# Toon de rand van de wereld

Verander de **SnakeWorld** constructor**:**

public SnakeWorld()

{

super(25, 20, 32);

addObject(new Border(), 0, 0);

}

Tweede versie van de **SnakeWorld** constructor**:**

public SnakeWorld()

{

super(25, 20, 32);

for (int x = 0; x < getWidth(); x ++ ) {

addObject(new Border(), x, 0);

addObject(new Border(), x, getHeight()-1);

}

for (int y = 1; y < getHeight()-1; y ++) {

addObject(new Border(), 0, y);

addObject(new Border(), getWidth()-1, y);

}

}

# Teken Bobby's hoofd

Voeg de volgende eigenschap toe aan de klasse **SnakeWorld**:

private LinkedList<SnakeBody> snake = new LinkedList<SnakeBody>();

Voeg het volgende toe aan het begin van de klasse **SnakeWorld**:

import java.util.\*;

Voeg deze lijnen toe aan de **SnakeWorld** constructor:

SnakeBody body = new SnakeBody();

snake.add(body);

addObject(body, 2, 2);

# Bobby's bewegingen

Voeg de volgende lijnen toe aan de eigenschappen van de klasse **SnakeWorld**:

private int dx = 1;

private int dy = 0;

Voeg deze methode toe aan de klasse **SnakeWorld**:

public void act()

{

//on remplace l'image de la tête

SnakeBody head = snake.getLast();

head.setImage("tail.png");

//crée une nouvelle tête

SnakeBody newHead = new SnakeBody();

int newHeadX = head.getX()+dx;

int newHeadY = head.getY()+dy;

//ajoute la nouvelle tête à la liste et au world

addObject(newHead, newHeadX, newHeadY);

snake.add(newHead);

}

# De lengte van de staart van de slang

Voeg de volgende eigenschap to aan de klasse **SnakeWorld**:

private int tailCounter = 5;

Voeg de volgende lijnen toe aan de methode **act**() van de klasse **SnakeWorld**:

if (tailCounter == 0) {

SnakeBody tail = snake.removeFirst();

removeObject(tail);

} else {

tailCounter--;

}

# Veranderen van richting

Voeg deze methode toe aan de klasse **'SnakeWorld'**:

private void changeDirection() {

if (Greenfoot.isKeyDown("left") && dx == 0 ) {

dx = -1;

dy = 0;

} else if (Greenfoot.isKeyDown("right") && dx == 0 ) {

dx = 1;

dy = 0;

} else if (Greenfoot.isKeyDown("down") && dy == 0 ) {

dx = 0;

dy = 1;

} else if (Greenfoot.isKeyDown("up") && dy == 0 ) {

dx = 0;

dy = -1;

}

}

Voeg de volgende lijn toe aan het begin van de methode **act**() van de klasse **SnakeWorld**:

changeDirection();

# Botsingen

Voeg de volgende eigenschap to aan de klasse **SnakeWorld**:

private boolean dead = false;

Voeg de volgende lijnen toe aan de methode **act**() van de klasse **SnakeWorld**:

if (dead) {

return;

}

Voeg deze methode toe aan de klasse **SnakeWorld**:

public void dead() {

dead = true;

}

Voeg de volgende lijnen toe aan de methode **act**() van de klasse **SnakeWorld** (pas op: kopiëer de code op de juiste plaats):

List<Block> blocks = getObjectsAt(newHeadX, newHeadY, Block.class);

for(Block block : blocks) {

block.collision(this);

}

Voeg deze method toe aan de klasse **Block**:

public void collision(SnakeWorld world) {

world.dead();

}

# Een appel toevoegen

Voeg de volgende lijnen toe aan de SnakeWorld constructor:

Apple apple = new Apple();

addObject(apple,

Greenfoot.getRandomNumber(getWidth()-2)+1,

Greenfoot.getRandomNumber(getHeight()-2)+1);

# Botsingen met een appel

Voeg deze methode toe aan de klasse **Apple**:

public void collision(SnakeWorld world) {

world.grow(2);

setLocation(

Greenfoot.getRandomNumber(getWorld().getWidth()-2)+1,

Greenfoot.getRandomNumber(getWorld().getHeight()-2)+1);

}

Voeg deze methode toe aan de klasse **SnakeWorld**:

public void grow(int i) {

tailCounter = tailCounter + i;

}

# Geluid toevoegen

Voeg de volgende lijn toe in de methode **collision**() van de **Apple** klasse:

Greenfoot.playSound("slurp.mp3");

Voeg de volgende lijn toe aan de methode **collision**() van de **Block** klasse:

Greenfoot.playSound("dead.mp3");