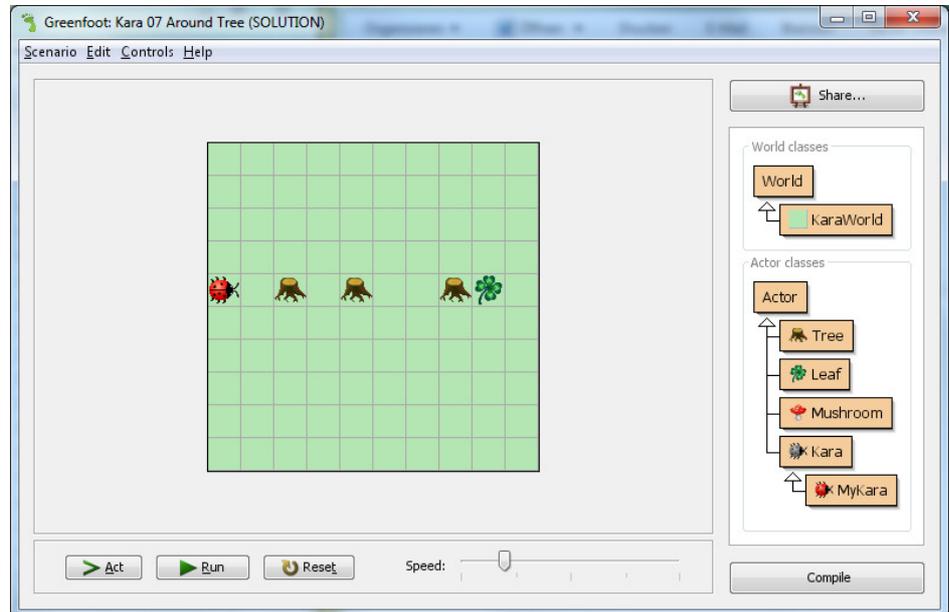


Instructions for Teachers

This document contains general information and hints to work with *GreenfootKara* as a teacher.

Example Task “Around Tree” from chapter 1:



Example from the game “Kara Sokoban” from chapter 4: (Kara must push all mushrooms on leaves)



Why Greenfoot and Kara

The **concept of Kara** has proven itself over years for the introduction to programming. Of great value are also the many exercises that already exist for Kara.

Greenfoot is a development environment for Java. It allows a very rapid development of interactive, graphical projects.

As a teacher you can provide Greenfoot scenarios on any level of difficulty for students to work with. **GreenfootKara** contains many such scenarios to work with the ladybug Kara.

Greenfoot is great for novice programmers because one doesn't have to deal with many of the big difficulties (like `public static void main(String [])` and such). Nevertheless, Greenfoot is real Java and leaves all options open. Thus, advanced students can also realize very complex projects.

➔ **GreenfootKara combines the proven concept of Kara with Greenfoot's flexibility and ease of use.**

Instructions for GreenfootKara

To use the Kara-Scenario Greenfoot must be installed first.

The newest Greenfoot versions can be found on the Greenfoot website <http://www.greenfoot.org/download/>. Run the setup program and follow the installation instructions.

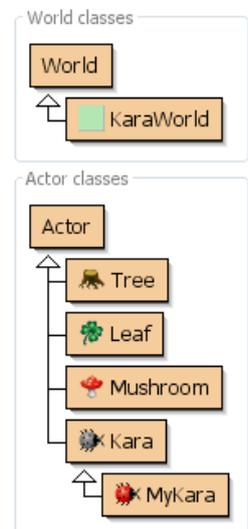
After installation, any Kara-Scenario may be opened using the “**project.greenfoot**” file in the scenario folder. For each exercise there is a scenario available (see folder **scenarios-chapter-1**, **scenarios-schapter-1-solutions**, etc.). In each scenario, the world of Kara (with trees, clovers, etc.), is already prepared for each exercise.

The Kara-Scenario

The Kara-scenario and thus the possibilities of Kara remain the same for all exercises. The possibilities of Kara are the following):

Actions	
<code>move()</code>	Kara makes a step in the current direction
<code>turnLeft()</code>	Kara turns left by 90 degrees
<code>turnRight()</code>	Kara turns right by 90 degrees
<code>putLeaf()</code>	Kara puts down a leaf
<code>removeLeaf()</code>	Kara picks up a leaf

Sensors	
<code>onLeaf()</code>	Kara checks if he stands on a leaf
<code>treeFront()</code>	Kara checks if there is a tree in front of him
<code>treeLeft()</code>	Kara checks if there is a tree on his left side
<code>treeRight()</code>	Kara checks if there is a tree on his right side
<code>mushroomFront()</code>	Kara checks if there is a mushroom in front of him



In Chapter 4 Kara has a few additional methods so that a Sokoban game can be programmed.

In Chapter 5 Kara has a few additional methods to show messages and to ask the user for input.

The classes Kara und MyKara

The most important classes are **Kara** and **MyKara**. The class Kara includes all the functionality of the beetle Kara as seen above. But **Programming is always done in the class MyKara** which, through inheritance, can access all the methods of Kara. Thus, the complexity of Kara’s methods are hidden from the students at first.

Later, the students may choose to find out how the methods were implemented in Kara itself. For this step it is recommended that they first view the class in the *Documentation* mode. This only shows the Javadoc comments in the Greenfoot editor.

Starting with Mouse Only

Objects of classes can be instantiated through *right-click, new()* and then be placed in the world. (Tip: Through pressing the Shift key you can place multiple instances in the world without using the context menu).

On first contact with GreenfootKara it is helpful to **only use the mouse**. If you right-click on a Kara-object, all available methods are shown and be selected with the mouse. This way one can get accustomed with how Kara works.

Programming

After the first contact with the mouse, the programming can be done inside the `act()` -method of `MyKara`. This method is executed when pressing the Act button. When the Run button is pressed, then the `act()` -method is called repeatedly.

Tips

The Editor

- Often the students have trouble with cleanly structuring the code. The editor helps you format by an auto-layout feature found inside the Edit menu.
- Ctrl-Space will open a pop-up for **auto completion**.
- Top right in the editor you can switch the view from *Source Code* to **Documentation**.
- In the menu *Options / Preferences ...* the font size can be changed (e.g. for presentation with a projector).

World Setup Files

In the file `WorldSetup.txt` the world can be defined. The file can also be named differently. If changed, you will have to adjust the constant `WORLD_SETUP_FILE` in the `KaraWorld` class.

A world setup file may contain multiple worlds. Each world must start with the following three lines:

```
World: [Your title]
X: [Width of the world]
Y: [Height of the world]
[Actors]
```

Actors are represented as follows:

- Tree by #
- Kara by @
- Leaf by .
- Mushroom by \$
- A mushroom on a leaf by *
- Kara on a leaf by +

Tipp: Create the world inside the Greenfoot editor and use right-click on the world | `saveWorldSetupToFile()` or `printWorldSetupToConsole()` to save the created world.

Screen Output and User Input

There are several ways to interact with the user through Greenfoot over Input/Output:

- `System.out.println(...)` will write something on the console.
- With a Swing dialog (e.g. JOptionPane): This is used in **KaraIO** (Chapter 5).
- Drawing labels: This is the most complex version but also elegant as the input and output appears directly on the world and not in a pop-up dialog. An example can be found in **Kara Sokoban**.

Sharing a Scenario with others (Deployment)

With Greenfoot, scenarios can easily be exported and shared with others. There are three ways to this:

- Upload the scenario on the Greenfoot Gallery (<http://greenfootgallery.org>). There it can be directly started as an applet. Even the highscore of the Sokoban game should work.
- Export the scenario as a runnable jar application.
- Create a custom website with the scenario.

Known Bug – the World Disappears

It is possible that Greenfoot blocks and the world of Kara is no longer drawn. Even recompiling or pressing reset does not help.

Reason: This happens if the program remains too long inside the `act` method and then the user tries to interrupt the program. This is a problem that is known for Greenfoot but it is difficult to solve. Indeed, it is hardly possible in Java to stop a running method from the outside.

Solution: Exit and restart Greenfoot. Alternatively, you can open the debugger and click on "Terminate". This exits the scenario and automatically starts it again.

Recommended Books and Additional Links

I recommend taking a look at the book by Michael Kölling "Introduction to Programming with Greenfoot". It can either be used as inspiration for the teacher or as a textbook for the whole class.

Our Education Blog (where new versions of GreenfootKara are announced)

- <http://edu.makery.ch>

Links for Kara and Greenfoot:

- GreenfootKara: <http://www.swisseduc.ch/informatik/karatojava/greenfootkara/>
- GameGridKara: <http://www.swisseduc.ch/informatik/karatojava/gamegridkara/>
- Worksheets and exercises for Kara: <http://www.swisseduc.ch/informatik/karatojava/javakara/material/>
- Main Website for Greenfoot: <http://www.greenfoot.org/>
- Greenfoot forum for teachers: <http://greenroom.greenfoot.org>