Demo ✓ 5

```
/*
 * Chris Tjon, Monster Class, Wed 4pm
 */
package monster;

import java.awt.*;

public class Monster {

    private int x;
    private int y;
    private int size; //in pixels
    private Color eyeColor;  //data type of color variable is the Color class.
    private Color bodyColor;

    public Monster(int desiredXCoord, int desiredYCoord, int desiredSize)
    {
        x = desiredXCoord;
        y = desiredYCoord;
        size = desiredSize;
        eyeColor = Color.red;  //I decided not to allow the user to set the colors
        bodyColor = Color.black;
    }

    public void run(int howFar)
    {
        x += howFar;
    }

    public void reset(Graphics g, int desiredXCoord, int desiredYCoord, int desiredSize)
    {
        x = desiredXCoord;
        y = desiredYCoord;
        size = desiredSize;
        eyeColor = Color.red;
        bodyColor = Color.black;
    }

    public void puffUp(Graphics g)
    {
        size = size * 2;
        display(g);
        for (int i = 0; i < 999999999; i++);
        eyeColor = Color.white;
        bodyColor = Color.white;
        display(g);
        eyeColor = Color.red;
        bodyColor = Color.black;
        size = size / 2;
    }

    public void jump(Graphics g)
    {
        x += 25;
        y += 150;
    }

    public void growl(Graphics g)
    {
        g.setColor(Color.blue);
        g.drawString("Growl", x, y);
    }

    public void becomeAngry(Graphics g)
    {
        eyeColor = Color.black;
        bodyColor = Color.red;
        display(g);
        for (int i = 0; i < 999999999; i++);
        eyeColor = Color.red;
        bodyColor = Color.black;
        display(g);
    }


    public void display(Graphics g)
    {
        // BODY
        g.setColor(bodyColor);
        int monsterWidth = size;
        int monsterHeight = size * 2 / 3;
        g.fillOval(x, y, monsterWidth, monsterHeight);
        // EYES

        g.setColor(eyeColor);
        int eyeWidth = monsterWidth / 10;
        int eyeHeight = eyeWidth * 2 / 3;
        int eye1X = x + monsterWidth / 6;
        int eye1Y = y + monsterHeight / 3;
        g.fillOval( eye1X, eye1Y, eyeWidth, eyeHeight );
        int eye2X = eye1X + monsterWidth / 2;
        int eye2Y = eye1Y;
        g.fillOval( eye2X, eye2Y, eyeWidth, eyeHeight );
    }
}
```

```
/*
 * Chris Tjon, Monster Controller, Wed 4pm
 */


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

public class MonsterController extends Applet implements ActionListener
{

   Monster fred; //declare an identifier to hold the object you will instantiate

   Button runButton;
   Button angryButton;
   Button growlButton;
   Button resetButton;
   Button jumpButton;
   Button puffUpButton;
   boolean runPressed = false;
   boolean angryPressed = false;
   boolean growlPressed = false;
   boolean jumpPressed = false;
   boolean puffUpPressed = false;
   boolean resetPressed = false;

   public void init() {
        /*  new instantiates (creates) the object from your class pattern.
            At the time your object is instantiated, java runs it's
            constructor
            method, passing it the parameters in parentheses. */
        runButton = new Button("Run");
        add(runButton);
        runButton.addActionListener(this);

        angryButton = new Button("Angry");
        add(angryButton);
        angryButton.addActionListener(this);

        growlButton = new Button("Growl");
        add(growlButton);
        growlButton.addActionListener(this);

        jumpButton = new Button("Jump");
        add(jumpButton);
        jumpButton.addActionListener(this);

        puffUpButton = new Button("Puff Up");
        add(puffUpButton);
        puffUpButton.addActionListener(this);

        resetButton = new Button("Reset");
        add(resetButton);
        resetButton.addActionListener(this);

        fred = new Monster(50, 50, 75);
        /*fred now exists and you can execute his methods,
               *making him do tricks - such as displaying and growling
        */
    }

    public void actionPerformed(ActionEvent event)
    {
        if (event.getSource() == runButton)
            runPressed = true;
        if (event.getSource() == angryButton)
            angryPressed = true;
        if (event.getSource() == growlButton)
            growlPressed = true;
        if (event.getSource() == jumpButton)
            jumpPressed = true;
        if (event.getSource() == puffUpButton)
            puffUpPressed = true;
        if (event.getSource() == resetButton)
            resetPressed = true;
        repaint();
    }

    public void paint(Graphics g) {

        if (runPressed)
        {
            fred.run(20);
            runPressed = false;
        }

        if (angryPressed)
        {
            fred.becomeAngry(g);
            angryPressed = false;
        }

        if (growlPressed)
        {
            fred.growl(g);
            growlPressed = false;
        }

        if (jumpPressed)
        {
            fred.jump(g);
            jumpPressed = false;
        }

        if (puffUpPressed)
        {
            fred.puffUp(g);
            puffUpPressed = false;
        }

        if (resetPressed)
        {
            fred.reset(g, 50, 50, 75);
            resetPressed = false;
        }
        fred.display(g); // displays fred
    }
}
```