package javaCodeGeeks;

/\*

 \* A checkbox example with event listener

 \*/

import javax.swing.JFrame;

import javax.swing.JLabel;

import javax.swing.JPanel;

import javax.swing.BorderFactory;

import javax.swing.JCheckBox;

import java.awt.GridLayout;

import java.awt.event.ItemEvent;

import java.awt.event.ItemListener;

public class checkboxWithEvent {

    // Create different checkbox

    public static JCheckBox checkbox1 = new JCheckBox("Alex");

    public static JCheckBox checkbox2 = new JCheckBox("Jessica");

    public static JCheckBox checkbox3 = new JCheckBox("Lily");

    public static JCheckBox checkbox4 = new JCheckBox("Steven");

    public static void main(String[] args) {

        // Create and set up a frame window

        JFrame.setDefaultLookAndFeelDecorated(true);

        JFrame frame = new JFrame("Checkbox with event listener");

        frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

        // Define the panel to hold the checkbox

        JPanel panel1 = new JPanel();

        JPanel panel2 = new JPanel();

        JLabel msg = new JLabel("", JLabel.CENTER);

        // Set up the title for the panel

        panel1.setBorder(BorderFactory.createTitledBorder("Name"));

        panel2.setBorder(BorderFactory.createTitledBorder("Output"));

        // Add the checkbox into the panels

        panel1.add(checkbox1);

        panel1.add(checkbox2);

        panel1.add(checkbox3);

        panel1.add(checkbox4);

        panel2.add(msg);

        // Add action listener

        checkbox1.addItemListener(new ItemListener() {

            public void itemStateChanged(ItemEvent e) {

                if (e.getStateChange() == 1) {

                    msg.setText("Alex is selected");

                } else {

                    msg.setText("Alex is unselected");

                }

            }

        });

        checkbox2.addItemListener(new ItemListener() {

            public void itemStateChanged(ItemEvent e) {

                if (e.getStateChange() == 1) {

                    msg.setText("Jessica is selected");

                } else {

                    msg.setText("Jessica is unselected");

                }

            }

        });

        checkbox3.addItemListener(new ItemListener() {

            public void itemStateChanged(ItemEvent e) {

                if (e.getStateChange() == 1) {

                    msg.setText("Lily is selected");

                } else {

                    msg.setText("Lily is unselected");

                }

            }

        });

        checkbox4.addItemListener(new ItemListener() {

            public void itemStateChanged(ItemEvent e) {

                if (e.getStateChange() == 1) {

                    msg.setText("Steven is selected");

                } else {

                    msg.setText("Steven is unselected");

                }

            }

        });

        // Add the panel into the frame

        frame.setLayout(new GridLayout(2, 1));

        frame.add(panel1);

        frame.add(panel2);

        // Set the window to be visible as the default to be false

        frame.pack();

        frame.setVisible(true);

    }

}

import java.awt.event.\*;

import java.awt.\*;

import javax.swing.\*;

class solve extends JFrame implements ItemListener {

    // frame

    static JFrame f;

    // label

    static JLabel l, l1;

    // combobox

    static JComboBox c1;

    // main class

    public static void main(String[] args)

    {

        // create a new frame

        f = new JFrame("frame");

        // create a object

        solve s = new solve();

        // set layout of frame

        f.setLayout(new FlowLayout());

        // array of string contating cities

        String s1[] = { "Jalpaiguri", "Mumbai", "Noida", "Kolkata", "New Delhi" };

        // create checkbox

        c1 = new JComboBox(s1);

        // add ItemListener

        c1.addItemListener(s);

        // create labels

        l = new JLabel("select your city ");

        l1 = new JLabel("Jalpaiguri selected");

        // set color of text

        l.setForeground(Color.red);

        l1.setForeground(Color.blue);

        // create a new panel

        JPanel p = new JPanel();

        p.add(l);

        // add combobox to panel

        p.add(c1);

        p.add(l1);

        // add panel to frame

        f.add(p);

        // set the size of frame

        f.setSize(400, 300);

        f.show();

    }

    public void itemStateChanged(ItemEvent e)

    {

        // if the state combobox is changed

        if (e.getSource() == c1) {

            l1.setText(c1.getSelectedItem() + " selected");

        }

    }

}

// Radio buttons

import java.awt.FlowLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.ButtonGroup;

import javax.swing.JButton;

import javax.swing.JFrame;

import javax.swing.JRadioButton;

public class GetSelectedJRadioButtonFromButtonGroup extends JFrame implements ActionListener {

    private static final long serialVersionUID = 1L;

    private JRadioButton java;

    private JRadioButton c;

    private JRadioButton net;

    private JButton button;

    private ButtonGroup buttonGroup;

    public GetSelectedJRadioButtonFromButtonGroup() {

        // set flow layout for the frame

        this.getContentPane().setLayout(new FlowLayout());

        java = new JRadioButton("Java");

        java.setActionCommand("Java");

        c = new JRadioButton("C/C++");

        c.setActionCommand("c");

        net = new JRadioButton(".NET");

        net.setActionCommand("net");

        java.setSelected(true);

        button = new JButton("Check");

        button.addActionListener(this);

        buttonGroup = new ButtonGroup();

        //add radio buttons

        buttonGroup.add(java);

        buttonGroup.add(c);

        buttonGroup.add(net);

        add(java);

        add(c);

        add(net);

        add(button);

    }

    @Override

    public void actionPerformed(ActionEvent e) {

        if (e.getActionCommand().equals("Check")) {

            System.out.println("Selected Radio Button: " + buttonGroup.getSelection().getActionCommand());

        }

    }

    private static void createAndShowGUI() {

  //Create and set up the window.

  JFrame frame = new GetSelectedJRadioButtonFromButtonGroup();

  //Display the window.

  frame.pack();

  frame.setVisible(true);

  frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

    }

    public static void main(String[] args) {

  //Schedule a job for the event-dispatching thread:

  //creating and showing this application's GUI.

  javax.swing.SwingUtilities.invokeLater(new Runnable() {

public void run() {

    createAndShowGUI();

}

  });

    }

}

import java.awt.Component;

import java.awt.GridLayout;

import java.awt.event.ActionEvent;

import java.awt.event.ActionListener;

import javax.swing.ButtonGroup;

import javax.swing.JButton;

import javax.swing.JCheckBox;

import javax.swing.JFrame;

import javax.swing.JPanel;

import javax.swing.JRadioButton;

public class Main {

public static void main(String[] args) {

JFrame frame = new JFrame();

JPanel entreePanel = new JPanel();

final ButtonGroup entreeGroup = new ButtonGroup();

JRadioButton radioButton;

entreePanel.add(radioButton = new JRadioButton("A"));

radioButton.setActionCommand("A");

entreeGroup.add(radioButton);

entreePanel.add(radioButton = new JRadioButton("B"));

radioButton.setActionCommand("B");

entreeGroup.add(radioButton);

entreePanel.add(radioButton = new JRadioButton("C", true));

radioButton.setActionCommand("C");

entreeGroup.add(radioButton);

final JPanel condimentsPanel = new JPanel();

condimentsPanel.add(new JCheckBox("Ketchup"));

condimentsPanel.add(new JCheckBox("Mustard"));

condimentsPanel.add(new JCheckBox("Pickles"));

JPanel orderPanel = new JPanel();

JButton orderButton = new JButton("Place Order");

orderPanel.add(orderButton);

frame.setLayout(new GridLayout(3, 1));

frame.add(entreePanel);

frame.add(condimentsPanel);

frame.add(orderPanel);

orderButton.addActionListener(new ActionListener() {

public void actionPerformed(ActionEvent ae) {

String entree = entreeGroup.getSelection().getActionCommand();

System.out.println(entree + " sandwich");

Component[] components = condimentsPanel.getComponents();

for (Component c : components) {

JCheckBox cb = (JCheckBox) c;

if (cb.isSelected())

System.out.println("With " + cb.getText());

}

}

});

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setSize(300, 150);

frame.setVisible(true);

}

}

**import** java.awt.Component;

**import** java.awt.GridLayout;

**import** java.awt.event.ActionEvent;

**import** java.awt.event.ActionListener;

**import** javax.swing.ButtonGroup;

**import** javax.swing.JButton;

**import** javax.swing.JCheckBox;

**import** javax.swing.JFrame;

**import** javax.swing.JPanel;

**import** javax.swing.JRadioButton;

**public** **class** Main {

**public** **static** **void** main(String[] args) {

JFrame frame = **new** JFrame();

JPanel entreePanel = **new** JPanel();

**final** ButtonGroup entreeGroup = **new** ButtonGroup();

JRadioButton radioButton;

entreePanel.add(radioButton = **new** JRadioButton("A"));

radioButton.setActionCommand("A");

entreeGroup.add(radioButton);

entreePanel.add(radioButton = **new** JRadioButton("B"));

radioButton.setActionCommand("B");

entreeGroup.add(radioButton);

entreePanel.add(radioButton = **new** JRadioButton("C", true));

radioButton.setActionCommand("C");

entreeGroup.add(radioButton);

**final** JPanel condimentsPanel = **new** JPanel();

condimentsPanel.add(**new** JCheckBox("Ketchup"));

condimentsPanel.add(**new** JCheckBox("Mustard"));

condimentsPanel.add(**new** JCheckBox("Pickles"));

JPanel orderPanel = **new** JPanel();

JButton orderButton = **new** JButton("Place Order");

orderPanel.add(orderButton);

frame.setLayout(**new** GridLayout(3, 1));

frame.add(entreePanel);

frame.add(condimentsPanel);

frame.add(orderPanel);

orderButton.addActionListener(**new** ActionListener() {

**public** **void** actionPerformed(ActionEvent ae) {

String entree = entreeGroup.getSelection().getActionCommand();

System.out.println(entree + " sandwich");

Component[] components = condimentsPanel.getComponents();

**for** (Component c : components) {

JCheckBox cb = (JCheckBox) c;

**if** (cb.isSelected())

System.out.println("With " + cb.getText());

}

}

});

frame.setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

frame.setSize(300, 150);

frame.setVisible(true);

}

}