DIỆN

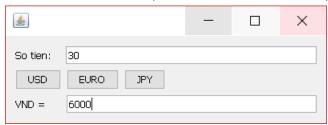


Yêu cầu:

- 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 2:

- 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



Bài tập 3:

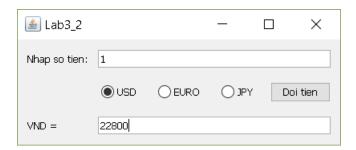


Yêu cầu:

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

Bài tập 4:

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



Bài tập 5:

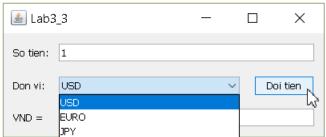


Yêu cầu:

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

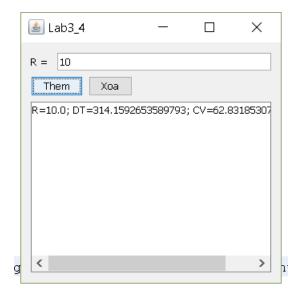
Bài tập 6:

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



Bài tập 7:

Thiết kế giao diện như sau, viết code và xử lý cho chương trinh sau:



Yêu cầu:

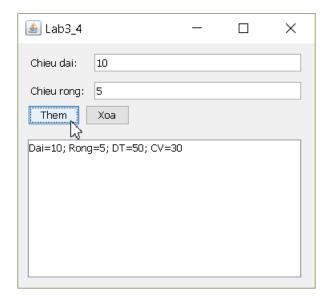
- Chạy chương trình, quan sát kết quả
- Hãy viết lệnh cho nút Xoa: xóa item đã chọn trong list.

Hướng dẫn:

- 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.

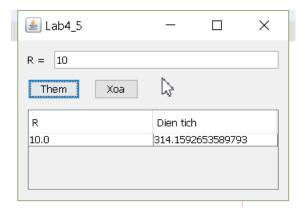
Bài tập 8:

- 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 tương đã chon.



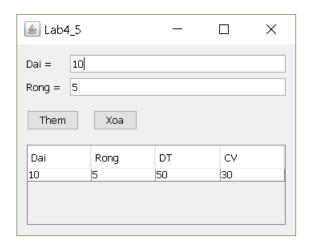
Bài tập 9:

Thiết kế giao diện và hàm xử lý như sau



<u>Bài tập 10:</u>

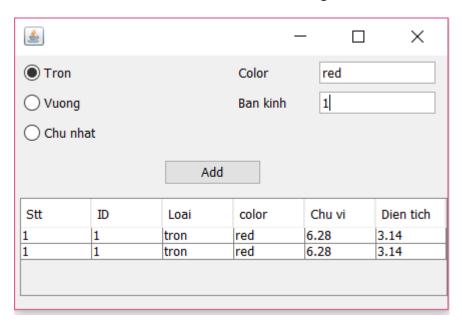
- Sử dụng lại lớp HinhChuNhat ở bài tập 4.
- Thiết kế Frame như hình, viết code xử lý cho nút **Them**: thêm đối tượng HinhChuNhat vào JTable, nút **Xoa**: xóa đối tượng đã chọn.



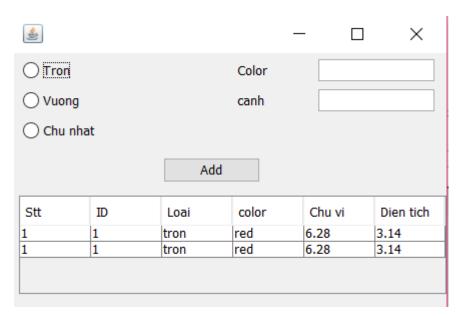
LAB 3: LẬP TRÌNH GIAO DIỆN

Bài tập 11: xây dựng chương trình có giao diện như sau:

- Khi nhấn vào radiobutton "Tron" thì giao diện như hình

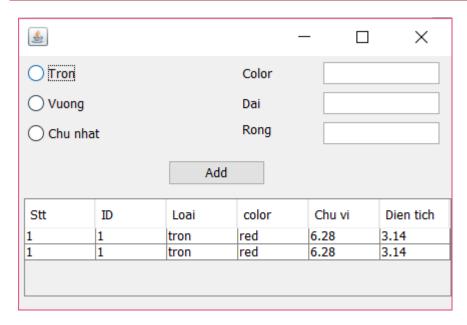


- Khi nhấn vào radio "Vuong" thì giao diện



- Khi nhấn vào radio "Chu nhat" thì giao diện như sau:

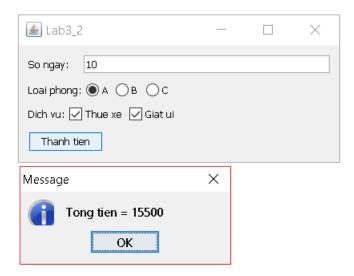
LAB 3: LẬP TRÌNH GIAO DIỆN



- Khi nhấn button Add thì thêm vào Array<đối tượng> với :
- Stt tăng dần
- Id random
- Loại tùy thuộc vào radio button
- Màu lấy từ editText Color
- Chu vi là hàm tính toán biết chu vi của tròn, vuông, chữ nhật có công thức :2pi*r, 4*a, 2*(dai+rong)
- Diện tích tương tự là: pi*r^2,a^2,dai*rong
- Viết các class cần thiết để viết chương trình này

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

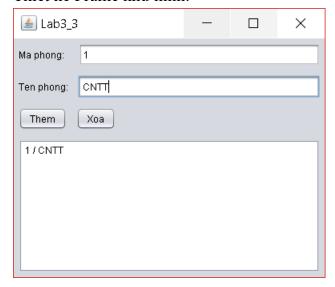
Dịch vụ: 300 thuê xe, 200 giặt ủi

Thành tiền = Giá loại phòng $x s \hat{o} ngày + các dịch vụ đi kèm$

2. Tạo lớp PhongBan như sau:



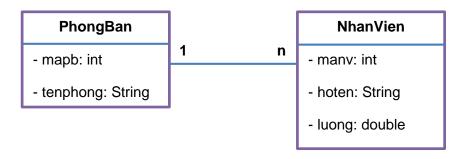
Thiết kế Frame như hình:



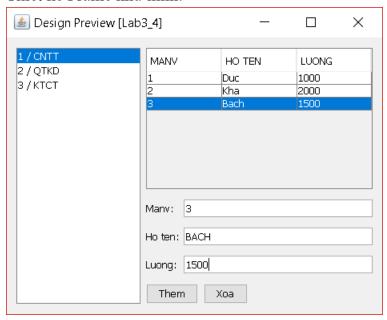
Yêu cầu:

- 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 chon.

3. Tạo sơ đồ lớp như sau:

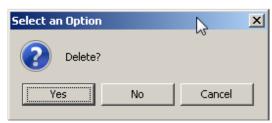


- Thiết kế Frame như hình:



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.