Object Oriented Programming
Assignment 03 - Creating Toolbars and Buttons

March 14, 2023

1 Assignment Goal

The purpose of this assignment is to use Object Oriented Programming into creating an effective user interface that contains drop-down menus, title bars, buttons and layers.

You can access Java Documentation at this link.

For all the graphics, you will need to read the Java Documentation at this link.

2 The Task

This assignment is all about doing the ground work for making the final product of this course. Even though Java comes with its own library for making the UI elements (called swing) we will be making our own elements by making full use of object-oriented programming techniques. This assignment will help you in understanding animations as well and will hopefully provide a complete understanding about how different buttons and menu items work in a desktop application.

The assignment was created using some of the code from zetcode. You can look into it if you want to get some understanding of the code or make some future projects but it will not help you in completing this assignment. The screen is continuously being refreshed, so you can create animations if you want or even a game but that is not the requirement of this assignment.

You will be creating the following items:

- Active buttons (which will return to depressed state after being pressed)
- Toggle buttons (which will stay in pressed until they are not being used anymore)
- Text boxes
- Menu buttons (these will be part of the menus and title bars
- Toolbars (these will contain buttons and will themselves be part of a window)
- Windows (These will contain toolbars and buttons)

You can look at MSPaint program to have an understanding of what you are going to be making. Do keep in mind that this assignment is just for creating the components and all the implementation will come in the next assignment (Such as drawing of shapes, saving files, undo and redo). If you feel adventurous then you can start doing the implementation as well but it will not hold any marks currently.

2.1 File Menu

You will create a menu bar which will have two items, **File** and **Edit**. The Clicking on the **File** menu item will initiate a drop-down menu from which you will be able to select: **New**, **Open** and **Save** menu items. Clicking on **New** will refresh the entire screen and lists. Clicking on **Open** will show you a window with a list of files that can be opened and Clicking on **Save** will let you save the file as a text file with the time-stamp as the file name.

Selecting the **Edit** menu item will drop down a menu which will have **Undo** and **Redo** commands. These commands will only work on the currently selected layer. All menu items should have keyboard shortcuts assigned to them. The following figure should be used as reference and may not be followed exactly:

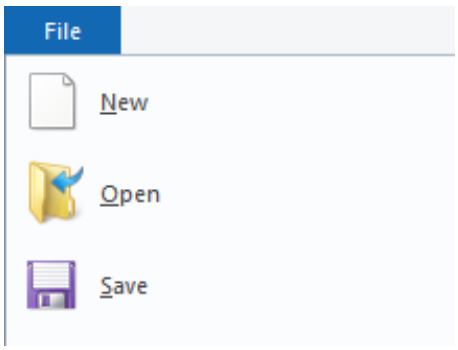

Figure 1: File menu

2.2 Shapes Toolbar & Buttons

You will create a Shapes Toolbar which will contain all the buttons that will contain the following shapes:

- Right angle triangle
- Equilateral triangle
- Rectangle
- Circle
- Hexagon
- Pentagram

The following figure should be used as reference and may not be followed exactly:

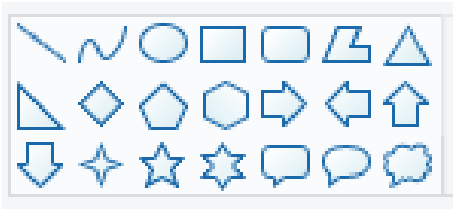

Figure 2: Shapes Toolbar

2.3 Colors Toolbar & Window

You will need to create a colors toolbar as shown below. Color1 here will be stroke color and Color2 will be fill color so label them as such:

Figure 3: Colors Toolbar

By clicking on the color gradient a color window will be created for selecting color from a gradient as shown below. Keep in mind that once a color has been selected from the color window and added to the color gradient, it will be shown within the color toolbar. If you are wondering how the color gradient can be made, you will need to come up with your own algorithm by reading the color values in MSPaint. Using **ColorGradient** class is forbidden.

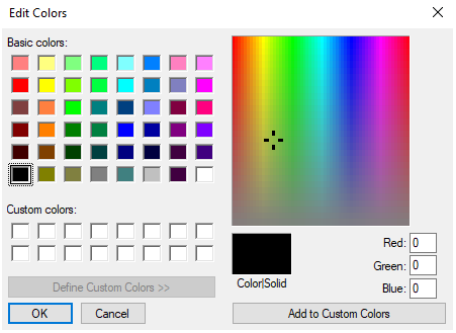

Figure 4: Colors Window

2.4 Layers Toolbar

On the right side of the application, there will be a Layers toolbar in which you can add and remove layers. Just keep in mind that you will always draw on the top-most layer. Everything you will draw will be added to this layer and if the layer is deleted, all the items that were in the layer will also be deleted. You will have buttons that will raise or lower the layer. Think of them as linkedlists within linkedlists.

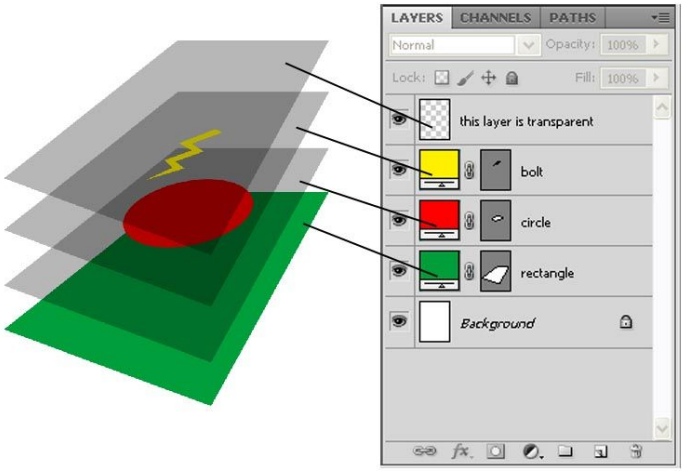

Figure 5: Layers

3 Important Information

- You need to make the buttons and elements such that a user knows exactly what has been selected or the current state of various buttons or menu items.
- Buttons for Undo and Redo need to be created as well, also for free drawing and stroke width.
- You create free drawing by creating a linkedlist and drawing circles which are stored in the linked list. The linkedlist itself is then pushed in the shapes list. This can be done by using interfaces as the linkedlist is not a shape. This implementation will be required in next assignment.
- You will need to use polymorphism and inheritance where-ever necessary. For example, there are multiple drop-down menus and multiple toolbars and buttons. Reuse code as much as possible... do not duplicate code in different classes.
- You can add buttons where-ever required or found necessary.

- You can use the built-in Java linkedlists or Arraylists if you want.
- For buttons, you can either draw the icons yourself or use images as done in the attached code.
- The attached code just shows mouse click interaction, shows how a button can be interacted with mouse clicks and key presses. You can use this code as a starting point for your own implementation.
- You can make use of all of your previous code or the lab code that you made provided that it was your own code. Do not use anyone else's implementation.

4 Submission

The deadline of the assignment is 2nd of April at 11:55 pm. There will be no extensions and no late submissions will be accepted. You should complete the assignment at least a day before the submission deadline.

You will zip the source folder and name it as your_name_ERP.zip. Suppose your name is John Doe and your ERP is 12345, then the name of your file should be John_Doe_12345.zip. You will then upload it to LMS before the deadline.

- If you upload the wrong file, you will not get any marks.
- If at the last moment LMS does not work, you will not get marks. LMS tends to fail at the last moment as everyone is uploading files. The load makes it crash so avoid uploading the files at the last moment. If you choose to take that risk, then it will be your choice.
- Do not email me your assignments or upload them on dropbox, they will not be checked
- If the size of the files is too big, you are uploading files that are not required so remove those files from the zip file and then upload them.
- If your files are not in a zip file, they will not be checked.

Keep in mind that any attempt at plagiarism or collusion will cause you to automatically fail the course, hence write your own code.