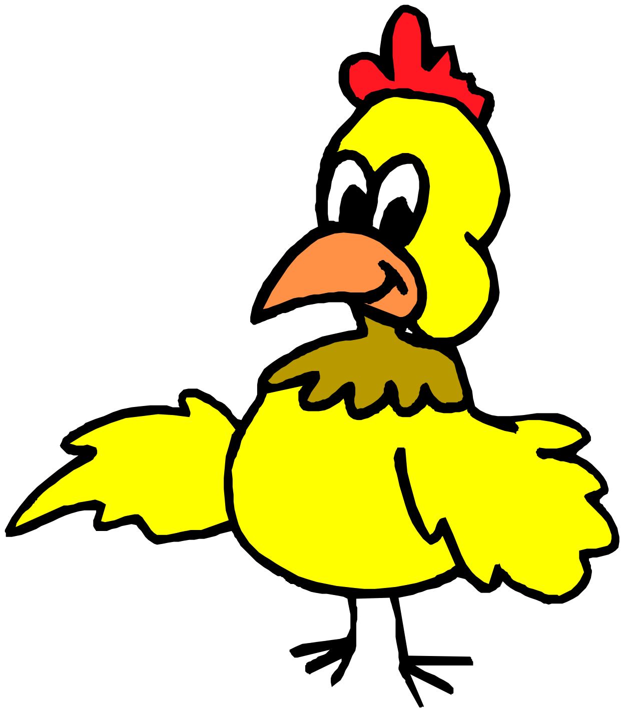
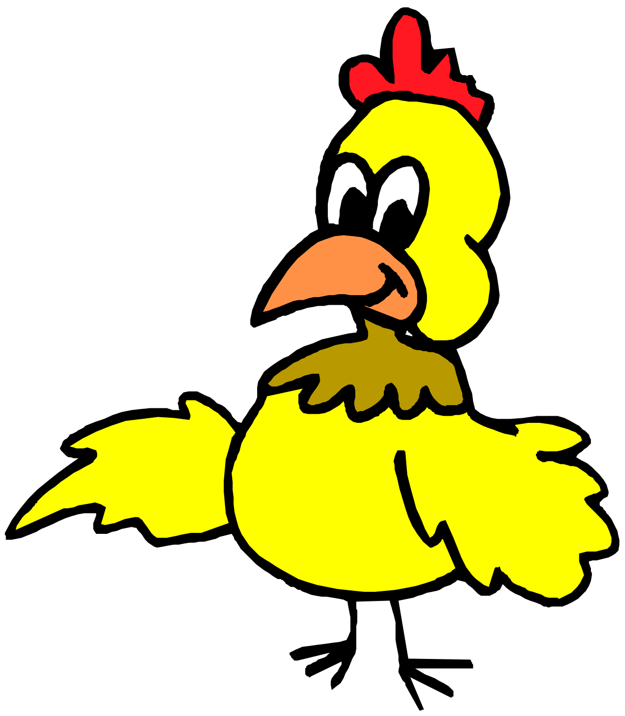
OOP Methods Parameters

Objects



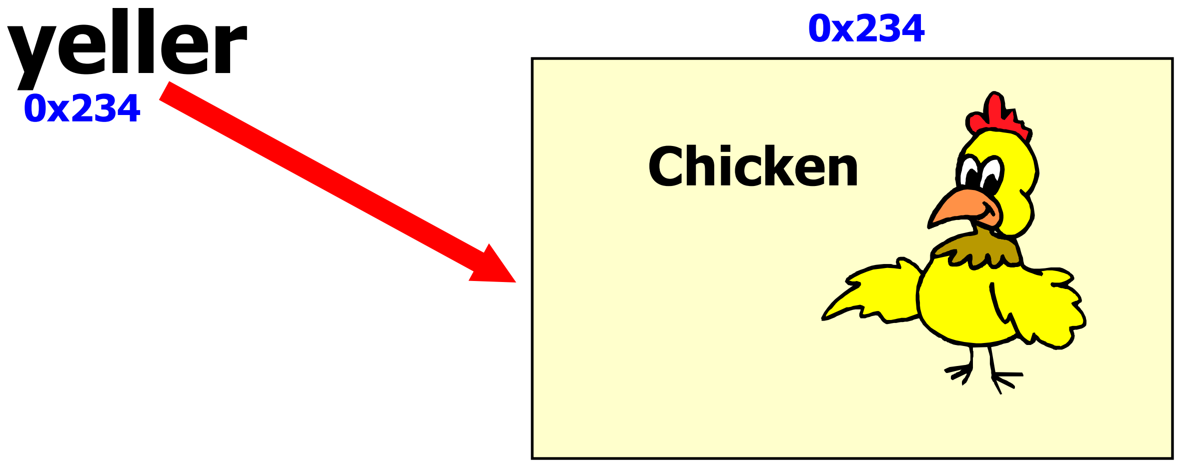
Object Instantiation

```
Chicken yeller = new Chicken();
```



Object Instantiation

```
Chicken yeller = new Chicken();
```



yeller is a reference variable that refers to a Chicken object.

Methods

What is a method?

A method is a storage location for related program statements. When called, a method usually performs a specific task.

```
System.out.println( )
```

What methods have we used?

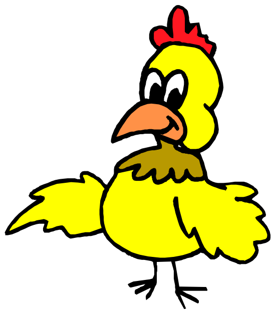
dude.goHome()

keyboard.nextInt( )

System.out.println( )

methods

```
public void speak()
{
  out.println("cluck-cluck");
}
```



OUTPUT

```
cluck-cluck
```

methods

access	return type	name	params
code			

```
public            void        speak(    )
{
  System.out.println("cluck-cluck");
}
```

What does public mean?

All members with public access can be accessed or modified inside and outside of the class where they are defined.

chicken

```
public class Chicken
{
   public void speak()
   {
      out.println("cluck-cluck");
   }

   public static void main(String[] args)
   {
      Chicken red = new Chicken();
      red.speak();
      red.speak();
      red.speak();
   }
}
```

OUTPUT

```
cluck-cluck
cluck-cluck
cluck-cluck
```

Open chicken.java

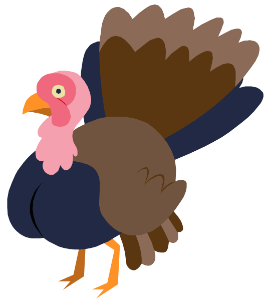
turkey

```
public class Turkey
{
   public void speak()
   {
      out.println("gobble-gobble");
   }

   public void sayName()
   {
      out.println("big bird");
   }
}
```

OUTPUT

```
gobble-gobble
big bird
gobble-gobble
big bird
gobble-gobble
```

```
//code in the main of another class
Turkey bird = new Turkey();
bird.speak();
bird.sayName();
bird.speak();
bird.sayName();
bird.speak();
```

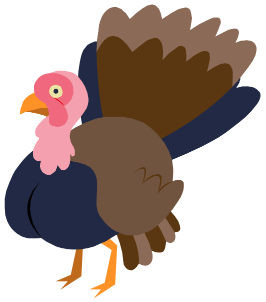


turkey

```
public class Turkey
{
   public void speak()
   {
      out.println("gobble-gobble");
   }

   public void sayName()
   {
      out.println("big bird");
      speak();
   }
}

//code in the main of another class
Turkey bird = new Turkey();
bird.speak();
bird.sayName();
bird.speak();
bird.sayName();
bird.speak();
```

OUTPUT

```
gobble-gobble
big bird
gobble-gobble
gobble-gobble
big bird
gobble-gobble
gobble-gobble
```



Open
turkey.java
turkeyrunner.java

Start work on the labs

Constructors and Graphics methods

Constructors

Constructors always have the same name as the class.

```
GraphOne test = new GraphOne();

Monster rob = new Monster();
```

Constructors

reference variable

```
Scanner keyboard =
                   new Scanner(System.in);
```

object instantiation / constructor call

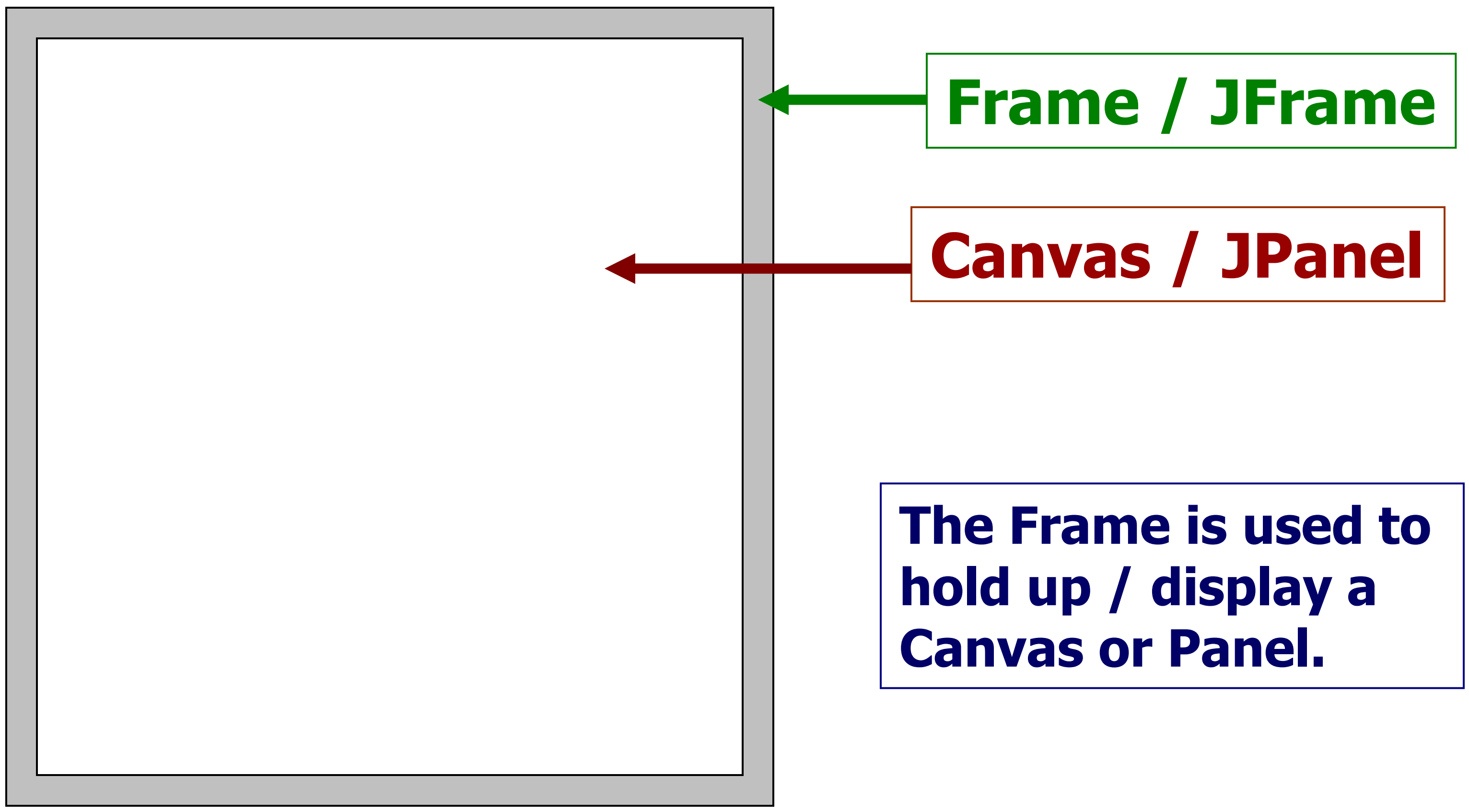
Constructors

```
public class GraphicsRunner extends JFrame
{
  private static final int WIDTH = 640;
  private static final int HEIGHT = 480;

  public GraphicsRunner()
  {
    setSize(WIDTH,HEIGHT);
    getContentPane().add(  new Circles()  );
    setVisible(true);
  }

  public static void main( String args[] )
  {
    GraphicsRunner run = new GraphicsRunner();
  }
}
```

the constructor

constructor call

Frame



The Frame is used to hold up / display a Canvas or Panel.

paint()

```
public class Circles extends Canvas
{

   //constructors

   public void paint( Graphics window )
   {
      window.setColor(Color.BLACK);
      window.drawString("Circles", 50, 50);

      window.setColor(Color.BLUE);
      window.drawOval(500,300,40,40);
   }

   //other methods

}
```

paint

paint() is called automatically when you instantiate the class containing the paint method.

When an event is triggered that requires a redraw, paint is called again.

To call paint() without a Graphics parameter, you can use the repaint() method.

Open

graphicsrunner.java

circles.java

Parameters and Graphics methods

Graphics
frequently used methods

Name	Use
setColor(x)	sets the current drawing color to x
drawString(s,x,y)	draws String s at spot x,y
drawOval(x,y,w,h)	draws an unfilled oval at spot x,y that is w wide and h tall
fillOval(x,y,w,h)	draws a filled oval at spot x,y that is w wide and h tall

```
import java.awt.Graphics;
import java.awt.Color;
import javax.swing.JFrame;
```

passing parameters

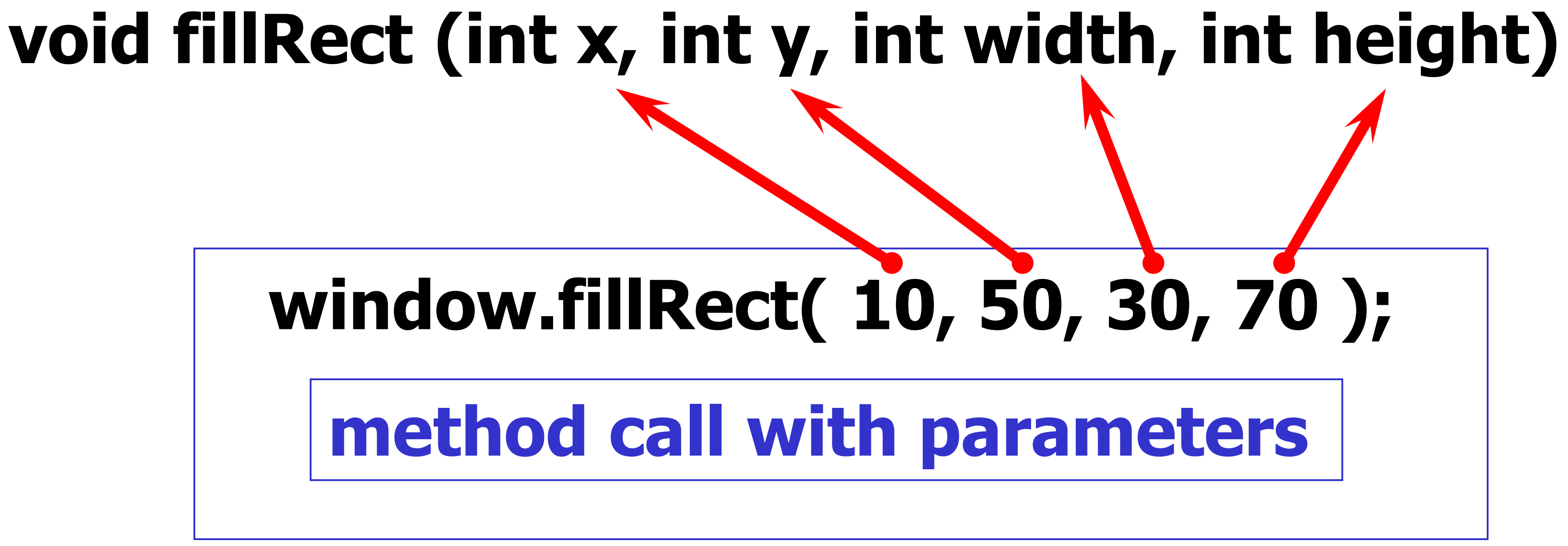
A parameter/argument is a channel used to pass information to a method. setColor() is a method of the Graphics class the receives a Color.

```
void setColor(Color theColor)
```

```
window.setColor( Color.RED );
```

method call with parameter

passing parameters



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

window.fillRect( 10, 50, 30, 70 );
```

method call with parameters

passing parameters

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

     window.fillRect( 10, 50, 30, 70 );
```

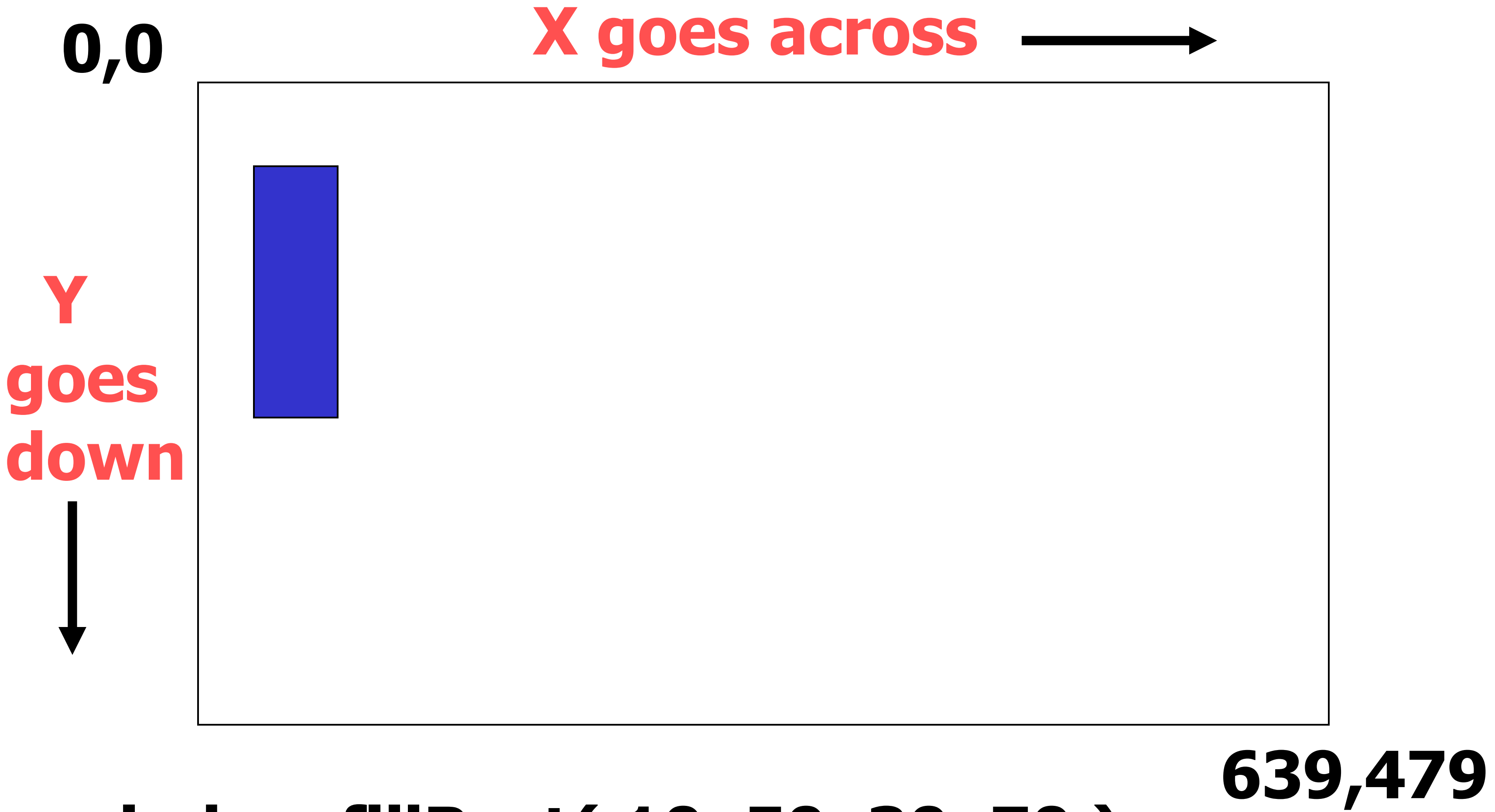
The call to fillRect would draw a rectangle at position 10,50 with a width of 30 and a height of 70.

Graphics
frequently used methods

Name	Use
drawLine(a,b,c,d)	draws a line starting at point a,b and going to point c,d
drawRect(x,y,w,h)	draws an unfilled rectangle at spot x,y that is w wide and h tall
fillRect(x,y,w,h)	draws a filled rectangle at spot x,y that is w wide and h tall

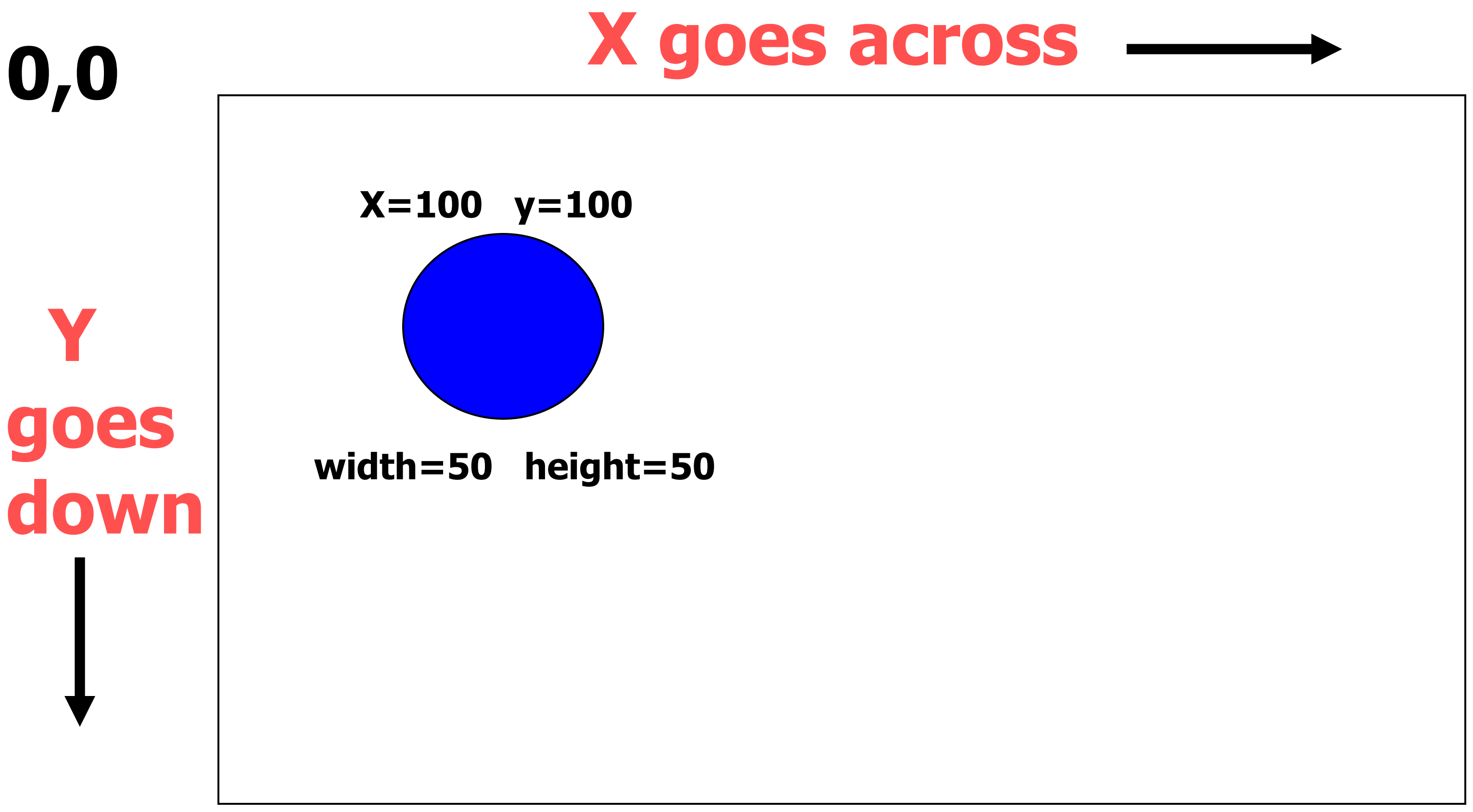
```
import java.awt.Graphics;
import java.awt.Color;
import javax.swing.JFrame;
```

The Graphics Screen



```
window.fillRect( 10, 50, 30, 70 );
```

The Graphics Screen



```
window.fillOval( 100, 100, 50, 50 );
```

Rectangles

```
public void paint( Graphics window )
{
   window.setColor(Color.BLUE);
   window.fillRect(150, 300, 100, 20);
   window.setColor(Color.GRAY);
   window.drawRect(200,80,50,50);
}
```

Open rectangles.java

Open lines.java

Graphics
frequently used methods

Name	Use
drawArc(x,y,w,h,startAngle,arcAngle)	draws an arc at spot x,y that is w wide and h tall
fillArc(x,y,w,h,startAngle,arcAngle)	draws a filled arc at spot x,y that is w wide and h tall
startAngle specifies the start of the arc arcAngle specifies the length of the arc	

```
import java.awt.Graphics;
import java.awt.Color;
import javax.swing.JFrame;
```

Open arcs.java

Open fonts.java

Open colors.java

Continue work on the labs