Company GUI Read Write

Objectives

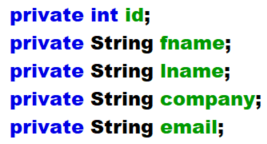
* Read and write a file in a GUI
* Read and write a structured sequential file
* Walk through the records

# Setup

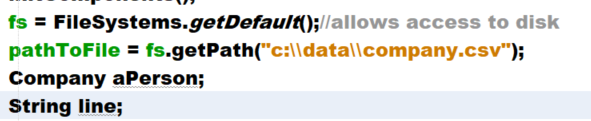
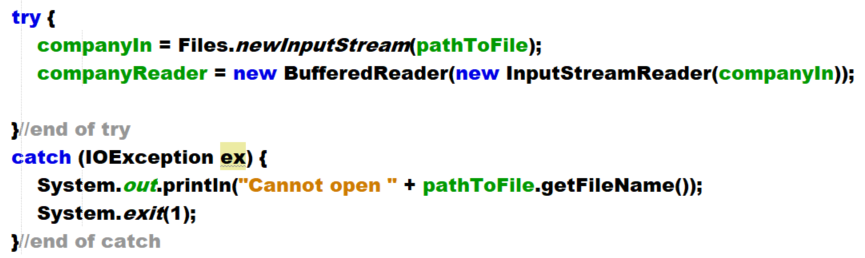
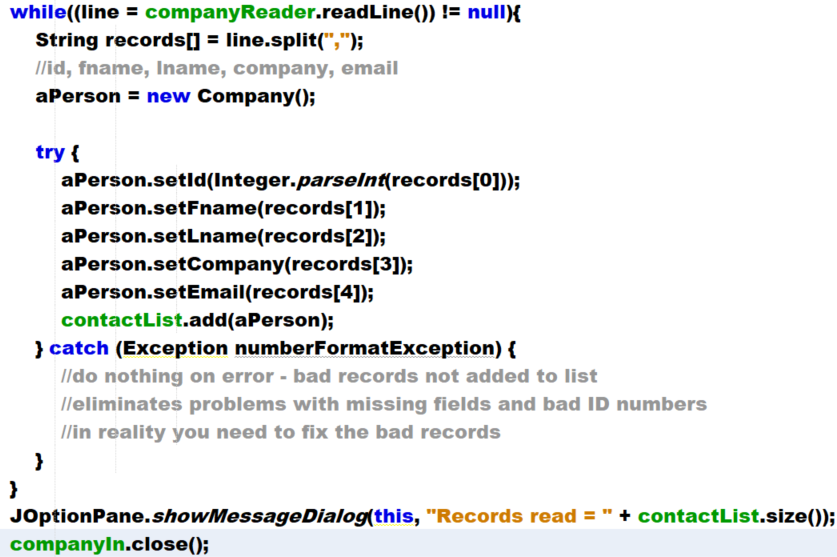
* Create a new Java project or package
* Download the guiCompany file from BB
* Extract the files
* Move the CompanyGUI.java and CompanyGUI.form to your package
* Create a c:\data folder
* Move the company.csv file to c:\data

# Code – Create a class for the data

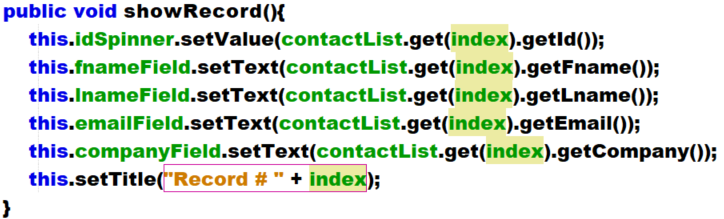
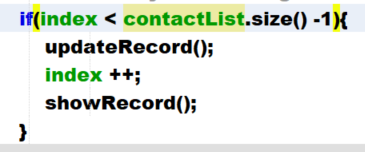
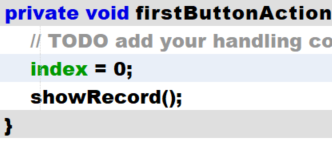
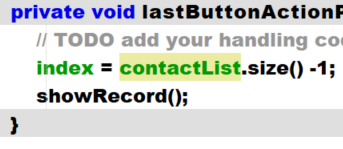
Look at the company.csv in a text document or Excel  
Visual Studio Code has a great extension called Rainbow csv  
Anytime you are dealing with data like this, you should class it.

* Build a class called Company  
  
* Create setters, getters, both constructors, toString

# Code – Read the file in to the gui

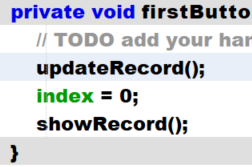
* Create these variables  
  
* Add to top of constructor, after initComponents  
  
* Open the file  
  
* Add after the companyReader line – this will walk through the file, convert to a class, add to the list – fix imports as needed  
  
* Run
* Once it works, you can remove the JOptionPane

Code – walk through the list

* We will display the record on the screen.
* The index will be used to display the object in the list one step at a time
* Create this method  
  
* Call at end of constructor, where JOptionPane was  
  
* Create next button  
  
* Create previous button  
  
* Create first, last button  
   

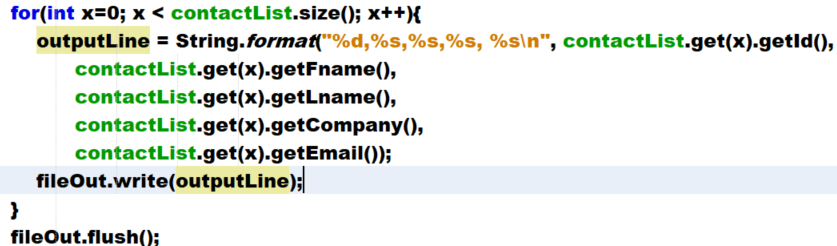
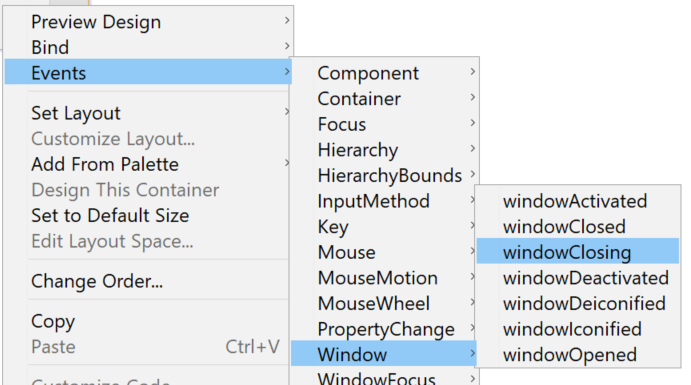
# Handle updating of values

The user can edit the values in the text boxes. We need to write them back into the list.

* Create this method  
  
* Call in each motion button, **before** the if or the index change  
   
* Run and test, ensure that the data updates on the screen

# Write back to disk

When the program finishes, we need to write back to disk. Could also write with every movement.

* Before you begin, make a backup of the company.csv file
* Create this method
* 
* Add this above the fileOut.close()  
  
* Create the windowClosing event
  + Go to the form
  + Rt click an empty section of the form. Select events  
    
  + Add these two function calls to the method  
    