514832-12.18-54EP AID:42438|12/03/2017

Program requirement:

* Have 3 radio buttons (red, orange, blue) and one Circle
* On clicking red button Circle colour should be changed to RED respectively for colour orange and blue

Program:

**import** java.awt.Color;

**import** java.awt.Container;

**import** java.awt.FlowLayout;

**import** java.awt.Graphics;

**import** java.awt.event.ItemEvent;

**import** java.awt.event.ItemListener;

**import** javax.swing.ButtonGroup;

**import** javax.swing.JFrame;

**import** javax.swing.JRadioButton;

//this program will demonstrate changing the circle

//by clicking on radio buttons

**public** **class** ChangeCircleColor **extends** JFrame {

//Declared required components

**private** Container contents;

**private** JRadioButton red,orange,blue;

**private** ButtonGroup buttonGroup;

//here it will be the default color of Circle

**private** Color circleFill=Color.***BLACK***;

//Constructor

**public** ChangeCircleColor(){

//initializing the declared variables and setting

//Container properties

**super**("changing circle color");

contents=getContentPane();

contents.setLayout(**new** FlowLayout());

red=**new** JRadioButton("Red");

orange=**new** JRadioButton("Orange");

blue=**new** JRadioButton("blue");

//Adding all button in button group for mutually exclusive

buttonGroup = **new** ButtonGroup();

buttonGroup.add(red);

buttonGroup.add(orange);

buttonGroup.add(blue);

//Adding all buttons to container

contents.add(red);

contents.add(orange);

contents.add(blue);

//Created listener object

ChangeColoListener listener = **new** ChangeColoListener();

//added all event generated component to listener

red.addItemListener(listener);

orange.addItemListener(listener);

blue.addItemListener(listener);

setSize(300,250);

setVisible(**true**);

//it will create Circle on container

repaint();

}

//Method for creating Circle using Graphic

**public** **void** paint(Graphics circle){

circle.setColor(circleFill);

circle.fillOval(80,100,120,110);

}

//inner class implementing itemlistener for event generation

**private** **class** ChangeColoListener **implements** ItemListener{

//checking the correct source and trigerring the

//required components as per event generation

@Override

**public** **void** itemStateChanged(ItemEvent e) {

**if**(e.getSource()==red){

circleFill=Color.***RED***;

//calling every time repaint method for

//making change in circle

repaint();

}

**if**(e.getSource()==orange){

circleFill=Color.***ORANGE***;

repaint();

}

**if**(e.getSource()==blue){

circleFill=Color.***BLUE***;

repaint();

}

}

}

//main method for running program

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

ChangeCircleColor obj = **new** ChangeCircleColor();

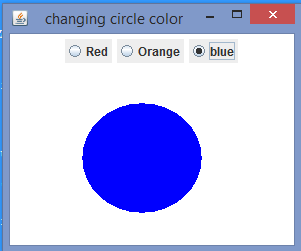
obj.setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

}

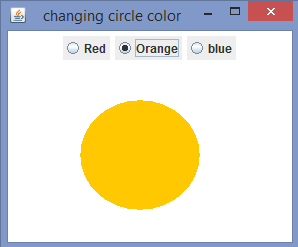
}

Output:

On selecting radio button blue



On selecting radio button orange



On selecting radio button red

