COMPUTER ENGINEERING

OOPM ODD SEM 2021-22/EXPERIMENT 7

NAME:- GAURAV AMARNANI (D7A, 67)

Experiment - 7

To implement applets

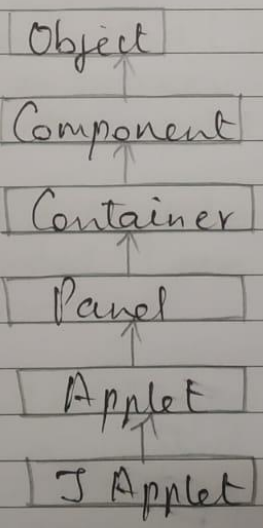
AIM: To implement applet

Theory:

Java Applet

Applet is a special type of program that is embedded in a webpage to generate the dynamic content. It runs inside the browser and works at client side.

Heirarchy of Applet



As displayed in the above diagram, Applet class extends Panel. Panel class extends container which is the subclass of Components.

Lifecycle of Java Applet:

- Applet is initialized
- Applet is Started
- Applet is Painted
- Applet is stopped
- Applet is destroyed

The java applet.Applet class provides 4 lifecycle Methods and java.awt.Component class provides lifecycle Methods for an applet

java.applet.Applet class

- public void start().- is invoked after the init Method or browser is maximized. It is used to start the applet

- public void stop().- is used to stop the applet It is invoked when Applet is stop or browser is minimized.

- public void destroy().- is used to destroy the Applet. It is invoked only once.

java awt Component class

- public void paint (Graphics g): is used to paint the applet. It provides Graphics class object that can be used for drawing oval, rectangle, arc, etc

Conclusion:

By performing this experiment we understand the concept of Applet, how we can generate Dynamic content using applet and various graphics Method.

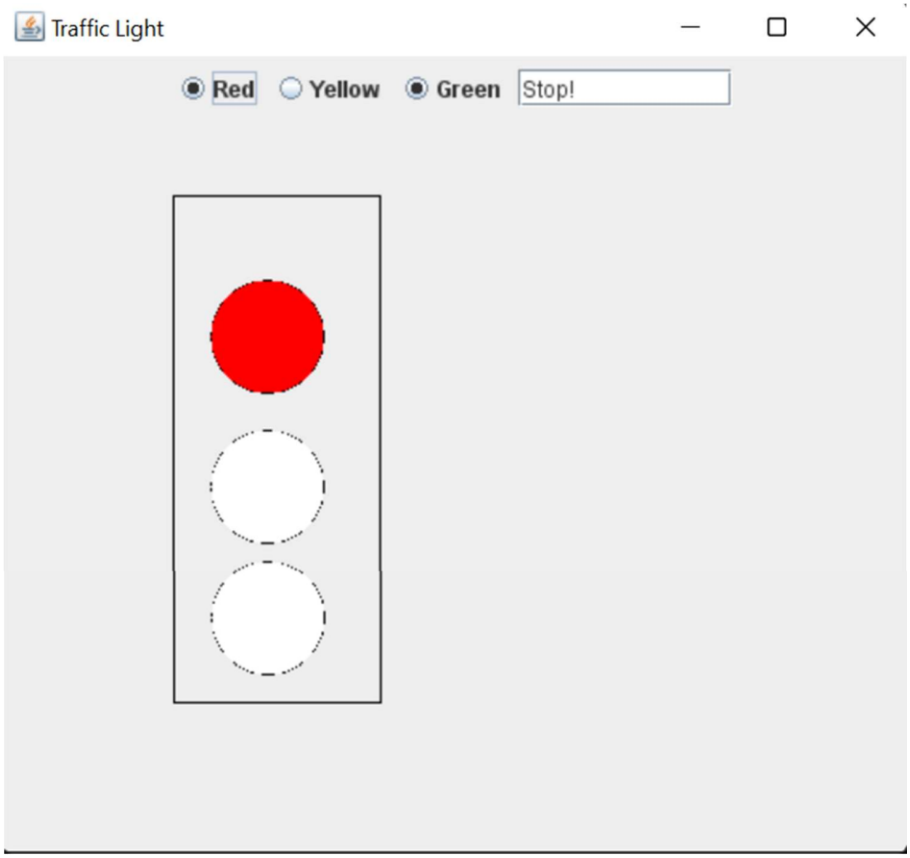
Program:

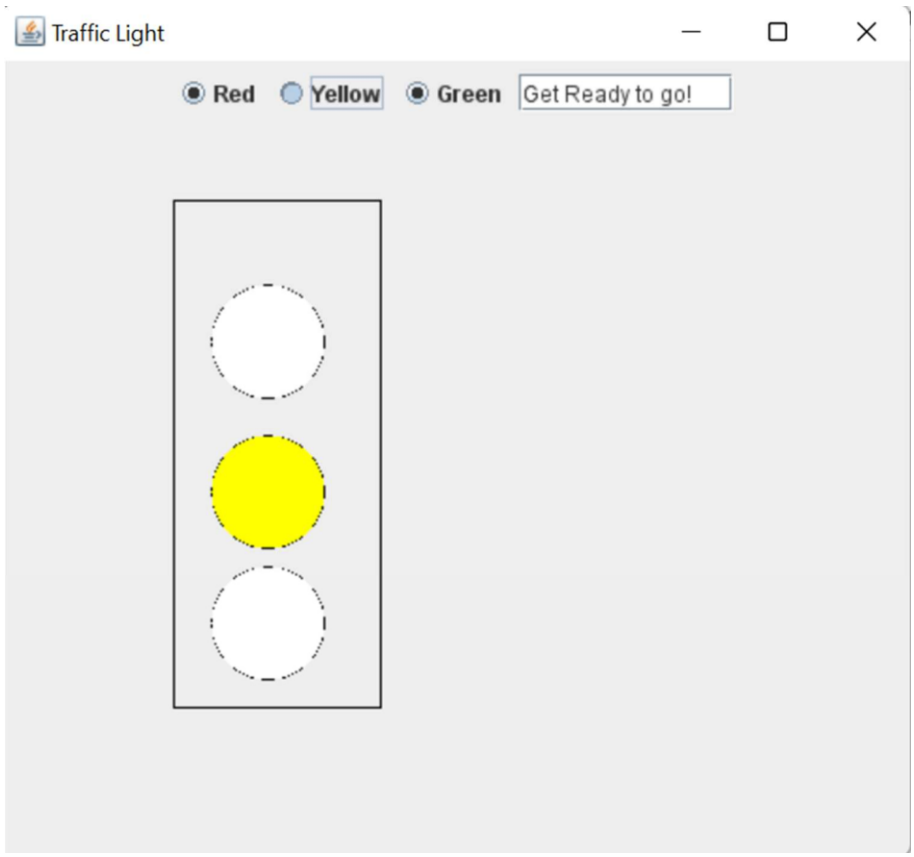
```
import java.awt.*;
import java.awt.event.*;
import java.util.*;
import javax.swing.*;
public class Traffic_Signal extends JFrame implements ItemListener {
JRadioButton jr1;
JRadioButton jr2;
JRadioButton jr3;
JTextField j1 = new JTextField(10);
ButtonGroup b = new ButtonGroup();
String msg = " ";
int x = 0, y = 0, z = 0;
public Traffic_Signal(String msg){
super(msg);
setLayout(new FlowLayout());
jr1 = new JRadioButton("Red");
jr2 = new JRadioButton("Yellow");
jr3 = new JRadioButton("Green");
jr1.addItemListener(this);
jr2.addItemListener(this);
jr3.addItemListener(this);
add(jr1);
add(jr2);
add(jr3);
b.add(jr1);
b.add(jr2);
b.add(jr3);
add(j1);
addWindowListener(new WindowAdapter() {
public void windowClosing(WindowEvent e){
System.exit(0);
}
});
}
public void itemStateChanged(ItemEvent ie){
if (ie.getSource() == jr1) {
if (ie.getStateChange() == 1) {
msg = "Stop!";
x = 1;
```

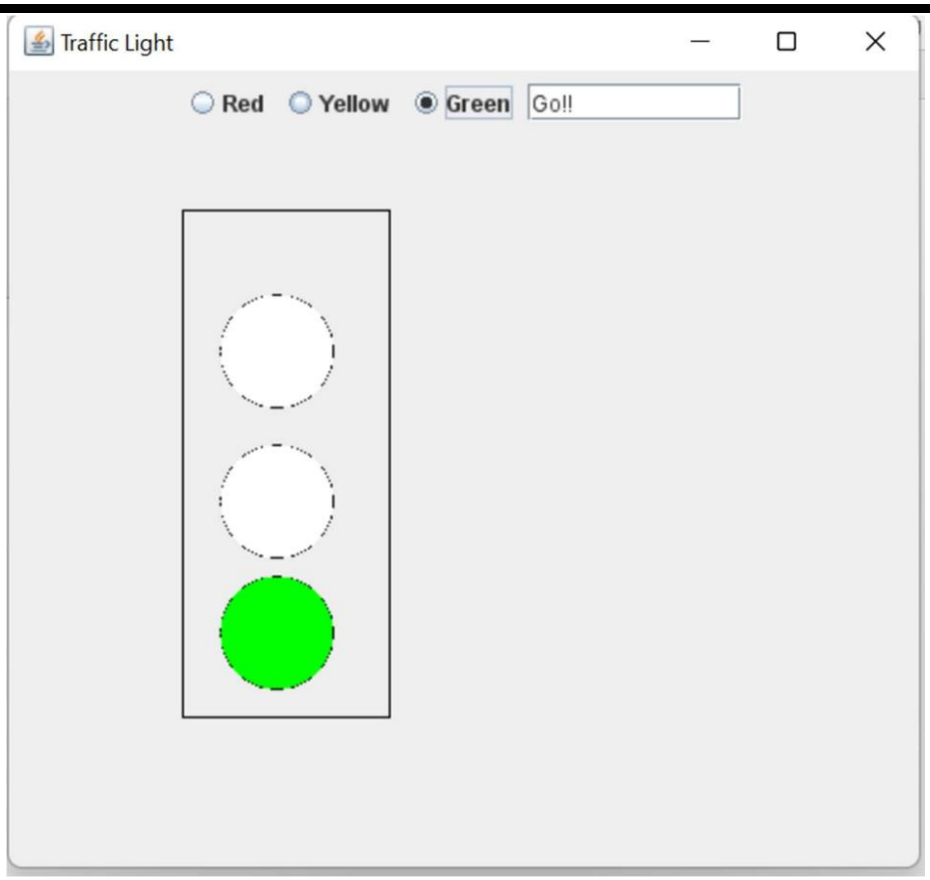
```
repaint();
}
else
msg = "";
}
if (ie.getSource() == jr2) {
if (ie.getStateChange() == 1) {
msg = "Get Ready to go!";
y = 1;

repaint();
}
else
msg = "";
}
if (ie.getSource() == jr3) {
if (ie.getStateChange() == 1) {
msg = "Go!!";
z = 1;
repaint();
}
else
msg = "";
}
j1.setText(msg);
}
public void paint(Graphics g){
g.drawRect(100, 105, 110, 270);
g.drawOval(120, 150, 60, 60);
g.drawOval(120, 230, 60, 60);
g.drawOval(120, 300, 60, 60);
if (x == 1) {
g.setColor(Color.RED);
g.fillOval(120, 150, 60, 60);
g.setColor(Color.WHITE);
g.fillOval(120, 230, 60, 60);
g.setColor(Color.WHITE);
g.fillOval(120, 300, 60, 60);
x = 0;
}
if (y == 1) {
```

```
g.setColor(Color.WHITE);
g.fillOval(120, 150, 60, 60);
g.setColor(Color.YELLOW);
g.fillOval(120, 230, 60, 60);
g.setColor(Color.WHITE);
g.fillOval(120, 300, 60, 60);
y = 0;
}
if (z == 1) {
g.setColor(Color.WHITE);
g.fillOval(120, 150, 60, 60);
g.setColor(Color.WHITE);
g.fillOval(120, 230, 60, 60);
g.setColor(Color.GREEN);
g.fillOval(120, 300, 60, 60);
z = 0;
}
}
public static void main(String args[]){
JFrame jf = new Traffic_Signal("Traffic Light");
jf.setSize(500, 500);
jf.setVisible(true);
}
}
```

Output-


Traffic Light
Red
Yellow
Green
Stop!


Traffic Light
Red
Yellow
Green
Get Ready to go!


Traffic Light
Red
Yellow
Green
Go!!