Anthony Meunier

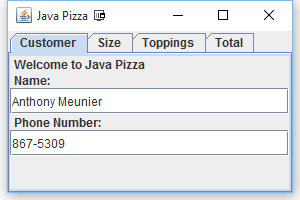
DeVry University

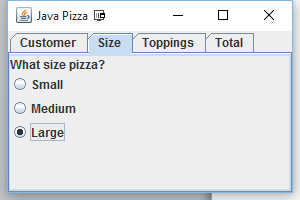
CIS 355A

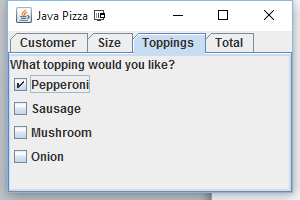
Week 6 iLab

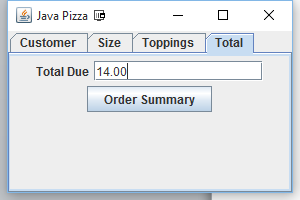
Swing and Database Connection

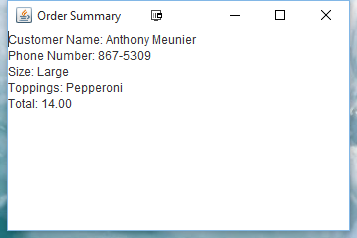
Step 1: JavaPizza











/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Program Name: JavaPizza.java

Programmer's Name: Anthony Meunier

Program Description: Application will collect customer data and preferences for ordering a pizza.

Will compute cost based on size, and toppinds selected.

Displays order summary which shows all input and selections.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**import** java.awt.Container;

**import** java.awt.FlowLayout;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** javax.swing.BoxLayout;

**import** javax.swing.ButtonGroup;

**import** javax.swing.ButtonModel;

**import** javax.swing.JButton;

**import** javax.swing.JCheckBox;

**import** javax.swing.JFrame;

**import** javax.swing.JLabel;

**import** javax.swing.JOptionPane;

**import** javax.swing.JPanel;

**import** javax.swing.JRadioButton;

**import** javax.swing.JTabbedPane;

**import** javax.swing.JTextArea;

**import** javax.swing.JTextField;

**import** javax.swing.event.ChangeEvent;

**import** javax.swing.event.ChangeListener;

**public** **class** JavaPizza **extends** JFrame **implements** ActionListener, ChangeListener {

/\*\*

\*

\*/

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

JTextField nameTextField, phoneTextField, totalTextField; // Text fields

JCheckBox pepperoniCheckBox, sausageCheckBox, mushroomCheckBox, onionCheckBox; // Check boxes

JButton summaryButton = **new** JButton("Order Summary"); // Summary Button

ButtonGroup btngroup = **new** ButtonGroup();

**private** **final** **int** SMALL\_PRICE = 8;

**private** **final** **int** MEDIUM\_PRICE = 10;

**private** **final** **int** LARGE\_PRICE = 12;

String pizzaSize = "";

String pizzaToppings = "";

**int** totalPrice;

JavaPizza() {

**super**("Java Pizza");

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

addComponentsToPane(getContentPane());

setSize(300, 200);

//pack();

setVisible(**true**);

}

**public** **void** addComponentsToPane(Container contentPane) {

JTabbedPane tabbedPane = **new** JTabbedPane();

// Add customer tab to the frame

JPanel customerPanel = customerTab();

tabbedPane.addTab("Customer", customerPanel);

// Add size tab to the frame

JPanel sizePanel = sizeTab();

tabbedPane.addTab("Size", sizePanel);

// Add topping tab to the frame

JPanel toppingPanel = toppingTab();

tabbedPane.addTab("Toppings", toppingPanel);

// Add total tab to the frame

JPanel totalPanel = totalTab();

tabbedPane.addTab("Total", totalPanel);

tabbedPane.addChangeListener(**this**);

add(tabbedPane);

}

@Override

**public** **void** stateChanged(ChangeEvent e) {

// Variable for number of toppings selected

**int** noOfToppings = 0;

**int** price = 0; // Price of the pizza

pizzaToppings = ""; // clear toppings

JTabbedPane sourceTabbedPane = (JTabbedPane) e.getSource();

**int** index = sourceTabbedPane.getSelectedIndex();

// Calculate if total tab is selected

**if** (sourceTabbedPane.getTitleAt(index).equalsIgnoreCase("Total")) {

// Get the size of the pizza

ButtonModel b = btngroup.getSelection();

**if** (b != **null**) {

pizzaSize = b.getActionCommand();

}

// If size is not selected, display a message to select size

**if** (pizzaSize.isEmpty()) {

JOptionPane.*showMessageDialog*(**null**, "Select size of your Pizza");

} **else** {

// calculate price of pizza

**if** (pizzaSize.equalsIgnoreCase("Small")) {

price = SMALL\_PRICE;

} **else** **if** (pizzaSize.equalsIgnoreCase("Medium")) {

price = MEDIUM\_PRICE;

} **else** {

price = LARGE\_PRICE;

}

// add toppings to the pizza

**if** (pepperoniCheckBox.isSelected()) {

pizzaToppings += pepperoniCheckBox.getText();

noOfToppings += 2;

}

**if** (sausageCheckBox.isSelected()) {

pizzaToppings += sausageCheckBox.getText();

noOfToppings += 2;

}

**if** (mushroomCheckBox.isSelected()) {

pizzaToppings += mushroomCheckBox.getText();

noOfToppings += 2;

}

**if** (onionCheckBox.isSelected()) {

pizzaToppings += onionCheckBox.getText();

noOfToppings += 2;

}

}

}

// Calculate total price

totalPrice = price + noOfToppings;

totalTextField.setText(String.*valueOf*(totalPrice));

}

@Override

**public** **void** actionPerformed(ActionEvent e) {

// Create a new frame and display summary of the order

JFrame frame = **new** JFrame("Order Summary");

frame.setDefaultCloseOperation(JFrame.***DISPOSE\_ON\_CLOSE***);

frame.setSize(250, 200);

JTextArea summaryTextArea = **new** JTextArea();

String result = "Customer Name: " + nameTextField.getText() + "\nPhone Number: " + phoneTextField.getText() + "\nSize: " + pizzaSize + "\nToppings: " + pizzaToppings + "\nTotal: " + totalPrice;

summaryTextArea.setText(result);

frame.getContentPane().add(summaryTextArea);

frame.setVisible(**true**);

}

**public** JPanel customerTab() {

JPanel panel = **new** JPanel();

panel.setLayout(**new** BoxLayout(panel, BoxLayout.***Y\_AXIS***));

panel.add(**new** JLabel("Welcome to Java Pizza"));

panel.add(**new** JLabel("Name: "));

nameTextField = **new** JTextField(15);

panel.add(nameTextField);

panel.add(**new** JLabel("Phone Number: "));

phoneTextField = **new** JTextField(15);

panel.add(phoneTextField);

panel.add(**new** JLabel(" "));

panel.add(**new** JLabel(" "));

**return** panel;

}

**public** JPanel sizeTab() {

JPanel panel = **new** JPanel();

panel.setLayout(**new** BoxLayout(panel, BoxLayout.***Y\_AXIS***));

panel.add(**new** JLabel("What size pizza?", JLabel.***CENTER***));

JRadioButton smallOption = **new** JRadioButton("Small");

smallOption.setActionCommand("Small");

JRadioButton mediumOption = **new** JRadioButton("Medium");

mediumOption.setActionCommand("Medium");

JRadioButton largeOption = **new** JRadioButton("Large");

largeOption.setActionCommand("Large");

btngroup.add(smallOption);

btngroup.add(mediumOption);

btngroup.add(largeOption);

panel.add(smallOption);

panel.add(mediumOption);

panel.add(largeOption);

**return** panel;

}

**public** JPanel toppingTab() {

JPanel panel = **new** JPanel();

panel.add(**new** JLabel("What toppings would you like?", JLabel.***CENTER***));

panel.setLayout(**new** BoxLayout(panel, BoxLayout.***Y\_AXIS***));

pepperoniCheckBox = **new** JCheckBox("Pepperoni");

sausageCheckBox = **new** JCheckBox("Sausage");

mushroomCheckBox = **new** JCheckBox("Mushroom");

onionCheckBox = **new** JCheckBox("Onion");

panel.add(pepperoniCheckBox);

panel.add(sausageCheckBox);

panel.add(mushroomCheckBox);

panel.add(onionCheckBox);

**return** panel;

}

**public** JPanel totalTab() {

JPanel panel = **new** JPanel();

panel.setLayout(**new** FlowLayout());

panel.add(**new** JLabel("Total Due"));

totalTextField = **new** JTextField(15);

panel.add(totalTextField);

panel.add(summaryButton);

summaryButton.addActionListener(**this**);

**return** panel;

}

**public** **static** **void** main(String[] args) {

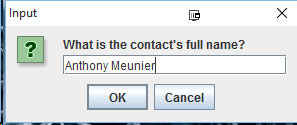
@SuppressWarnings("unused")

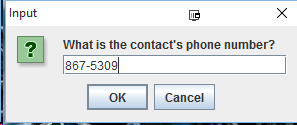
JavaPizza pizza = **new** JavaPizza();

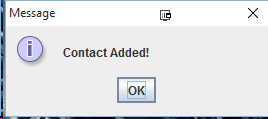
}

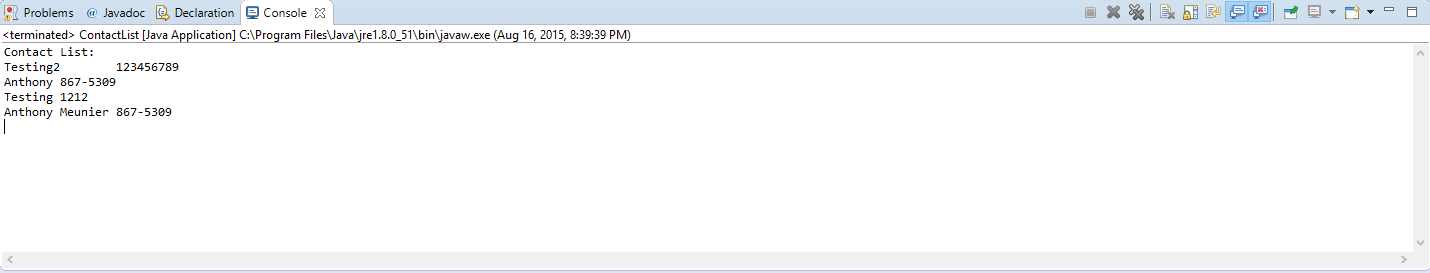
}

Step 2: Contact List









/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Program Name: JavaPizza.java

Programmer's Name: Anthony Meunier

Program Description: Application which makes successful connection to database and

adds contacts into contact table and displays all contacts and their info in table.

Uses JOptionPanes to ask for contact data and displays confirmation message when contact is added.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

**import** java.sql.Connection;

**import** java.sql.DriverManager;

**import** java.sql.ResultSet;

**import** java.sql.SQLException;

**import** java.sql.Statement;

**import** java.util.Properties;

**import** javax.swing.JOptionPane;

**public** **class** ContactList {

/\*\*

\* Create a database connection

\*/

**public** **static** Connection getConnection() {

// driver name and database URL

String DRIVER\_CLASS\_NAME = "com.mysql.jdbc.Driver";

String DB\_CONN\_STRING = "jdbc:mysql://Devry.edupe.net:4300/CIS355A-iLab6";

// Database credentials

Properties info = **new** Properties();

info.put("user", "3370");

info.put("password", "9kkHdVWx9TlBjG74");

Connection result = **null**;

**try** {

Class.*forName*(DRIVER\_CLASS\_NAME).newInstance();

} **catch** (ClassNotFoundException | InstantiationException | IllegalAccessException ex) {

System.***out***.println("Error loading driver: " + DRIVER\_CLASS\_NAME);

}

**try** {

result = DriverManager.*getConnection*(DB\_CONN\_STRING, info);

} **catch** (SQLException ex) {

System.***out***.println("Driver loaded, but cannot connect to database: " + DB\_CONN\_STRING);

}

**return** result;

}

**public** **void** addContact() {

String name = JOptionPane.*showInputDialog*("What is the contact's full name?");

String phone = JOptionPane.*showInputDialog*("What is the contact's phone number?");

Connection conn = *getConnection*();

Statement stmt = **null**;

**try** {

stmt = conn.createStatement();

String sql = "INSERT INTO Contact "

+ "VALUES ('" + name + "', '" + phone + " ')";

stmt.executeUpdate(sql);

JOptionPane.*showMessageDialog*(**null**, "Contact Added!");

stmt.close();

} **catch** (Exception se) {

se.printStackTrace();

}

}

**public** **void** viewContact() {

Connection conn = *getConnection*();

Statement stmt = **null**;

**try** {

stmt = conn.createStatement();

String select;

select = "SELECT \* from Contact";

ResultSet rs = stmt.executeQuery(select);

System.***out***.println("Contact List:");

**while** (rs.next()) {

System.***out***.print(rs.getString(1) + "\t");

System.***out***.println(rs.getString(2));

}

stmt.close();

conn.close();

} **catch** (SQLException e) {

System.***out***.println(e.getMessage());

}

}

**public** **static** **void** main(String[] args) {

ContactList contact = **new** ContactList();

contact.addContact();

contact.viewContact();

}

}