```
/*Surya Dantuluri
 * April 17th 2017
 *
 * Digestive System Game
 *
 * Welcome screen starts and button is under title
 *
 * Human, health bar and foods are present. A drawing on the human pops up
     trying to signify that s(he) should  feed the human food.
 *
 * Should have 6 different classes signifying each of the stages: Mouth,
     Esophogous, Stomach, Duodenum, Small intestine, *Large intestine
 *
 *-------
 * Game starts off with welcome screen.
 * Human points to the food options on the right
 * Health bar is shown.
 * Virtual human player urges normal human player to eat food by showing food
     on the right.
 *
 */

 import java.awt.Graphics;//add graphics
 import java.awt.Graphics2D;
 import java.awt.Image;
 import java.awt.Toolkit;//add toolkit to move images around
 import java.awt.event.MouseEvent;
 import java.awt.event.MouseMotionListener;
import javax.swing.JFrame;  //import javax.swing.*;
import javax.swing.JPanel;//add jpanel

//import java.awt.event.MouseEvent;

import java.awt.Color;      //import java.awt.*;
import java.awt.Graphics;
import java.awt.Font;
import java.awt.Dimension;
////////////////////////// import Classes needed for
    Layouts ///////////////////////
import java.awt.BorderLayout;
import java.awt.GridLayout;
import java.awt.CardLayout;
import java.awt.FlowLayout;
import javax.swing.JButton;
import javax.swing.JRadioButton;

import java.awt.event.ActionListener;
import java.awt.event.ActionEvent;
import javax.swing.JCheckBox;

public class EatHealthy
{
    JFrame frame;
    public EatHealthy()
    {
        landingPage landPage;
```

```
    healthPanel panBar;
    // Create a JFrame with BorderLayout
    frame = new JFrame("FlowLayout");   // Create the JFrame

    // notice this has DISPOSE.  What is the effect of that?
    frame.setDefaultCloseOperation(flowFrame.DISPOSE_ON_CLOSE);
    frame.setSize(800, 800);//set size of frame
    frame.setLocation(0, 0);//set location of frmae

    // Initialize panels
    //send stuff to  panBar = new RadioProbability(1, Color.RED);
    //send stuff to panFL2 = new ScrollRisk(2, Color.BLUE);

    ////////////////////////////////////////////////
    // setPreferredSize for all panels - 1st we will run the prog with out
        this.
    //
    landPage.setPreferredSize(new Dimension(800,800));
    panBar.setPreferredSize(new Dimension(220,70));


    frame.getContentPane().add(landPage);
    frame.getContentPane().add(panBar);


    // Make the JFrame visible
    frame.setVisible(true);
}

public static void main(String[] args)
{
    System.out.println("I will remember to log out.");
    EatHealthy eaty = new EatHealthy();
    //gig.run();
}

//public void run()
//{
//makeFlowLayout();     // FlowLayout window
//}


//public void makeFlowLayout()
//{

//}

class Mouth extends JPanel implements ActionListener
{
  //variables
    public Mouth(int numIn, Color colorIn)      //numbers on the panel
    {
        JRadioButton radiobutton = new JRadioButton("1");
        radiobutton.addActionListener(this);//set radio button of mouth
            panel to add quiz questions
        add(radiobutton);
```

```
            JRadioButton radiobutton2 = new JRadioButton("2");//quiz questions
            radiobutton2.addActionListener(this);
            add(radiobutton2);
            JRadioButton radiobutton3 = new JRadioButton("3");//quiz questions
            radiobutton3.addActionListener(this);
            add(radiobutton3);
            JRadioButton radiobutton4 = new JRadioButton("4");//quiz quistions
            radiobutton4.addActionListener(this);
            add(radiobutton4);

        }
        public void paintComponent(Graphics g)//paintComponent of Mouth Class
        {
            //super.paintComponent(g);
            //g.setFont(new Font("Arial", Font.BOLD, 18));
            //g.setColor(Color.RED);
            //g.drawString(num, 10, 20);
        }

        public void actionPerformed(ActionEvent evt)
        {
            String command=button.getText();
            if(command.equals("Press my belly."))
            {
                pressed=true;
                button.setText("reset");
            }

            else
            {
                button.setText("Press my belly.");
            }
            repaint();

        }
    }   // end class Mouth


}

class Stomach extends JComponent implements MouseMotionListener
{

// drag jcomponent to move image
  static int imageWidth = 60, imageHeight = 60;
  int imageX, imageY;

  Image image;

  public Esophogous(Image i) {
    image = i;//include image
    addMouseMotionListener(this);
  }

  public void mouseDragged(MouseEvent e) {
    imageX = e.getX();
```

```
      imageY = e.getY();
      repaint();
    }

    public void mouseMoved(MouseEvent e) {
    }

    public void paint(Graphics g) {
      Graphics2D g2 = (Graphics2D) g;

      g2.drawImage(image, imageX, imageY, this);
    }

    public static void main(String[] args) {
      String imageFile = "A.jpg";
      // Turn off double buffering
      RepaintManager.currentManager(null).setDoubleBufferingEnabled(false);

      Image image = Toolkit.getDefaultToolkit().getImage(DragImage.class.
          getResource(imageFile));//include toolkit to drag images around
      image = image.getScaledInstance(imageWidth, imageHeight, Image.
          SCALE_DEFAULT);//include width and height of the image
      JFrame frame = new JFrame("DragImage");//make new fram but CHANGE TO PANEL
      frame.getContentPane().add(new DragImage(image));
      frame.setSize(300, 300);
      frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
      frame.setVisible(true);
    }
  }

  class SmallIntestine extends JComponent implements MouseMotionListener
  {

  // drag jcomponent to move image
    static int imageWidth = 60, imageHeight = 60;
    int imageX, imageY;

    Image image;

    public SmallIntestine(Image i) {
      image = i;
      addMouseMotionListener(this);
    }

    public void mouseDragged(MouseEvent e) {
      imageX = e.getX();
      imageY = e.getY();
      repaint();
    }

    public void mouseMoved(MouseEvent e) {
    }

    public void paint(Graphics g) {
      Graphics2D g2 = (Graphics2D) g;
```

```
    g2.drawImage(image, imageX, imageY, this);
  }

  public static void main(String[] args) {
    String imageFile = "A.jpg";
    // Turn off double buffering
    RepaintManager.currentManager(null).setDoubleBufferingEnabled(false);

    Image image = Toolkit.getDefaultToolkit().getImage(DragImage.class.
        getResource(imageFile));//include toolkit to drag images around
    image = image.getScaledInstance(imageWidth, imageHeight, Image.
        SCALE_DEFAULT);//include width and height of the image
    JFrame frame = new JFrame("DragImage");//make new fram but CHANGE TO PANEL
    frame.getContentPane().add(new DragImage(image));
    frame.setSize(300, 300);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    frame.setVisible(true);
  }
}

class LargeIntestine extends JComponent implements MouseMotionListener
{

// drag jcomponent to move image
  static int imageWidth = 60, imageHeight = 60;
  int imageX, imageY;

  Image image;

  public LargeIntestine(Image i) {
    image = i;
    addMouseMotionListener(this);
  }

  public void mouseDragged(MouseEvent e) {
    imageX = e.getX();
    imageY = e.getY();
    repaint();
  }

  public void mouseMoved(MouseEvent e) {
  }

  public void paint(Graphics g) {
    Graphics2D g2 = (Graphics2D) g;

    g2.drawImage(image, imageX, imageY, this);
  }

  public static void main(String[] args) {
    String imageFile = "A.jpg";
    // Turn off double buffering
    RepaintManager.currentManager(null).setDoubleBufferingEnabled(false);

    Image image = Toolkit.getDefaultToolkit().getImage(DragImage.class.
        getResource(imageFile));//include toolkit to drag images around
```

```
    image = image.getScaledInstance(imageWidth, imageHeight, Image.
        SCALE_DEFAULT);//include width and height of the image
    JFrame frame = new JFrame("DragImage");//make new fram but CHANGE TO PANEL
    frame.getContentPane().add(new DragImage(image));
    frame.setSize(300, 300);
    frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
    frame.setVisible(true);
  }
}
```