

Practical Object oriented design (Text Editor Design)

Design a simple text editor that support only one open document(for an example, you can represent it as stringbuffer) and operations listed below. You might need to create singleton Clipboard object for cut and paste operations. The clipboard object needs a method to load a string and a method to get a copy of the string it holds. Editor must support following operations:

OpenFile: Open operation allows the user to open any text file in the editor. It should load the contents of the file into document model object.

SaveFile: Save operation allows the user to save the modifications performed in the document model object back to the file.

InsertText: Insert operation allows the user to insert text in a document. It will take cursor position and text as input parameters and inserts the text at cursor position inside document. For example, if the document contained the text "012345" and the parameters were 2 and "abc" the result should be "012abc345".

DeleteText: Delete operation allows the user to select a region of the text and delete it. It will take two integers which are the positions of the first and last characters to be deleted (assuming the positions are numbered from zero). For example, if my document contains "0123456789" and the parameters to DeleteText are 2 and 4, the document should end up containing "0156789".

Cut: Cut operation allows the user to cut a selected region of text in a document and place it in the clipboard. It's parameters and its effect on the document are the same as DeleteText, but it also loads the text into the Clipboard so we can use it later.

Paste: Paste operation allows the user to get stuff out of the clipboard onto document. It's parameters and its effect on the document are the same as InsertText, but it should get the string to be inserted from the clipboard.

Copy: Copy operation allows the user to select a region of text and paste it another place. Like in most real systems there is nothing to undo for this operation.

Find: Find operation allows the user to search for a text in the document.

Find/Replace: It allows the user to perform search and replace of certain text in the document.

Practical Object oriented design (Text Editor Design)

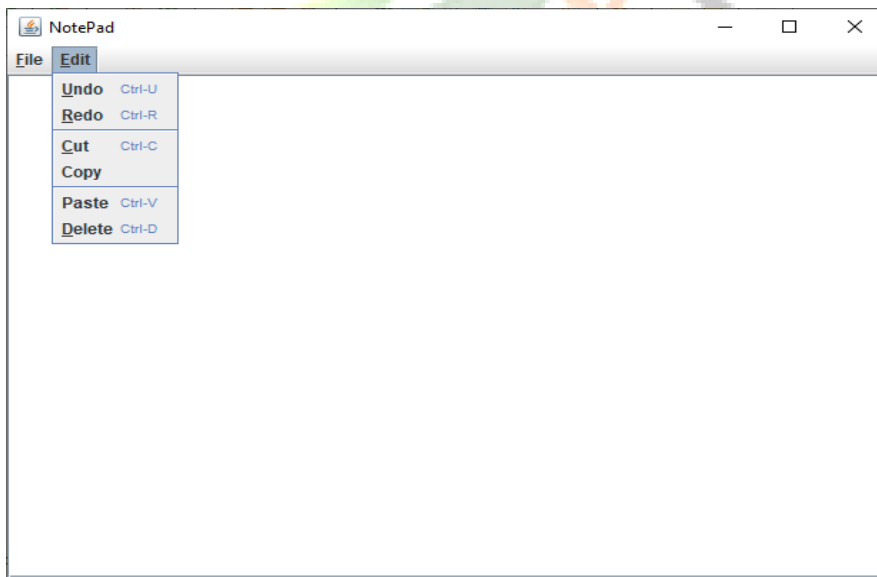
Milestone1: Design and implement the domain model for text editor that supports above operations.

Milestone2: Extend your domain model with undo/redo functionality using command pattern.

Undo/Redo: The editor must also support infinite undo and redo for insert, delete, cut and paste operations.

Milestone3: Develop a command-line interface to interact with the editor.

Milestone4: Develop a GUI interface to interact with the editor. Use MVC Architecture to separate domain model from graphical interface with controller as mediator. Sample GUI layout is given here:



Practical Object oriented design (Text Editor Design)

