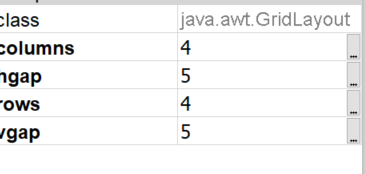
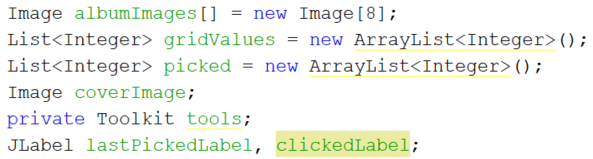
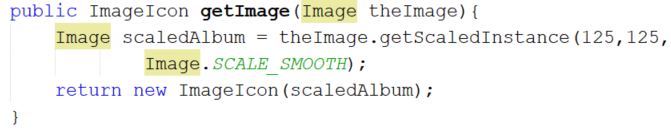
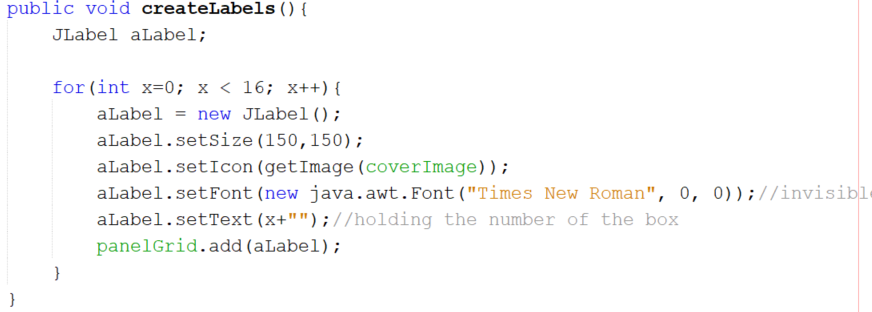
Java Concentration Lecture

Objectives

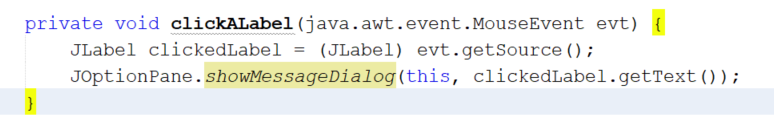
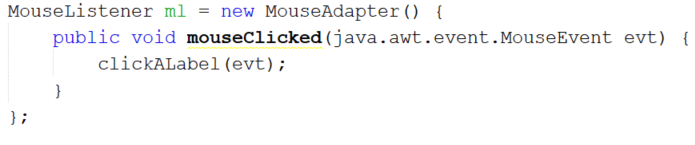
* Use arrays to track objects on screen
* Create custom mouse events
* Create dynamic objects in code

1. Add panel to middle
   1. Set minimum and preferred size to 620\*620
   2. Set layout to 6rid 4x4
   3. 
2. Drag the 9 images into the package

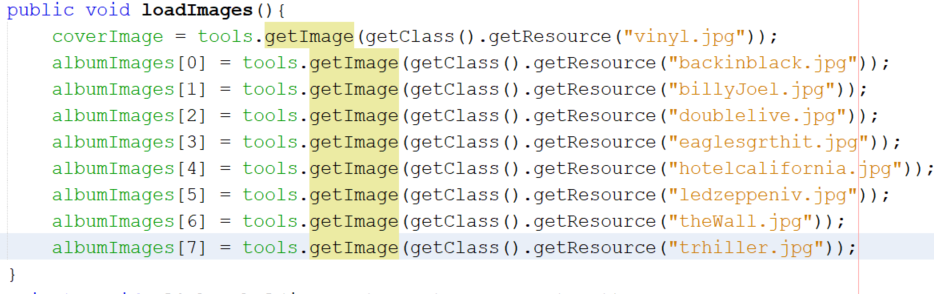
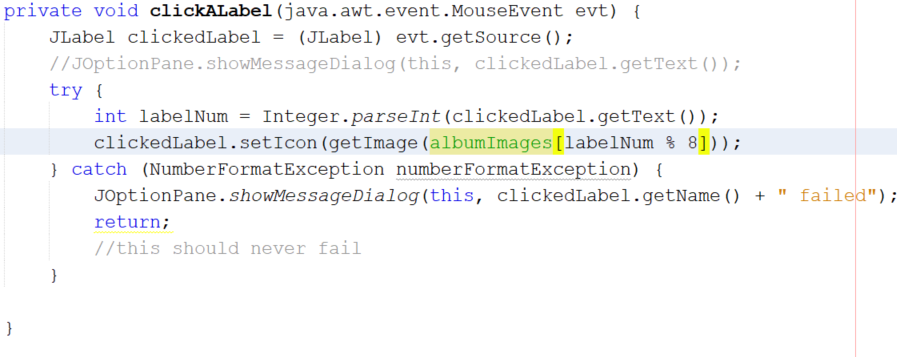
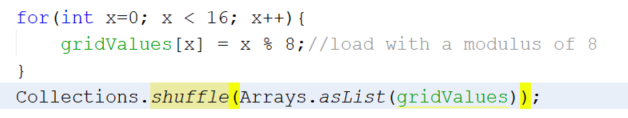
# Go to code

1. Add these variables for now
2. 
3. Constructor
   1. 
4. Create method getImage() - scale the image  
   
5. Create createLabels()
   1. 
   2. Run, should have 16 labels on the screen

# Respond to mouse click

1. Create a custom event handler function
   1. 
   2. Demonstrate evt and what it allows
2. Create a MouseListener variable
   1. 
3. Add the handler to the object in createLabels
   1. Put this before add
   2. 
   3. It is hard to remember this code, easy way is to add a object, add the event in GUI then copy the code – demonstrate
   4. Could put the code here instead of clickALabel
4. Run, click the labels – should see the number

# Get images into the labels

1. Create a new method named loadImages()
   1. Move the coverImage line from the constructor
   2. Calll loadImages() in the constructor
   3. Create this method – not the most elegant, but it works  
      
2. Modify clickLabels to show the images based on mod
3. 
4. Now let us randomly place the images
   1. We will use the shuffle method of Collections.
   2. Add this to loadImages  
      
5. Modify clickALabel  
     
   
6. This is kludgy – avoiding threads
   1. Add a jButton
   2. AQdd this to top of clickALabel  
      
   3. Add this into the try
7. Add to the button  
   
8. Solve a glitch with multiple selections
   1. Add these two methods
   2. 
9. Make these calls
   1. ClickLable before the jLabel
   2. 
   3. Button at the end
   4. 