

Computer Programming Fundamentals

CS 152

Professor: Leah Buechley

TAs: Melody Horn, Noah Garcia, Andrew Geyko, Juan Ormaza

Time: MWF 10:00-10:50am

https://handandmachine.cs.unm.edu/classes/CS152_Fall2021/

**USE PIAZZA FOR QUESTIONS
DURING LECTURE**

DUE TUESDAY: ASSIGNMENT 2

- Due Tuesday 9/6 by 9:30am
- Use the Screen.java code from class
- Submit via UNM Learn

**LETS DO SOME MORE
INTERESTING PROGRAMMING**

IntelliJ or Replit

IntelliJ PROJECT SETUP

1. Create a new project
2. Make sure you have the right SDK selected (Java 16)
3. Click “Next”
4. Click “Next”

IntelliJ PROJECT SETUP

1. Give your project a name and make sure it's being saved in the correct location. The folder where you're storing all your code for this class
2. For PCs:
C:\Users\YOUR_ACCOUNT\CS152Java
3. Type in "ScreenExample" for Project name
4. Make sure "ScreenExample" is at the end of the text you see in Project location. Add this text if it isn't there.

CREATE A NEW FILE

- Create a new Java Class file in the src directory.
- Name it “Screen.java”
- Copy the code from this location into your file:
https://handandmachine.cs.unm.edu/classes/CS152_Fall2021/sampleCode/Screen.java
- Follow link from schedule on website

Screen.java

```
/*  
*****  
* Author: Leah Buechley  
* For course: CS 152 - Computer Programming Fundamentals  
* Date: 8/2021  
*  
* A simple class to make it easy to get started with graphics programming in Java.  
* Inspired by Processing. Trying to support a similar experience.  
* Refer to Java graphics documentation for information on drawing:  
* https://docs.oracle.com/en/java/javase/16/docs/api/java.desktop/java/awt/Graphics.html  
*****  
*/  
  
import javax.swing.*;  
import java.awt.*;  
import java.awt.event.*;  
import java.awt.image.BufferStrategy;  
  
class Screen extends Canvas implements KeyListener, MouseMotionListener, MouseListener {  
    int width, height;  
    int mouseX, mouseY;  
    Color backgroundColor;  
    String name;  
    BufferStrategy bufferStrategy;  
    char keyPressed;  
    int keyCode;  
    boolean mouseClicked;  
    boolean mousePressed;  
    int numberOfClicks;  
}
```

CREATE ANOTHER FILE

- Create a new Java Class file in the src directory.
- Name it “ScreenExample.java”
- Copy the code from this location into your file:

https://handandmachine.cs.unm.edu/classes/CS152_Fall2021/sampleCode/ScreenExample.java

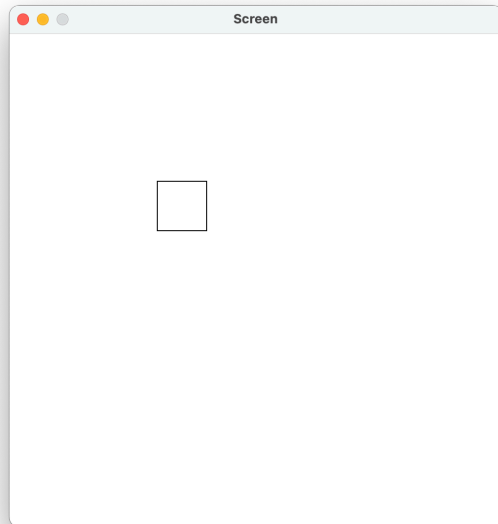
- Follow link from schedule on website

ScreenExample.java

```
/*  
*****  
* Author: Leah Buechley  
* Date: 8/2021  
* This is an example to help you use the Screen class  
* Refer to Java graphics documentation for information on drawing:  
* https://docs.oracle.com/en/java/javase/16/docs/api/java.desktop/java/awt/Graphics.html  
*****  
*/  
  
import java.awt.*;  
  
public class ScreenExample {  
    //Create a screen/window to draw in  
    static Screen screen= new Screen();  
  
    //Main just paints the screen over and over forever  
    public static void main(String[] args) {  
        while (true) {  
            paint();  
        }  
    }  
  
    //The paint() method is where all the interesting stuff happens  
    public static void paint() {  
        //clear the screen  
        screen.clearScreen();  
        Graphics g = screen.getGraphics();  
  
        //Do all drawing here  
        g.setColor(Color.BLACK);  
        g.drawRect(150,150,50,50);  
    }  
}
```

COMPILE & RUN ScreenExample

- In ScreenExample file
- Click on the green arrow next to the main method
- Click “Run ScreenExample.main()”



WHAT'S THIS PROGRAM DOING?

LETS LOOK AT THE CODE

ScreenExample.java

```
/*  
*****  
* Author: Leah Buechley  
* Date: 8/2021  
* This is an example to help you use the Screen class  
* Refer to Java graphics documentation for information on drawing:  
* https://docs.oracle.com/en/java/javase/16/docs/api/java.desktop/java/awt/Graphics.html  
*****  
*/  
  
import java.awt.*;  
  
public class ScreenExample {  
    //Create a screen/window to draw in  
    static Screen screen= new Screen();  
  
    //Main just paints the screen over and over forever  
    public static void main(String[] args) {  
        while (true) {  
            paint();  
        }  
    }  
}
```

WRITING CODE: FOCUS

```
//The paint() method is where all the interesting stuff happens
public static void paint() {
    //clear the screen
    screen.clearScreen();
    Graphics g = screen.getGraphics();

    //Do all drawing here
    g.setColor(Color.BLACK);
    g.drawRect(150,150,50,50);

    //update the screen with the drawing that you made
    screen.update(g);
}
```


CHANGING THE RECTANGLE

RECTANGLE: SIZE

```
//The paint() method is where all the interesting stuff happens
public static void paint() {
    //clear the screen
    screen.clearScreen();
    Graphics g = screen.getGraphics();

    //Do all drawing here
    g.setColor(Color.BLACK);
    g.drawRect(150,150,100,100);

    //update the screen with the drawing that you made
    screen.update(g);
}
```

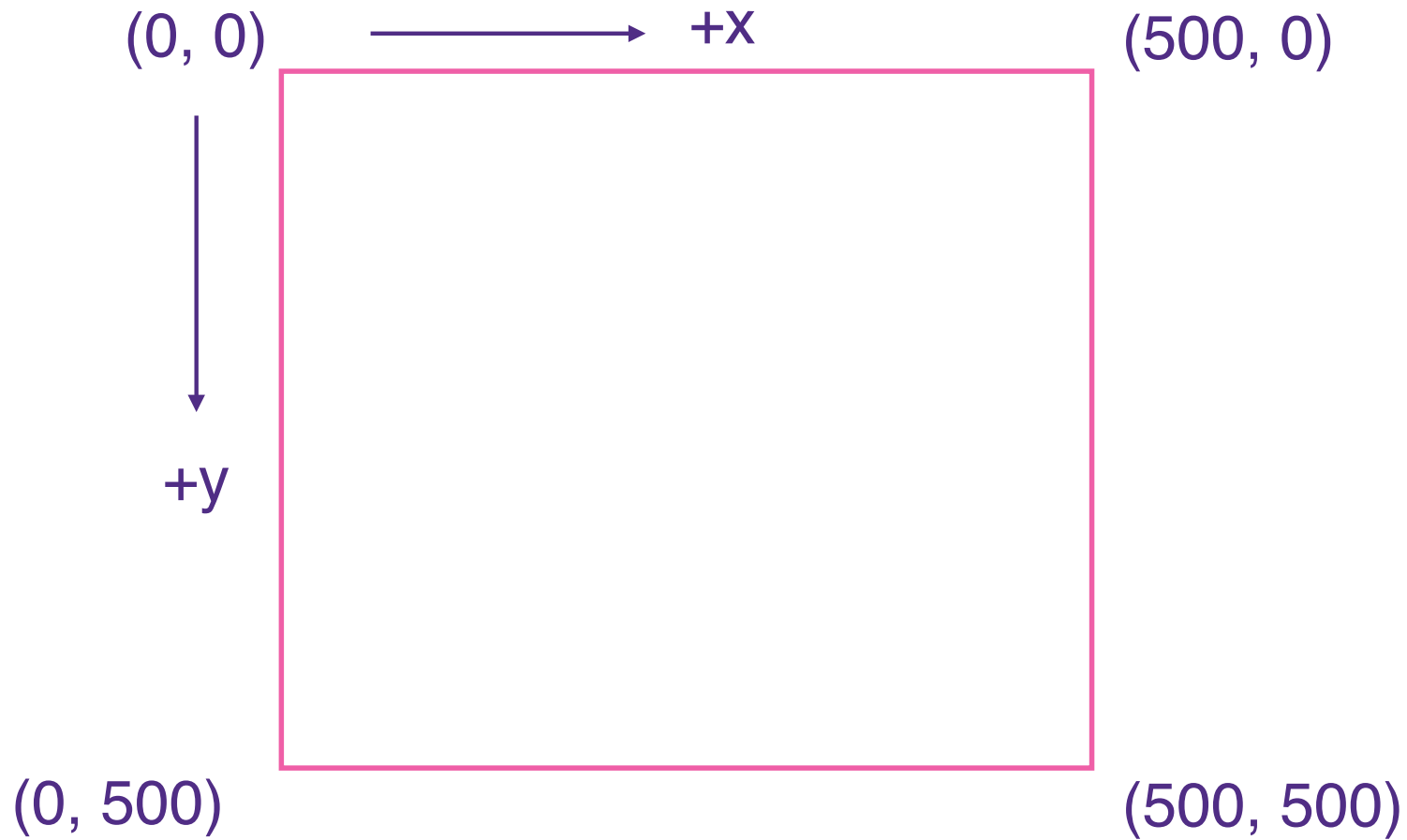
RECTANGLE: POSITION

```
//The paint() method is where all the interesting stuff happens
public static void paint() {
    //clear the screen
    screen.clearScreen();
    Graphics g = screen.getGraphics();

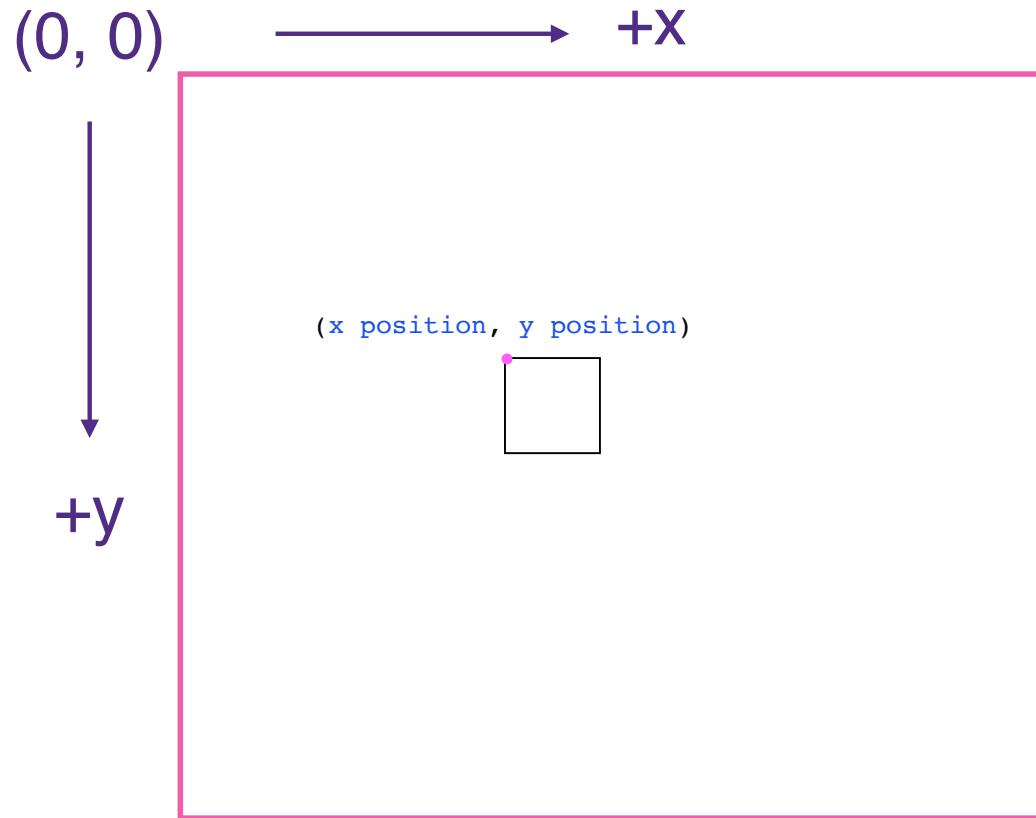
    //Do all drawing here
    g.setColor(Color.BLACK);
    g.drawRect(200, 150, 100, 100);

    //update the screen with the drawing that you made
    screen.update(g);
}
```

CS COORDINATE SYSTEM

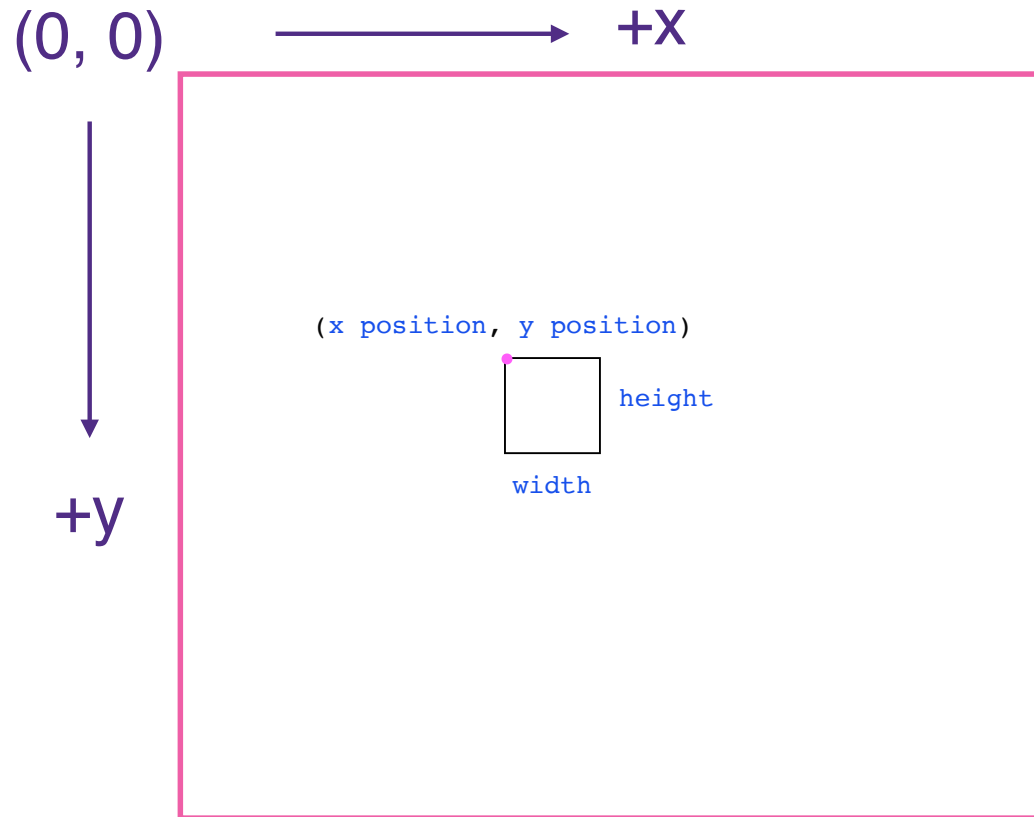


CS COORDINATE SYSTEM



```
g.drawRect(x position,y position,100,100);
```

CS COORDINATE SYSTEM



```
g.drawRect(x position,y position, width, height);
```

questions?

INTERACTIVITY

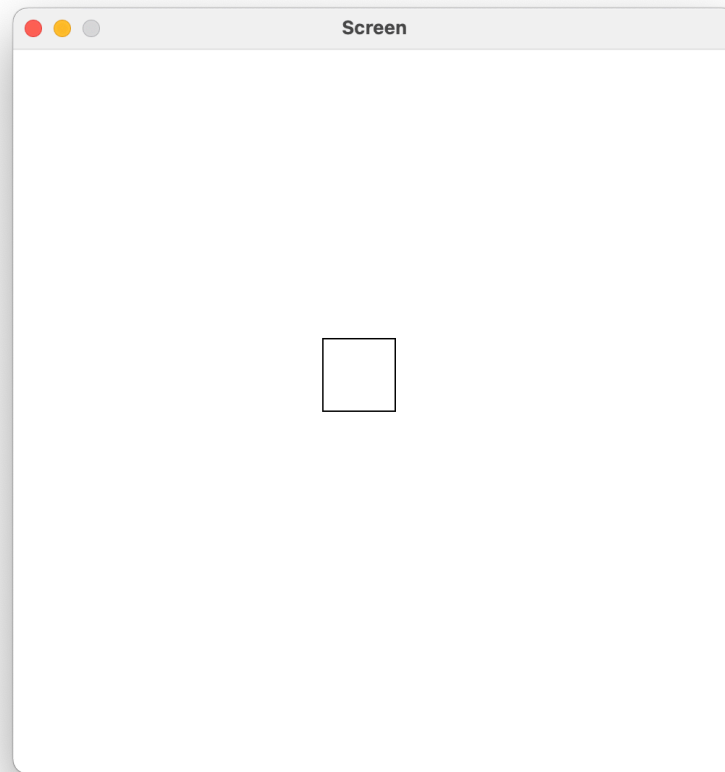
INTERACTIVITY

```
//The paint() method is where all the interesting stuff happens
public static void paint() {
    //clear the screen
    screen.clearScreen();
    Graphics g = screen.getGraphics();

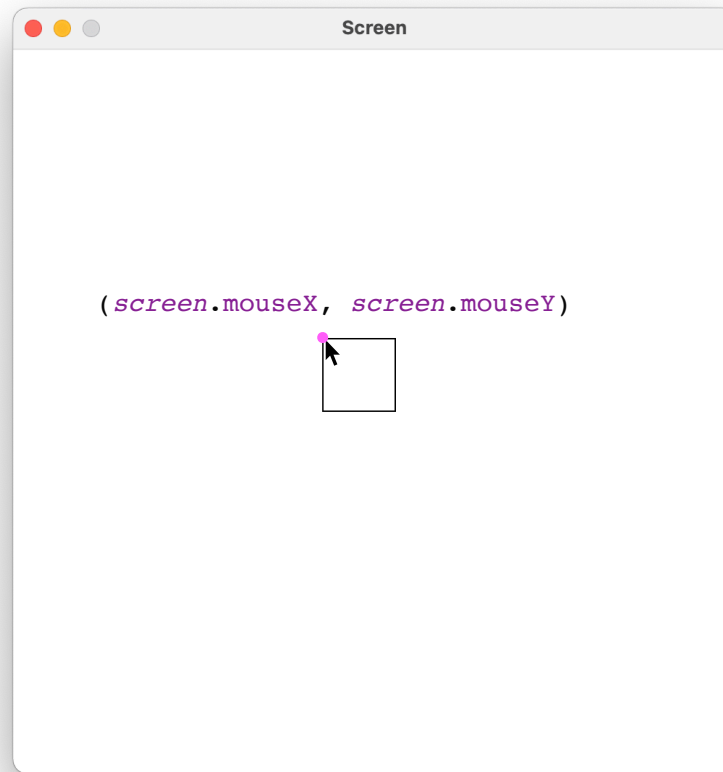
    //Do all drawing here
    g.setColor(Color.BLACK);
    g.drawRect(screen.mouseX, screen.mouseY 50,50);

    //update the screen with the drawing that you made
    screen.update(g);
}
```

INTERACTIVITY



INTERACTIVITY

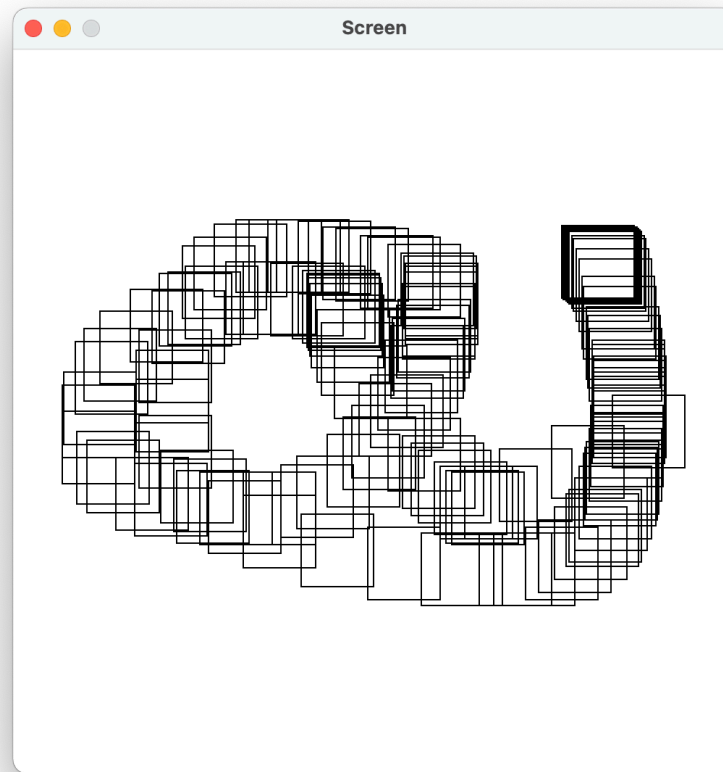


INTERACTIVITY

```
//The paint() method is where all the interesting stuff happens  
public static void paint() {  
    //clear the screen  
    //screen.clearScreen();  
    Graphics g = screen.getGraphics();  
  
    //Do all drawing here  
    g.setColor(Color.BLACK);  
    g.drawRect(screen.mouseX, screen.mouseY, 50,50);  
  
    //update the screen with the drawing that you made  
    screen.update(g);  
}
```

PREDICTIONS?

INTERACTIVITY



WHY?

```
public class ScreenExample {
    //Create a screen/window to draw in
    static Screen screen= new Screen();

    //Main just paints the screen over and over forever
    public static void main(String[] args) {
        while (true) {
            paint();
        }
    }

    //The paint() method is where all the interesting stuff happens
    public static void paint() {
        //clear the screen
        screen.clearScreen();
        Graphics g = screen.getGraphics();

        //Do all drawing here
        g.setColor(Color.BLACK);
        g.drawRect(screen.mouseX, screen.mouseY, 50,50);

        //update the screen with the drawing that you made
        screen.update(g);
    }
}
```


questions?

A FILLED RECTANGLE

```
public class ScreenExample {
    //Create a screen/window to draw in
    static Screen screen= new Screen();

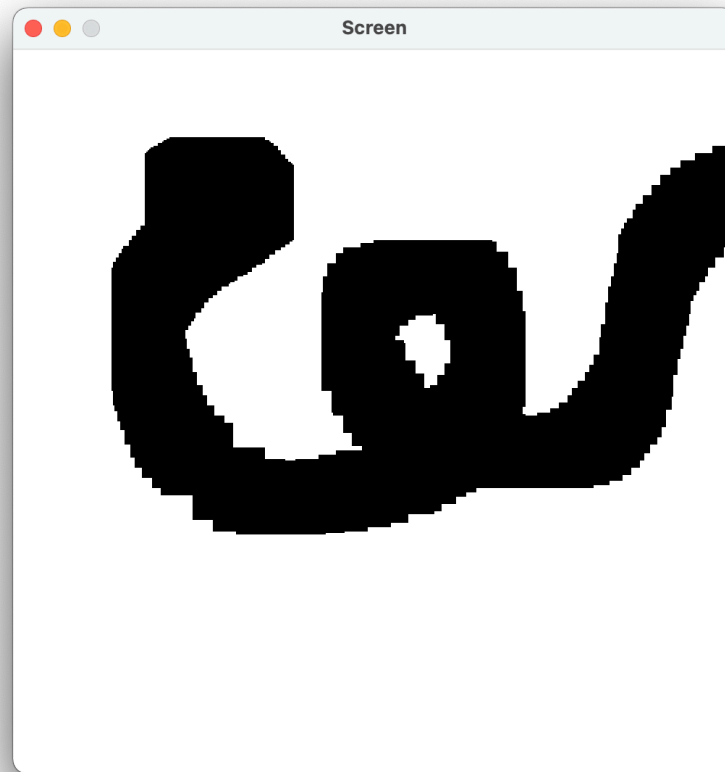
    //Main just paints the screen over and over forever
    public static void main(String[] args) {
        while (true) {
            paint();
        }
    }

    //The paint() method is where all the interesting stuff happens
    public static void paint() {
        //clear the screen
        //screen.clearScreen();
        Graphics g = screen.getGraphics();

        //Do all drawing here
        g.setColor(Color.BLACK);
        g.fillRect(screen.mouseX, screen.mouseY, 50,50);

        //update the screen with the drawing that you made
        screen.update(g);
    }
}
```

A FILLED RECTANGLE



CHANGING THE COLOR

```
public class ScreenExample {
    //Create a screen/window to draw in
    static Screen screen= new Screen();

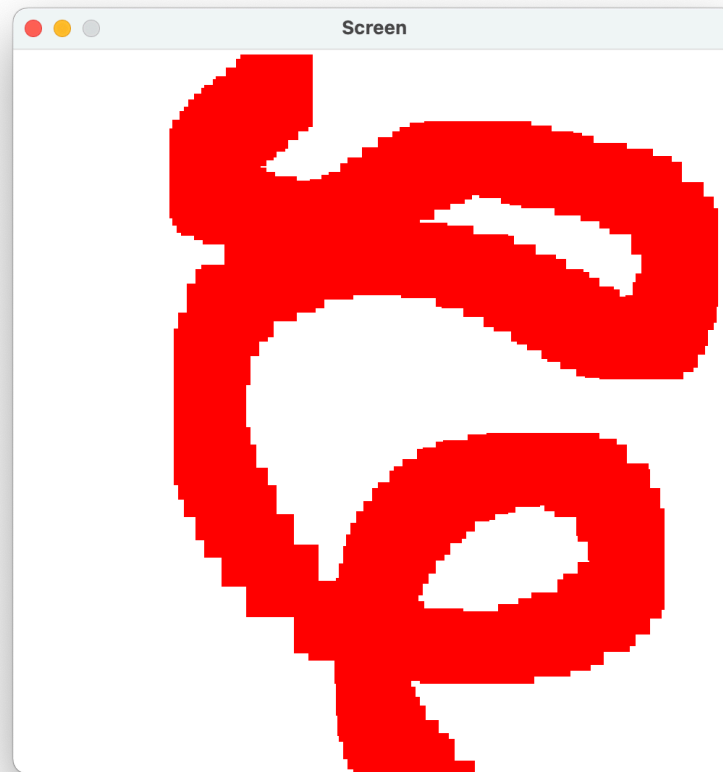
    //Main just paints the screen over and over forever
    public static void main(String[] args) {
        while (true) {
            paint();
        }
    }

    //The paint() method is where all the interesting stuff happens
    public static void paint() {
        //clear the screen
        //screen.clearScreen();
        Graphics g = screen.getGraphics();

        //Do all drawing here
        g.setColor(Color.RED);
        g.fillRect(screen.mouseX, screen.mouseY, 50,50);

        //update the screen with the drawing that you made
        screen.update(g);
    }
}
```

A RED RECTANGLE



questions?

THE SCREEN

THE SCREEN

```
public class ScreenExample {
    //Create a screen/window to draw in
    static Screen screen= new Screen();

    //Main just paints the screen over and over forever
    public static void main(String[] args) {
        while (true) {
            paint();
        }
    }

    //The paint() method is where all the interesting stuff happens
    public static void paint() {
        //clear the screen
        screen.clearScreen();
        Graphics g = screen.getGraphics();

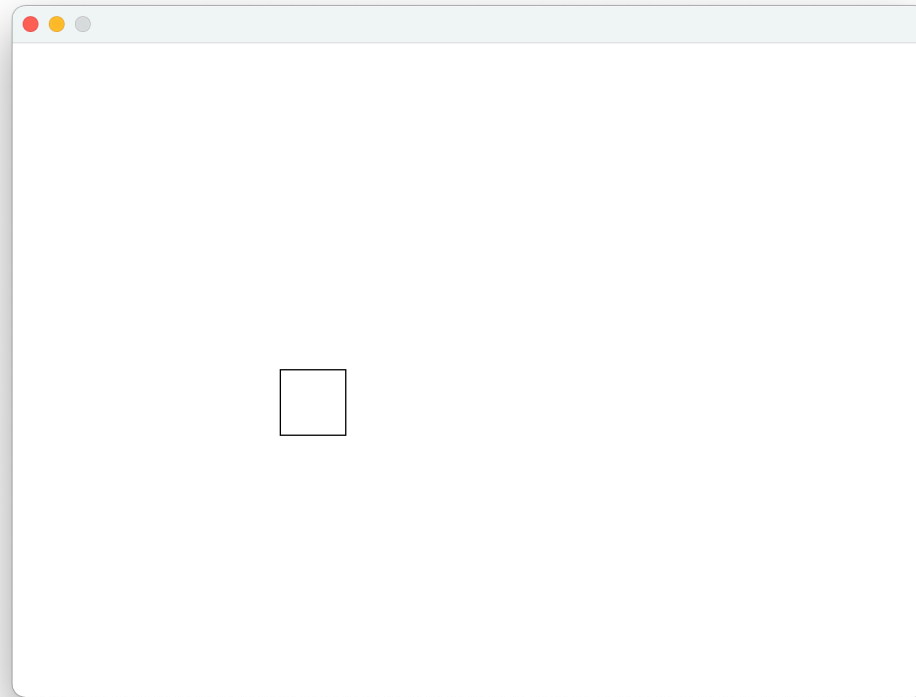
        //Do all drawing here
        g.setColor(Color.PINK);
        g.drawRect(screen.mouseX, screen.mouseY, 50,50);

        //update the screen with the drawing that you made
        screen.update(g);
    }
}
```

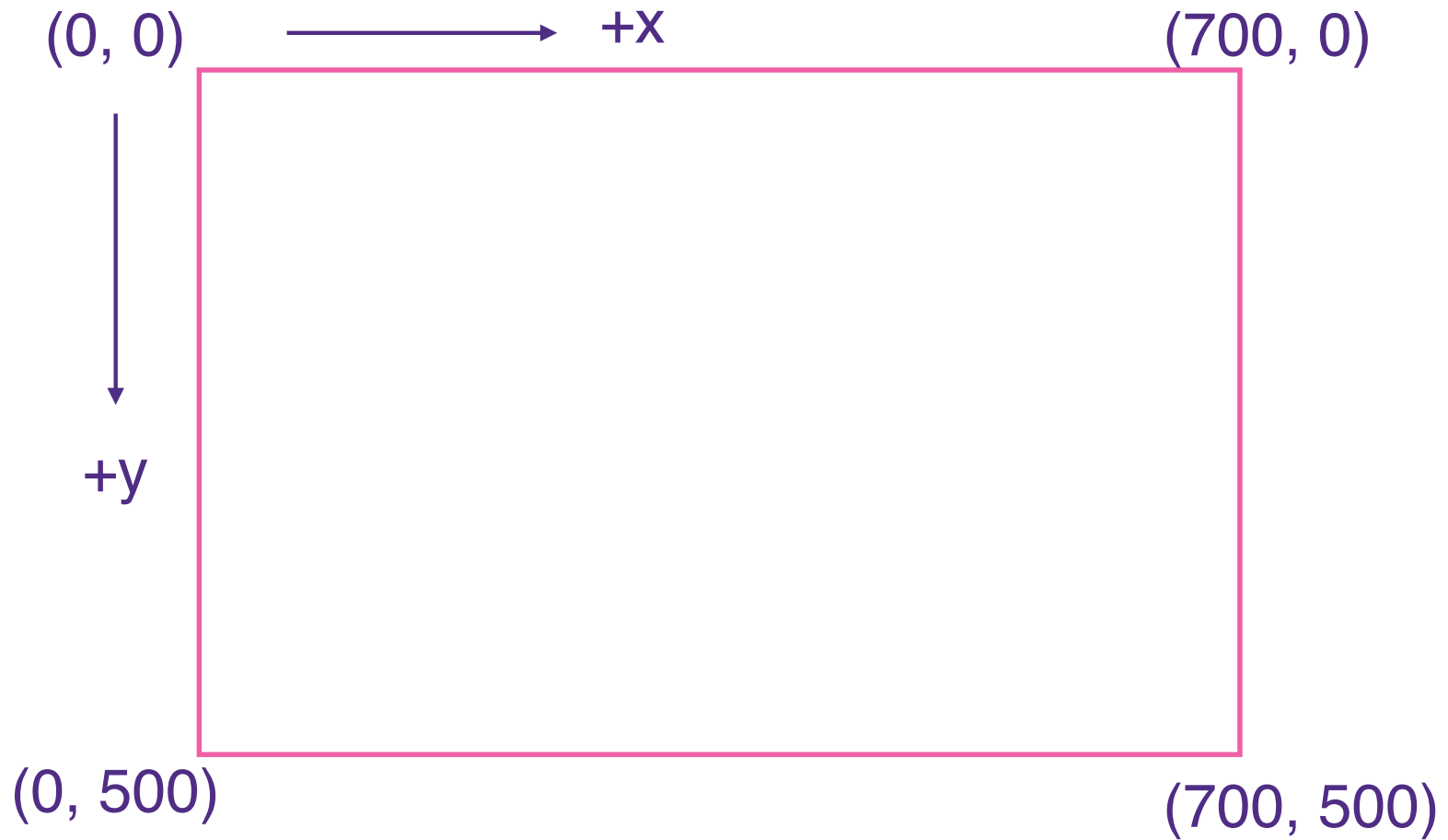
THE SCREEN

```
public class ScreenExample {  
    //Create a screen/window to draw in  
    static Screen screen= new Screen(700,500);  
  
    //Main just paints the screen over and over forever  
    public static void main(String[] args) {  
        while (true) {  
            paint();  
        }  
    }  
  
    //The paint() method is where all the interesting stuff happens  
    public static void paint() {  
        //clear the screen  
        screen.clearScreen();  
        Graphics g = screen.getGraphics();  
  
        //Do all drawing here  
        g.setColor(Color.PINK);  
        g.drawRect(screen.mouseX, screen.mouseY, 50,50);  
  
        //update the screen with the drawing that you made  
        screen.update(g);  
    }  
}
```

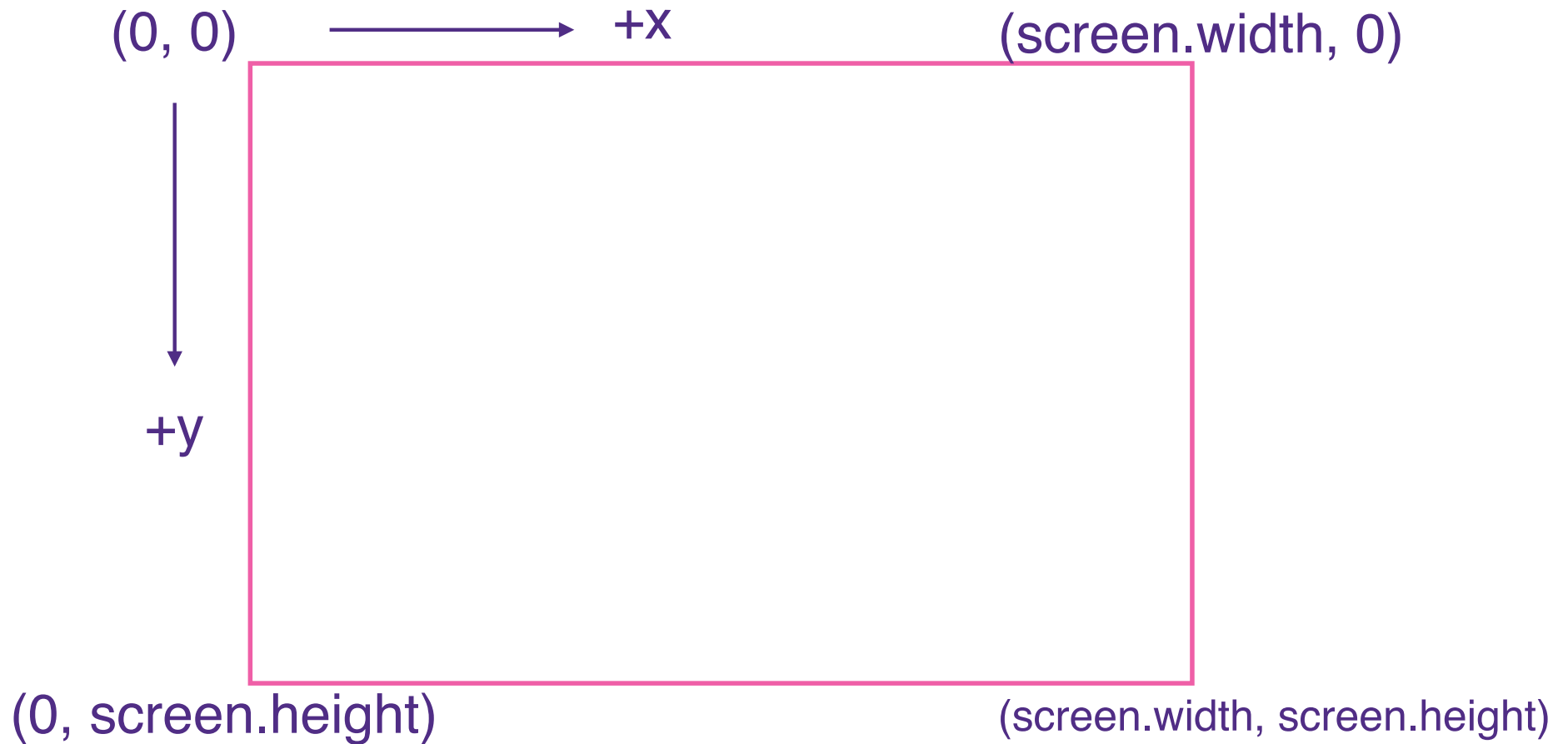
THE SCREEN



CS COORDINATE SYSTEM



CS COORDINATE SYSTEM



THE SCREEN

```
//Create a screen/window to draw in
static Screen screen= new Screen(700,500);

//Main just paints the screen over and over forever
public static void main(String[] args) {
    while (true) {
        paint();
    }
}

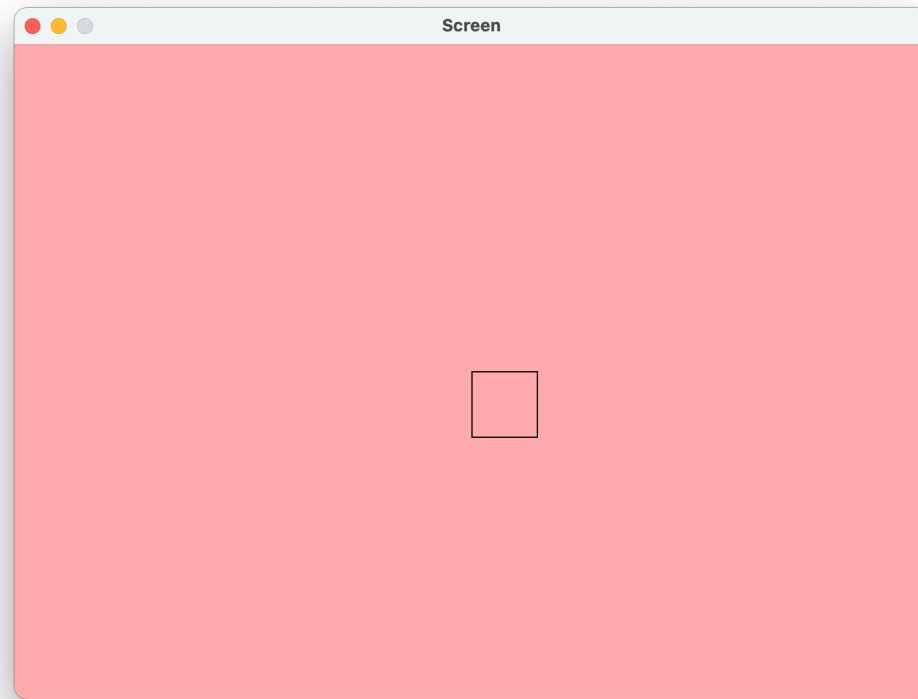
//The paint() method is where all the interesting stuff happens
public static void paint() {
    //clear the screen
    screen.clearScreen();
    screen.setBackground(Color.PINK);

    Graphics g = screen.getGraphics();

    //Do all drawing here
    g.setColor(Color.PINK);
    g.drawRect(screen.mouseX, screen.mouseY, 50,50);

    //update the screen with the drawing that you made
    screen.update(g);
}
}
```

THE SCREEN



How would we draw a rectangle in the center of the screen?

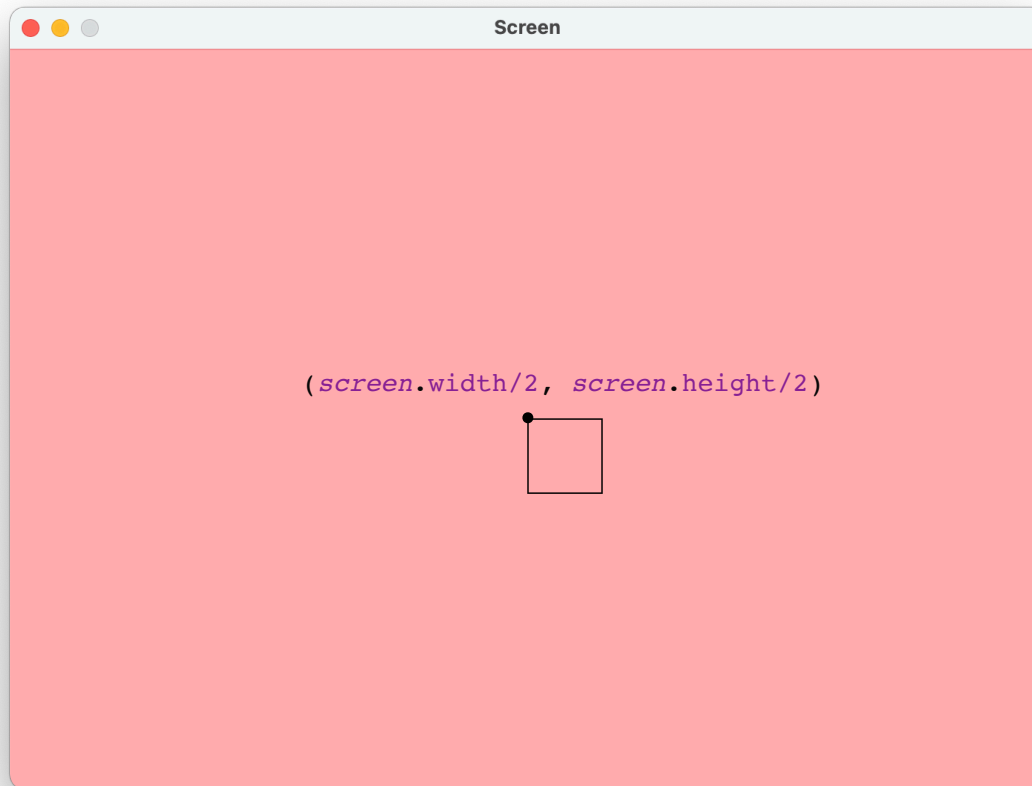
CENTERED RECTANGLE?

```
//the paint() method is where all the interesting stuff happens
public static void paint() {
    //clear the screen
    screen.clearScreen();
    Graphics g = screen.getGraphics();
    screen.setBackground(Color.PINK);

    //Do all drawing here
    g.setColor(Color.BLACK);
    g.drawRect(screen.width/2, screen.height/2, 50,50);

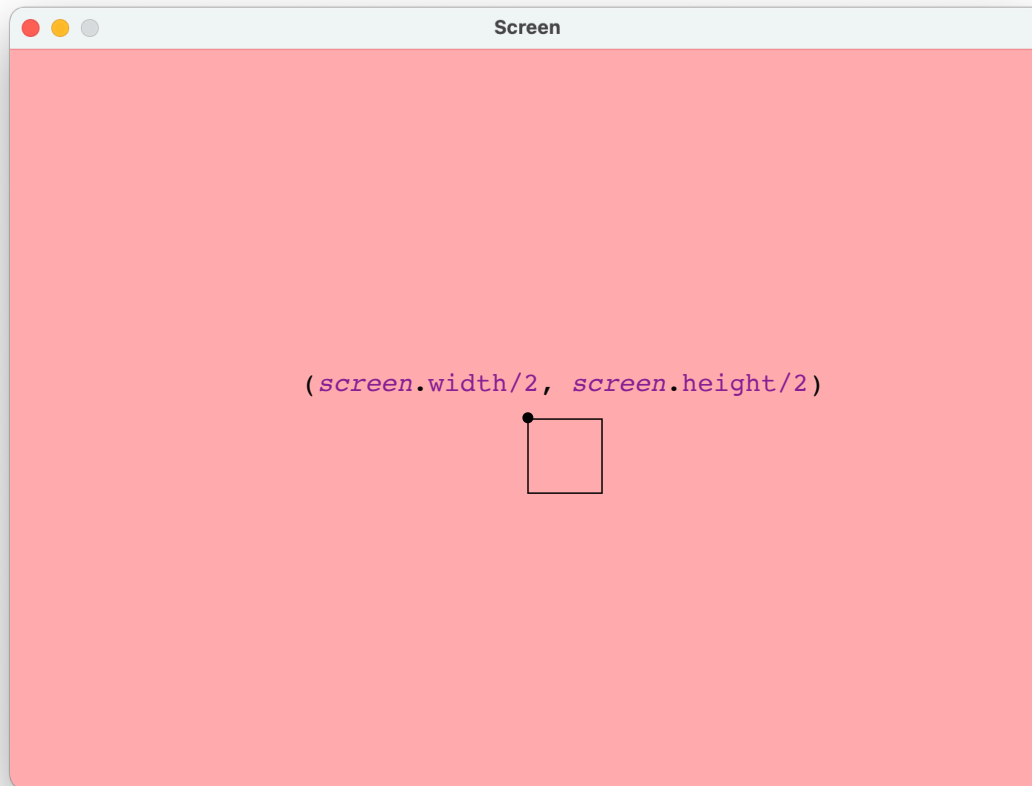
    //update the screen with the drawing that you made
    screen.update(g);
}
```

CENTERED RECTANGLE?



The upper left corner is centered.
How would we center the shape?

CENTERED RECTANGLE?



CENTERED RECTANGLE?



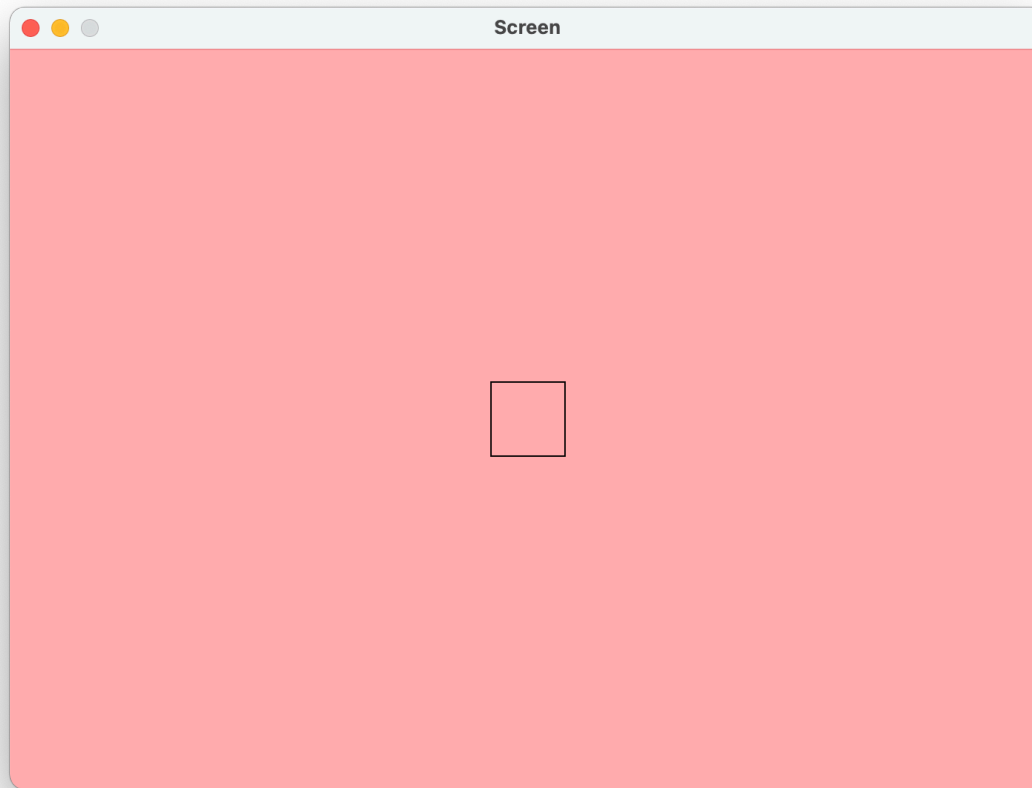
CENTERED RECTANGLE

```
//the paint() method is where all the interesting stuff happens
public static void paint() {
    //clear the screen
    screen.clearScreen();
    Graphics g = screen.getGraphics();
    screen.setBackground(Color.PINK);

    //Do all drawing here
    g.setColor(Color.BLACK);
    g.drawRect(screen.width/2-25, screen.height/2-25, 50, 50);

    //update the screen with the drawing that you made
    screen.update(g);
}
```

CENTERED RECTANGLE



HOW CAN WE IMPROVE THIS CODE?

```
//the paint() method is where all the interesting stuff happens
public static void paint() {
    //clear the screen
    screen.clearScreen();
    Graphics g = screen.getGraphics();
    screen.setBackground(Color.PINK);

    //Do all drawing here
    g.setColor(Color.BLACK);
    g.drawRect(screen.width/2-25, screen.height/2-25, 50,50);

    //update the screen with the drawing that you made
    screen.update(g);
}
```


VARIABLES!

SUGGESTIONS?

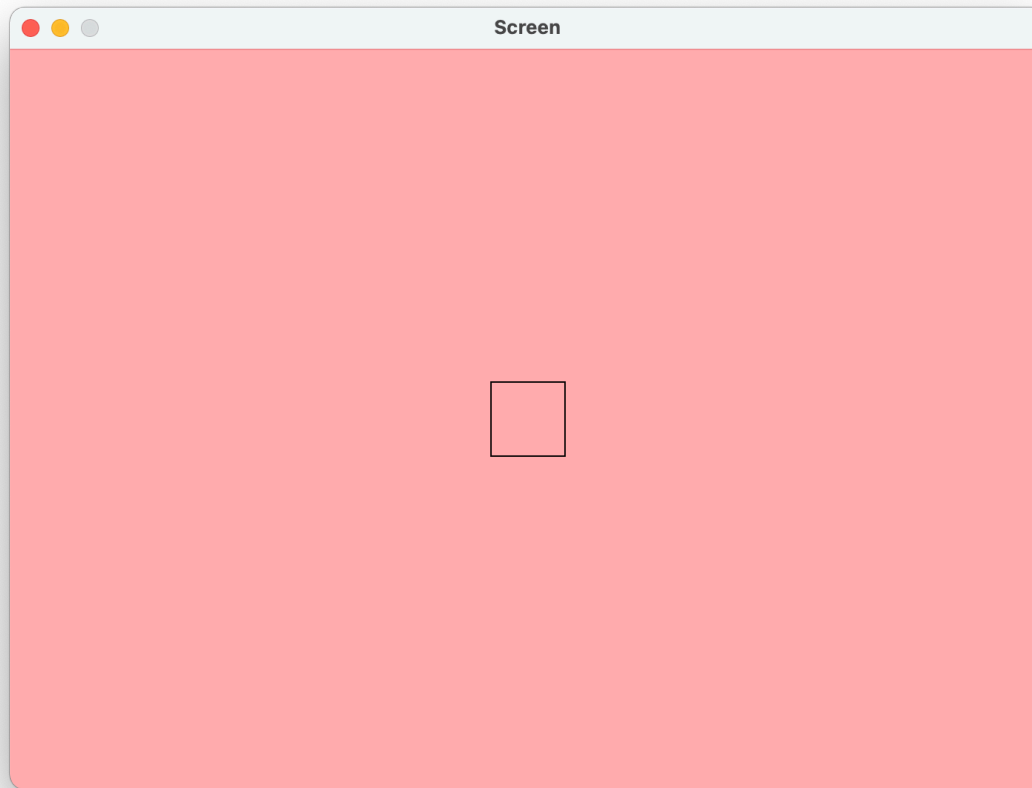
IMPROVED CODE

```
//The paint() method is where all the interesting stuff happens
public static void paint() {
    //clear the screen
    screen.clearScreen();
    Graphics g = screen.getGraphics();
    screen.setBackground(Color.PINK);

    //Do all drawing here
    g.setColor(Color.BLACK);
    int rectWidth = 50;
    int rectHeight = 50;
    g.drawRect(screen.width/2-rectWidth/2, screen.height/2-rectHeight/2, rectWidth, rectHeight);

    //update the screen with the drawing that you made
    screen.update(g);
}
```

CENTERED RECTANGLE



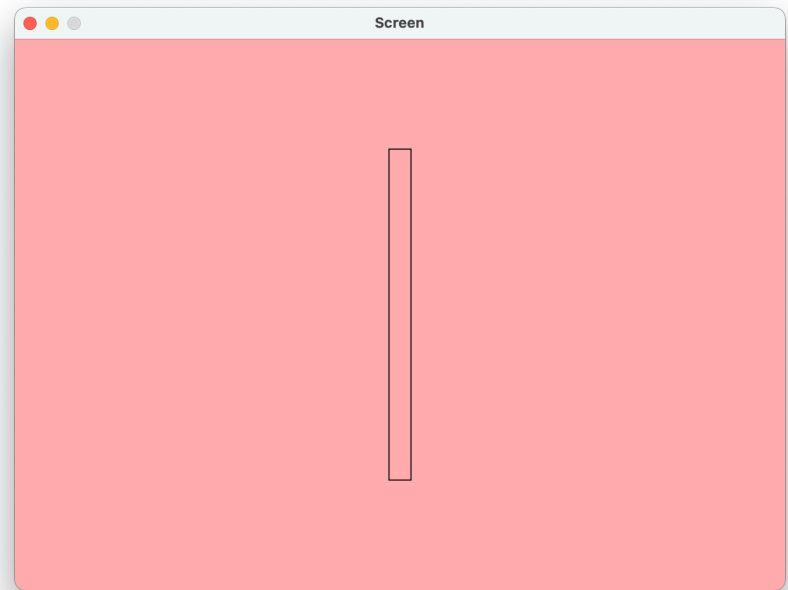
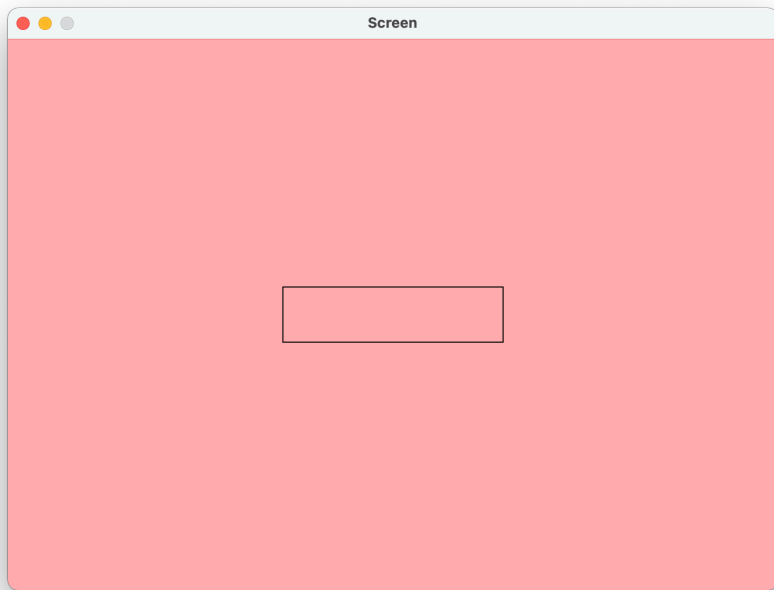
IMPROVED CODE

```
//The paint() method is where all the interesting stuff happens
public static void paint() {
    //clear the screen
    screen.clearScreen();
    Graphics g = screen.getGraphics();
    screen.setBackground(Color.PINK);

    //Do all drawing here
    g.setColor(Color.BLACK);
    int rectWidth = 200;
    int rectHeight = 50;
    g.drawRect(screen.width/2-rectWidth/2, screen.height/2-rectHeight/2, rectWidth, rectHeight);

    //update the screen with the drawing that you made
    screen.update(g);
}
```

CENTERED RECTANGLES



questions?

DUE TUESDAY: ASSIGNMENT 2

- Due Tuesday 9/6 by 9:30am
- Use the Screen.java code from class
- Submit via UNM Learn

Thank you!

CS 152

Professor: Leah Buechley

TAs: Melody Horn, Noah Garcia, Andrew Geyko, Juan Ormaza

Time: MWF 10:00-10:50am

https://handandmachine.cs.unm.edu/classes/CS152_Fall2021/