

1. Modify the text's Remote Control With Undo so that multiple undo's and redo's function correctly.

In the text example each slot of the remote control has a command object. Each of these is just reused whenever the user presses a button.

Consider making a copy of (or clone) a command object and then later on use that copy.

Consider a stack as an abstract data type that serves as a collection of elements, with two principal operations:

push, adds an element to the top

pop, removes the element at the top.

A stack is sometimes called LIFO (for last in, first out). `Stack` is a class in `java.util`.

Consider and use this approach (from Holub on Patterns, page 380-381, handed out in class) where command objects are managed in two stacks: an undo stack and a redo stack:

When a command is to be *executed* (as opposed to *redo* or *undo*), *do* the command and then *push* it onto the *undo* stack.

When *redoing* a command *pop* it from the *redo* stack, *do* it, and then *push* it onto the *undo* stack.

When *undoing* a command *pop* it from the *undo* stack, *undo* it, and then *push* it onto the *redo* stack.

2. Test your new Remote Control with the operations to be given at a later date.

Submit the .java files you have modified via email to 3913-001@acs.uwinnipeg.ca

Every .java file you code or modify has documentation to show you as the author.