# ITDEV-140 Final Project Report

Chris Otto

Fall 2014

## Instructions

The application is a Java application that runs on the user’s machine. They need the database titled LIST in order to run the application. The .netbeans-derby file folder has been included with this submission from development. When launched the user is created with a jlist box and four text boxes for them to enter information. Below the textboxes are three buttons so that the user can add, update and delete information in the database. The user will enter text to add a new item, select an item from the jlist to update and also select an item from the jlist to delete.

## Overview and Description

The reason I chose to use a database is because it is the easiest to retrieve information from. By just simply adding a connecting I am able to set queries that will manipulate the data with data generated from the user in the application. For populating the jlist I used a simple select query where it selected the names of the items in the table and displayed them to the user. Also I had to add a different listener other than action performed for the jlist. The reason that the jlist was chosen over a combobox is that with the jlist all the items are displayed right to the user and they do not need to go into it in order to what items are there. For the button functionality for manipulating the data I used insert into for add, update for update and delete from for delete. This went pretty smoothly as I have SQL background and just had to pass the variables used in the application to the queries. One of the classes I created was an action listener class to handle the functionality of the buttons. Main creates the application and runs it. The largestInt class is used to get the largest ID from the table and then increment that so that when the user adds something new it goes into the database as it should. The class, fillList, will repopulate the jlist with the new information after an action has taken place, meaning add, update or delete. The load class will fill the textboxes with data from the selected jlist item for edit or delete.

## Lessons Learned

The toughest part of creating the application was adjusting to differences that the jlist has compared to combobox. I went through a lot of online searching in order to figure out how I wanted to program the application so that it would run the smoothest and made the most sense. StackOverflow was a great resource. Once I figured out how to use the jlist correctly and what methods it already had with it, I was able to go through the rest of the application pretty smoothly. The only other thing I had issues with was getting null pointers when a data item was out of scope meaning that at that point in the application there was nothing there.

## Life After

I had not really considered about using this application in a portfolio while I was writing it. That thought only entered my mind when it was listed as a retrospective item in this report. I would consider because it displays that I can code a java application and manipulate SQL data. Although the code I wrote may not be up to the standards at said company I would think that they would take into consideration it being a project. I created a personal website for the final for ITDEV-160, I would most likely that screenshots and code snippets and post them to my portfolio page to display the work done. I would not put the application itself on the site however.