**Refactoring:**

The refactoring method I chose was to move a class/jFrame from one folder to another. I didn’t have a folder to move it to however I still refactored it to demonstrate what refactoring did to it.

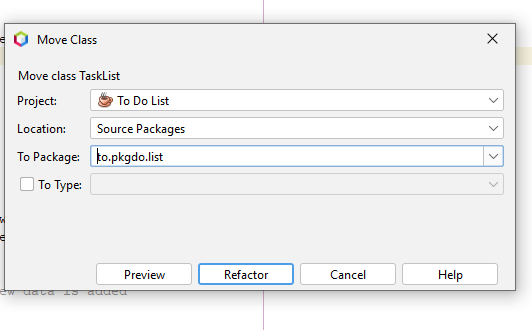
According to (Davison, G, n.d.) the reason to do this is “Classes are often created in a packages close to where they are being used, this can make sense until the class starts to be re-used by other parts of the product. The package in question might also have just become too big. (I have a preference that my packages never have more than about 10 classes)

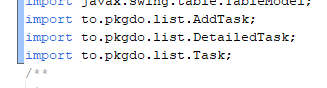
It is often better to move to this class to a package more related to it in form or function. This can help remove complex package level dependencies and make it easier for developers to find and re-use classes.

If there are many dependencies for the class within its own package, then Extract Class could be used first to split out the relevant parts.

Another example where is this used often is to move String resource objects into sub a res package to simplify localization compilation.”

**Screen Shots**

****

****

**AFTER REFEACTORING**

****

**BEFORE REFECTORING**

**Bibliography**

Davison, G. (n.d.). *Move Class* refactoring.com https://www.refactoring.com/catalog/moveClass.html