```
Loading vc-cvs...
package caltool.edit;

import mvp.*;

/****
 *
 * Class Edit is the model class for the Calendar Tool edit handling.  It
 * contains methods for all of the operations defined on the Edit menu, which
 * constitute the functional command group for basic editing.
 *
 */
public class Edit extends Model {

    /**
     * Construct this.
     */
    public Edit(View v) {
      super(v);
    }

    /**
     * Undo the most recent scheduling operation.  Only one level of undo/redo
     * is specified here.  Successive invocations of undo toggle between the
     * current and previous states of the calendar.  Note that the calendar
     * editing operations support undo by saving a snapshot of the input
     * calendar in the previous_state component of the workspace.
     */
    public void undo() {
        System.out.println("In Edit.undo");
    }

    /**
     * Redo the last undo.
     */
    public void redo() {
        System.out.println("In Edit.redo");
    }

    /**
     * Redo the given operation the given number of times.
     */
    public void repeat(CalendarToolOperation operation, int repeatTimes) {
        System.out.println("In Edit.repeat");
    }

    /**
     * The currently selected text segment is copied into the clipboard and
     * removed from its context.  The workspace selection in set to empty.
     */
    public void cut() {
        System.out.println("In Edit.cut");
    }

    /**
     * The currently selected text segment is copied into the clipboard;
     */
    public void copy() {
        System.out.println("In Edit.copy");
    }

    /**
     * Paste the contents of the clipboard into the currently selected start
     * position, replacing any selected text from start to end position.
     */
    public void paste() {
        System.out.println("In Edit.paste");
    }

    /**
     * Delete the currently selected text.  This function can be specialized by
     * subclasses to delete a tool-specific object.  */
    public void delete() {
        System.out.println("In Edit.delete");
    }

    /**
     * Select all text in the current workspace window.
     */
    public void selectAll() {
        System.out.println("In Edit.selectAll");
    }

    /**
     * Find the given text string in all active windows.
     */
    public void find(String text) {
        System.out.println("In Edit.find");
    }

    /** Data represenation forthcoming */

}
```