Slides for the Week

CS273 Laboratory 9

This week's lab focuses on classes

This is not a pipe.

It is a digital projection of a painting of a pipe.

This is not a pipe.

It is a digital projection of someone else's rendering of Nintendo's digitally rendered pipe.

Ultimately everything on a computer is a simulation or model of something else.

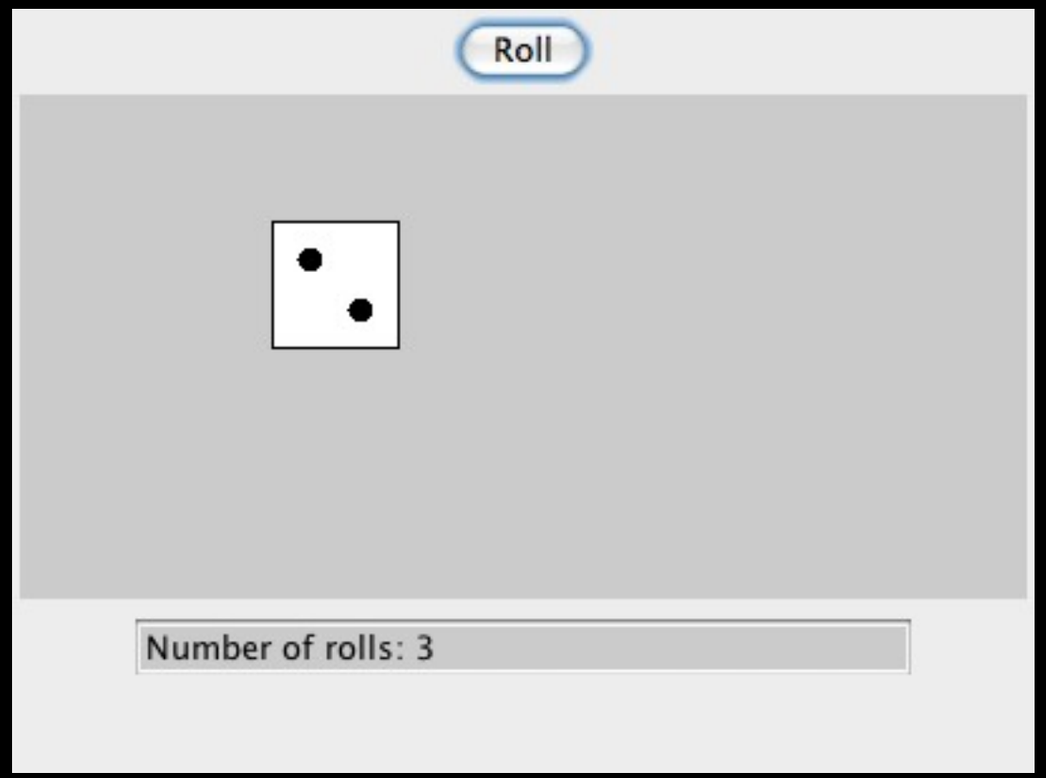
In order for the computer to do this effectively the programmer must **define** what makes a pipe a pipe.

```
// Simple Pipe constructor for Pipe class.
public Pipe(double height, double radius, Color
color) {
    h = height;
    r = radius;
    c = color;
}

...

// Meanwhile, one can use the constructor to
// create and initialize a pipe.
Pipe myPipe = new Pipe(30.0, 0.25, Color.red);
```

In this lab, you will define a class in Die.java to represent a single cube-shaped die.

The data:

- x and y position on graphics object (where to draw)
- value showing on die
- optionally a size that tells the dimensions of a die in pixels.



The actions:

- Constructor, roll, paint, reRoll setSize, drawSpot, and more.

These are defined in the starter code:

- **instance variables**: what data are in the object.

- **methods**: the actions that the object can perform.

```
public class Die {

  // Checkpoint 1: Private instance variables
  private int x;
  private int y;
  private int currentValue;

  private int size;

  public Die (int newX, int newY)
  {
      x = newX;
      y = newY;
  }

  public void setSize(int s)
  {

  }

  private void drawSpot(int xcord, int ycord, Graphics g)
  {

  }

...

}
```

One of the methods to be implemented is called **`paint`**

It takes a graphics object and draws itself upon it at its x and y location.

This means that, depending upon how the paint method is called, the die will be drawn in a different place and in a different way.

```
public void paint(Graphics g)
  {

   ...

      // Draw a white rectangle
         g.setColor(Color.WHITE);
         g.fillRect(x,y,size,size);

  ...

}
```

Another of the methods to be implemented is called **drawSpot**

It takes 3 parameters. It draws a SINGLE spot. The x,y coordinates that it receives are relative to the die, **not** the graphics object. So, the method will have to do some translation.

Please do **not** do this translation manually in the paint method.

```
private void drawSpot(int xcord, int
ycord, Graphics g)
  {
   ...

  }
```

Because the Die.java class will have different capabilities at each checkpoint, there are different test programs for each checkpoint. Therefore some test programs won't compile right away. This is OK.

- Checkpoint #1, #2 = RunAB
- Checkpoint #3 = RunC
- Checkpoint #4 = RunD
- Checkpoint #5, #6 = RunE

- A common question asked in this lab, ***"How do I convert an int to a String?"***

```
int myInt = 5;
String myValue = “” + i;
```

Good Luck!

If you have any questions the TAs and I are happy to help.