

Lesson 6: Defining Methods

Try It: Practice Activities

Objectives

- Describe effective placement of methods in a super or subclass
- Simplify programming by creating and calling defined methods

Vocabulary

Identify the vocabulary word for each definition below.

	A new method that a class didn't already possess; these methods are written in the class's source code below the act method.
--	------------------------------------------------------------------------------------------------------------------------------

Project

1. Continue the project from Section 3 Lesson 5.
2. Open the Project named FrogFlyL5. From the Scenario Menu, choose "Save as" and use the name FrogFlyL6. This will keep a copy from Lesson 5 in case there is a need to refer to it later.
3. Compile the Greenfoot project.
4. Open the code editor for the Fly subclass.
5. Change the distance in the move() statement in the act() method to 50.
6. Add an atWorldEdge() method that will return true if a Fly is on the edge of the World.
7. Add an if statement to the act() method in the Fly class to check the atWorldEdge() method. If it returns true, have the Fly turn right 90 degrees.
8. Open the code editor for the Frog subclass.
9. Add an atWorldEdge() method that will turn a Frog around if it is on the edge of the World.
10. Add an if statement to the act() method in the Frog class to call the atWorldEdge() method, and turn right 90 degrees if the Frog is on the edge.
11. Change the distance in the move() statement in the act() method to 50.
12. Open the code editor for the FrogWorld.
13. Use the addObject() method to add one Frog instance and one Fly instance to the

world.

14. Compile your code and test the `atWorldEdge()` method for both the fly and the frog by pressing the Up and Right arrow keys.
15. Test and debug. Recompile as necessary.
16. Save your project and exit.

Optional Practice

1. Create a new scenario with three Actor subclasses. Create a new method called `atWorldEdge()` for each subclass that will test if the object is on the edge. Turn the object 180 degrees if the world edge is reached. Use the `isKeyDown()` method or `move()` and `turn()` statements.