**Course Project**

**DeVry University**

**College of Engineering and Information Sciences**

**Course Number: CIS355A**

# Module 3: Use Advanced Controls in Your GUI Application

# Objectives

* Add radio buttons or checkboxes to your GUI application.
* Add a menu to your GUI application.
* Add images to your GUI application.

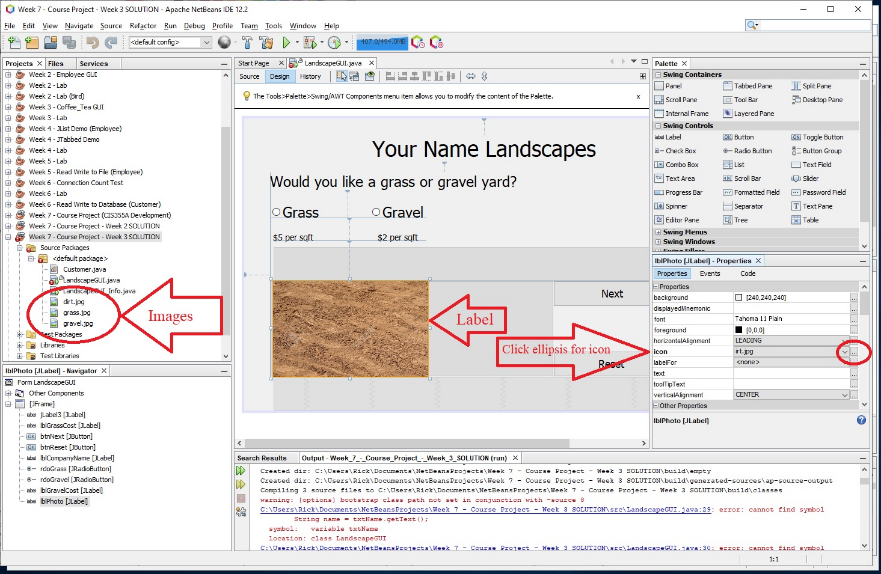
We have a basic GUI application. So far, so good. Now, let’s add some advanced GUI controls and images to make our application more professional.

Look at your design. Let’s implement the first page (Page 1) of your Visio wireframe diagram. Because we do not know how to do tabs yet, let’s create a new JFrame to hold our radio buttons and images. Copy your current LandscapeGUI.java file and name it “LandscapeGUI\_Info.java.” Then, on the LandscapeGUI.java file, delete or rearrange as needed to create your first screen (not including the menu or tabs or panel).

Finally, when you click the next button, launch the second JFrame (LandscapeGUI\_Info.java). This way, you will have a multiwindow application.

Let’s get started.

# Steps

1. We are going to use images in our project this week. The first step is to add images to the project. Open Google and search for dirt, then grass, and then gravel. Be sure to use “creative commons” in your search so it finds free images that we can use without paying royalties. For example, Google “dirt creative commons”. Find and download a dirt image, then a grass image, and then gravel image.
2. Open each image using paint. Resize the image to 400 px wide and save it as a JPG file. For example, see below.
   1. Open the dirt image in paint.
   2. Click the resize command on the Home tab.
   3. Click the pixels radio button.
   4. Change the horizontal value to 400 (Vertical will automatically change.)
   5. Click the OK button.
   6. Click the file tab.
   7. Click save as and then JPEG Picture.
   8. Change the file name to dirt.jpg and the save as type to JPEG.
   9. Navigate to your desktop and then click the save button.
3. You should have the three files resized, renamed, and saved to your desktop (dirt.jpg, grass.jpg, and gravel.jpg). Open NetBeans. Click the + sign next to your Course Project to open it. Then, click the + sign next to source packages to open it. Finally, click the + sign next to <default package> to open it. Drag the images from your desktop into <default package>. The images are now part of your project. If you check the project directory, you will find the images in the same folder as your LandscapeGUI.java file.
4. Right-click on the LandscapeGUI.java file in the projects pane (left side window). Choose copy and then right-click it again and choose Paste 🡪 Refactor Copy. In the new name field, enter LandscapeGUI\_Info and then click the refactor button.
5. Close the LandscapeGUI\_Info.java tab at the top. Then, double-click on the LandscapeGUI.java file on the projects pane to open it. Confirm that the LandscapeGUI.java file is open by reading the tab at the top. Delete the following old methods that we will not need in LandscapeGUI.java. Be careful to delete the entire method, and nothing more.  
   1. validateInputs( )
   2. createCustomer( )
   3. submitOrder( )
6. Rename, move, add, and delete items until your LandscapeGUI looks like the first page (Page 1) of your Visio wireframe diagram. Note: Use a label (JLabel) to hold the image. You can change the icon property of the label to select the image. When you finish creating the LandscapeGUI frame, it should look something like this (notice the next and reset buttons, which should be called btnNext and btnReset respectively). Please note: we are not going to do the tabs this week. Please skip the tabs and the menu items for now.  
     
   
7. The two radio buttons must be placed into a ButtonGroup object, so only one radio button can be selected at a time. Drag a ButtonGroup item from the swing controls and drop it on the form. Now, look at the navigator that is located at the bottom, left corner of the IDE. Notice that the ButtonGroup is located in the other components branch. Open the branch if you need to do so and rename the ButtonGroup to btgYardType (right-click 🡪 Properties 🡪 Code tab 🡪 Variable Name).  
   1. Add the radio buttons to the ButtonGroup.
      1. Click on the rdoGrass radiobutton on your GUI. Change the buttonGroup property to btgYardType.
      2. Click on the rdoGravel radio button on your GUI. Change the buttonGroup property to btgYardType.
8. Menus are easy with Net Beans if you know the overall structure of how menus work. The menu bar extends across the top. The menu bar holds menu objects. The menu name will appear at the top of the menu. Then, the menu objects hold menu items. The menu items will appear when you click on a menu object.
   1. Open the swing menu controls by clicking the plus sign (+) on the left of swing menus (located on the right side with the other controls).
   2. Drag the menu bar to the top of the form and drop it.
      1. Use the navigator and rename the menu bar as mnbMain
   3. The menu bar will automatically contain a file menu and an edit menu.
      1. Right-click on the edit menu and rename it to Order to match our Visio wireframe diagram. Right-click again and choose properties. Click the code tab and change the variable name to mnuOrder.
      2. Right-click on the file menu and choose properties. Click the code tab and change the variable name to mnuFile.
   4. Drag a menu item and drop it on top of the file menu. Rename the menu item to exit. Right-click on the exit menu item and change the variable name to mniExit (notice the mni).
   5. Drag a menu item and drop it on top of the order menu. Rename the menu item to submit order. Right-click on the submit order menu item and change the variable name to mniSubmitOrder.
   6. Drag another menu item and drop it on top of the order menu. Rename the menu item to reset. Right-click on the reset menu item and choose change variable name and change it to mniReset.
9. Let’s create the event handlers for our buttons and our menu items. Right-click on the next button and choose Events 🡪 Action 🡪 actionPerformed. When you click on actionPerformed, Net Beans will take you to the event method, or more likely, to the bottom of the code window. If it takes you to the bottom of the code window, find the event method. If your button is called btnNext, then the event method will be called btnNextActionPerformed. Add this code to the event method for the next button. (Delete any code that may already be there from our copy and paste.)

// create the second GUI form

LandscapeGUI\_Info infoForm = new LandscapeGUI\_Info();

// set default close operation to "dispose" so application stays open

infoForm.setDefaultCloseOperation(JFrame.DISPOSE\_ON\_CLOSE);

// launch the second GUI form by making it visible

infoForm.setVisible(true);

1. Go back to the design window of LandscapeGUI. Right-click on the reset button and choose Events 🡪 Action 🡪 actionPerformed. When you click on actionPerformed, Net Beans will take you to the event method, or more likely, to the bottom of the code window. If it takes you to the bottom of the code window, find the event method. If your button is called btnReset, then the event method will be called btnResetActionPerformed. Add this code to the event method for the reset button. (Delete any code that may already be there from our copy and paste.)  
     
   reset( );
2. Notice that we are only calling the reset( ) method. We want to write the code one time and then call it from multiple locations. Add this reset( ) method code to the bottom of the code window. Be careful and stay inside of the last curly brace, which closes the class.

private void reset()

{

// clear the radiobutton selection

btgYardType.clearSelection();

// put the image back to "Dirt"

lblPhoto.setIcon(new ImageIcon(this.getClass().getResource("dirt.jpg")));

}

1. Let’s set the events for the radiobuttons. Right-click on the rdoGrass radiobutton and choose Events 🡪 Action 🡪 actionPerformed. When you click on actionPerformed, find the event method, which is probably called rdoGrassActionPerformed. Add this code to it so the image is changed to grass.jpg when rdoGrass is selected.

if (rdoGrass.isSelected())

{

// look in directory where class exists, and get location of "grass.jpg"

URL location = this.getClass().getResource("grass.jpg");

// create image using the location of "grass.jpg"

ImageIcon icon = new ImageIcon(location);

// set the image icon to the label

lblPhoto.setIcon(icon);

}

1. Right-click on the rdoGravel radiobutton and choose Events 🡪 Action 🡪 actionPerformed. When you click on actionPerformed, find the event method, which is probably called rdoGravelActionPerformed. We can change the picture with one line instead of the three lines of code that we used for the rdoGrass radio button. Add this code to the rdoGravel event method so the image changed to gravel.jpg when rdoGravel is selected.

if (rdoGravel.isSelected())

{

// find image, create image icon, and set image icon to label

lblPhoto.setIcon(new ImageIcon(this.getClass().getResource("gravel.jpg")));

}

1. Run your application and test it. Are you having fun with it?
2. Stop the application from running, if it is running. Go to the design view. Click on the file menu located on your application’s menu bar to open it up. Click on the exit menu item to select it. Right-click on the exit menu item and select Events 🡪 Action 🡪 actionPerformed. When you click on actionPerformed, find the event method, which is probably called mniExitActionPerformed. Add this line to exit the application when the exit menu item is clicked.  
     
   // exit the application  
   System.exit(0);
3. In the design view, click on the order menu to open it up. Click on the submit order menu item to select it. Right-click on the submit order menu item and select Events 🡪 Action 🡪 actionPerformed. When you click on actionPerformed, find the event method, which is probably called mniSubmitOrderActionPerformed. Add this line to show a stub when the submit order menu item is clicked.  
     
   JOptionPane.showMessageDialog(this, "Method is not complete.");
4. In the design view, click on the order menu again to open it up. Click on the reset menu item to select it. Right-click on the reset menu item and select Events 🡪 Action 🡪 actionPerformed. When you click on actionPerformed, find the event method, which is probably called mniResetActionPerformed. Add this line to call the reset( ) method when the reset menu item is clicked.   
     
   reset( );
5. Run your application and test it. Radio buttons, menus, and images really add a lot of value to your applications.

# Course Project Deliverables for Week 3

* Close Net Beans so your zip file will have the latest code. Then, go to the project folder. Right-click on the project folder and choose Send to 🡪 Compressed (zipped) file. Then, rename the zipped file to include your name. Submit the zipped file on Canvas.
  + Week 3 Advanced GUI – Your Name.zip