

# for Loop and Lists

9/23/2009

# Opening Discussion

- Do you have any questions about the quiz?
- Minute Essay comments
  - Greenfoot on campus and lab access.
  - Have people come back out of the houses at random times, but only if they went in.
  - When do we leave Greenfoot?
  - MonsterLemurs as pets or circle houses waiting for prey.
  - Can we break a computer?

# More Minute Essays

- Can we see the code from a popular game?
- Build houses in a pattern.
- Game of sharks and minnows with people, lemurs, and houses.

# Objectives for the Week

- The first objective is to be able to repeat code without doing cut and paste.
- We want to use this to have the people walk toward the closest house.
- Then we want to make it so each house can only hold one person.
- With a little elaboration we could make an interesting game.

# Repeating Code

- So far, if I have asked you to do something several times you have just copied the code.
- This is inefficient and inflexible. Can't do it a variable number of times.
- Let's consider a new method in city:
  - `void addBuildings(int howMany)`
- Copy and paste can't do this.
- The solution is a construct called a loop.

# The for Loop

- The first loop we will learn about is the for loop. These are the most commonly used loops in Java and can do anything you want.
  - `for(init; condition; iterator) {...}`
- We will start off with just counting.
  - `for(int i=0; i<number; i++) {...}`
- The variable doesn't have to be `i`.
- The statement `i++` is shorthand for `i=i+1`.
- Now let's add the buildings.

# Lists

- Right now one variable refers to one object. We want to be able to deal with collections of objects.
- The Greenfoot API has methods in World and Actor that tell us about Actors. These all return lists. Let's look at them.
- We can also look at List in the full API. The main methods we need now are `get(int index)` and `size()`.
- List indexes start at zero.

# Generic Information

- When we declare a List we can tell Java the type of things that will be in the List. This is called a generic.
- In our case, the generic should match the type of the class we pass into the method.



# House Hunting

- Let's work on a method in our person called `moveToNearestHouse()`.
- First, we should see if we can figure out how to run through all the buildings.
- Once we can do that we need to figure out how to determine which one is closest.

# Minute Essay

- What questions do you have about the things that we covered today? How comfortable do you feel working in Greenfoot?