



Abdul Al-Faraj

Student ID: 11014963

School Of Computer Science

Electronic and Computer Engineering

Java GUI Assignment

Introduction

In this project, JEditorPane is used to browse basic HTML pages. As it is known that, this library is already made by the JAVA API, however, it's been said that JWebPane, was going to be released to allow better CSS and HTML performance for java developers. Furthermore, the limitation of JEditorPane and the HTML render is very limited and inefficient it does not support any new HTML released, like HTML5, nor Javascript, Flash or any other modern tool used in popular browsers such as chrome and firefox.

Features

On the other hand, the application can be launched using Java executable file that you can locate in folder called Application Executable File. The software, regardless time limitation, has few good features:

- Home button.
- URL Text field, to enter any wanted URL that the user wants to navigate to.
- Reload (Refresh) button.
- Go button, to navigate to user-specified however user can also just press enter.
- Back button, will return to previous page.
- Forward button, will go forward to page that been browsed.
- Search button and search field for yahoo search engine, could not use Google because it does not permit direct search query.
- Menu Bar:
 - 1) File Menu: can create new window (shortcut ctrl+N) and exist with confirmation and it's shortcut is ctrl+X. This feature will only close the current window in case if there is more than one window opened.
 - 2) Help Menu: allow the user to read about! This will display information about the browser.

Class Representation and Organization

In my design of this browser I choose each element to be separated in different class to allow easy configuration and simplicity when it comes to change any element. The class that uses all other classes is the Browser class, and I have a driver which will run the Browser as follows in the diagram on the next page. As the diagram shows, each of the classes is separated, and they work independently. I choose this design, to allow flexibility in the design of my application.

Furthermore, each class is documented using the JavaDocs and it can be located in folder Documentation. The classes and each class is documented the same way any class in Java API is represented and documented.

