CSSE 220 Day 6

Intro to Java Graphics

Check out IntroToJavaGraphics and BiggestFan projects from SVN

Announcement

- Exam is one week from today
  - Wednesday Sept. 23
- More details to follow

Outline

- Static (by examples)
- Live coding: a Java graphics program

```java
public class Car {

    double mileage;

    //other stuff

    public double getMilesTravelled() {
        return this.mileage;
    }

    public static double convertMilesToKm(double numberOfMiles) {
        return numberOfMiles * 1.609344f;
    }

}

//Elsewhere…

//requires you to have a car object
Car myCar = new Car();
//requires you to have a car object
System.out.println(myCar.getMilesTravelled());//output depends on code
//can be called on the class Car itself
System.out.println(Car.convertMilesToKm(77));//output is 123.919488
```

```java
public class Bicycle {

    private int speed;
    private static int numCreated = 0;

    public Bicycle(int speed) {
        this.speed = speed;
        Bicycle.numCreated++;
    }
    public int getSpeed() {
        return this.speed;
    }
    public static int getNumCreated() {
        return Bicycle.numCreated;
    }

}
//No requirement to have a Bicycle yet…
System.out.println(Bicycle.getNumCreated());
Bicycle myBike1 = new Bicycle(18);
Bicycle myBike2 = new Bicycle(1);
System.out.println(Bicycle.getNumCreated() + " " + myBike1.getSpeed());
0
2 18
```

Simplest Java Graphics Program

```java
import javax.swing.JFrame;
/**
 * From Ch 2, Big Java.
 * @author Cay Horstmann
 */
public class EmptyFrameViewer {
    /**
     * Draws a frame.
     * @param args ignored
     */
    public static void main(String[] args) {
        JFrame frame = new JFrame();
        frame.setSize(300,400);
        frame.setTitle("An Empty Frame");
        frame.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        frame.setVisible(true);
    }
}
```

This code is already in your project for today

Creates a graphics frame object

Configures it

Display the frame

Tells Java to exit program when user closes the frame

MyViewer and MyComponent (Based on RectangleViewer and RectangleComponent from Big Java)

LIVE CODING

Other Shapes

- `new Ellipse2D.Double(double x, double y,`
  `                      double w, double h)`
- `new Line2D.Double(double x1, double y1,`
  `                   double x2, double y2)`
- `new Point2D.Double(double x, double y)`
- `new Line2D.Double(Point2D p1, Point2D p2)`
- `new Arc2D.Double(double x, double y,`
  `                 double w, double h,`
  `                 double start, double extent,`
  `                 int type)`

- Try these!
  - Add an ellipse and both kinds of lines to MyComponent

Using translate and rotate successfully

- Translate and rotate to adjust the "state" of the pen
- It is usually easier to move the pen, than draw in a fixed configuration around (0,0), then move the pen back
- Make (0,0) your center of rotation
  - can change the point of origin using translate() so you can rotate different portions of the component

Work on the biggest fan code

- We'll walk through it together to explain how the classes work

- Then you should modify the fan to print one blade vertically – use transform to move (0,0) to the **center** of the fan and then draw from there

Making Faces

- Due Monday Sep 21, 11:59 PM
- Implement a class that draws a face of a given size at a given location. You should also be able to mutate (translate & rotate) it.
- Similar to Biggest Fan (but more complex)
1. Specification (in HW)
2. Code (incrementally)
3. This project can be difficult