

INSY 4306 ADVANCED SYSTEMS DEVELOPMENT

ASSINGMENT -3

20 points

1. INSTRUCTIONS

- **For the late submissions, 5% of the total points will be deducted per hour automatically on Canvas.**
- You can only use the techniques that you have learned so far. Other techniques will not be accepted. **For other techniques, 10 pts will be deducted.**
- Do not drag and drop components. **It is not acceptable. You will get 0 points.**
- In this assignment, upload the following file:
 - **YourFirstNameYourLastName.zip /.rar [including, MyMenuFrame.java, MyMenuFrameTest.java, MyMenuFrame.jar]**
- Each question is **independent** of each other.
- **Do not forget to add comments to explain how your codes are working! Short comments are acceptable.**
- **Write your codes individually! Do not copy of any of them from someone else!**
- **NOTE: If you are using any IDE (Netbeans, Eclipse, etc.), please delete the statement `package xxxxx;` (and save it again), from your application. Otherwise, I will get a compilation error, and you will lose 1 pt for each file giving a compilation error. It is your responsibility.**

2. GRADING POLICY

- **Case 1:**

- For each question:
- I will compile your .java files. **If any compilation error occurs, 1 pt will be deducted from each file including compilation errors.**
- After that, I will check your algorithms whether they are correct or not. For example; if it says find odd and even numbers. I will check whether it really finds both even and odd numbers. **This part will be evaluated based on your work.**
- You can only use the techniques that you have learned so far. Other techniques will not be accepted. **For other techniques, 10 pts will be deducted.**
- Do not drag and drop components. **It is not acceptable. You will get 0 point.**
- Additionally, comments will be checked whether they clearly and briefly explain what you have done. **If comments are missing or not clear, enough, or brief 1 pt will be deducted.**

- **Case 2:**

- For each question:
- If there is not any compilation error:
 - I will try each case scenario as stated in each question. For example; if it says find odd and even numbers. I will try both even and odd numbers. **This part will be evaluated based on your work.**
 - You can only use the techniques that you have learned so far. Other techniques will not be accepted. **For other techniques, 10 pts will be deducted.**
 - Do not drag and drop components. **It is not acceptable. You will get 0 point.**
 - Additionally, comments will be checked whether they clearly and briefly explain what you have done. **If comments are missing or not clear, enough, or brief 1 pt will be deducted.**

- **Case 3:**

- **If you do not upload a .java file, I will not evaluate your answer.**

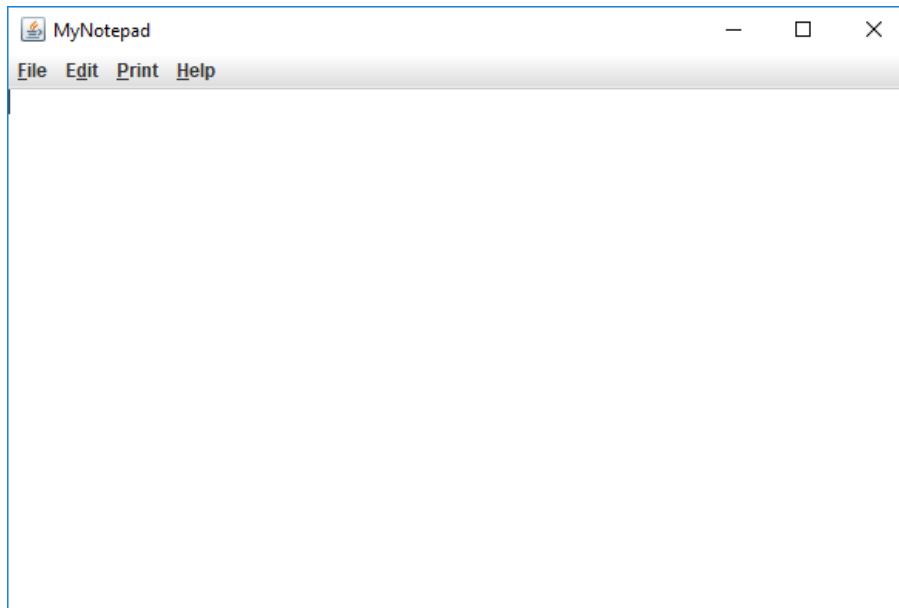
- **Case 4:**

- **If it is determined that you copy the codes from someone else, you will get 0 pt.**

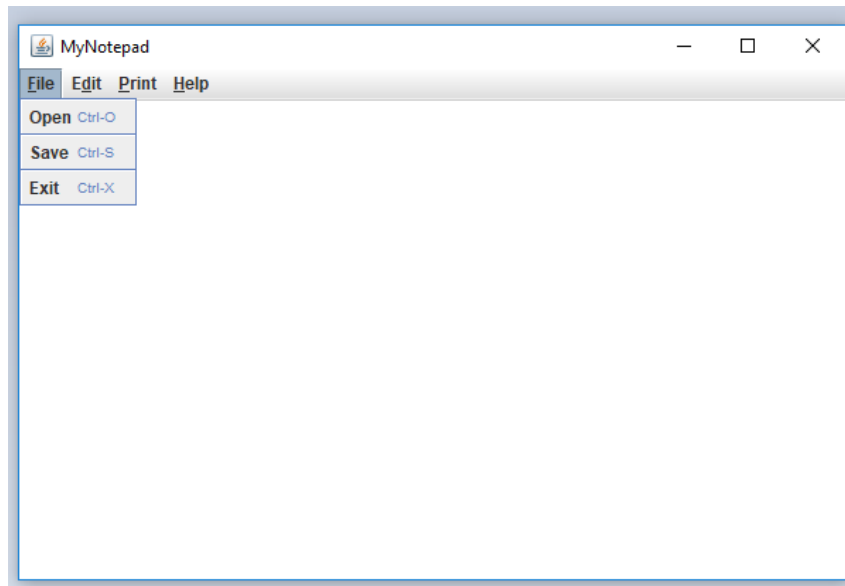
QUESTIONS

Note: In each question, assume that the user enters correct inputs. You do not handle exceptions. Do not drag and drop your components. You must create Java class, not Java Form. Do not drag and drop components!

1. Write a simple Notepad application. (20 pts).



- a. The title of the frame is “MyNotepad”.
- b. Create and add border layout.
- c. Create a text area to display contents. Add the text area to the center of the border layout.
- d. Create a menu bar.
- e. Create a file menu. Set mnemonic for file menu. It is “F”. File menu includes three menu items.
- f. Add a separator between each menu item in the file menu.



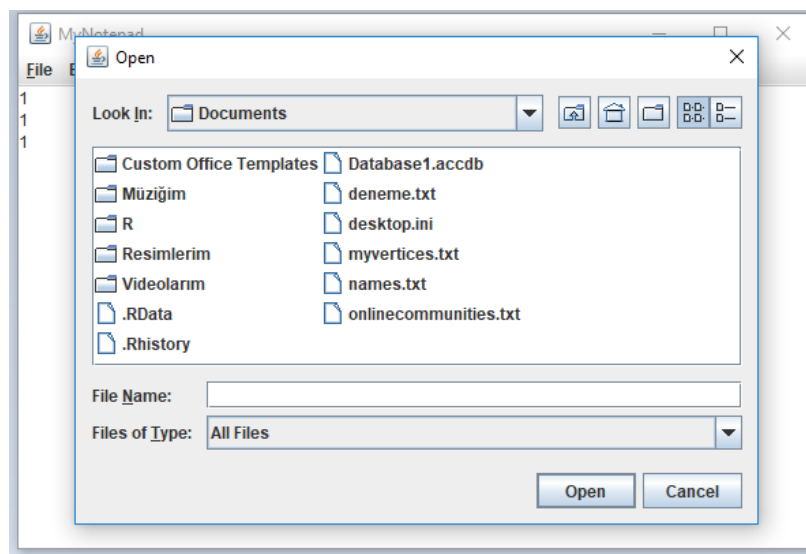
i. Open

1. Add a short cut for the menu item. It is ctrl+O.

a. Use this syntax:

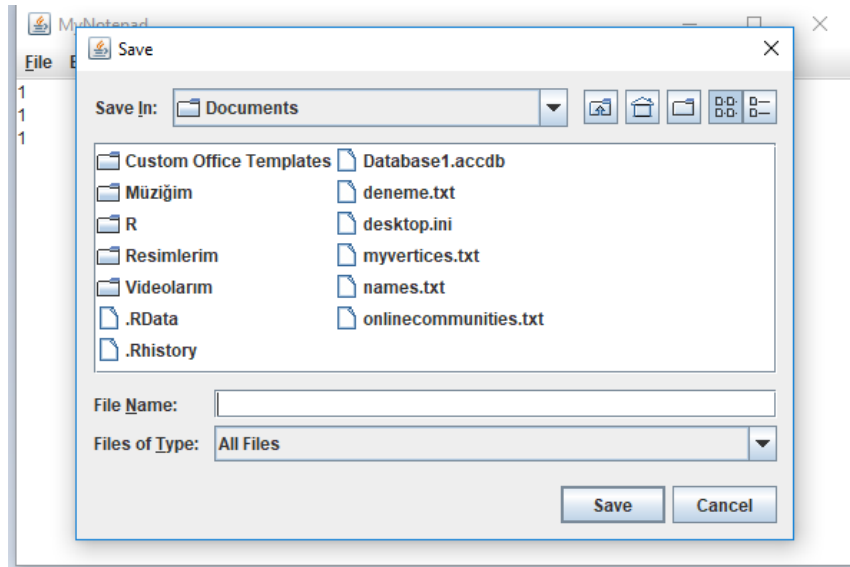
`MenuItem.setAccelerator(KeyStroke.getKeyStroke('O', CTRL_DOWN_MASK));`

2. When a user clicks it (**an action event occurs**) a file chooser is created, and an open dialog box is opened. The user selects a text file to open it and then clicks “Open” or the user can click on “Cancel”. Use try block with resources. (*Hint: add catch block automatically, click add catch clause*).



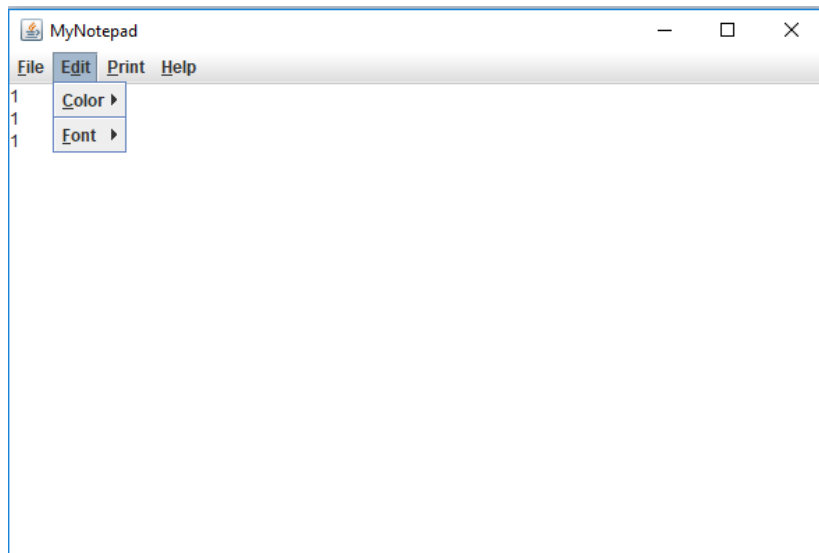
ii. Save

1. Add a short cut for the menu item. It is ctrl+S.
2. When a user clicks it (**an action event occurs**) a file chooser is created, and a save dialog box is opened. The user writes the name of text file to save it and then clicks “Save” or the user can click on “Cancel”. User try block with resources. (*Hint: add catch block automatically, click add catch clause*).

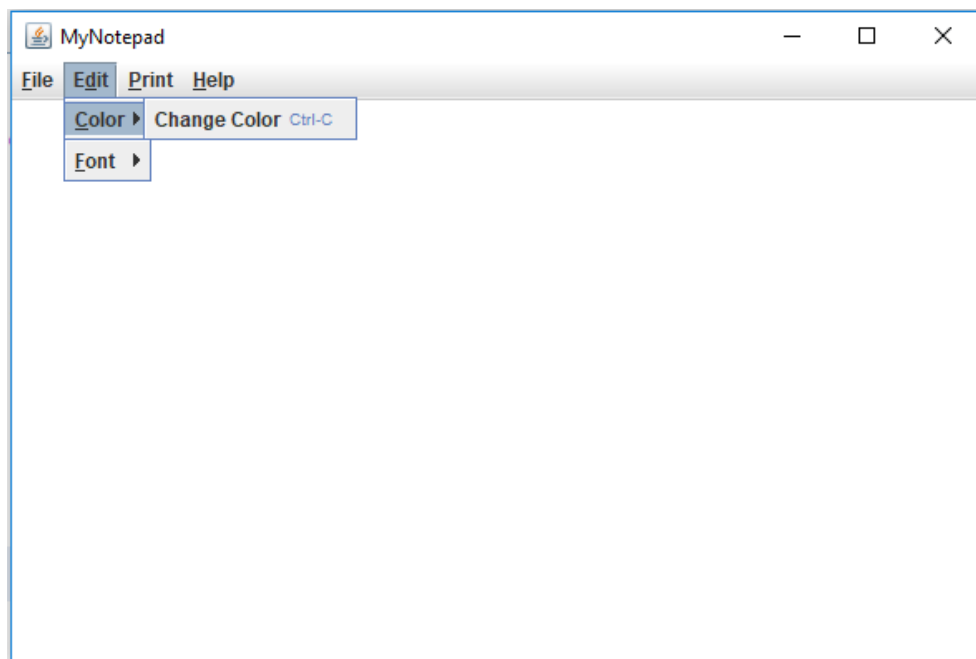


iii. Exit

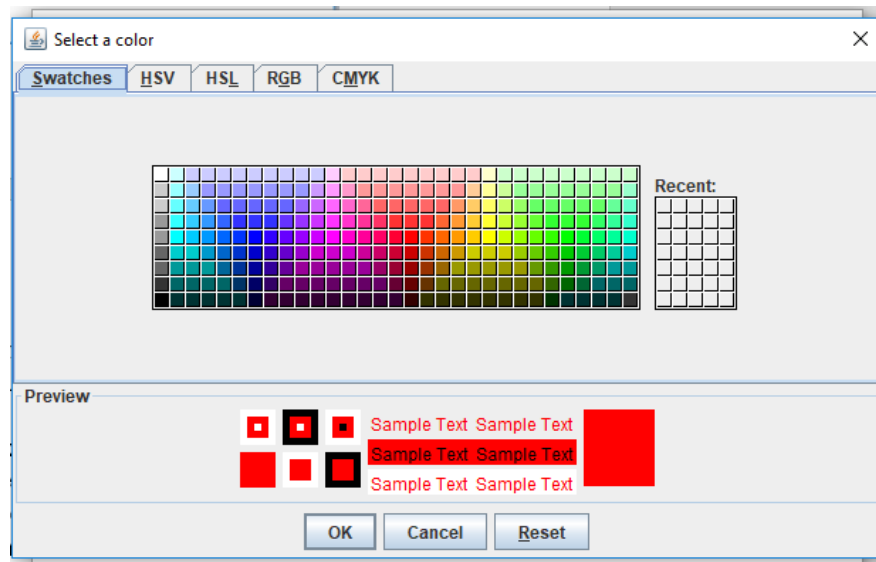
1. Add a short cut for the menu item. It is ctrl+X.
 2. When a user clicks it (**an action event occurs**), it terminates the application.
- g. Create edit menu. Set mnemonic for file menu. It is “D”. Edit menu includes two menus: Color and Font.
- h. Add a separator between each sub menu.



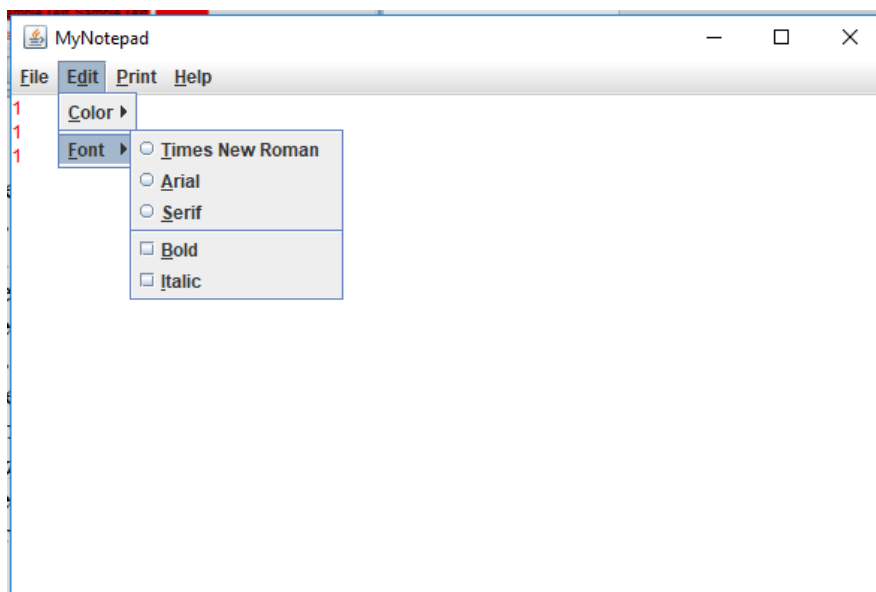
- i. Create color menu. Set mnemonic for Color menu. It is "C".
 - i. Create change color menu item. Add a short cut for the menu item. It is ctrl+C.



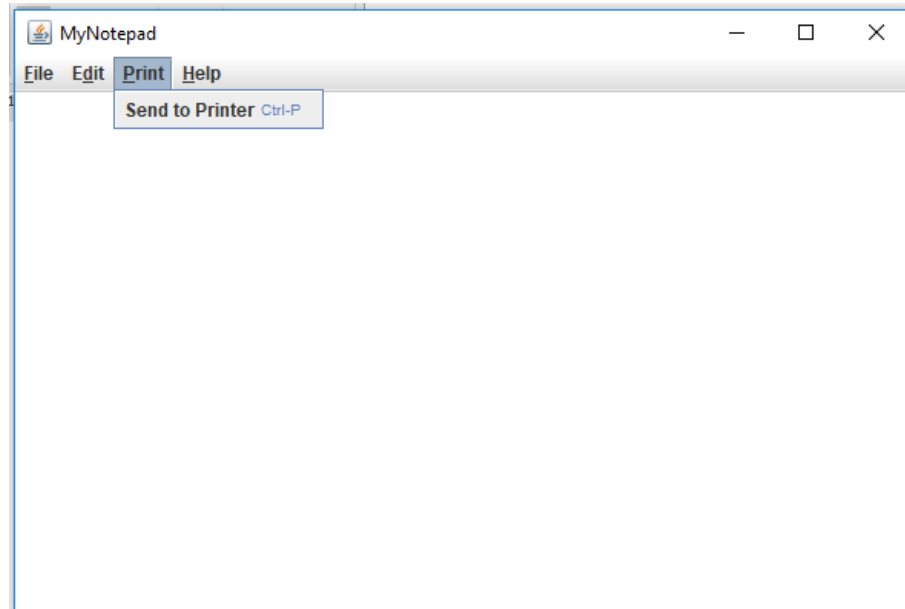
- ii. When a user clicks change color, a color chooser is opened, and the user selects a color. By default, red color is selected. Then, the color of the text is changed.



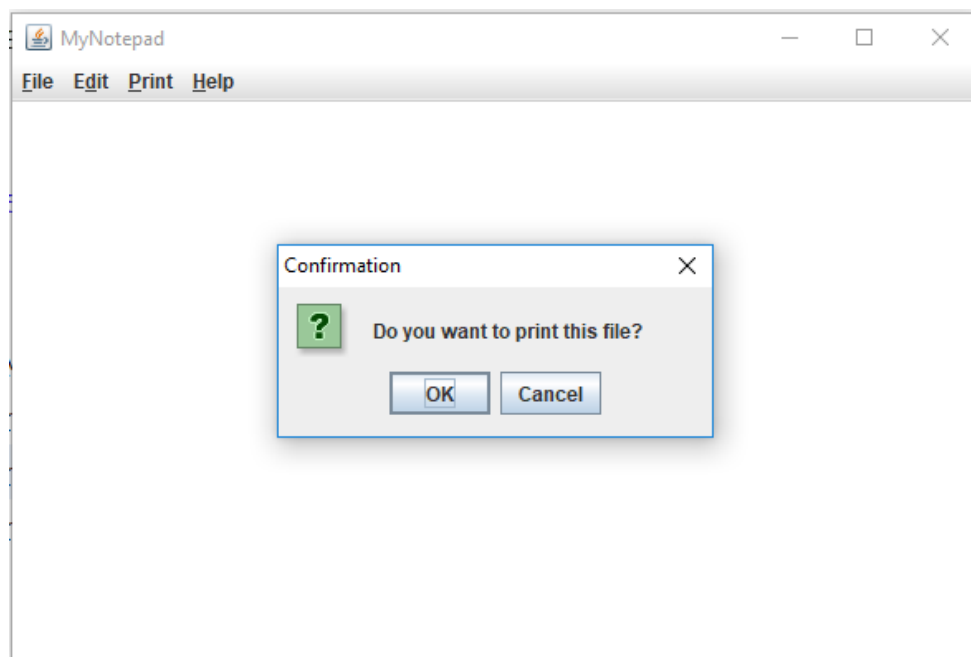
- j. Create font menu. Set mnemonic for Font menu. It is “F”.
- This menu includes three radio button menu items: Times New Roman, Arial, Serif. (*Hint: you also need a button group*). When a user selects one of these fonts (**an action event occurs**), the font of the text will change. Set font size as 20.
 - This menu includes also two check box menu items: Bold, Italic. A User can select both, only bold one, italic one, or unselect any of them (**an item event occurs**). Based on the user selection, set the font. (*Hint: if one of them is unselected, set the font as plain*). Set font size as 20.
 - Add a separator between radio button menu items and check box menu items.

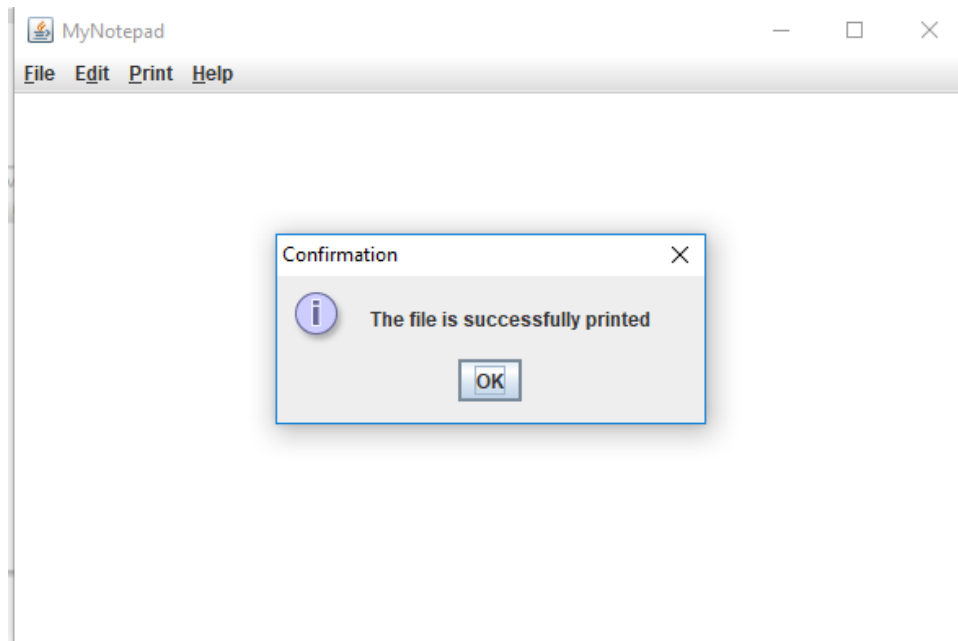


- k. Create Print menu. Set mnemonic for print menu. It is “P”.
 - i. Printer menu includes one menu item: Send to Printer. Add a short cut for the menu item. It is ctrl+P.

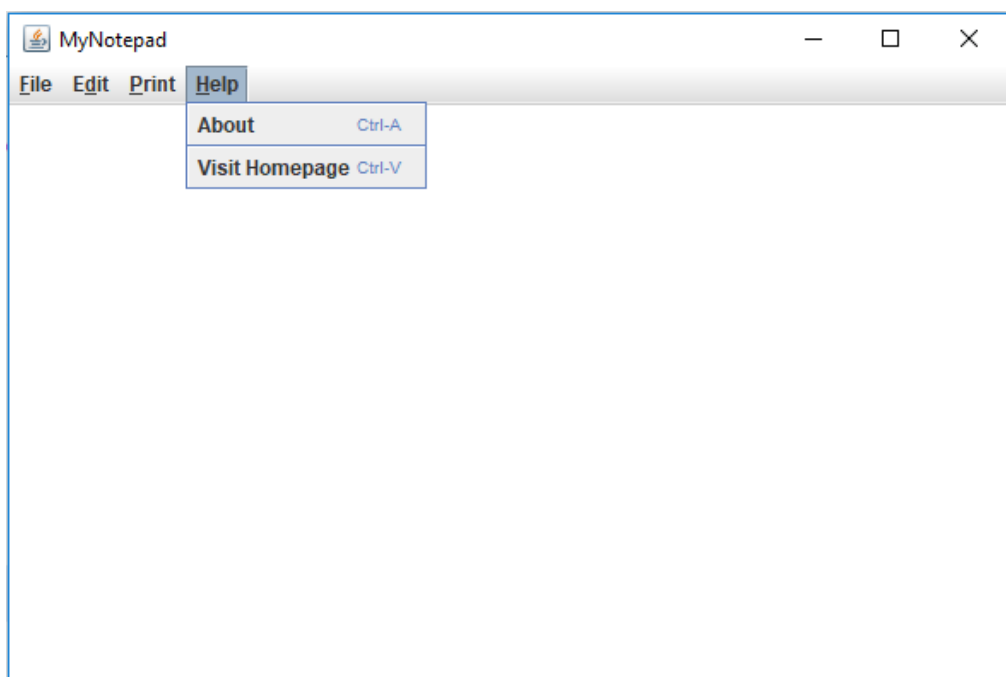


- ii. When a user clicks it (**an action event occurs**), display an Option Dialog. Display the message in the figure. If the user clicks “Ok”, display a message dialog box (set information icon). Display the message in the figure. If the user clicks cancel, make the current frame as visible.



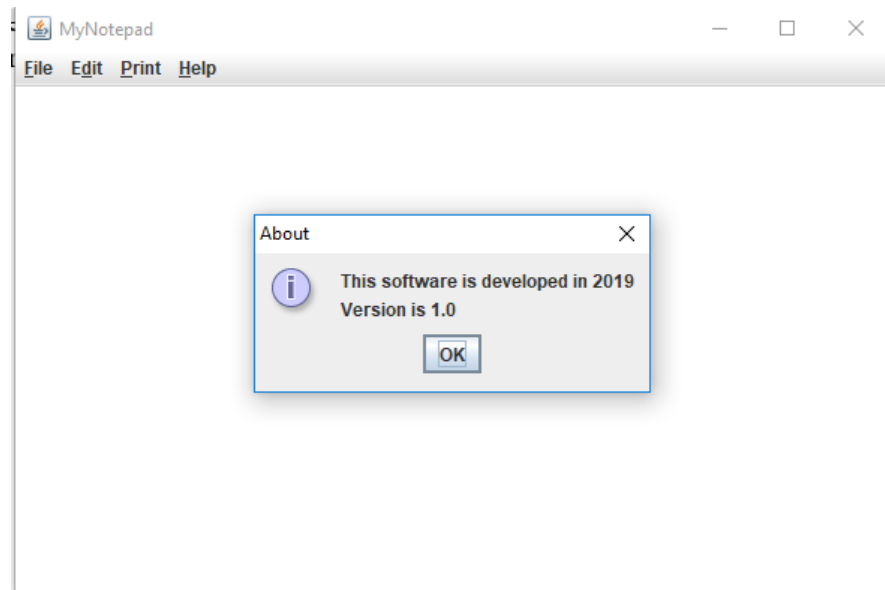


- l. Create a help menu. Set mnemonic for help menu. It is "H".
- m. Help menu includes two menu items: About, Visit Homepage. Add a separator between these menu items.



- i. Create a menu item which is about. Add a short cut for the menu item. It is ctrl+A.

1. When a user clicks it (an action event occurs), display a show message dialog box. Display the message shown in the figure. Display information icon.



- ii. Create a menu item which is visit homepage. Add a short cut for the menu item. It is ctrl+V.
 1. When a user clicks it (an action event occurs), the user will be navigated to <http://www.microsoft.com>.
 2. For the navigation, create a static method (copy it):

```
public static void openWebpage (String urlString) {  
    try {  
        Desktop.getDesktop().browse(new  
            URL(urlString).toURI());  
    }  
    catch (Exception e) {  
        e.printStackTrace();  
    }  
}
```

Then in the action performed method, call this static method and provide the url String.

- n. When it is necessary, import the necessary classes and interfaces.
- o. Set frame as 600*400.