The FontAnimation Application

Learning Objectives

The development process of the FontAnimation application will help the students to:

- Create a simple Java based GUI application
- Understand the concept of applet programming
- Implement Graphics and multi-threading programming concepts.

Understanding the Calculator Application

The FontAnimation application presents the concepts of Applets, Graphics and multi-threading in java. The application creates an applet which draws a text and keeps on increasing its font size to a defined maximum limit and then reduces the font size back to the original size. The process is dynamic and continuous. The user is provided options to change the text and text color. Events and listeners have also been used in the application.

Creating the Calculator Application

Class FontAnimation uses Abstract Window Toolkit (AWT) controls to add textfields and buttons to provide options to users for dynamically changing the text and its color, while running the applet. To create the FontAnimation application, create a file named FontAnimation.java which inherits Applet and implements Listener ActionListener which listens to actions performed using JButton control.

Creating the Java File

The FontAnimation Class (FontAnimation.java)

Class FontAnimation inherits applet and uses thread programming to dynamically change the text size and its color. It contains the following methods:

- init(): init() Is a lifecycle method of applet and is used to initialize the applet object. Adds controls to the applet and registers the ActionListener for JButton control.

- actionPerformed(): Method of Listener ActionListener and is called on the click of JButton control. The method contains the code for getting the user input, namely, text and the text color properties and applying the user-entered values to the initial values. After updating the text, applet is painted again to show the changes made.

- paint(): Draws the text and sets its color and uses threads to continuously change the size of the text.

Generating the Class Files

The steps for generating the class file for FontAnimation application are:

- Place the FontAnimation.java file in the required directory.
- In the command prompt, go to the directory where the java file is stored.

- Compile the file using the command for compiling the java files:

  javac FontAnimation.java

Creating the HTML File

To use an applet, an HTML file has to be created which will call the Java Applet. To add Class FontAnimation to FontAnimation.html file, HTML tag <APPLET> is used in which the code attribute will contain the value “FontAnimation.class”.

Working with the FontAnimation application

The steps for working with the FontAnimation application are:

- In the command prompt, go to the directory containing the class files for FontAnimation.
- Enter the following command to run the Calculator application:

  appletviewer FontAnimation.html

- The applet will be displayed with the default text as “Hello” and color as “black”.
- The user can change the text and Red, Green and Blue values for color of the text. On clicking button “Apply”, the text changes according to the user entered values.
- During the running of the applet, the text size will continuously increase and decrease.

Code for the Calculator Application

FontAnimation.java

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

public class FontAnimation extends Applet implements ActionListener
{
	TextField text=new TextField(20);
	TextField red=new TextField(20);
	TextField gr=new TextField(20);
	TextField blue=new TextField(20);
	Button b1=new Button("Apply");
	String str="Hello";
	int size=10;
	Font f;
	boolean inc=true;
	int r,g,b;
	Color fcolor;

	public void init()
	{
		add(new Label("Enter Text to animate here"));
		add(text);
		add(new Label("Enter value for Red Color here"));
		add(red);
```

```
        red.setText("0");
        add(new Label("Enter value for Green Color here"));
        add(gr);
        gr.setText("0");
        add(new Label("Enter value for Blue Color here"));
        add(blue);
        blue.setText("0");
        add(b1);
        b1.addActionListener(this);
    }

    public void actionPerformed(ActionEvent e)
    {
        if(e.getSource()==b1)
        {
            str=text.getText();
            if(str=="")
                str="Hello";
            r=Integer.parseInt(red.getText());
            g=Integer.parseInt(gr.getText());
            b=Integer.parseInt(blue.getText());
            fcolor=new Color(r,g,b);
            repaint();
        }
    }

    public void paint(Graphics g)
    {
        f=new Font("Arial",Font.BOLD,size);
        g.setFont(f);
        g.setColor(fcolor);
        g.drawString(str,50,200);
        try
        {
            Thread.sleep(500);
        }
        catch(Exception e)
        {
            System.out.println(e.getMessage());
        }
        if(inc==true)
        {
            size+=10;
            if(size==100)
                inc=false;
        }
        else
        {
            size-=10;
            if(size==10)
                inc=true;
        }
        repaint();
    }
}
```

FontAnimation.html

```
<html>
<head>
<title>Applet Demonstrating Applet and Thread Programming in Java</title>
</head>
<body>
         <APPLET CODE=FontAnimation.class
        WIDTH=400 HEIGHT=800>
         </APPLET>
</body>
</html>
```

Exercises:

1. Write the java code to add other graphics controls to an applet like line, rectangle and others.