Java Graphics

Friday, November 30, 2007

CS111 Computer Programming

Department of Computer Science
Wellesley College

Java Programs

Java programs can be **applets** or **applications**.

Most of the programs that we've developed this semester have been run as **applications** (e.g., by pressing **F2** or "**Run**" in Dr. Java to execute a class's `main` method).

An **applet**, an instance of the Applet of JApplet class, is a program that can run inside a window (like a browser window).

When an **applet** is created, a `main` method is *not* invoked.

The browser expects an applet to respond to the following message:

```
paint()
```

Check.java

Check.class


Check.html


Check.java

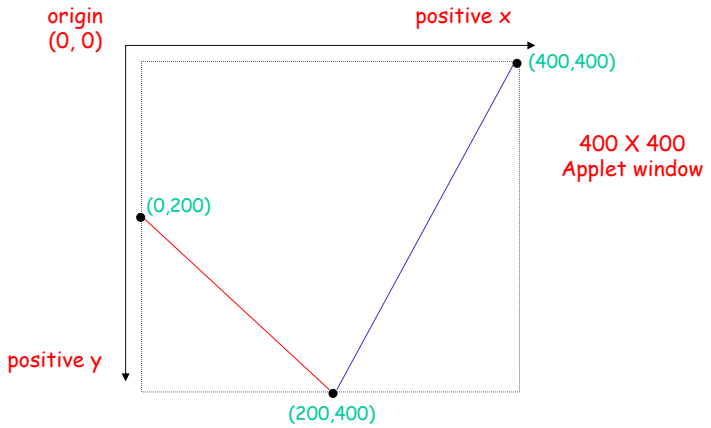
```
import java.awt.*;
import javax.swing.*;  // for JApplet
public class Check extends JApplet {
  public void paint (Graphics g) {
    g.setColor(Color.red);
    g.drawLine(0, 200, 200, 400);
    g.setColor(Color.blue);
    g.drawLine(200, 400, 400, 0);
  }
}
```

Graphics object provided by Applet to which we send all drawing commands

Imagine 'g' is a painting machine. You're telling it what to paint and where. It does all the work.

Check.html

```
<html>
  <body>
    <applet code="Check.class" width=400 height=400></applet>
  </body>
</html>
```



Graphics contract

public void setColor(Color c)
Sets this graphics context's current color to the specified color.

public void drawLine(int x1, int y1, int x2, int y2)
Draws a line, using the current color, between the points (x1, y1) and (x2, y2) in this graphics context's coordinate system.

public void drawRect(int x, int y, int width, int height)
Draws the outline of the specified rectangle.

public void fillRect(int x, int y, int width, int height)
Fills the specified rectangle.

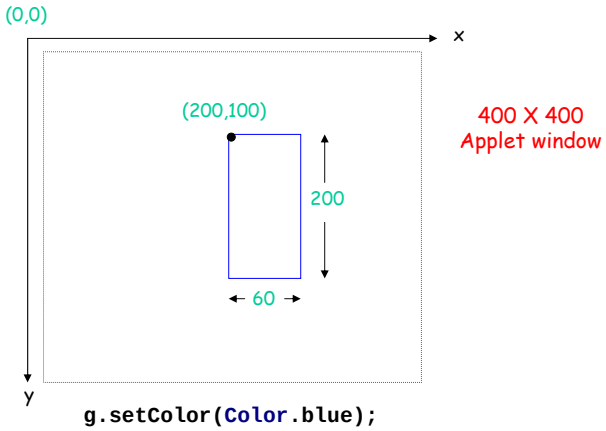
public void drawOval(int x, int y, int width, int height)
Draws the outline of an oval.

public void fillOval(int x, int y, int width, int height)
Fills the specified oval.

public void drawString(String str, int x, int y)
Draws the text given by the specified string, using this graphics context's current font and color.

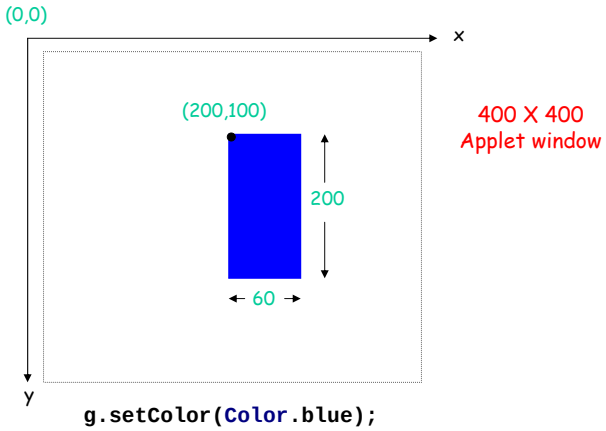
...

drawRect(int x, int y, int width, int height)



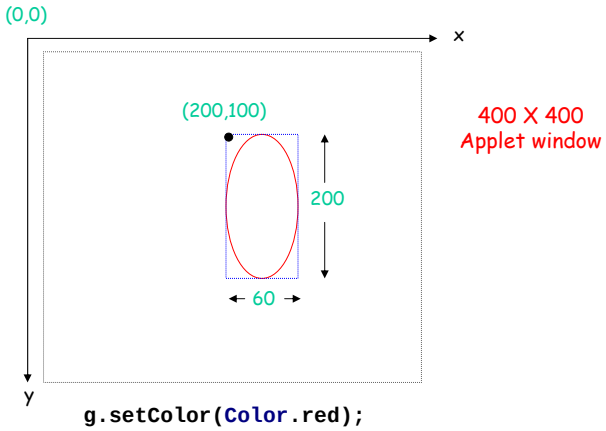

```
g.setColor(Color.blue);
g.drawRect(200,100,60,200);
```

`fillRect(int x, int y, int width, int height)`



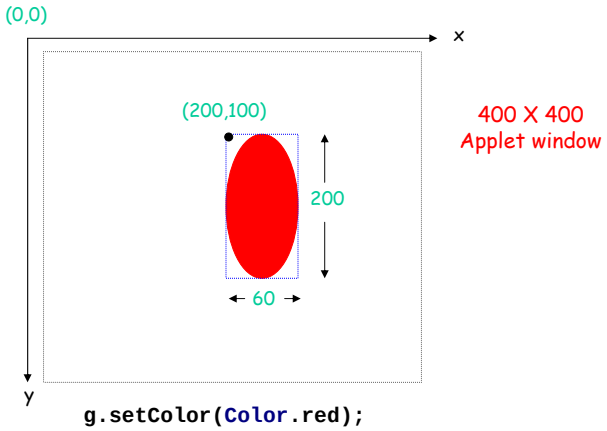

```
g.setColor(Color.blue);
g.fillRect(200,100,60,200);
```


`drawOval(int x, int y, int width, int height)`



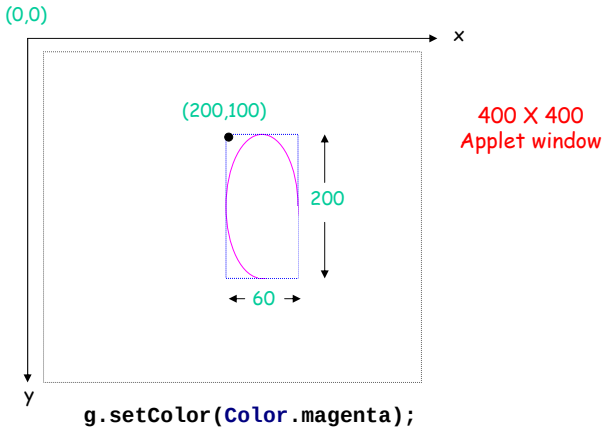

```
g.setColor(Color.red);
g.drawOval(200,100,60,200);
```

fillOval(int x, int y, int width, int height)



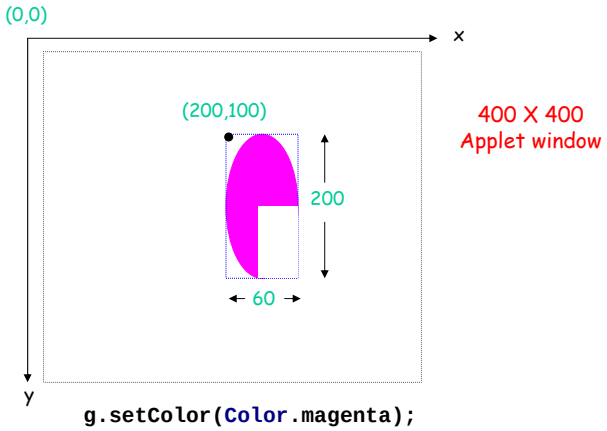

```
g.setColor(Color.red);
g.fillOval(200,100,60,200);
```


drawArc(int x, int y, int width, int height,
int startAngle, int arcAngle)



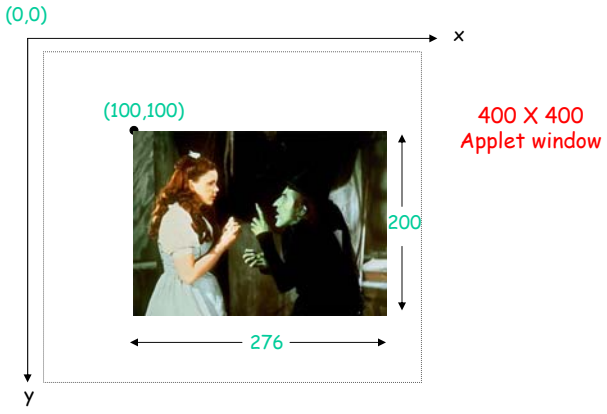

```
g.setColor(Color.magenta);
g.drawArc(200,100,60,200,0,270);
```

fillArc(int x, int y, int width, int height, int startAngle, int arcAngle)




```
g.setColor(Color.magenta);
g.fillArc(200,100,60,200,0,270);
```


drawImage(Image img, int x, int y, int width, int height, ImageObserver observer)



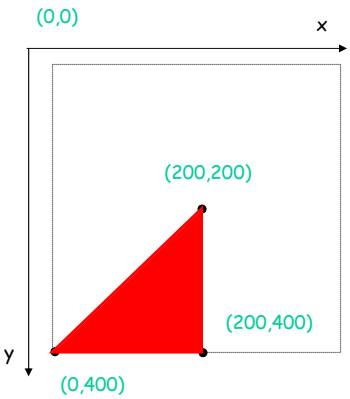

```
Image dorothy = getImage(getCodeBase(),"wizardOz.jpg");
g.drawImage(dorothy, 100, 100, 276, 200, this);
```

drawString(String str, int x, int y)


```
g.setColor(Color.darkGray);
g.drawString("There's no place like home!",100,100);
```


The Polygon class

- A member of the Polygon class has multiple vertices.
- We create a new Polygon with, you guessed it, a constructor, then add vertices.
- Polygons don't appear until asked.

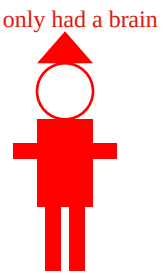


Creates an empty Poly →

and Poly gets the point →

```
Polygon anna = new Polygon();
anna.addPoint(0,400);
anna.addPoint(200,400);
anna.addPoint(200,200);
g.drawPolygon(anna);
g.fillPolygon(anna);
```

The Tin Man

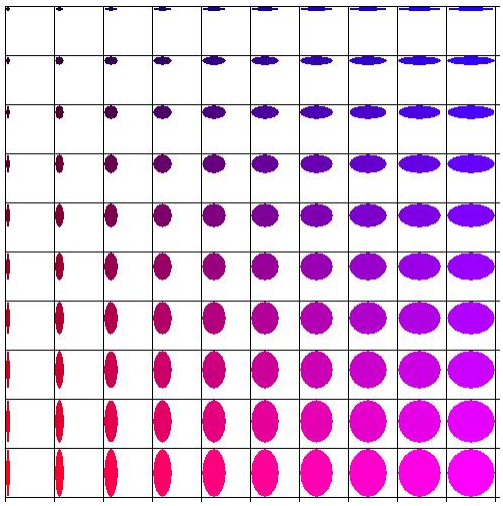



```
g.setColor(Color.red);
g.drawOval(100,100,50,50);  // Head
Polygon hat = new Polygon();
hat.addPoint(100,100);
hat.addPoint(125,70);
hat.addPoint(150,100);
g.fillPolygon(hat);
g.drawString("If I only had a brain", 80, 60);
g.fillRect(100,150,50,80);  // Body
g.fillRect(105,230,15,50);  // Leg
g.fillRect(130,230,15,50);  // Leg
g.fillRect(75,175,100,20);  // Arms
```


We don't need no stinkin' browser!

```
import java.awt.*;
import javax.swing.*;

public class TinManApplication {

  public static void main (String[] args) {
    // Enable window decorations
    JFrame.setDefaultLookAndFeelDecorated(true);

    // Create and set up the window
    JFrame frame = new JFrame("Portrait of a TinMan");
    frame.setSize(280, 400);

    // Add applet to frame
    frame.add(new TinMan(), BorderLayout.CENTER);

    // Display the window
    frame.setVisible(true);
  }

}
```

Ovals



Arachnophobia