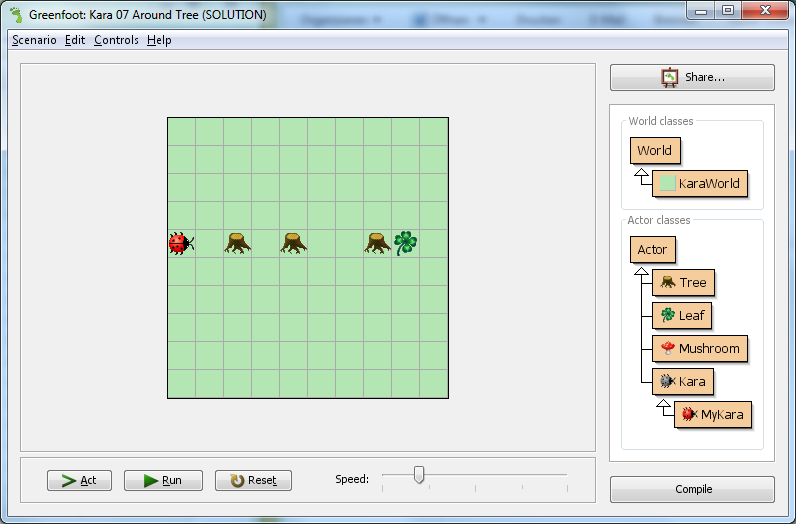
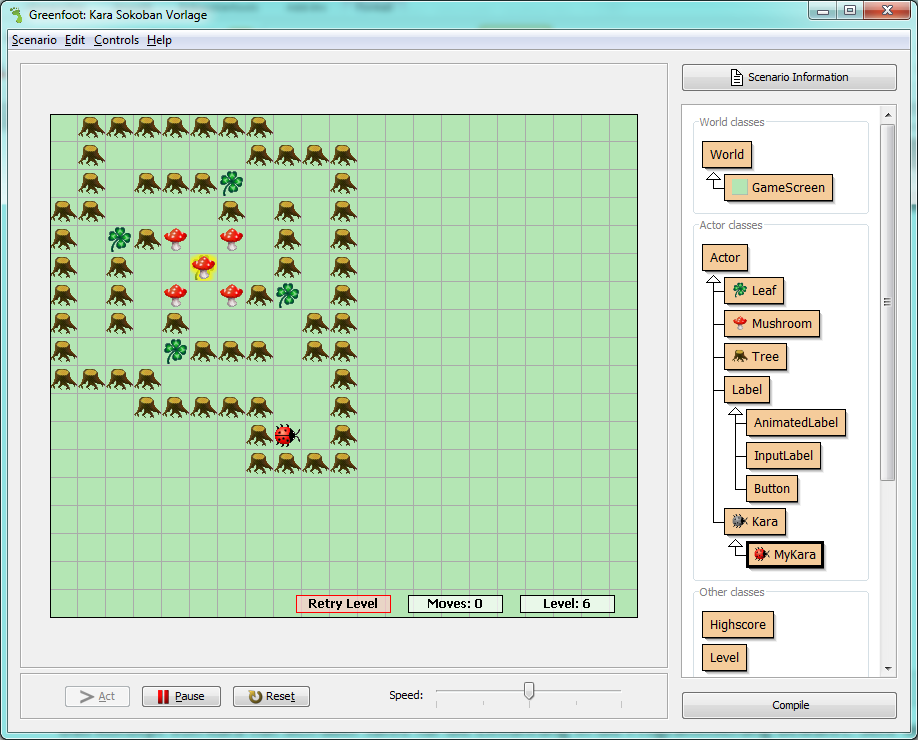
Instructions for Teachers

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 leafs)**

# 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** is such a scenario. It provides the functionality of Kara such as move(), putLeaf() onLeaf(), etc.

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.**

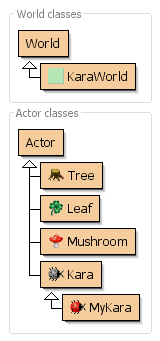
# Instructions for GreenfootKara

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 **schenarios-chapter-1**, **scenario-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 (siehe ***Base Scenario – Kara***):

|  |  |
| --- | --- |
| **Actions** | |
| move() | Kara makes a step in the current direction |
| turnLeft() | Kara turns left by 90 degrees |
| turnRight() | Kara turns right by 90 degrees |
| putLeaf() | Kara puts down a leaf |
| removeLeaf() | Kara picks up a leaf |

|  |  |
| --- | --- |
| **Sensors** | |
| onLeaf() | Kara checks if he stands on a leaf |
| treeFront() | Kara checks if there is a tree in front of him |
| treeLeft() | Kara checks if there is a tree on his left side |
| treeRight() | Kara checks if there is a tree on his right side |
| mushroomFront() | 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 (see ***Base Scenario – Kara Sokoban***).

In Chapter 5 Kara has a few additional methods to show messages and to ask the user for input (see ***Base Scenario – Kara with IO Methods***).

### 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 accessed 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 the 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).

## Keeping all Objects in the World

* The objects that were placed in the world can be saved so that they don’t disappear after every compile. This can be done by *right click on the world | Save the World*. This will automatically generate a method in KaraWorld which instantiates all the objects.

## Basic Settings oft KaraWorld

* - In the class **KaraWorld** the following settings can be made:
  + Number of horizontal cells
  + Number of vertical cells
  + Size of the cells (note: when changing the cell size, the images might need to be adjusted)
  + The initial speed

## 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 ***Base Scenario – Kara with IO Methods***.
* 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 base scenario **Base Scenario – 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.
* Export the scenario as a runnable jar application.
* Create a custom website with the scenario.

# Known Bugs

## 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.

**Links for Kara:**

* Main Website for Kara: <http://www.swisseduc.ch/informatik/karatojava/>
* Worksheets and good exercises for Kara: <http://www.swisseduc.ch/informatik/karatojava/javakara/material/>

**Links for Greenfoot:**

* Main Website for Greenfoot: <http://www.greenfoot.org/>
* Greenfoot forum for teachers: <http://greenroom.greenfoot.org>