Slide 1

Administrivia

- Reminder: Please keep the lab doors closed after hours. There have been reports of thefts.
- Homework 6 will be on the Web later today. Design due Thursday, code next Tuesday.

  Not all homeworks 4 and 5 are in — if you're having trouble, come talk to me; if you're just behind, try to catch up.
- My course next term (CSCI 3190, "Unix Power Tools") — description linked from my home page.

Slide 2

Graphics in Java —Custom Components

- Predefined components (`JButton`, etc.) do a lot, but what if you want something that's not provided? in particular, you want to control the image yourself?
- Make a custom component — define a subclass of a component that provides some of the needed functionality, and override the method that defines what's displayed.

  E.g., subclass `JPanel` and override `paintComponent`, to include your code to "paint" the panel.
- Call `repaint` when ready to redisplay.

Slide 3

Custom Painting

- Method to override is

  `public void paintComponent(Graphics g)`.

  `g` is a "graphics context" that you can draw on. (Actually it's a `Graphics2D`.) Tutorial recommends first calling `super.paintComponent(g)`.
- Can get dimensions of panel with `getSize`, `getHeight`, `getWidth`, `getInsets`.
- Can set colors, draw shapes, lines, text, etc., etc. — see `Graphics` and `Graphics2D` class. Coordinate system is similar to what you're using in your game. See code in `BasicBlock` for simple example.
- General advice — look over the methods of `Graphics` and `Graphics2D`; if confused, follow links to tutorial(s) and look for a suitable example to adapt.

Slide 4

- Let's look at an example . . .

Slide 5

Minute Essay

- None — quiz 4.