**Main GUI**

import java.awt.GridLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.\*;

@SuppressWarnings("serial")

public class MainGUI extends JFrame {

private JPanel buttons = new JPanel();

private JButton exit = new JButton("Exit");

private JButton addCust = new JButton("Add Customer");

private JButton viewCust = new JButton("View Customers");

private JButton invMenu = new JButton("View Inventory");

private GUIController control;

The above code is the graphic representation of the buttons that will show up on the display (Graphical User Interface). This is what the Pharmacist will see when adding information into the database. As can be seen above, the user has the options of “Adding Customer”, “View Customer”, and “View Inventory”.

public MainGUI() {

createButtons();

createPanel();

add(buttons);

setSize(300, 300);

setUndecorated(true);

}

The above code is formatting the GUI appearance, including the size and the panel to add the buttons.

public MainGUI(GUIController gc) {

this();

control = gc;

}

The above code is a constructor for the GUI Controller.

private void createPanel() {

buttons.setLayout(new GridLayout(4, 1, 0, 0));

buttons.add(addCust);

buttons.add(viewCust);

buttons.add(invMenu);

buttons.add(exit);

}

The above code creates the panel, and assigns each button the class to be accessed.

private void createButtons() {

exit.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent ae) {

control.setRun(false);

}

});

The above code adds the exit button's actionListener

addCust.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent ae) {

control.setMainRun(false);

control.setAddRun(true);

}

});

The above code adds the add button's actionListener

viewCust.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent ae) {

control.setMainRun(false);

control.setSearchCustRun(true);

}

});

The above code adds the view button's actionListener

invMenu.addActionListener(new ActionListener() {

@Override

public void actionPerformed(ActionEvent ae) {

control.setMainRun(false);

control.setInvRun(true);

}

});

The above code adds the inventory button's actionListener

}

}