GROUP C

Project

Team Members:

Harshit Oberoi

Tanya Khullar

Yi Qi

zhenzhouyu

Yu Gong

XiuruiHou

Xuan zhang

# Menus application

## Overview

This project is to demonstrate various menu items. For each menu item, we associate an action with it to change the corresponding attribute of a sample text and also specify special characters called mnemonicsthat can provide quick access to a menu or menu item from the keyboard.

## Overall GUI structure

The program uses JFrame as a top-level container, which is associated with a JMenuBar to place menus and a JLabel to display a sample text

On the JMenuBar, we have two top-level menus: FILE and FORMAT

JMenuItems are used to display normal menu items.

JRadioButtonMenuItems are used to display items with a radio button.

JCheckBoxMenuItems are used to display items with a check box.

JMenuItem

JMenuItem

JRadioButtonMenuItem

JRadioButtonMenuItem

COLOR (JMenu)

---------------------------------------------------- (Separator)

JCheckBoxMenuItem

...

---------------------------------------- (Separator)

JRadioButtonMenuItem

...

FONT (JMenu)

FORMAT (JMenu)

FILE (JMenu)

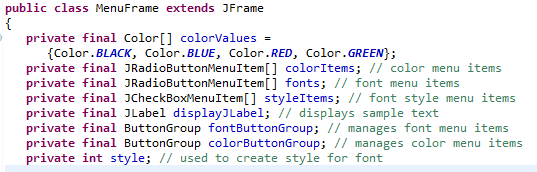
JMenuBar

JLabel

JFrame

## Detailed implementation

1. Use JFrame as a top container, and define needed components

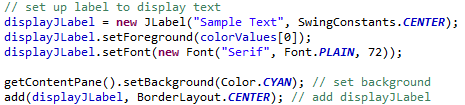


1. Define JMenuBar and JLabel, and add them to the JFrame.

For the JLabel, set the initial foreground color and font, and place it to the center position of the JFrame.

Also set the background color of the content pane



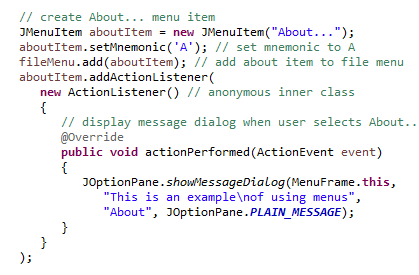


1. FILE menu

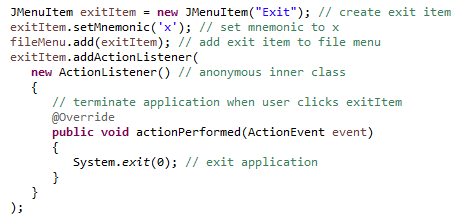
Create FILE menu and set mnemonic of this menu to “F”



Add menu item (”About...”), set mnemonic, and set corresponding action (show a dialog in the middle of JFrame)



Add menu item (”EXIT”), set mnemonic, and set corresponding action (exit program). Add FILE menu to the menu bar.





1. FORMAT menu

Create FORMAT menu, set mnemonic and add this menu to the menu bar.



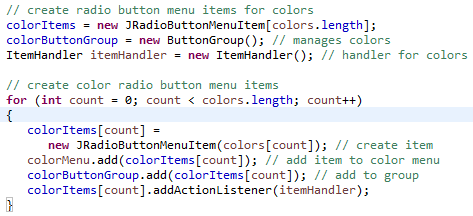


1. COLOR sub-menu

Create COLOR sub-menu, set mnemonic.



Use JRadioButtonMenuItem to implement a radio button for every color and add it to the sub-menu COLOR; add all buttons in a button group so that whenever select one button, the other buttons in the group will be deselected. Also add a listener (ItemHandler, we will discuss it later on) for every button



Select the first button by default. Add COLOR sub menu to FORMAT menu and also add a separator (a horizontal line)

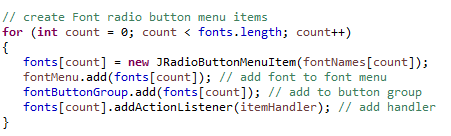


1. FONT sub-menu

Create FONT sub-menu, set mnemonic.



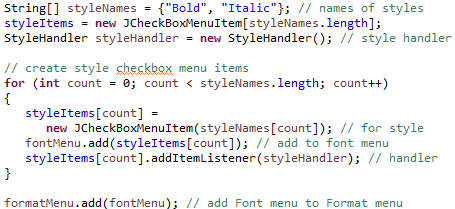
Use JRadioButtonMenuItem to implement a radio button for every font and add it to the sub-menu FONT; add all these buttons in a button group (the same function with the button group in COLOR sub-menu). Also add a listener (ItemHandler) for every button



Select the first button by default and add a separator (a horizontal line)

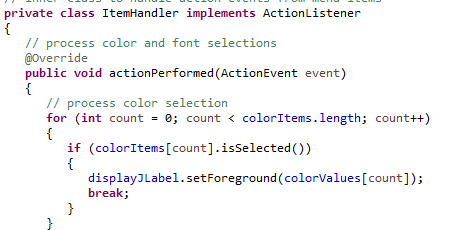


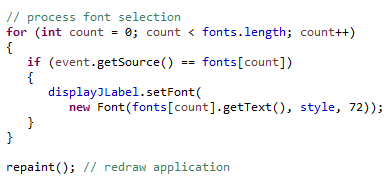
Use JCheckBoxMenuItem to implement a check box button for every style and add it to the sub-menu FONT; add a listener (StyleHandler) for every button. Note that we don’t need a button group here. (We can select more than one item when using a check box). Add FONT sub menu to FORMAT.



1. ItemHandler

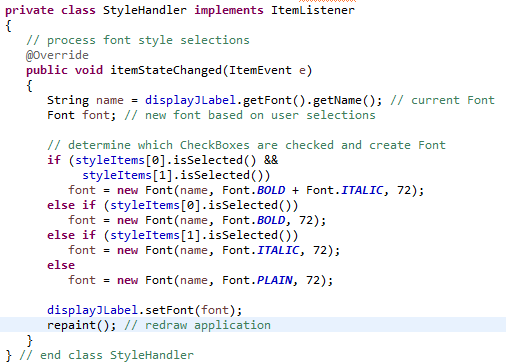
Whenever choose a color/font, update the color/font by using the selected color/font. (Note that the selected font was located by event.getSource, which returns the button that triggered the listener). After update color and font, repaint the application





1. StyleHandler

Change the style of the sample text when a user selected one or more styles on the menu. Note that there are many possible combinations of choices to deal with.

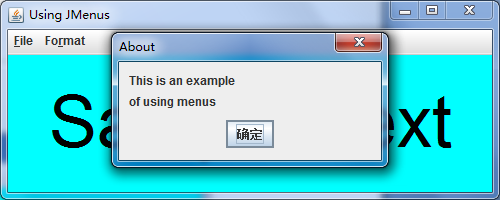


## Running Result

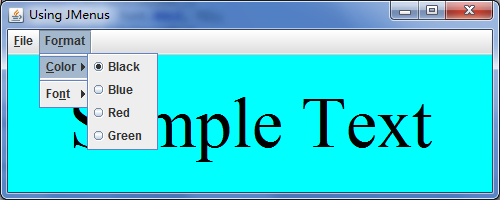
We can either use keyboard by pressing ALT + “F” or use mouse to click “File” menu to open the FILE menu.



By pressing ALT + “A”, we can see the “about” information



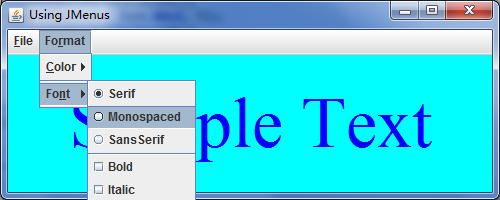
We can choose a color, black by default.

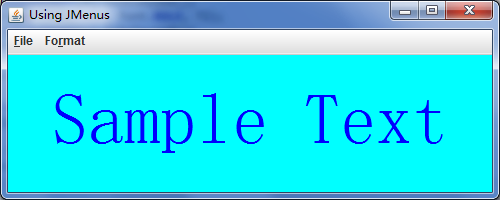


After we choose “blue”

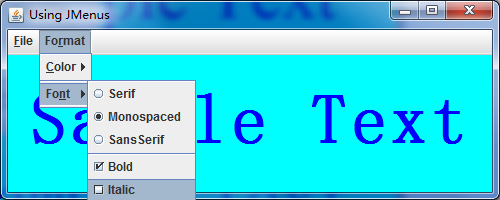


We can choose a different font.





Choose “Bold” style, and then choose another style.



Both of the two styles will be used.

