# Exercise02: Java Swing

**Objectives:**

- To learn to use JList and to play with the **model** and **view** for JList.
- To use JOptionPane
- To learn to use JTree and to understand it's model and view.
- To use basic JTable.

JList, JTree, and JTable are complex JComponents and have separate MODEL and VIEW elements that can be customized. In this Exercise, we will only work with JList and JTree. Read the example file (TreeFrame.java).

**Work with your group (or by yourself). Each group should upload only one submission.**

First, open blackboard, go to Course Contents, and then download exercise02.zip file into your workspace (U:\workspace or something like that!). Then, unzip.

# 1 Warm Up: Try Some Examples

1. First, open blackboard, go to Course Contents, and then download exercise02.zip file into your workspace (U:\workspace or something like that!). Then, unzip.
2. Open Eclipse, create a new Java Project, and then copy the "*.java" files from your unzipped folder to the src folder of your project.
3. Play with each of the given examples. Each of them has some "TODO" comments. Try following the instructions and observe what happens.

You should now be reasonably ok with JList, Dialogs, and JTree.

Note that the assignment assumes you have understood these examples.

# 2 The main assignment!

1: Run Eclipse and make sure that the workspace is set to U:\workspace (your COMS home directory).

2. Go to File->New->Project->Java Project. For the project name, type in “Lab2-Swing” and click Finish. You should see the project “Lab2-Swing” built and shown in the “Package Explorer”.

3. Right-click on Lab2-Swing, New->Package. Call the package “cs319”.

4. Now, choose the drop-down item labeled “Create new visual classes” from the main toolbar (it should be the second item from the left). Go to Swing->JFrame. Name the JFrame “Lab2Swing”. Also, make sure that Source Folder and Package are set correctly.

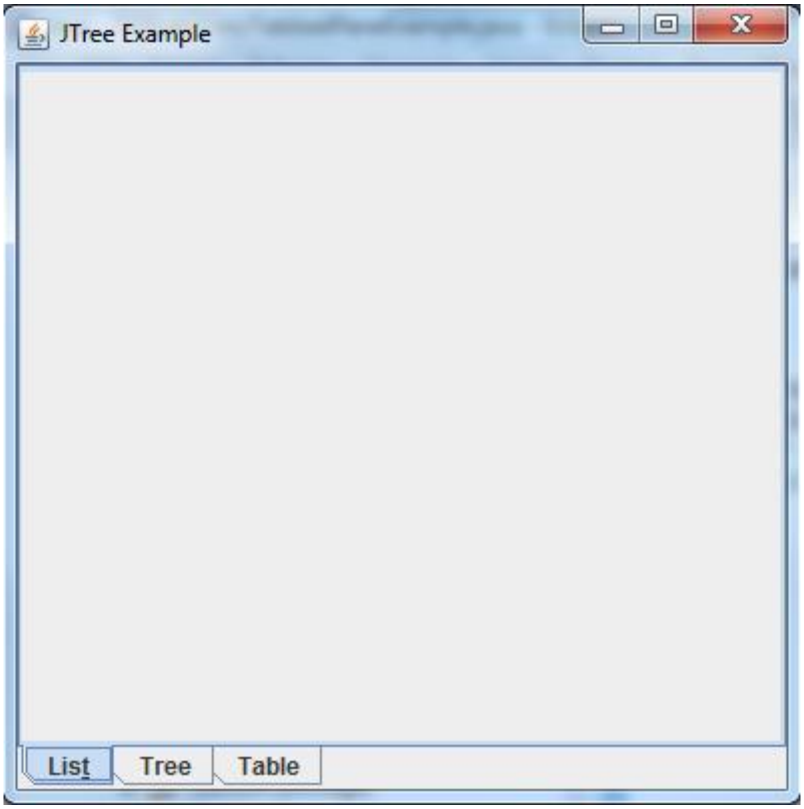
5. You will see a new file named "Lab2Swing.java" was created by Windows Builder with some java code in it.

6. At the bottom of the editor, there are two tabs: Source and Design. If you go to the Design view, you will get a visualized view of your application corresponding to what the user of the application will see. You will also have access to windows labeled “Structure”, “Properties”, and “Palette” that facilitate building and editing the JFrame visually.

7. Right-click on the empty window in the editor->Set Layout-> Absolute Layout.

## 2.1 Tabbed Swing Application

Have three tabs in the Lab2Swing (using JTabbedPane). The first tab is called List, the second one is called Tree, and the third one is called Table (which will be left EMPTY in this lab). Set the title of the window to “Tabbed Swing Application”. When the “X” button on the top right corner is pressed, the application terminates. It should be similar to the following:

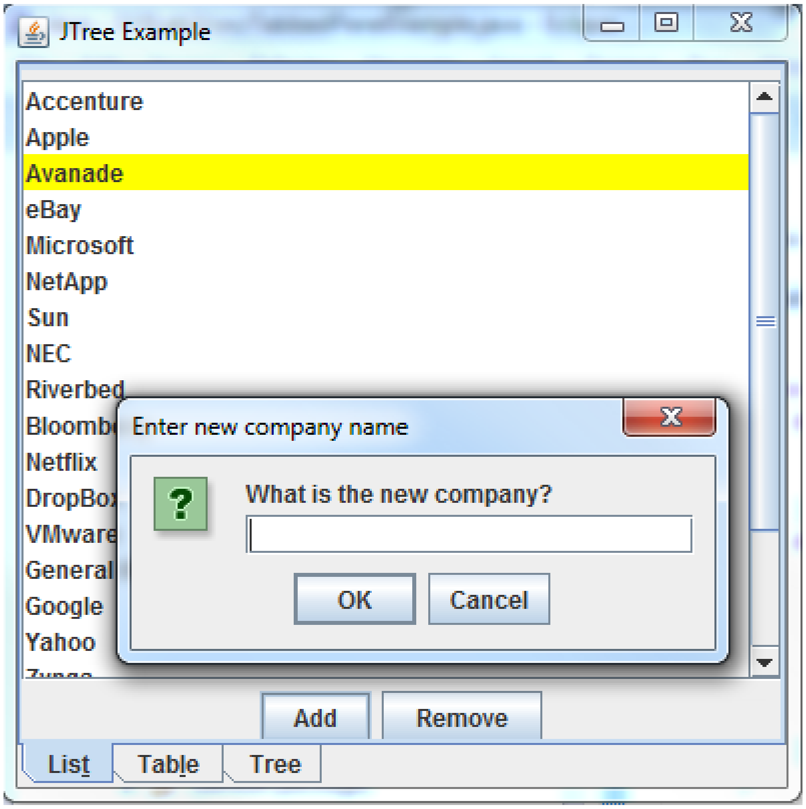


## 2.2 List Panel

The objective in this part is to create a List (i.e. JList component) in the first tab to read and show the contents of an external “.txt” file. The file is called “companies.txt” (located in the .zip file of the lab) and in each line of the file, there is a company name. It can be assumed that the file is located inside the current directory. You must create a class called DataModel which extends the javax.swing.AbstractListModel class in order to complete this part. The DataModel class you create must read entries from a file named “companies.txt” and fill an array (ArrayList) of Strings that will store the names of the companies. This DataModel class will be used as the model for the list.

When an item from the list is selected, the background color of the selected item should change to yellow. There must also be two buttons, Add and Remove. As the names suggest they will be responsible for adding/removing items to/from the list. Remember that any changes to the list must be reflected in "companies.txt" as well (i.e., if we add a new item to the list, it gets added to the file and if we remove an item, it gets removed from the file). When you press the Add button, a dialog box should pop up as shown in the figure and prompt the user for the new company name. The Remove button should remove the selected item from the list and "companies.txt". Finally, make sure the list is scrollable (make use of JScrollPane).

The end result would be similar to the following figure:



Accenture
Apple
Avanade
eBay
Microsoft
NetApp
Sun
NEC
Riverbed
Bloomb
Netflix
DropBo
VMware
General
Google
Yahoo

What is the new company?

## 2.3 Tree

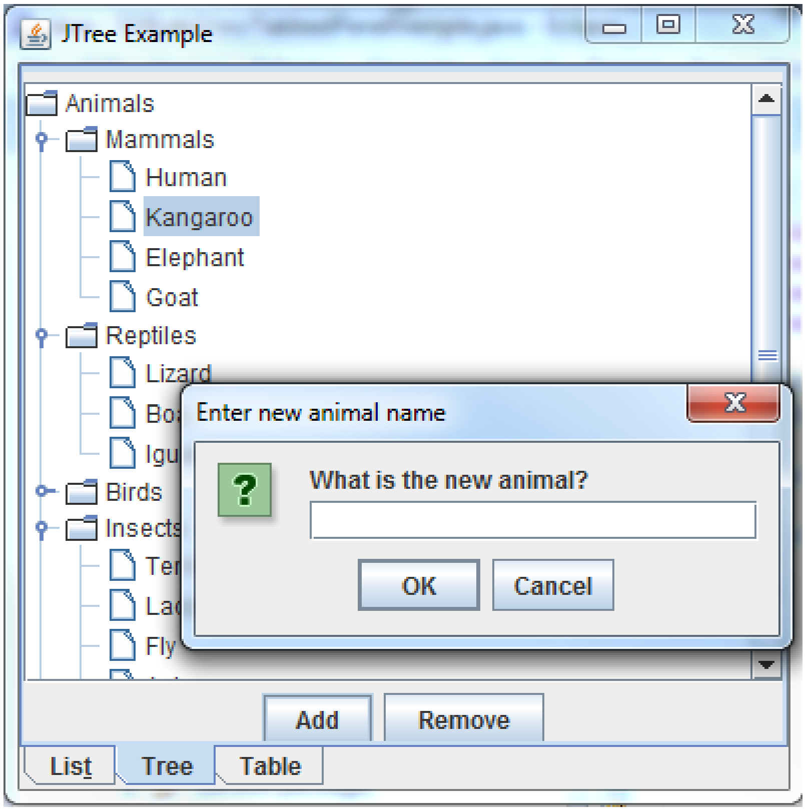
The objective in this part is to create a Tree (i.e. JTree component) in the second tab to build and show the following hierarchy:

Animals
├ Mammals
  ├ Human
  ├ Kangaroo

```
   ├ Elephant
   └ Goat
├ Reptiles
   ├ Lizard
   ├ Boa
   └ Iguana
├ Birds
   ├ Duck
   ├ Pigeon
   ├ Turkey
   └ Goose
├ Insects
   ├ Termite
   ├ Ladybug
   ├ Fly
   └ Ant
└ Fish
   ├ Sword Fish
   ├ Shark
   └ Eel
```

In addition, it should be possible to add and remove new animal to each of the categories (Mammals, Reptiles, Birds, Insects, Aquatic). The new animal should get added to the currently selected category. For example, if Reptiles (**or any of its children nodes**) is selected and Add button is pressed, the new animal should be added to Reptiles. The Remove button removes the selected item from the tree. If the selected item has any children, all of its children should be removed as well. If the selected item is the root of the tree, the entire tree is removed. Also, make sure the tree is scrollable.

The end result would be similar to the diagram in the following page.



## 3 Part 1: JTable [Extra Credit]

We have NOT read or learnt about JTable. However, there are lots of great tutorials on the web. Here is an example http://docs.oracle.com/javase/tutorial/uiswing/components/table.html
We will assume you will be able to read and learn to use JTable (which is similar to JList and JTree) from these.

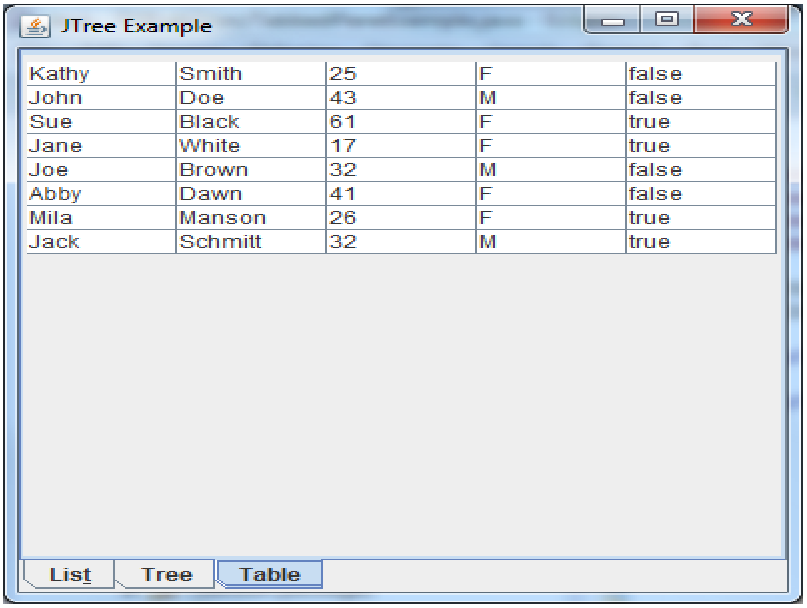
The objective in this part is to create a Table (i.e. JTable component) in a JPanel to read and show some contents. The following are the columns of the table:

```
String[] columnNames = {"First Name","Last Name","Age","Gender","Vegetarian"};
```

And the following are the contents that are to be shown in the table:

```
Object dataValues[][] = {
    {"Kathy", "Smith", 25, ‘F’, false}
    {"John", "Doe", 43, ‘M’, false}
    {"Sue", "Black", 61, ‘F’, true}
    {"Jane", "White", 17, ‘F’, true}
    {"Joe", "Brown", 32, ‘M’, false}
    {"Abby", "Dawn", 41, ‘F’, false}
    {"Mila", "Manson", 26, ‘F’, true}
    {"Jack", "Schmitt", 32, ‘M’, true}
}
```

Make sure the table is scrollable (make use of JScrollPane). The end result would like the following:



| | | | | |
|---|---|---|---|---|
| Kathy | Smith | 25 | F | false |
| John | Doe | 43 | M | false |
| Sue | Black | 61 | F | true |
| Jane | White | 17 | F | true |
| Joe | Brown | 32 | M | false |
| Abby | Dawn | 41 | F | false |
| Mila | Manson | 26 | F | true |
| Jack | Schmitt | 32 | M | true |

## CheckList for JTREE

**[ ] JTree is there and is scrollable**

**[ ] JTree includes all the nodes and the hierarchy as specified**

**[ ] Add and Remove buttons are there**

**[ ] Add button adds new items properly to the correct category**

**[ ] Remove button properly removes the selected item (and its children if any)**

**[ ] The JTree looks similar to the given figure**

## CheckList for JList

**[ ] JList is scrollable**

**[ ] JList shows the contents of the companies.txt file**

**[ ] DataModel class reads companies.txt file and fills the array inside the class**

**[ ] DataModel class extends AbstractListModel class**

**[ ] Add and Remove button are there**

**[ ] Add adds a new company and Remove removes a company name from both the list and companies.txt (write code in DataModel class to accomplish this)**

**[ ] When Add button is pressed, a dialog box is shown to ask about the new company name**

**[ ] Whenever an item is selected, the background color is changed to yellow.**

**[ ] The JList looks similar to the given figure.**

**<u>CheckList for JTable[Extra Credit]</u>**

**[ ] JTable is there and is scrollable**

**[ ] JTable has all five columns**

**[ ] JTable shows the contents properly**

**[ ] JTable looks similar to the given figure**

**[ ] Extra credit: make the table sync with a CSV (comma separated values) file (using model etc.) make sure to create a readme file for this part also.**

## 4 Submission:

Zip your Eclipse project and submit on black board. Remember there is only one submission per group. Make sure to include all the files that are needed in order to run your program(s).