

## CS202 Java Object Oriented Programming GUI Programming – Introduction to Java2D

Chengyu Sun  
California State University, Los Angeles

## The paint() Method

- ◆ How does Java draw the GUI components?
- ◆ `public void paint(Graphics g)`
  - A method of `java.awt.Component`
  - Inherits by all AWT and Swing components
  - Responsible for “drawing” the component

## Customize paint()

```
public class PaintBoard extends JFrame {  
    ... ..  
    public void paint( Graphics g )  
    {  
        super.paint(g);  
        // paint some more  
        ...  
    }  
}
```

## paint() Again

```
public class PaintBoard2D extends JFrame {  
    ... ..  
    public void paint( Graphics g )  
    {  
        super.paint(g);  
        Graphics2D g2 = (Graphics2D) g;  
        // paint some more  
        ...  
    }  
}
```

## Old Friends

- ◆ `Graphics2D` is a subclass of `Graphics`
  - `drawLine`, `drawString`
  - `draw/fillRect`, `draw/fillOval`, `draw/fillArc`
  - ...

## New Friends in `java.awt.geom`

- ◆ `Shape`
  - `Line2D`, `Arc2D`, `Ellipse2D`, `Rectangle2D`, `RoundedRectangle2D`
  - Curves
  - `GeneralPath` (generalized polygon)
  - `Area` (boolean of shapes )
- ◆ `g2.draw( Shape )`, `g2.fill( Shape )`
- ◆ <http://java.sun.com/docs/books/tutorial/2d/display/strokeandfill.html>

## Coordinates

- ◆ Float and double coordinates
  - Line2D.Float, Line2D.Double
  - Rectangle2D.Float, Rectangle2D.Double
  - Ellipse2D.Float, Ellipse2D.Double
- ◆ Why?

## Stroke – Line Width and More

- ◆ g2.setStroke( Stroke )
- ◆ BasicStroke
  - width
  - end caps
  - line joins
  - miter limit
  - dash pattern

## Paint – Fancy Colors

- ◆ g2.setPaint( Paint )
- ◆ Paint
  - Color
  - GradientPaint
    - ◆ (x1, y1, color1, x2, y2, color2)
  - TexturePaint

## Transformations

- ◆ *It's the coordinate system that moves.*
- ◆ Transformations
  - g2.translate(double x, double y)
  - g2.rotate(double theta, double x, double y)
  - g2.scale(double sx, double sy)

## Simple Animation

```
public void paint( Graphics g )
{
    ... ..
    try
    {
        Thread.sleep(100);
    }
    catch( Exception e )
    {}

    repaint();
}
```