Web Technology
Name: Veeransh Shah
Reg Id: 221070063

Aim:
Write in Java to create applets incorporating the following features:
Create a color palette with matrix of buttons
Set background and foreground of the control text area by selecting a color from color palette.
In order to select Foreground or background use check box control as radio buttons to set background image.

Theory:

To create an applet with a color palette and functionalities to set the background and foreground colors of a text area based on selected colors from the palette, you'll need to define a layout for arranging buttons representing different colors. Then, implement action listeners for the buttons to update the foreground and background colors of the text area accordingly. Additionally, use checkbox controls to toggle between setting the background or foreground color, ensuring only one option is selected at a time. Handle checkbox state changes to adjust the functionality of the color palette buttons.

Here's a concise outline of the steps:

1. Define a layout for the color palette buttons and implement action listeners to update the text area's colors based on the selected palette color. Use checkbox controls to toggle between setting the background or foreground color, ensuring single selection.
2. Handle checkbox state changes to appropriately adjust the behavior of the color palette buttons. By integrating these features, users can select colors from the palette to customize the background and foreground of the text area dynamically, enhancing the interactive experience of the applet.

Code:

```
import java.net.*;
import java.awt.*;
import java.awt.event.*;
import java.applet.*;

/*<applet code="Cpdemo" width=500 height=600>
 </applet>
*/
@SuppressWarnings("unused")
public class Cpdemo extends Applet implements ActionListener
{
```

```
 Panel p1,p2,p3;
 Checkbox c1,c2;
 CheckboxGroup cbg;
 TextArea ta;
 Button b1,b2,b3;

public void init()
 {
   setLayout(new BorderLayout());

   p1=new Panel(new FlowLayout());
   p2=new Panel(new GridLayout(2,2));
   p3=new Panel(new FlowLayout(FlowLayout.CENTER));

   cbg=new CheckboxGroup();

   c1=new Checkbox("FOREGROUND",cbg,true);
   c2=new Checkbox("BACKGROUND",cbg,false);

   p1.add(c1);
   p1.add(c2);

   add(p1,BorderLayout.NORTH);

   b1=new Button("RED");
   b2=new Button("BLUE");
   b3=new Button("GREEN");
   b1.setBackground(Color.RED);
   b2.setBackground(Color.BLUE);
   b3.setBackground(Color.GREEN);

   p2.add(b1);
   p2.add(b2);
   p2.add(b3);

   add(p2,BorderLayout.CENTER);

   ta=new TextArea("My first Applet project\nName:Pratham
Shah\nID:221070061",5,20);

   p2.add(ta);

   add(p3,BorderLayout.SOUTH);
```

```
   //c1.addItemListener(this);
   //c2.addItemListener(this);

   b1.addActionListener(this);
   b2.addActionListener(this);
   b3.addActionListener(this);
}
  public void actionPerformed(ActionEvent ae)
   {
     String str=ae.getActionCommand();
     String str1=cbg.getSelectedCheckbox().getLabel();
     System.out.println(str1);

     if(str1.equals("FOREGROUND"))
       {

         if(str=="RED")
          ta.setForeground(Color.red);

         if(str=="BLUE")
          ta.setForeground(Color.blue);

         if(str=="GREEN")
          ta.setForeground(Color.green);
    }

     else
     {
         if(str=="RED")
          ta.setBackground(Color.red);

         if(str=="BLUE")
          ta.setBackground(Color.blue);

         if(str=="GREEN")
          ta.setBackground(Color.green);
       }
   }
}
```

Conclusion

In conclusion, by incorporating a color palette and checkbox controls within a Java applet, we've facilitated dynamic customization of the background and foreground colors of a text area. This interactive feature enhances user experience and allows for greater control over the applet's appearance. The applet provides a simple yet effective interface for users to select colors and apply them to the text area, offering versatility in design. Through thoughtful integration of these elements, we've created a functional and visually appealing tool suitable for various interactive applications.