Exercise 2.7

1. White: new Color(255, 255, 255) Black: new Color(0,0,0)
2. Describe the roles and responsibilities of a frame, a panel, and a layout manager in a GUI Application
   1. Panel – is a container that can be filled with other objects. A panel is a rectangular region suitable for displaying yet other objects, such as a geometric shapes and images.
   2. Layout Manager –When a program adds an object to a container, the container’s layout manager controls the placement of the panels.
   3. Frame – A Java class that defines the windows for an application.
3. Where are panels displayed when a border layout is used to control their placement in a window?
   1. A border layout allows us to arrange up to five objects in positions that correspond to the directions N, E, S, W and center. If we add fewer than five objects, the layout manager stretches some of them to fill the unoccupied area.
4. Write a code segment that would be used to set the layout for adding panels to a 5-by-5 grid in a window you may assume that the panel’s content pane named pane
   1. Pane northPanel = new jPanel();

Critical Thinking p. 30

Copyrights protect the intellectual property of someone’s work and at stake is are the right to receive payment for work. Most recently copyrights have been open to include software and other forms of digital media. If a piece of software is copyrighted and is stolen or illegally downloaded, the perpetrator can be punished by law.

What is Java API and Javadoc?

For java the documentation, which contains the syntax, the semantics and how to use the language, can be found in the Java API. You can learn more about the Java API on the Oracle website, and also down a particular version of Java. Java is very big. It has dozens of classes, packages, a lots of methods.

Exercise 3.7

1. Write the code segment that would draw the following objects in the graphics context g:
   1. g.drawRect (45,20,10,100); g.fillRect(Color.red);
   2. g.drawLine(20,20,100,100);
   3. g.drawOval (100,100,50,50);
   4. g.drawLine (100,100,50,50); g.drawLine( 50,50, 200,200); g.drawLine(200,200, 100,100);
2. g.drawRect (45,20,10,100); g.fillRect(Color.blue); g.setBackground(Color.black);
3. The program uses x and y coordinates to express position relative to the system’s origin at (0,0) or center.
4. Font style, font size, font name.

Critical Thinking p.55

This is a good suggestion because the program as to cater to the customer’s needs. If a programmer doesn’t talk to the customer, the programmer doesn’t know what the customer would like the program to look like. Some of the questions I would ask are: Is this product for staff or the customer? What’s are some of the menus you would like to see? What do you want the color scheme to be? How much are you willing pay?

What is the Math Class in Java?

The math class is used for doing basic operations, such as logarithm, square root, and trig functions. The math class is more relaxed because it is not defined to return the bit-for-bit same results. This means over all better performants.

Projects:

2.7

**import** javax.swing.\*; // For JFrame and JPanel

**import** java.awt.\*; // For Color, Container, and GridLayout

**public** **class** GUIWindow26{

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

JFrame theGUI = **new** JFrame();

theGUI.setTitle("Fourth GUI Program");

theGUI.setSize(300, 200);

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

JPanel panel1 = **new** JPanel();

panel1.setBackground(Color.***white***);

JPanel panel2 = **new** JPanel();

panel2.setBackground(Color.***black***);

JPanel panel3 = **new** JPanel();

panel3.setBackground(Color.***black***);

JPanel panel4 = **new** JPanel();

panel4.setBackground(Color.***white***);

JPanel panel15 = **new** JPanel();

panel15.setBackground(Color.***white***);

JPanel panel16 = **new** JPanel();

panel16.setBackground(Color.***black***);

Container pane = theGUI.getContentPane();

pane.setLayout(**new** GridLayout(3, 3));

pane.add(panel1);

pane.add(panel2);

pane.add(panel3);

pane.add(panel4);

pane.add(panel15);

pane.add(panel16);

theGUI.setVisible(**true**);

}

}

3.6

**import** javax.swing.\*;

**import** java.awt.\*;

**public** **class** Illusion **extends** JPanel{

**public** Illusion(Color color){

setBackground(color);

}

**public** **void** paintComponent (Graphics g){

**super**.paintComponent(g);

// the lines on the x-axis

g.drawLine(20, 50,150, 50 );

g.drawLine(20, 20,150, 20);

// the lines on the y-axis

g.drawLine(150, 50,170, 65 );

g.drawLine(150, 50,170, 40 );

g.drawLine(10,60,20,50 );

g.drawLine(10,40, 20,50 );

g.drawLine(20,20,30,30 );

g.drawLine(30,10, 20,20 );

g.drawLine(140,10,150,20);

g.drawLine(150,20,140,30 );

}

}

3.7

**import** javax.swing.\*;

**import** java.awt.\*;

**public** **class** Center37 **extends** JPanel {

**public** Center37(Color color){

setBackground(color);

}

**public** **void** paintComponent (Graphics g){

**super**.paintComponent(g);

**int** x = getWidth() / 2 -40;

**int** y = getHeight() / 2 ;

String centerofpage = "(" + x + " ," + y + ")";

// g.fillArc(x,y,50,50,0,360);

Font font = **new** Font("Courier", Font.***BOLD***, 14);

g.setFont(font);

g.drawString(centerofpage,x, y);

}

}