Assignment 0602
The goal of this assignment is to combine practice in a number of things: GridBagLayout, event listening and handling, and some data transfer. There is also an extra credit element that sees how you can handle a dynamic element to Swing programming.

For Submission
Create a customizable text data entry panel that behaves in the following way:

• Initialize the panel with a String array or collection that represents the text fields that it should present. For example, if you name your class DataEntryPanel, the code fragment below should result in a Swing panel with 2 columns and 4 rows: the first column displays the provided prompts, and the second column displays JTextFields for each requested field.

  ```java
  String[] fields = { "Last Name", "First Name", "Phone", "Birthdate" };
  DataEntryPanel dep = new DataEntryPanel(fields);
  ```

• “Wrap” this panel into a demonstration program with two buttons: one for setting the data and another for displaying the data. When “Set Data” is clicked, the program should overwrite the fields with whatever set of Strings you choose. When “Display Data” is clicked, the program shows the current contents of the data entry panel (console output is OK, but if you can do it with more Swing components, then why not).

As usual, e-mail the code to me before class on June 2.

Extra Credit
Accomplishing this part of the assignment counts as an additional homework above and beyond the total assigned for the semester:

• Modify your data entry panel so that the fields it displays can change dynamically. That is, implement a method with a signature like this:

  ```java
  public void setFields(String[] fieldNames)
  ```

• Calling this method should rebuild the panel from the ground up, now showing the new set of request fields. You will need to look into the Container class for how to do this, and also read up on how to use the revalidate() and repaint() methods to tell a panel to update itself when significant changes are made to its child components.

To get the extra credit, you must also e-mail the code for this program before class on June 2. If you do fulfill the extra credit tasks, you don’t need to submit two batches of code; just provide a set of classes that has all of the requested functionality.

Reminder
Don’t forget to turn in your paper prospectus this Tuesday, May 31!