Übung: Set- und Get-Methoden

Entwerfen Sie ein Spiel, in dem Sie einen Frosch steuern (von links nach rechts). Von oben sollen zufallsgesteuert verschiedene Obst-Objekte fallen. Jedes Obst-Objekt hat eine bestimmte Energie in Kilokalorien. Diese Energie soll beim Erzeugen des Objekts zufallsgesteuert festgelegt werden (zwischen 1 und 25 Kilokalorien).

Ein Bild, das Text, Screenshot enthält.

Automatisch generierte Beschreibung

|  |  |
| --- | --- |
| **1.** | Ihre Aufgabe: Die Fruit-Objekte sollen eine bestimmte Menge Energie erhalten!   * Erstellen Sie in der Klasse Fruit ein Attribut *fruitEnergy*. * Entwerfen Sie den Konstruktor der Klasse *Fruit*. Beim Erzeugen eines Fruit-Objekts soll dieses Objekt zufallsgesteuert einen Energiewert zwischen 1 und 25 erhalten. |
| **2.** | Der Frosch soll in der Lage sein, die Energie der Fruit-Objekte aufzusammeln. Gehen Sie so vor:   * Geben Sie dem Frosch das Attribut *frogEnergy* und initialisieren Sie dieses mit 0! * Implementieren Sie in der *Fruit-Klasse* eine *setEnergy*- und eine *getEnergy*-Methode. |
| **3.** | Entwerfen Sie die Methode *checkCollisonWithFruit*! Diese soll folgendes tun:   * Wenn der Frosch mit einem Obst-Objekt zusammenstößt, soll er die Kilokalorien des Objekts „aufsammeln“. Das heißt, dass sein eigener Energievorrat entsprechend ansteigt. **Verwenden Sie hierzu die vorher programmierten set- und get-Methoden!** * Lassen Sie dies rechts oben auf dem Spielfeld ausgeben (mit der showText-Methode). * Der Energievorrat des Obst-Objekts soll auf 0 sinken. Entfernen Sie das Objekt dann aus der Welt. |
| **4.** | **Profi-Projekt**  Verändern Sie das Projekt so:  Bei jeder Berührung (also in jedem act-Zyklus) soll der Frosch nur 1 Kilokalorie fressen können. Je länger der Frosch also mit einer Frucht zusammenstößt, desto mehr Energie nimmt er auf (bis die Energie des Obstes aufgebraucht ist). |

import greenfoot.\*;

**public class Fruit extends Actor {**

private int fruitEnergy = 0;

**public Fruit() {**

**fruitEnergy = Greenfoot.getRandomNumber(25) + 1;**

**}**

public int getEnergy() {

return fruitEnergy;

}

public void setEnergy(int pEnergy) {

this.fruitEnergy = pEnergy;

}

public void act() {

setLocation(getX(), getY() + 5);

if (getY() > 590) {

this.getWorld().removeObject(this);

}

}

}

import greenfoot.\*;

public class Frog extends Actor {

**private int frogEnergy = 0;**

public void act() {

**this.getWorld().showText("Energie: " + frogEnergy, 520, 20);**

if (Greenfoot.isKeyDown("a")) {

move(-5);

}

if (Greenfoot.isKeyDown("d")) {

move(5);

}

**this.checkCollision();**

}

**public int getEnergy() {**

**return frogEnergy;**

**}**

**public void setEnergy(int energy) {**

**this.frogEnergy = energy;**

**}**

**private void checkCollision() {**

**Fruit myFruit = (Fruit)this.getOneIntersectingObject(Fruit.class);**

**if (myFruit != null && myFruit.getEnergy() > 0) {**

**frogEnergy = frogEnergy + 1;**

**myFruit.setEnergy(myFruit.getEnergy() - 1);**

**if (myFruit.getEnergy() == 0) {**

**getWorld().removeObject(myFruit);**

**}**

**}**

}

}