

## Laboratory 14: Drag and Drop

### [Java API](#)

In this lab, you will be working with the GUI interface from your first programming assignment. In particular, you will replace the check boxes used to select the toppings of the pizza with a JList. The user will be able to select a topping from the JList and drag it to the image of the pizza to add the desired topping to the pizza. The check boxes will still be needed behind the scenes (not visible) as the toppings must be added in a particular order to obtain the correct image for the pizza.

### Lab:

- Drag and Drop
- Observer
- Drawable

### Part I: ToppingList

Download the following files:

- [DecoratedPizza.java](#)
- [Pizza.java](#)
- [Crust.java](#)
- [Pepperoni.java](#)
- [Sausage.java](#)
- [Onions.java](#)
- [Peppers.java](#)
- [Mushrooms.java](#)
- [DrawPanel.java](#)
- [Drawable.java](#)
- [EasyGridBag.java](#)
- [CenterFrame.java](#)
- [ImageLoader.java](#)
- [SimpleDialogs.java](#)
- [PizzaGUI.java](#)
- [images.zip](#)
- [ToppingTransferable.java](#)
- [ToppingSelected.java](#)
- [ToppingList.java](#)

Extend JList to create a ToppingList class. Set the starting elements in ToppingList (**setListData**) to represent all of the toppings that are available (simply use Strings to do this). Implement DragGestureListener and DropTargetListener in ToppingList so that when a drop is accepted by the pizza, the corresponding topping entry in the list is removed. Later, you will also need to inform any listeners that a topping was added to the pizza.

Add the appropriate code to the GUI to enable drag and drop. Test your work to make sure that your drag and drop is working (the image will not appear yet, however).

## Part II: Drawable

The DrawPanel is a JPanel that contains graphics that must be periodically updated. The image of the pizza is displayed in a DrawPanel. When told to repaint by the JFrame, the DrawPanel calls the draw method on the object registered with it, passing along the DrawPanel's Graphics object to be drawn upon. The objects registered with DrawPanel must therefore implement the Drawable interface, and are responsible for drawing on the provided Graphics object.

The GUI needs to be registered with the DrawPanel, so the draw method in the GUI will be called when the DrawPanel needs to be updated (repainted). Register the GUI with the DrawPanel. The GUI will then simply tell the pizza to draw itself. Modify DecoratedPizza so that it implements Drawable and is able to draw itself (you will need to call an abstract method from the draw method in DecoratedPizza and use the ImageLoader).

## Part III: ToppingSelected

When a topping is selected from the ToppingList (and successfully dropped), the GUI should be notified. The GUI will check the appropriate (not visible) check box based on the topping selected. You will need to (re)build and (re)display the pizza when a change is made as the toppings must be added in a specific order to get the correct image.

## Part IV: Order

After reporting the price of the pizza, reset the GUI for a new pizza order.

