Southern Luzon State University



College Of Engineering

Computer Engineering Department

**Activity 18: JToggleButton**

**CpE05 -**  **Object Oriented Programming**

Name: Dave Jhared G. Paduada Date: May 17, 2024

Section/Schedule: IF T & TH 10:30 - 13:30 Score:\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**SOURCE CODE:**

// Activity 18: Create a JToggleButton that changes the background color of a JPanel when clicked.

// Write a program of the given output below. Whenever you press a button, it remains active unless you pressed again.

// There are three toggle buttons named red, blue and green. The maxximum color value

// for each color is 255 and when 1, 2, or 3 (all) buttons are active, the color of the panel is changed.

import javax.swing.\*;

import java.awt.\*;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

public class GUIActivity\_18\_JToggleButton extends JFrame implements ActionListener {

// We define the JToggleButton objects as class variables so that we can access them in the actionPerformed method.

private JToggleButton red, green, blue;

private JPanel rightPanel;

public GUIActivity\_18\_JToggleButton() {

setTitle("JToggleButton");

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

setSize(400, 250);

setLocationRelativeTo(null);

setLayout(new GridLayout(1, 2));

JPanel leftPanel = new JPanel(new GridLayout(3,1,0,10));

// Creation and placing an action

red = new JToggleButton("red");

red.addActionListener(this);

green = new JToggleButton("green");

green.addActionListener(this);

blue = new JToggleButton("blue");

blue.addActionListener(this);

leftPanel.add(red);

leftPanel.add(green);

leftPanel.add(blue);

leftPanel.setBorder(BorderFactory.createEmptyBorder(40, 40, 40, 40));

// We make all three buttons of equal size.

blue.setMaximumSize(green.getMaximumSize());

red.setMaximumSize(green.getMaximumSize());

rightPanel = new JPanel();

rightPanel.setLayout(new GridBagLayout());

JPanel displayPanel = new JPanel();

displayPanel.setPreferredSize(new Dimension(160, 160));

displayPanel.setBackground(Color.BLACK);

rightPanel.add(displayPanel, new GridBagConstraints());

add(leftPanel);

add(rightPanel);

setVisible(true);

}

@Override

public void actionPerformed(ActionEvent e) {

// In the actionPerformed method, we determine the current red, green, blue parts of the display background color.

JPanel display = (JPanel) rightPanel.getComponent(0);

Color color = display.getBackground();

int red = color.getRed();

int green = color.getGreen();

int blue = color.getBlue();

// We determine which button was toggled and update the color part of the RGB value accordingly

if (e.getActionCommand()==("red")) {

if (red == 0) {

red = 255;

} else {

red = 0;

}

} else if (e.getActionCommand()==("green")) {

if (green == 0) {

green = 255;

} else {

green = 0;

}

} else if (e.getActionCommand()==("blue")) {

if (blue == 0) {

blue = 255;

} else {

blue = 0;

}

}

// Here a new color is created and the display panel is updated to a new color.

Color setCol = new Color(red, green, blue);

display.setBackground(setCol);

}

public static void main(String[] args) {

new GUIActivity\_18\_JToggleButton();

}

}

**SAMPLE OUTPUT:**

