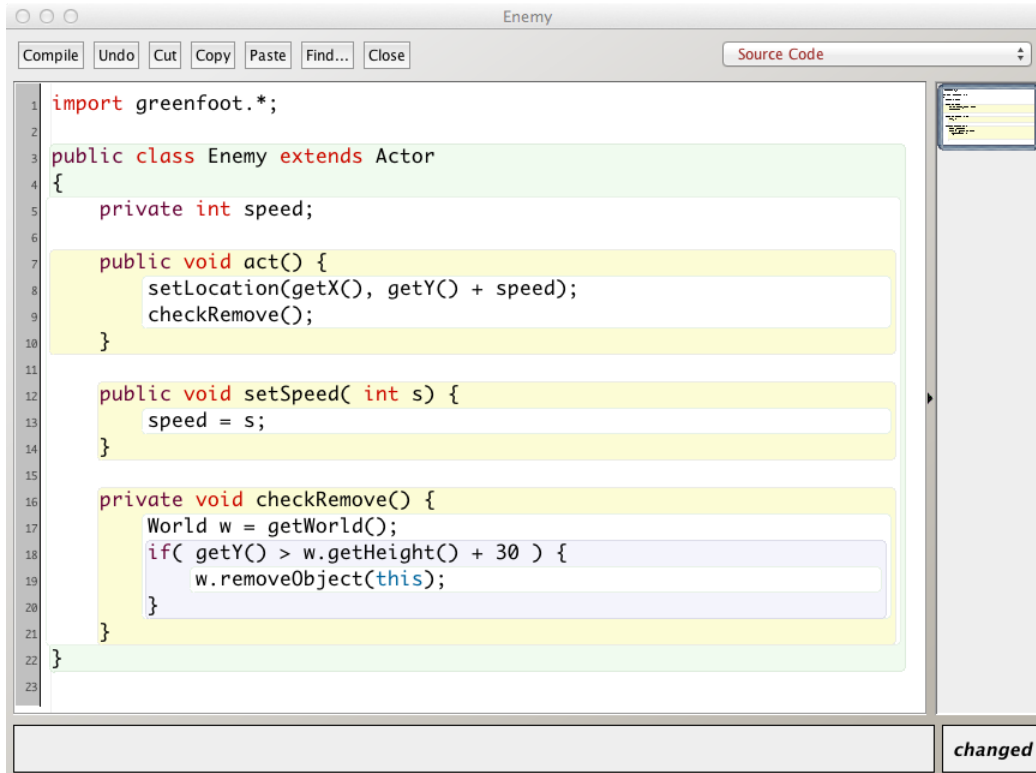


Chapter 1



The screenshot shows a code editor window titled "Enemy". The code defines a class `Enemy` that extends `Actor`. It includes a private `int` field `speed`, a public `act()` method that moves the actor and checks for removal, a public `setSpeed(int s)` method, and a private `checkRemove()` method that checks if the actor has fallen off the world.

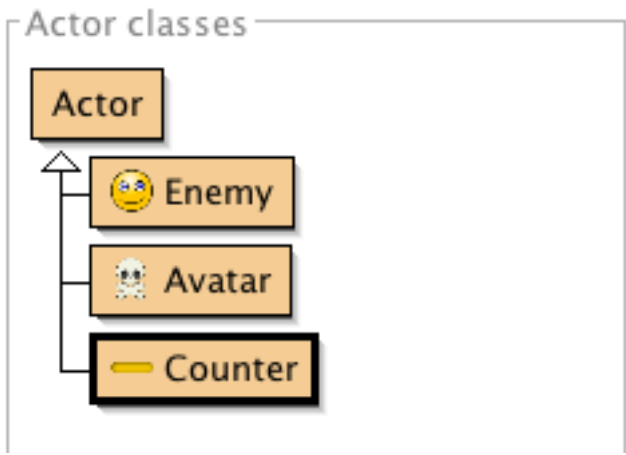
```
1 import greenfoot.*;
2
3 public class Enemy extends Actor
4 {
5     private int speed;
6
7     public void act() {
8         setLocation(getX(), getY() + speed);
9         checkRemove();
10    }
11
12    public void setSpeed( int s) {
13        speed = s;
14    }
15
16    private void checkRemove() {
17        World w = getWorld();
18        if( getY() > w.getHeight() + 30 ) {
19            w.removeObject(this);
20        }
21    }
22 }
23
```

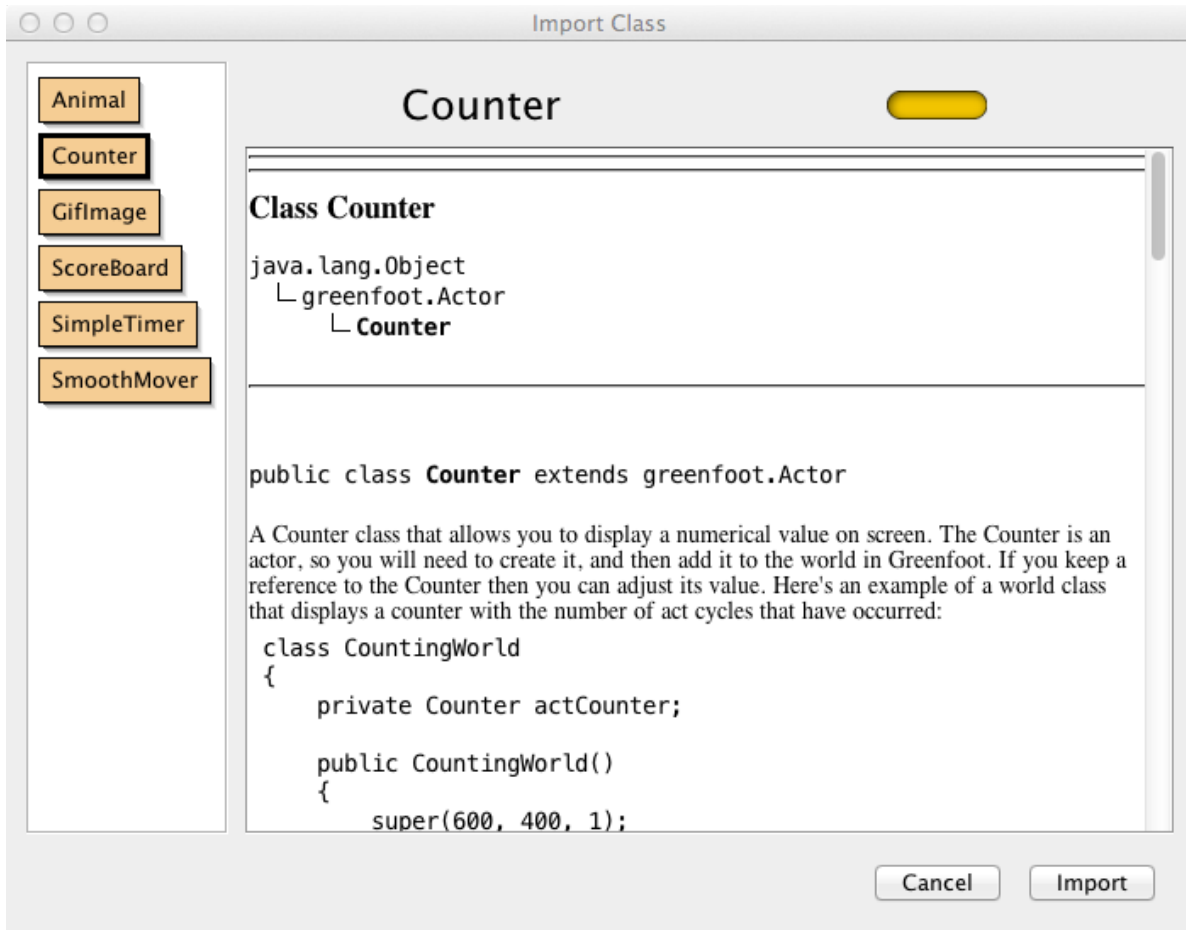
At the bottom right of the window, there is a status bar that says "changed".

```
2
3 public class AvoiderWorld extends World
4 {
5     private GreenfootSound bkgMusic;
6     private Counter scoreBoard;
7     private int enemySpawnRate = 20;
8     private int enemySpeed = 1;
9
```

```
2
3 public class AviderWorld extends World
4 {
5     private GreenfootSound bkgMusic;
6     private Counter scoreBoard;
7
8     public AviderWorld()
9     {
10         // Create a new world with 600x400 cell
11         super(600, 400, 1, false);

```





```
1 import greenfoot.*; // (World, Actor, GreenfootImage,
2
3 public class AvoiderWorld extends World
4 {
5     private GreenfootSound bkgMusic;
6
7     public AvoiderWorld()
8     {
9         // Create a new world with 600x400 cells with
10        super(600, 400, 1, false);
11
```

```
public AvoiderWorld()
{
    // Create a new world with 600x400 cells with a cell size of 1x1 pixels.
    super(600, 400, 1, false);

    // Initialize the music
    bkgMusic = new GreenfootSound("sounds/UFO_T-Balt.mp3");
    // Music Credit: http://www.newgrounds.com/audio/listen/504436 by T-balt
    bkgMusic.playLoop(); // Play the music

    prepare();
}
```

new AvoiderGameIntroScreen()

Open editor

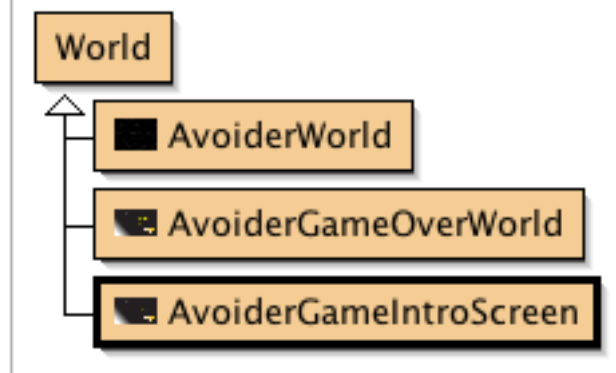
Set image...

Inspect

Remove

New subclass...

World classes

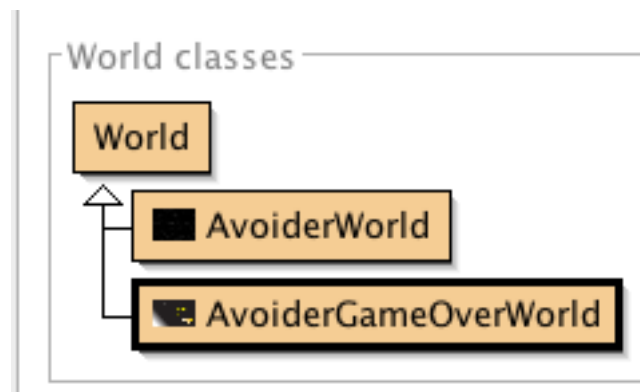


Avoider Game

Avoid stuff or die...



```
public void endGame() {  
    AvoiderGameOverWorld go = new AvoiderGameOverWorld();  
    Greenfoot.setWorld(go);  
}
```



Avoider Game

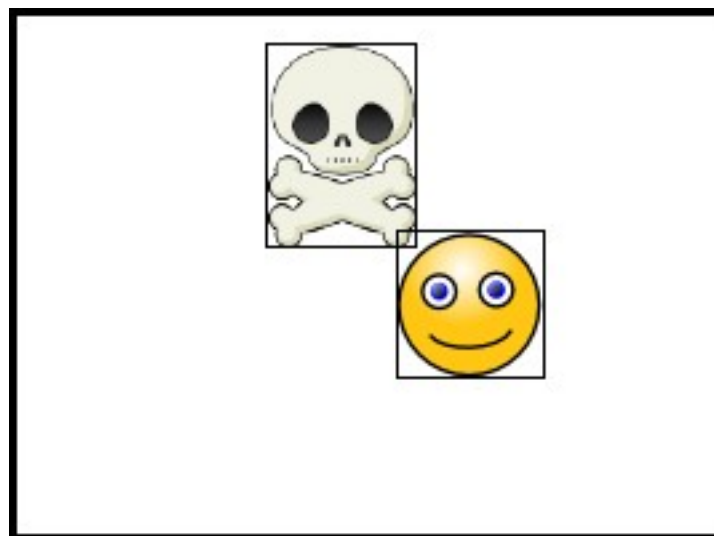
You didn't avoid stuff...

...so you died...

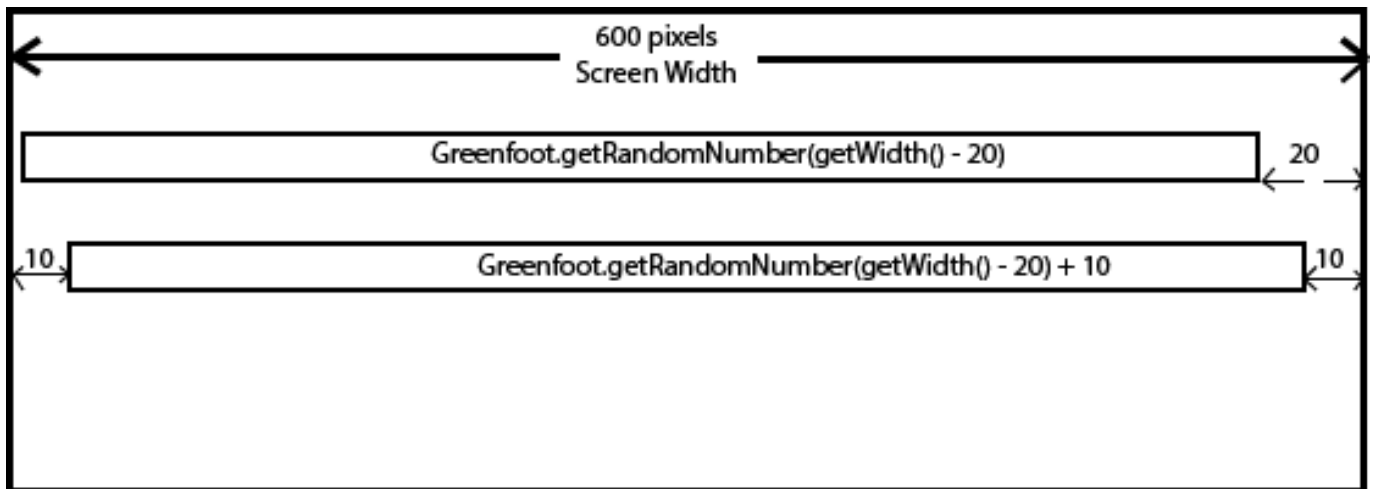


```
Avatar
Compile Undo Cut Copy Paste Find... Close Source Code
1 import greenfoot.*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)
2
3 public class Avatar extends Actor {
4
5     public void act() {
6         followMouse();
7         checkForCollisions();
8     }
9
10    private void checkForCollisions() {
11        Actor enemy = getOneIntersectingObject(Enemy.class);
12        if( enemy != null ) { // If not empty, we hit an Enemy
13            getWorld().removeObject(this);
14            Greenfoot.stop(); // For now, pause the World
15        }
16    }
17
18    private void followMouse() {
19        MouseInfo mi = Greenfoot.getMouseInfo();
20        // Check for null in case the mouse is off the screen
21        if( mi != null ) {
22            setLocation(mi.getX(), mi.getY());
23        }
24    }
25 }
26
```

changed



```
Enemy
Compile Undo Cut Copy Paste Find... Close Source Code
8 */
9 public class Enemy extends Actor
10 {
11     /**
12     * Act - do whatever the Enemy wants to do. This method is called whenever
13     * the 'Act' or 'Run' button gets pressed in the environment.
14     */
15     public void act() {
16         setLocation(getX(), getY() + 1);
17         checkRemove();
18     }
19
20     private void checkRemove() {
21         World w = getWorld();
22         if( getY() > w.getHeight() + 30 ) {
23             w.removeObject(this);
24         }
25     }
26 }
27
saved
```




```
9 public class Avatar extends Actor
10 {
11     /**
12     * Act - do whatever the Avatar wants to do. This method is called whenever
13     * the 'Act' or 'Run' button gets pressed in the environment.
14     */
15     public void act()
16     {
17         followMouse();
18     }
19
20     private void followMouse() {
21         MouseInfo mi = Greenfoot.getMouseInfo();
22         // Check for null in case the mouse is off the screen
23         if( mi != null ) {
24             setLocation(mi.getX(), mi.getY());
25         }
26     }
27 }
28
```

saved

```
import greenfoot.*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)
public class Avatar extends Actor
{
    /**
    * Act - do whatever the Avatar wants to do. This method is called whenever
    * the 'Act' or 'Run' button gets pressed in the environment.
    */
    public void act()
    {
        followMouse();
    }
}
```

cannot find symbol - method followMouse()

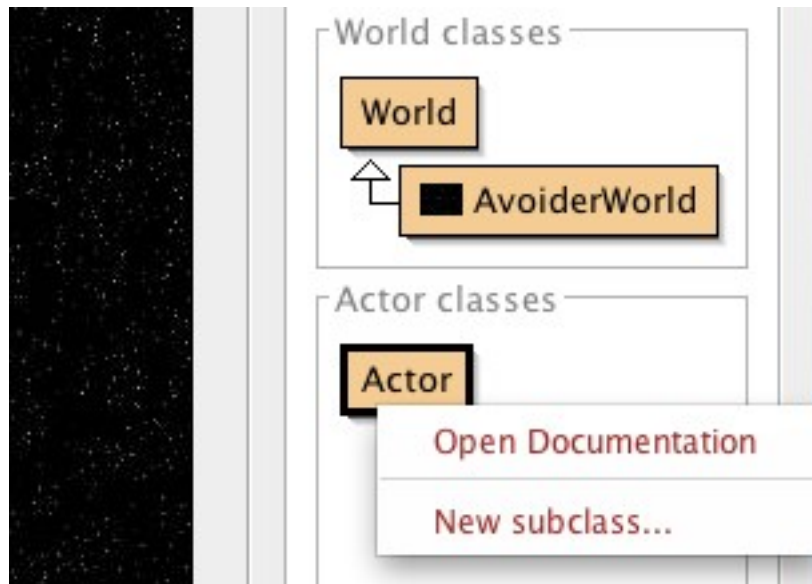
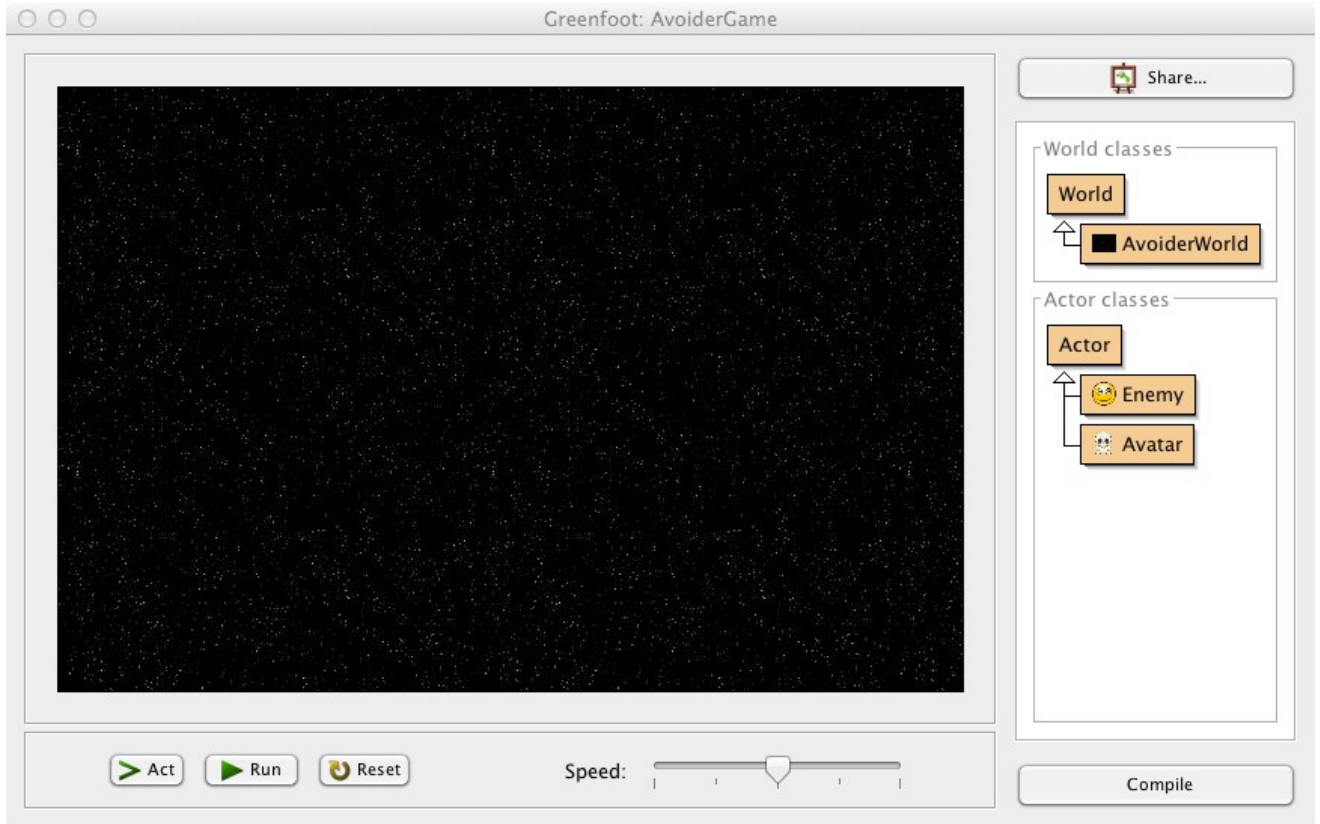
saved

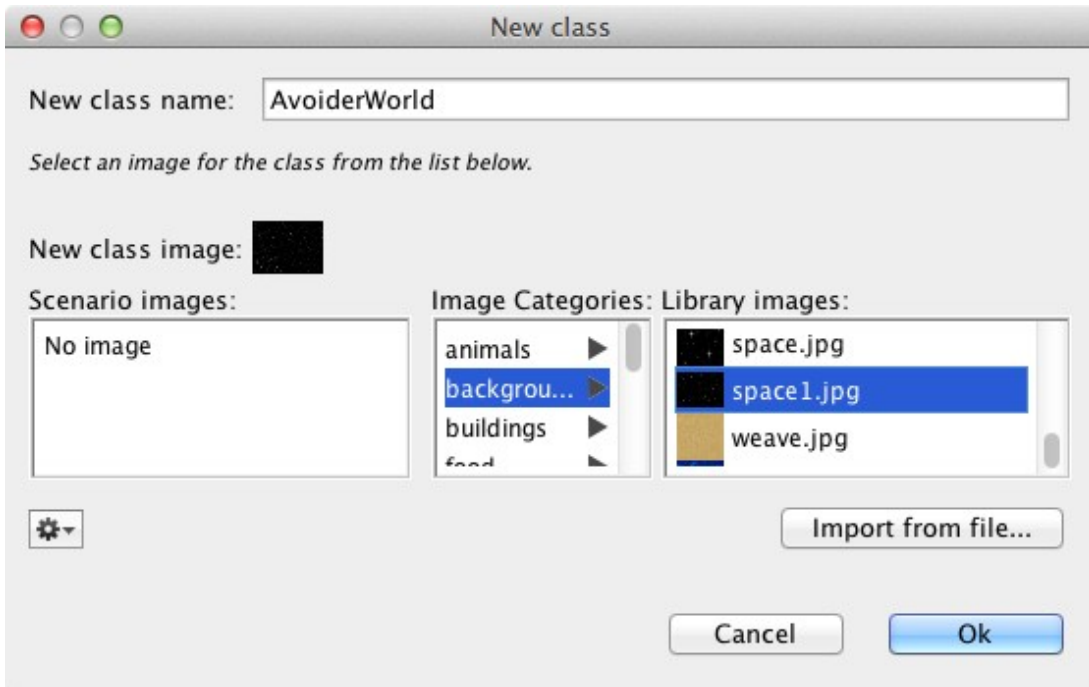
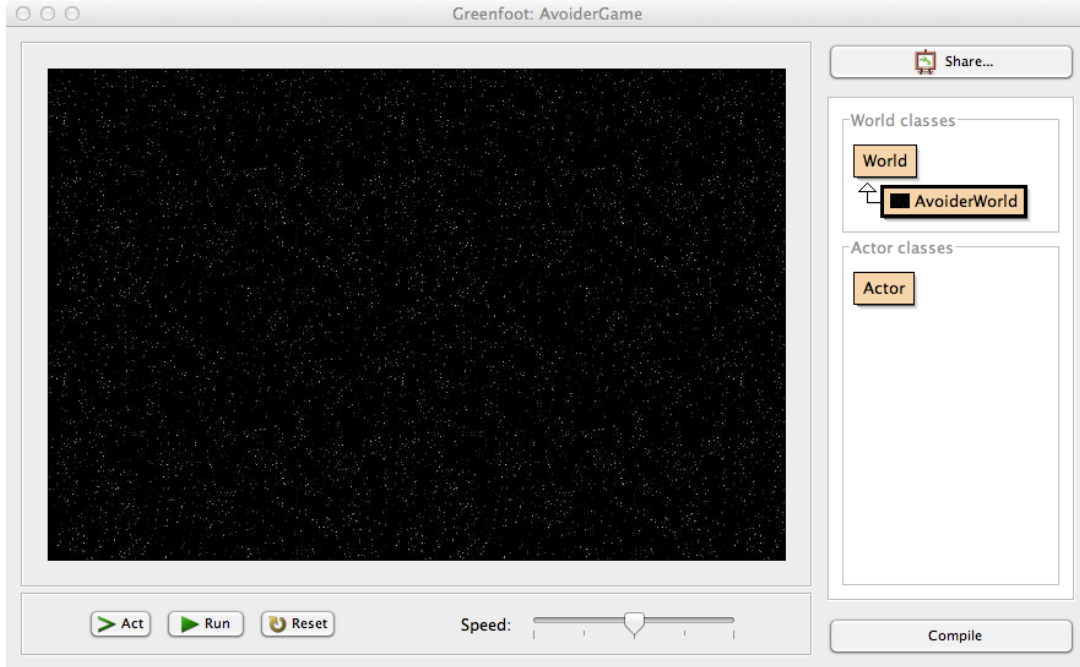
```
*/  
public void act()  
{  
    followMouse();  
}
```

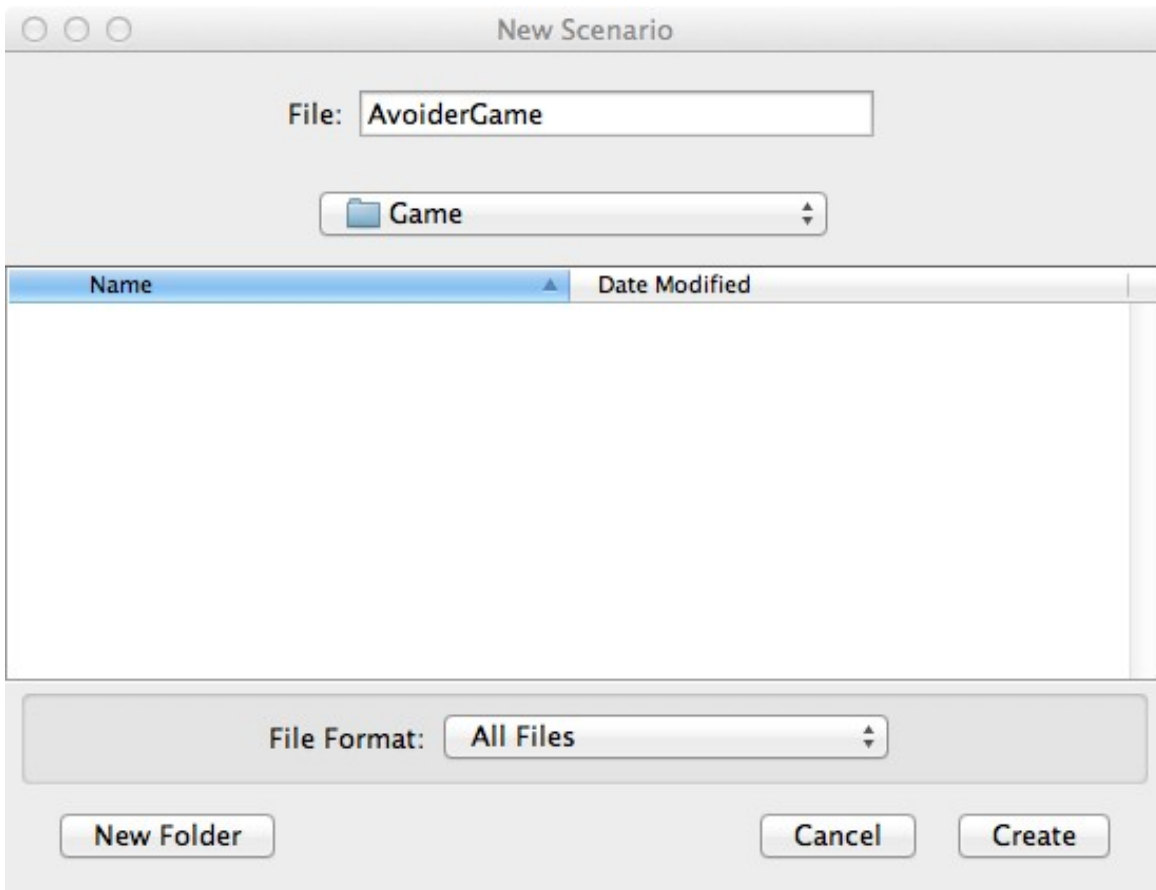
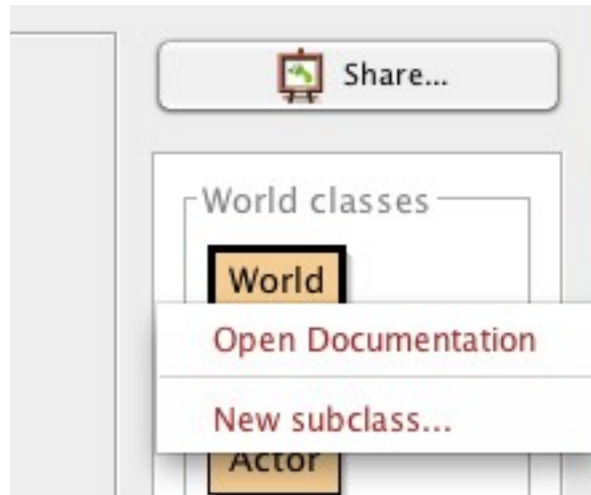
The screenshot shows a code editor window titled "Avatar". The window has a menu bar with "Compile", "Undo", "Cut", "Copy", "Paste", "Find...", and "Close". A "Source Code" dropdown menu is visible in the top right. The code is as follows:

```
1 import greenfoot.*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)  
2  
3 /**  
4  * Write a description of class Avatar here.  
5  *  
6  * @author (your name)  
7  * @version (a version number or a date)  
8  */  
9 public class Avatar extends Actor  
10 {  
11     /**  
12     * Act - do whatever the Avatar wants to do. This method is called whenever  
13     * the 'Act' or 'Run' button gets pressed in the environment.  
14     */  
15     public void act()  
16     {  
17         // Add your action code here.  
18     }  
19 }  
20
```

A "saved" button is located in the bottom right corner of the editor window.



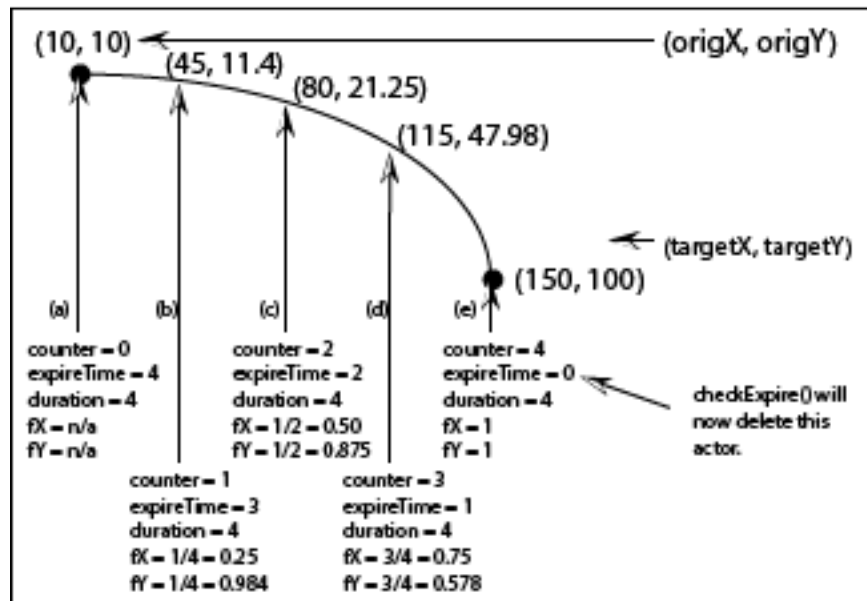




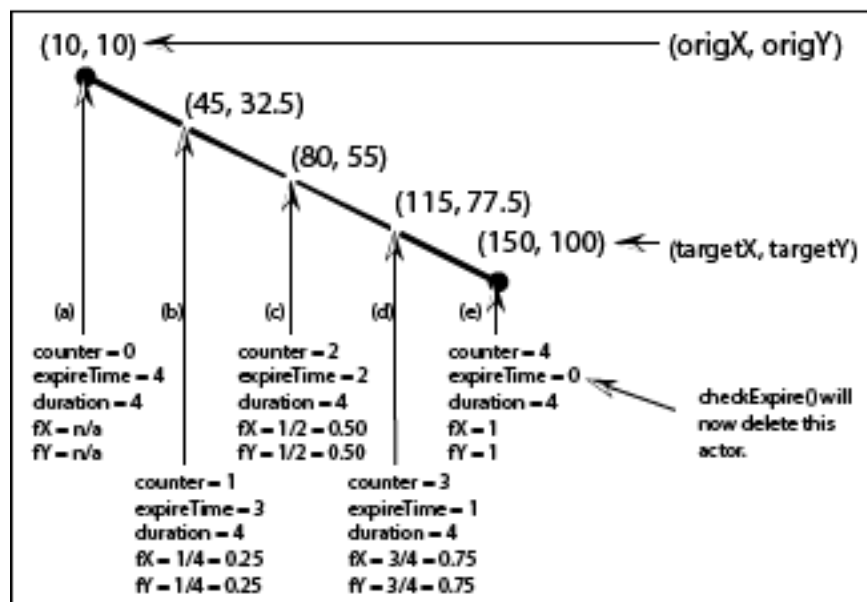
Chapter 2

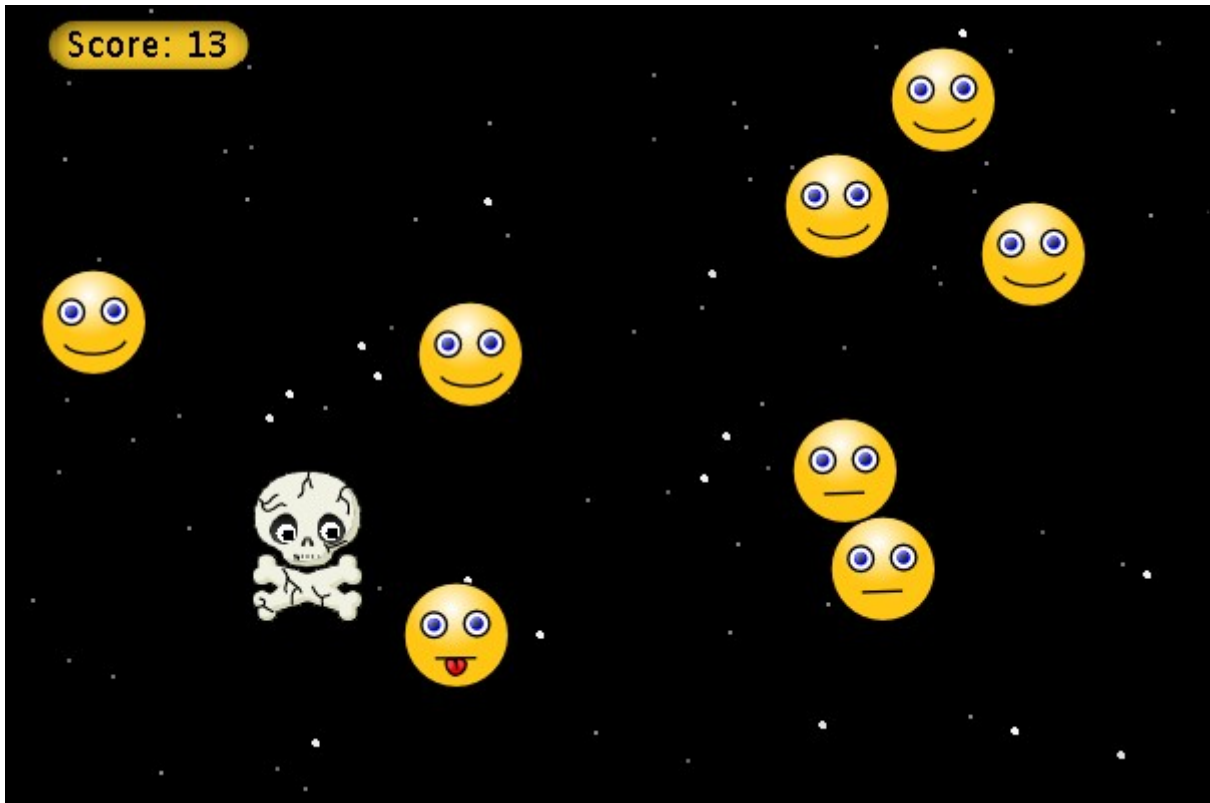
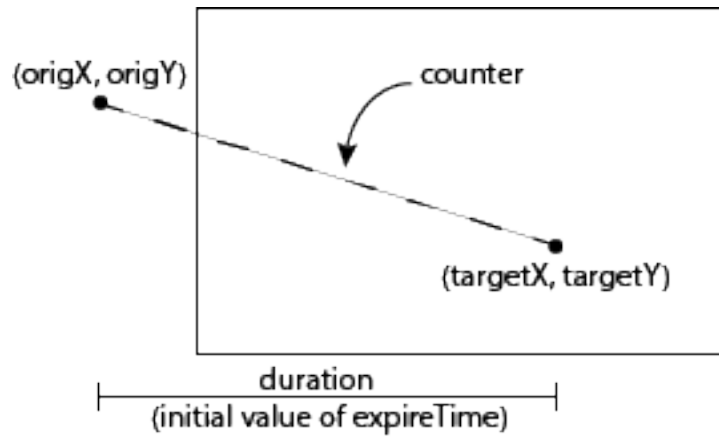


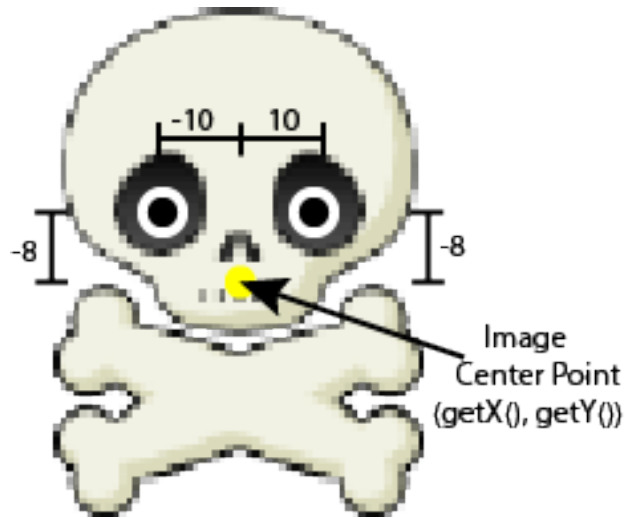
Exponential Easing Example

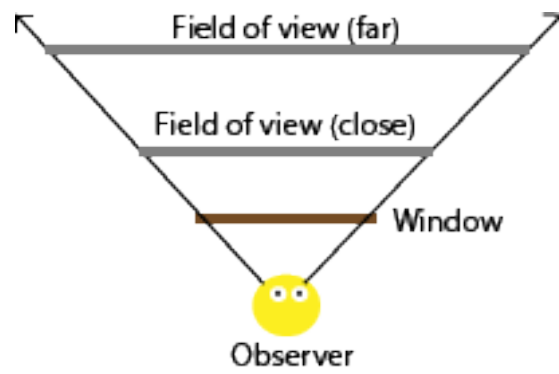
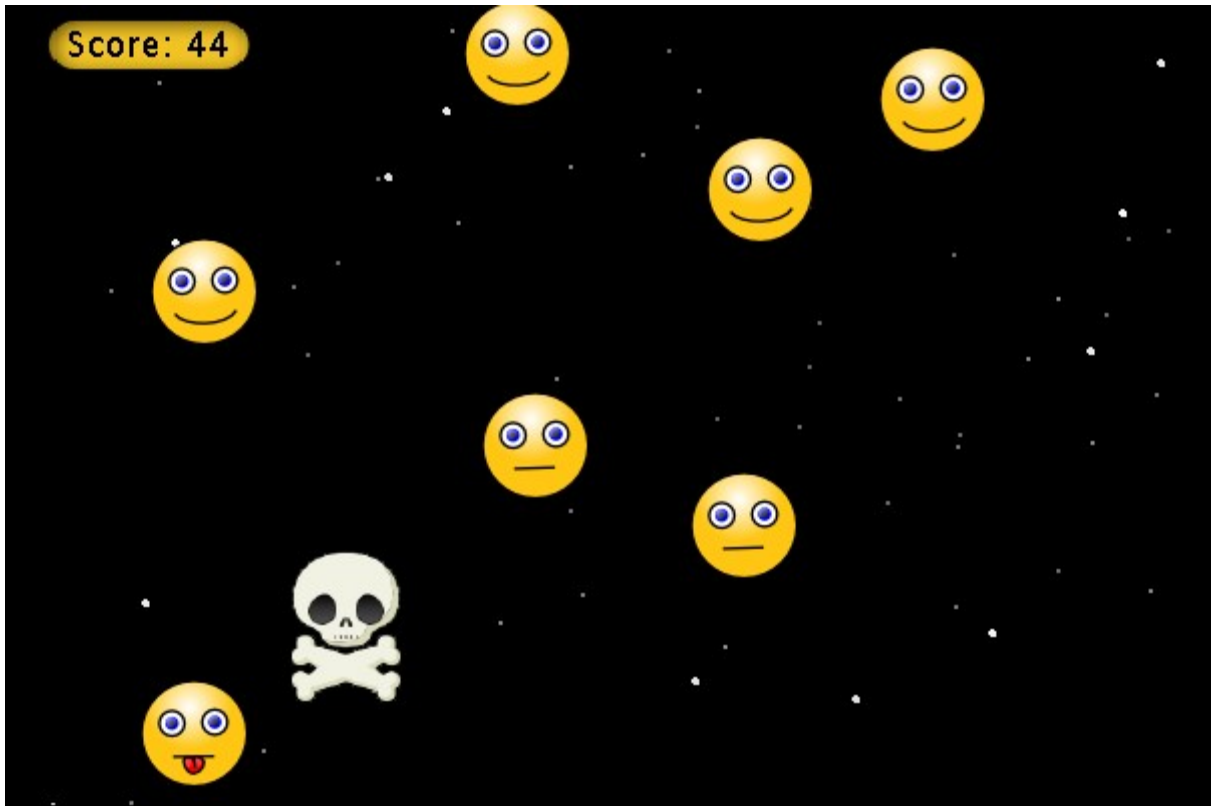


Linear Easing Example

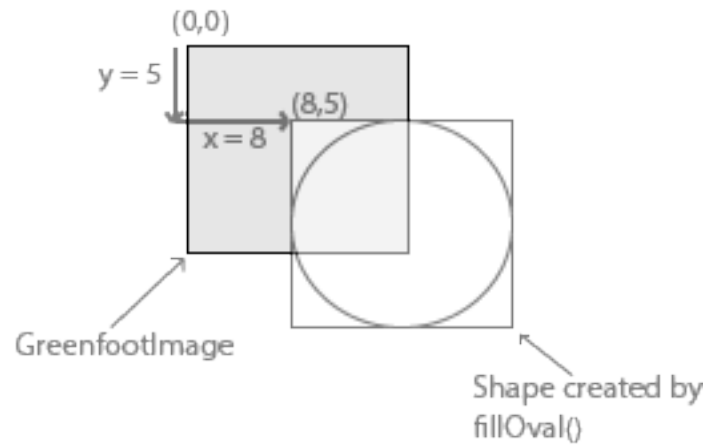


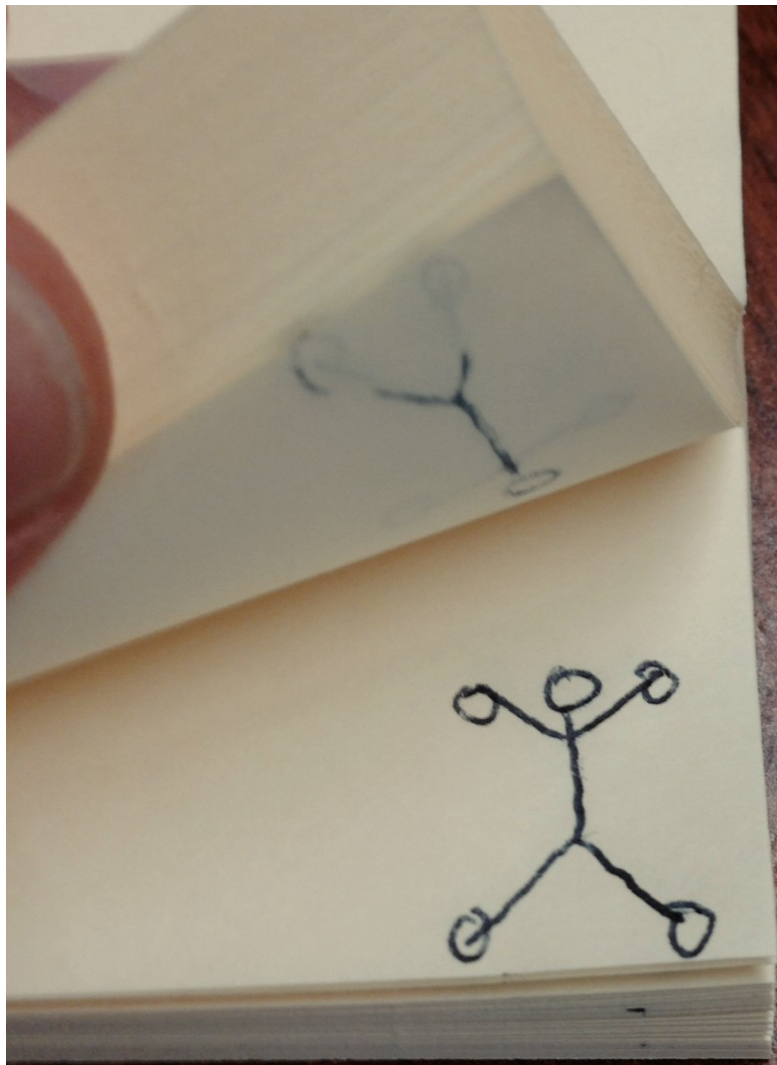
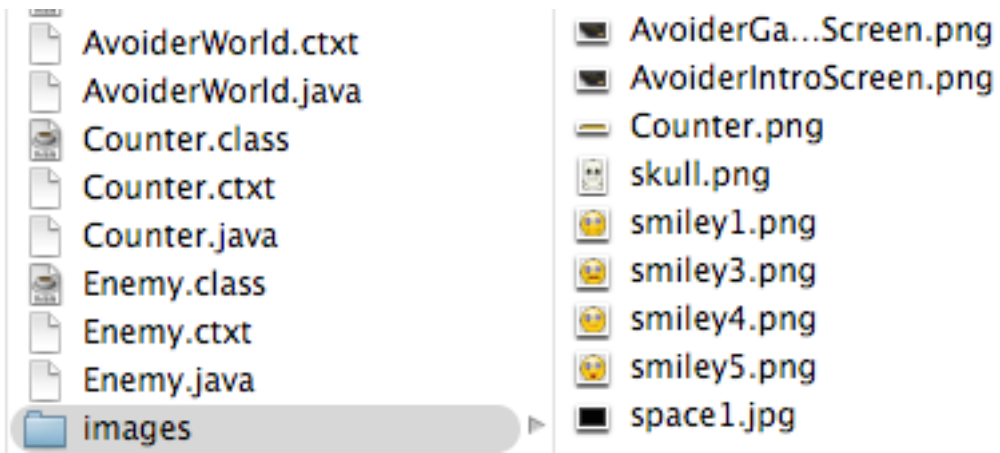


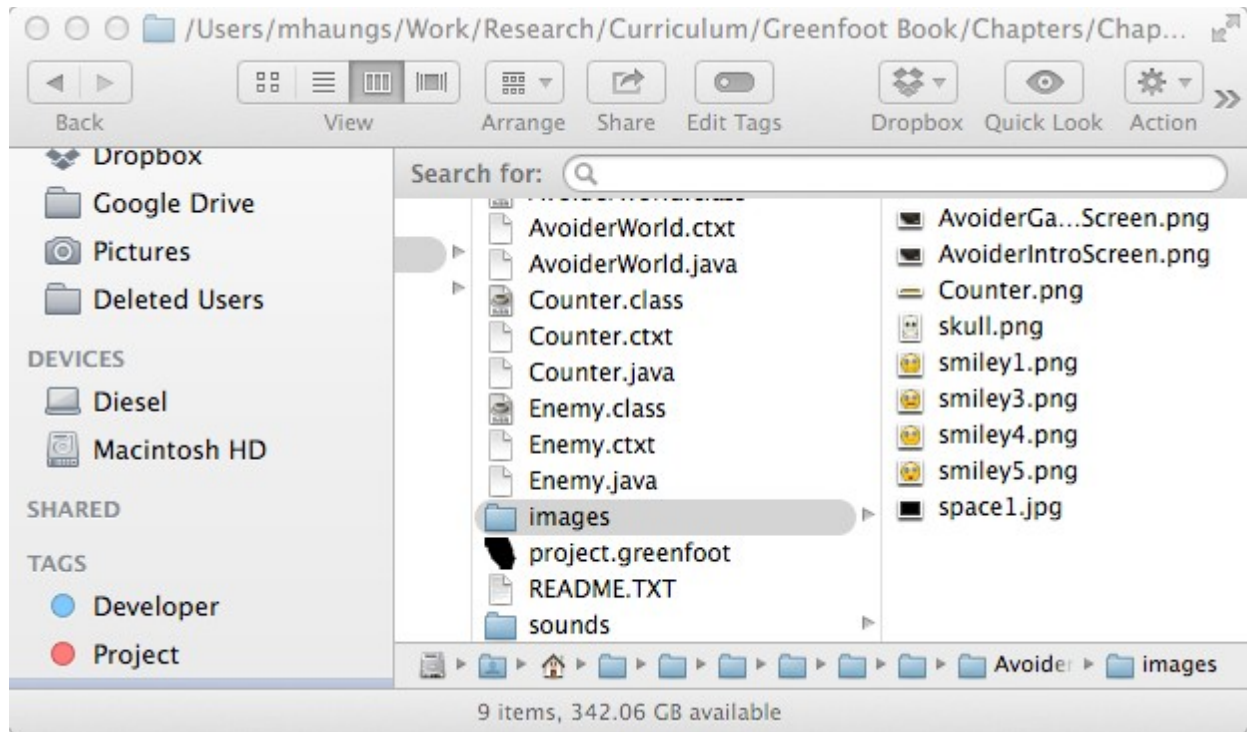




```
Star
Compile Undo Cut Copy Paste Find... Close Source Code
1 import greenfoot.*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)
2 import java.awt.Color;
3
4 public class Star extends Actor
5 {
6     int speed = 1;
7
8     public Star() {
9         GreenfootImage img = new GreenfootImage(10,10);
10        img.setColor(Color.white);
11        img.fillOval(0,0,10,10);
12        setImage(img);
13    }
14
15    public void act() {
16        setLocation(getX(), getY()+speed);
17        checkRemove();
18    }
19
20    public void setSpeed( int s ) {
21        speed = s;
22    }
23
24    private void checkRemove() {
25        World w = getWorld();
26        if( getY() > w.getHeight() + 30 ) {
27            w.removeObject(this);
28        }
29    }
30 }
changed
```







GreenfootImage (Greenfoot API)

file:///Applications/Greenfoot%202.3.0/ Reader

Greenfoot Home CalPoly PolyXpress Developer Notes Press This Recent

GreenfootImage (Greenfoot API)

All Classes

- [Actor](#)
- [Greenfoot](#)
- [GreenfootImage](#)
- [GreenfootSound](#)
- [MouseInfo](#)
- [UserInfo](#)
- [World](#)

Package **Class** **Tree** **Deprecated** **Index** **Help**

[PREV CLASS](#) [NEXT CLASS](#) [FRAMES](#) [NO FRAMES](#)
[SUMMARY: NESTED | FIELD | CONSTR | METHOD](#) [DETAIL: FIELD | CONSTR | METHOD](#)

greenfoot

Class GreenfootImage

java.lang.Object
 └─greenfoot.GreenfootImage

```
public class GreenfootImage
  extends java.lang.Object
```

An image to be shown on screen. The image may be loaded from an image file and/or drawn by using various drawing methods.

Version:
2.4

Author:
Poul Henriksen

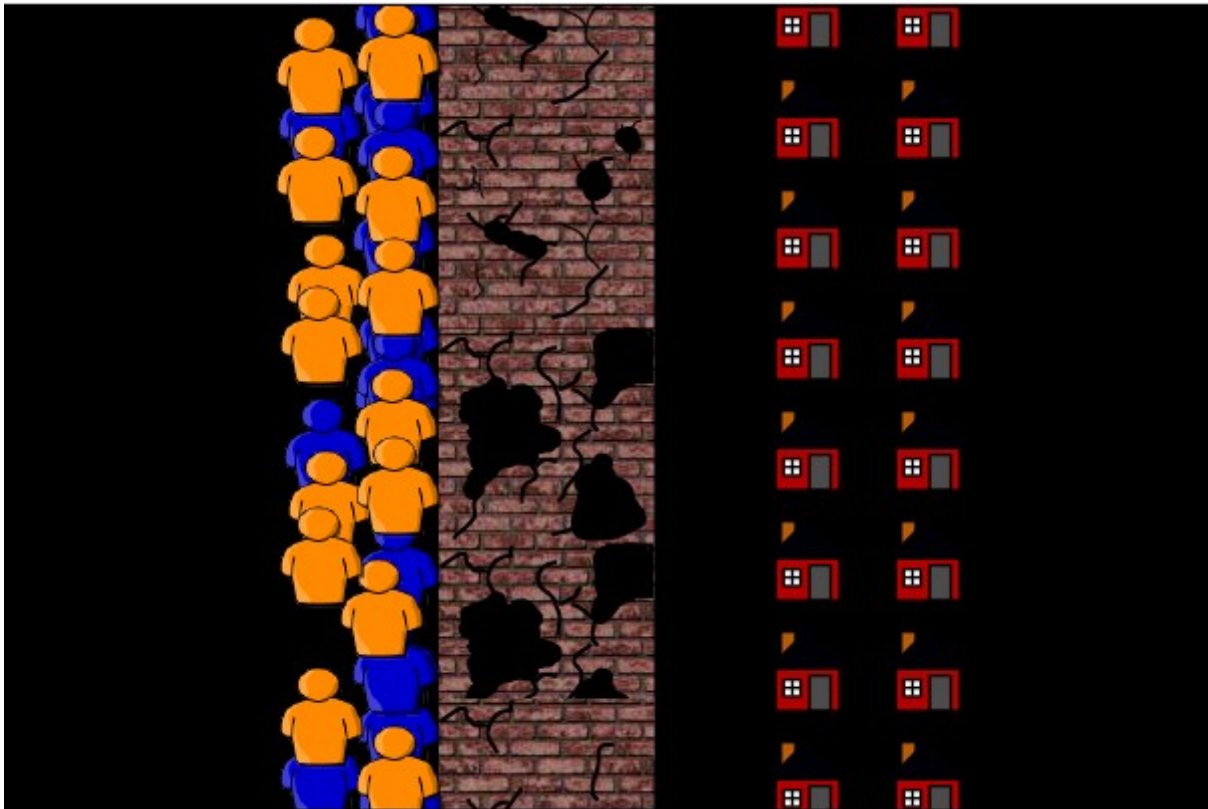
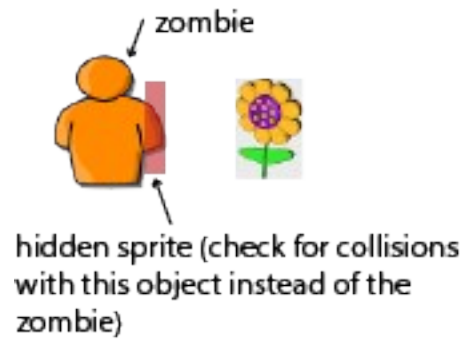
Constructor Summary

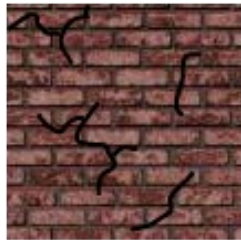
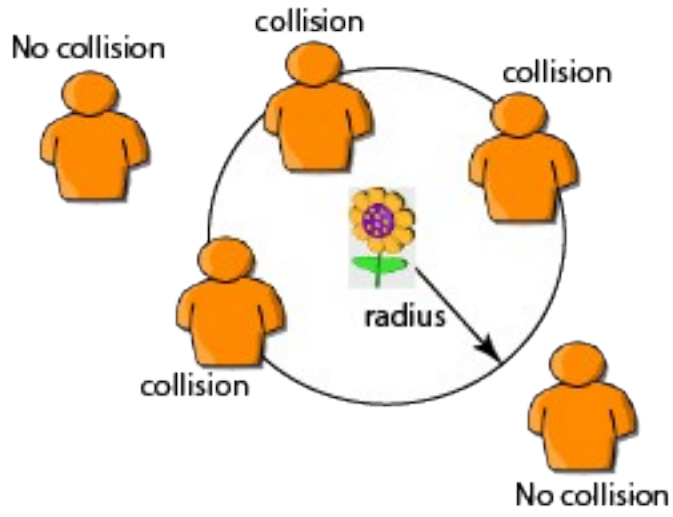
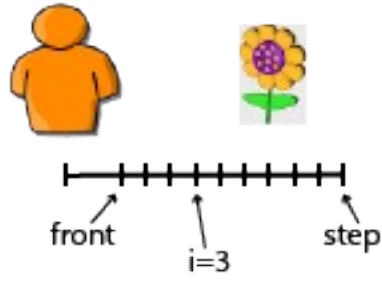
| | |
|--|--|
| GreenfootImage (GreenfootImage image) | Create a GreenfootImage from another GreenfootImage. |
| GreenfootImage (int width, int height) | Create an empty (transparent) image with the specified size. |

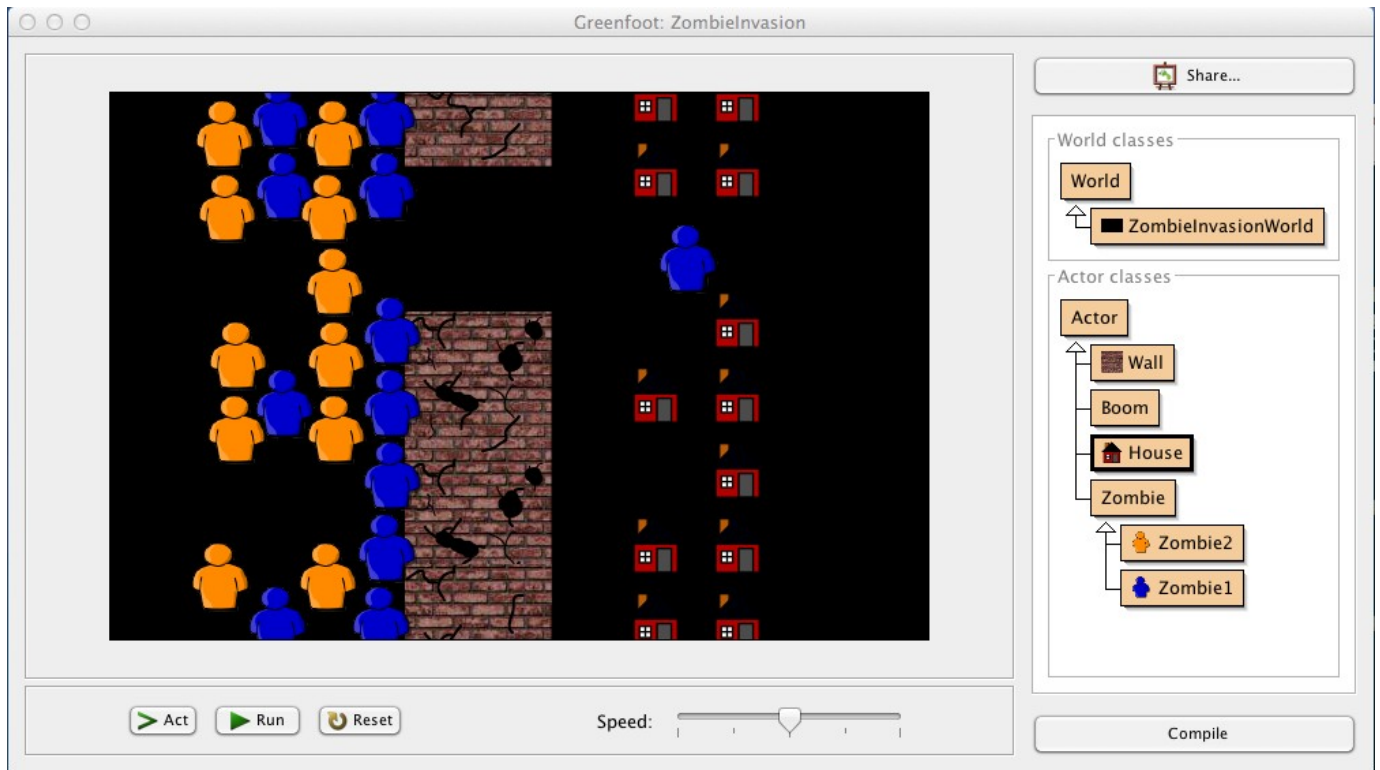
Package greenfoot

| Class Summary | |
|--------------------------------|---|
| Actor | An Actor is an object that exists in the Greenfoot world. |
| Greenfoot | This utility class provides methods to control the simulation and interact with the system. |
| GreenfootImage | An image to be shown on screen. |
| GreenfootSound | Represents audio that can be played in Greenfoot. |
| MouseInfo | This class contains information about the current status of the mouse. |
| UserInfo | The UserInfo class can be used to store data permanently on a server, and to share this data between different users, when the scenario runs on the Greenfoot web site. |
| World | World is the world that Actors live in. |

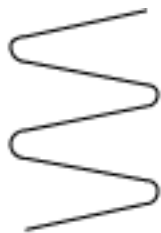
Chapter 3







(a) Sine Wave



(b) Sine Wave rotated 90 degrees



(c) Sine Wave Increasing Frequency



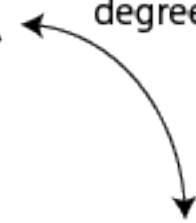
(d) Sine Wave Increasing Frequency Further

Chapter 4

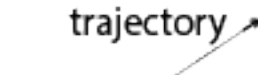
Turret



Rotate 30 degrees



future trajectory



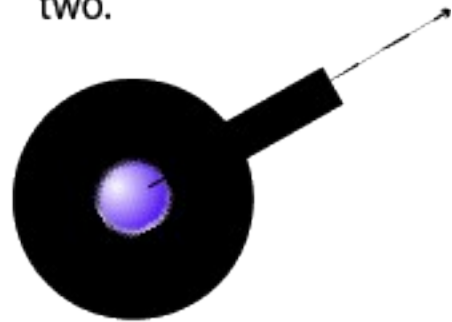
Ball



Rotate 30 degrees



Composing the two.



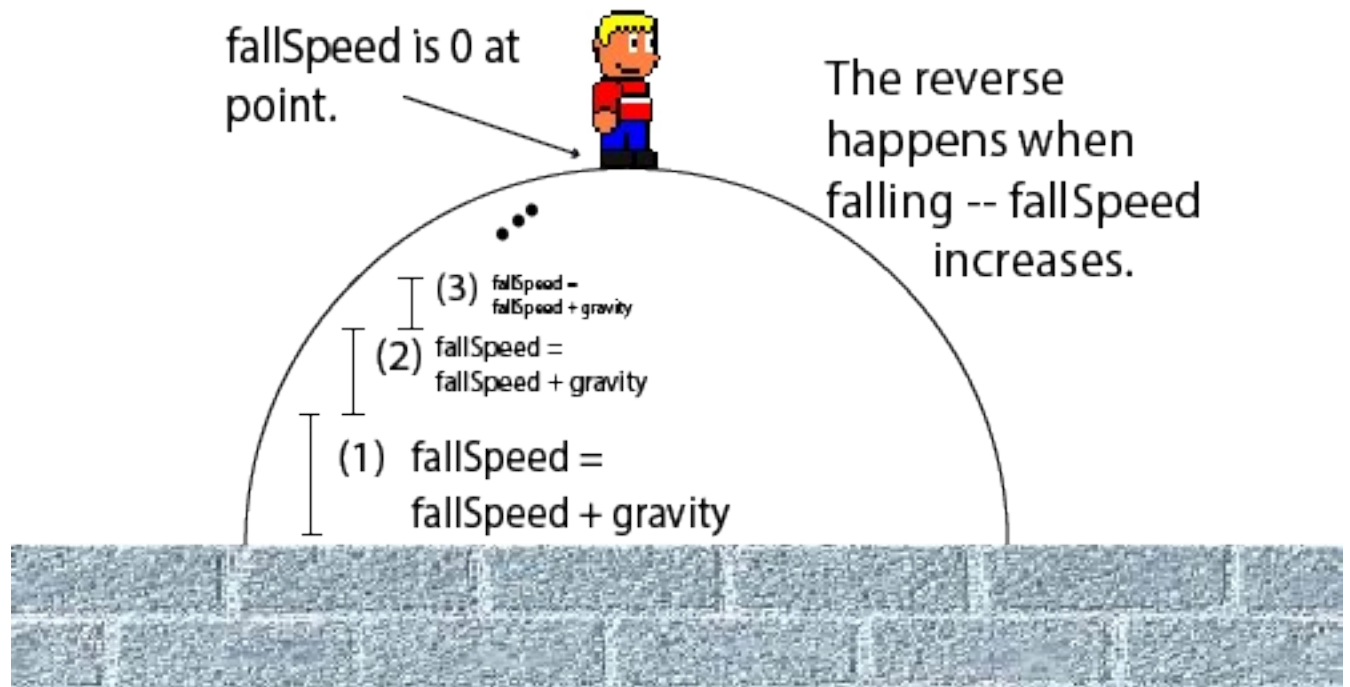
Point of impact

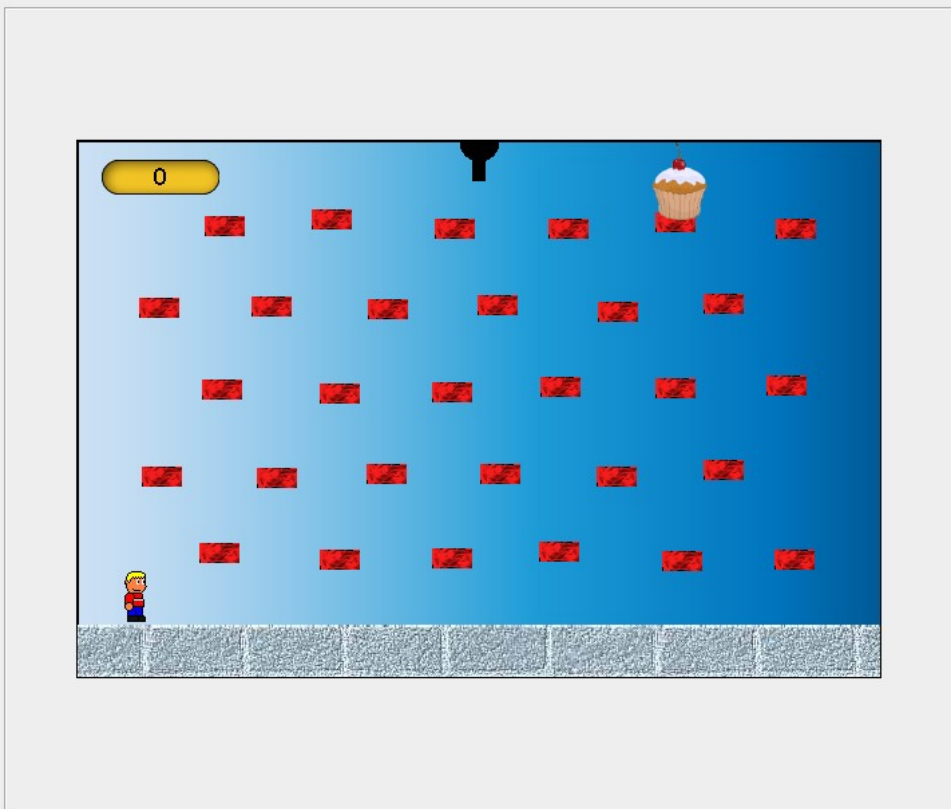


Phase 1: Distance is less than fallSpeed.



Phase 2: The distance is equal to the distance that was remaining in the jump, minus the loss of inertia.



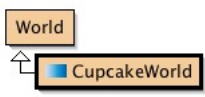


Act Run Reset

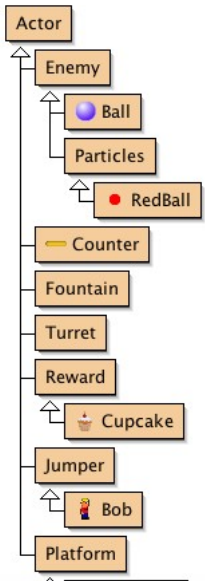
Speed:

Share...

World classes

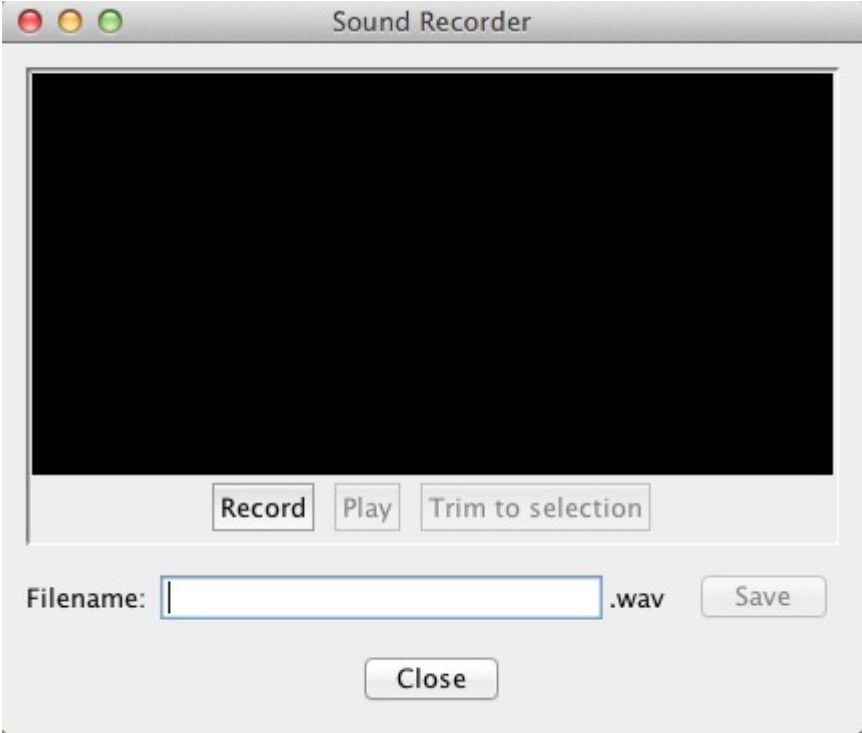


Actor classes



Compile

Chapter 5



Game:

Michael J. Williams, Original Actionscript Version
Michael Haungs, Greenfoot Version

Sound:

<http://www.newgrounds.com/audio/listen/504436> by T-balt

Images:

Default Greenfoot Images
Michael Haungs



Story:



This is a story about good vs evil. However, good looks evil and evil looks good. The sheer evil of the smiley-faced villains is enough to crack the bones of our skull-faced hero so you must avoid their touch at all cost. Other objects will appear on the screen. Avoid them if they look good.



You score more points the longer you survive the cheery horde.



Avoider Game

Avoid stuff or die...

Start Game
Credits Screen
Story Screen



Create new class

A class name cannot contain spaces or punctuation characters and should start with a capital letter.

New class name:

Cancel Ok

Avoider Game

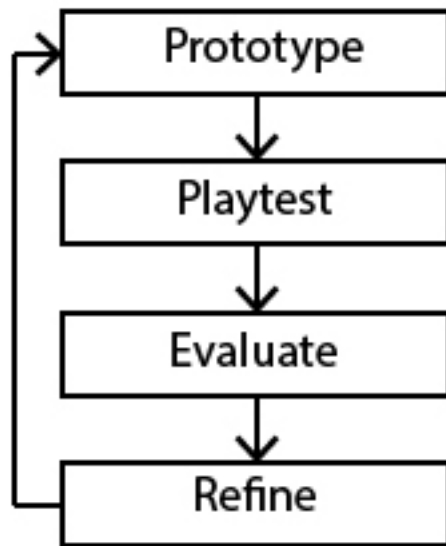
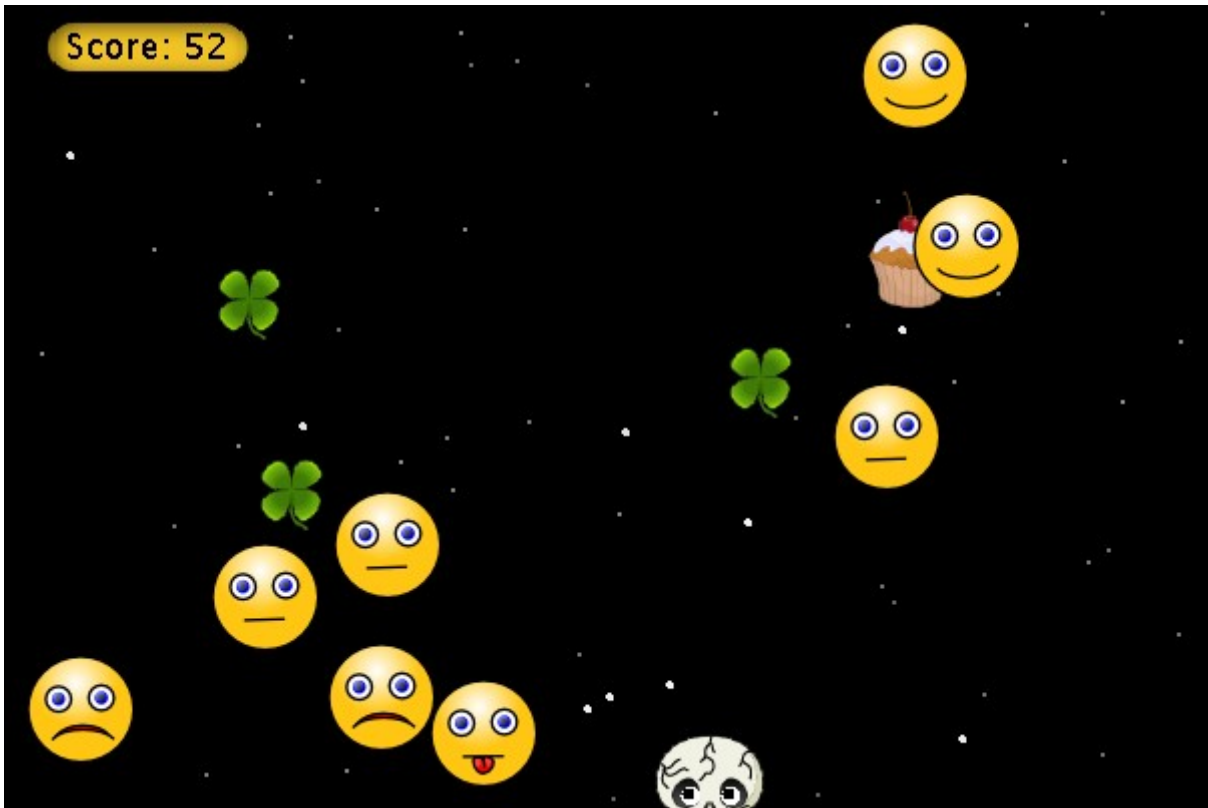
You didn't avoid stuff...

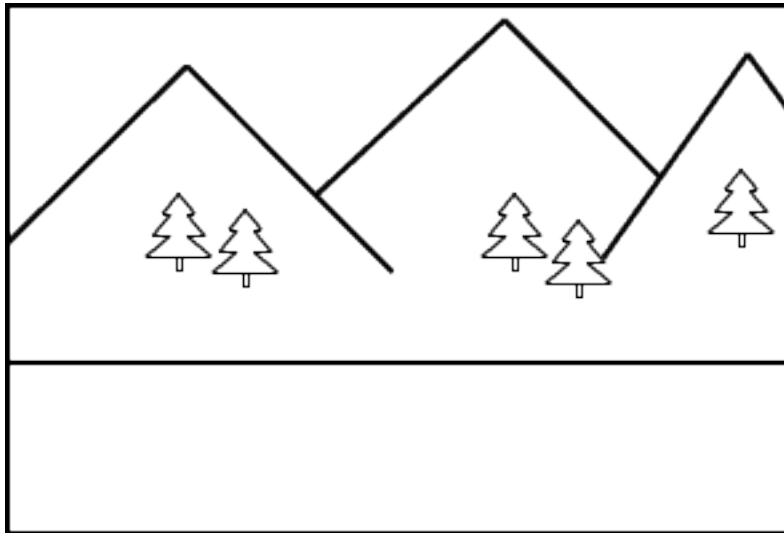
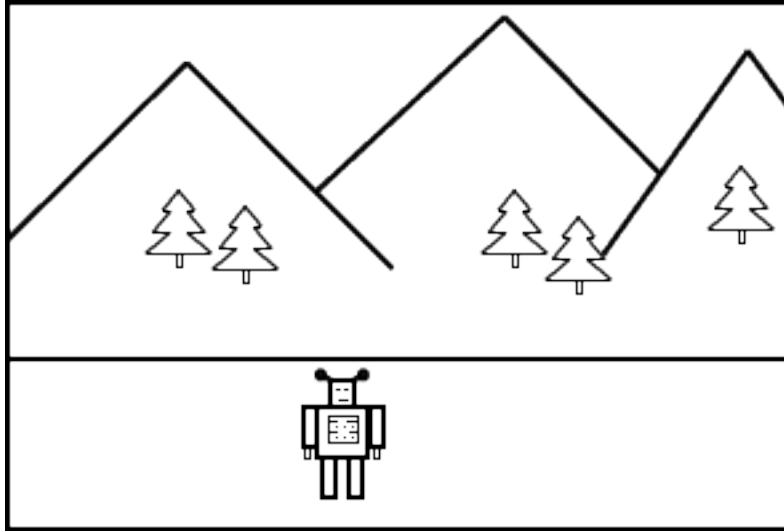
...so you died...

Your Score: 27

Your Best: 492

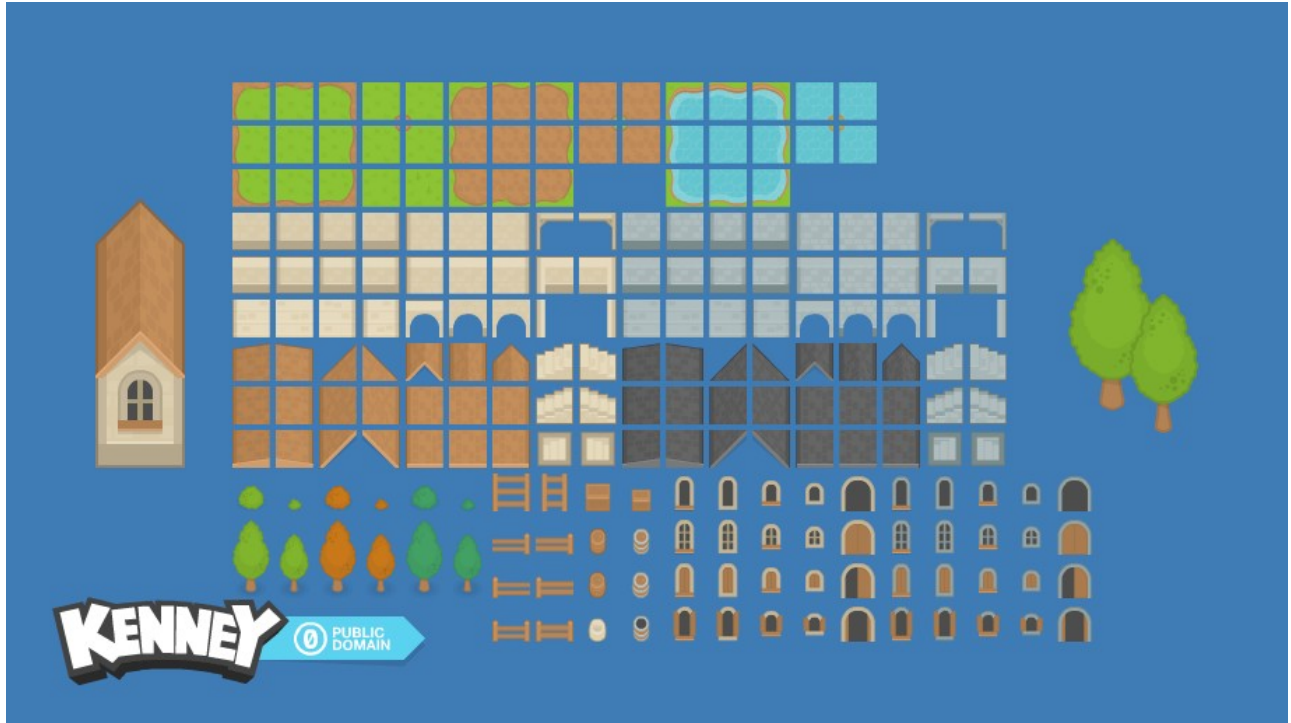


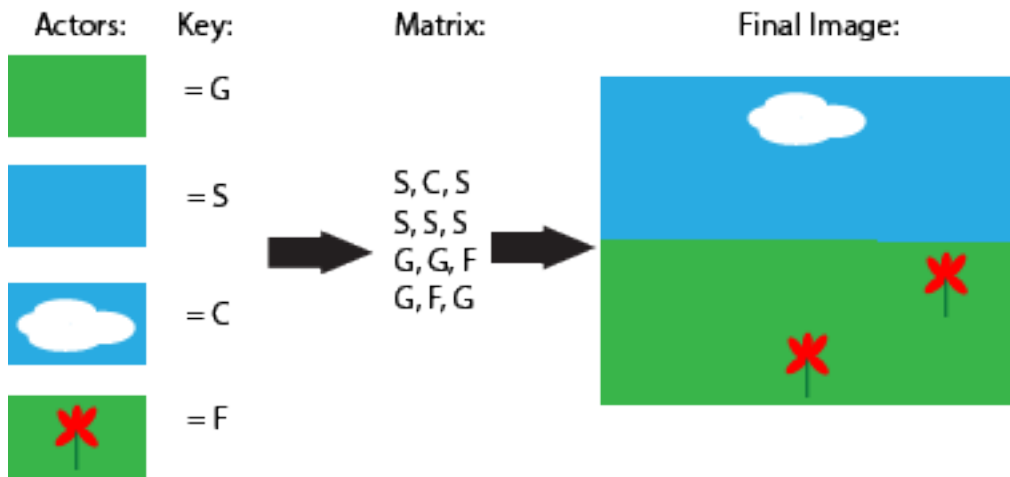
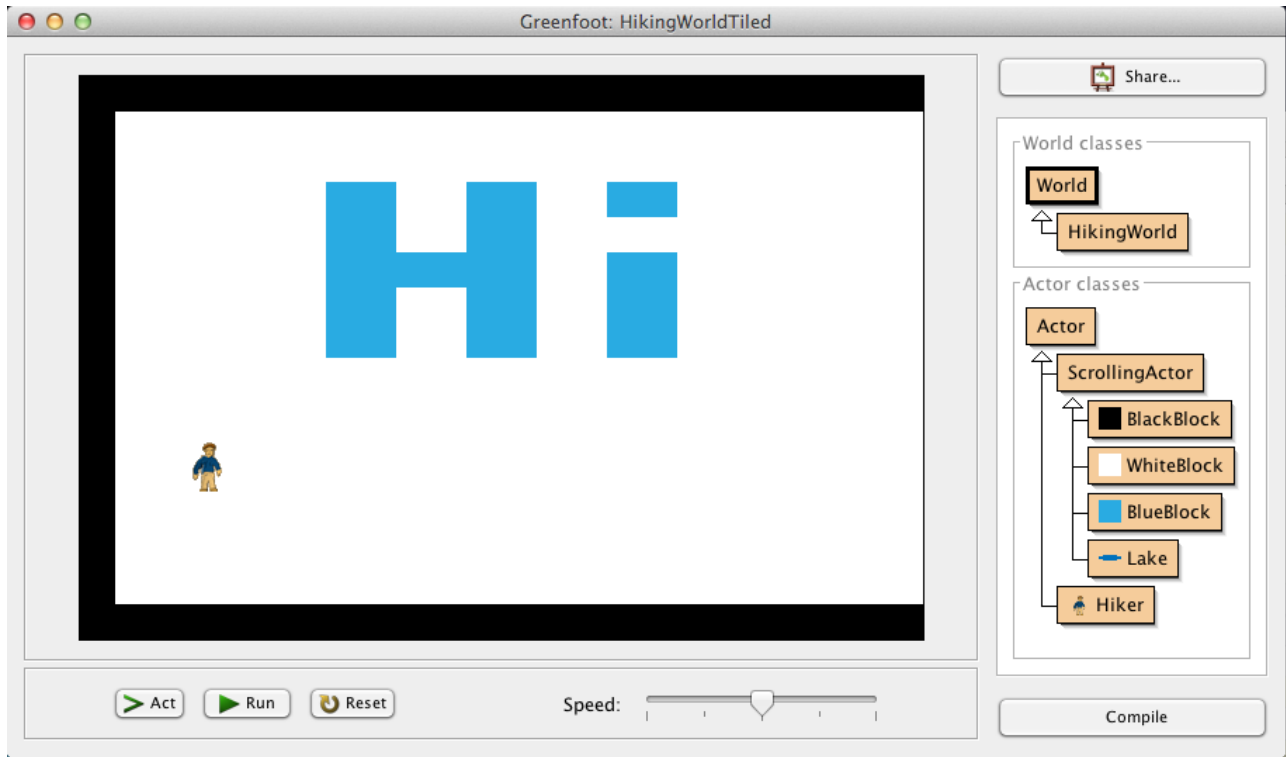




Chapter 6

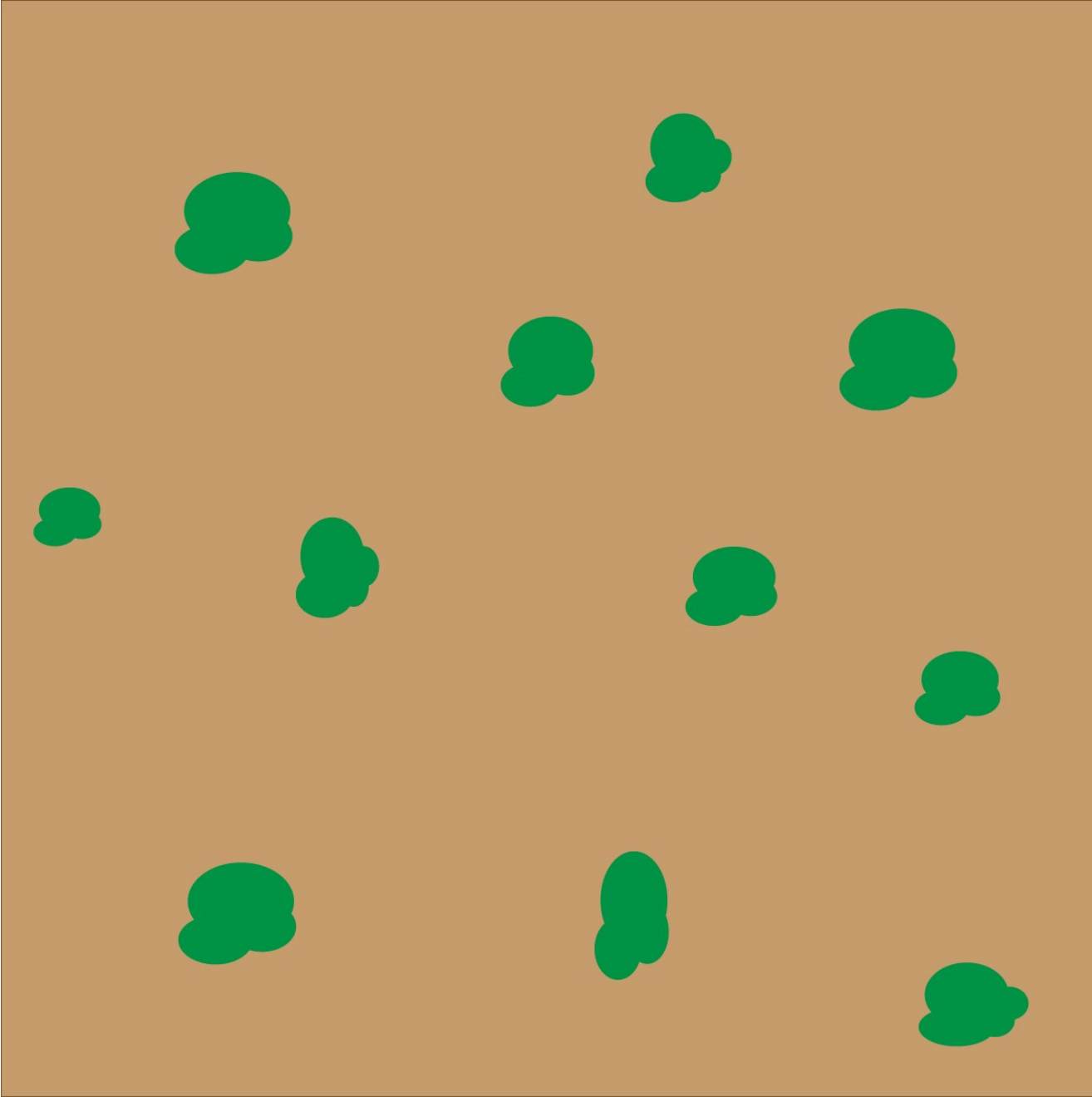


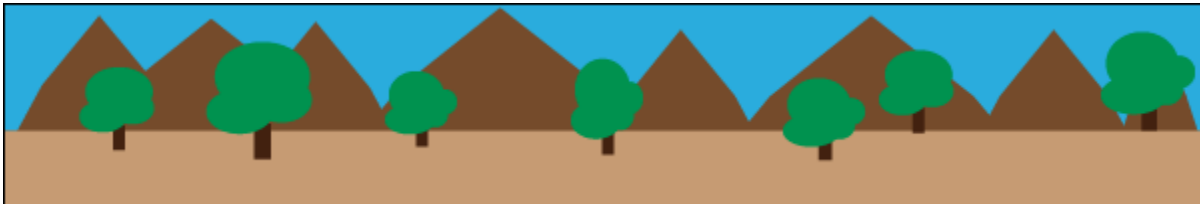
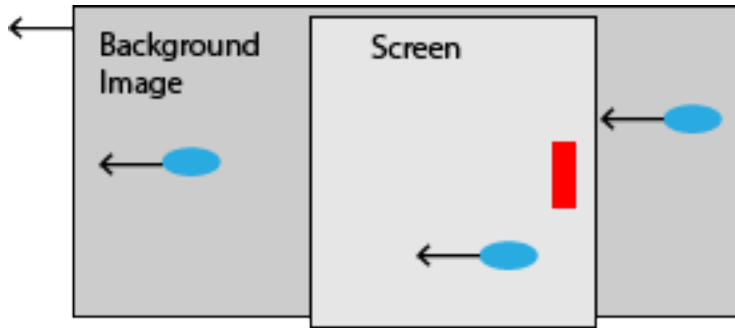












Greenfoot: HikingWorld

Share...

World classes

- World
- HikingWorld

Actor classes

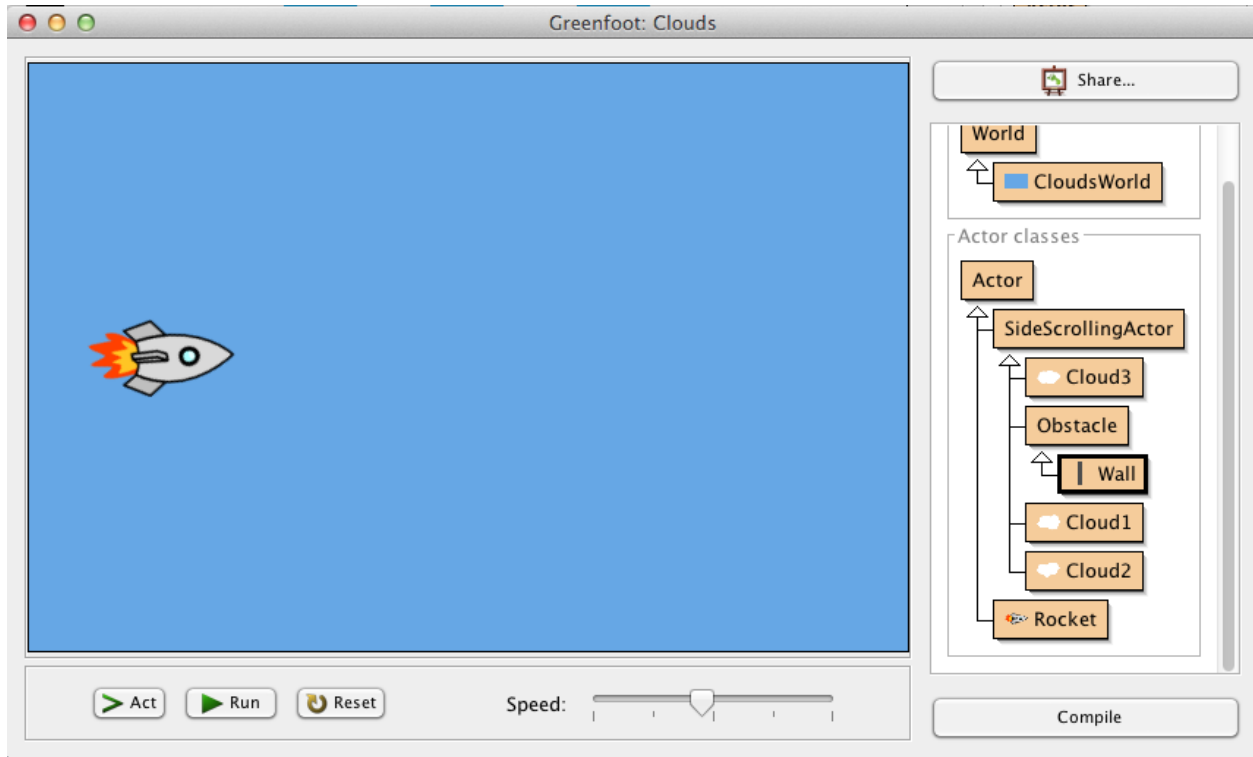
- Actor
- ScrollingActor
- HedgeHog
- Lemur
- Lake
- Hiker

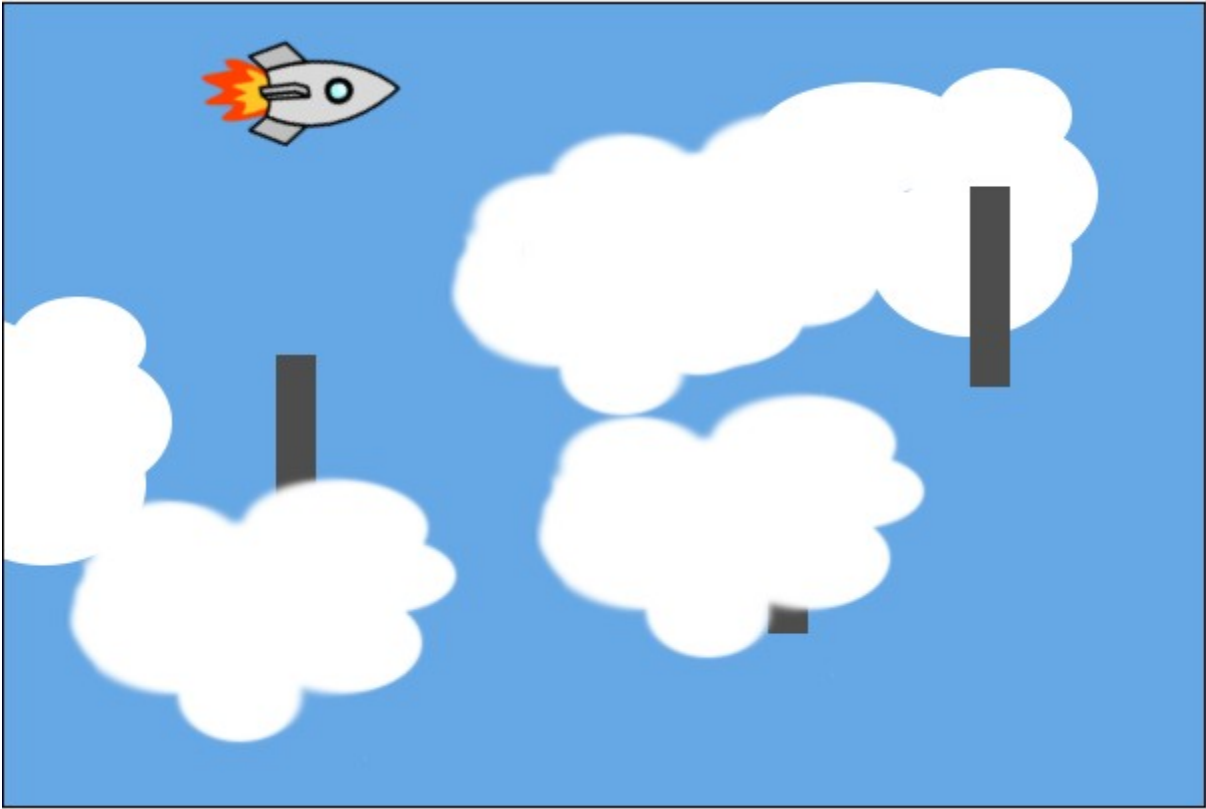
Act Run Reset

Speed: [Slider]

Compile

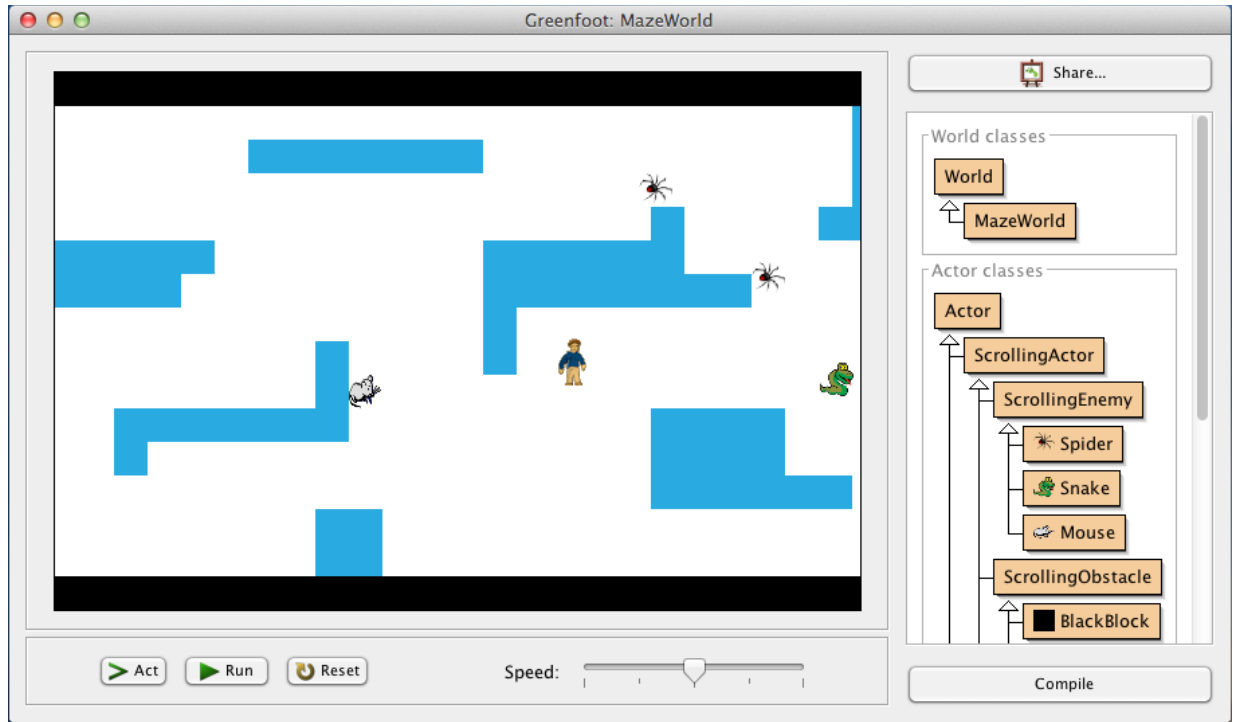
The screenshot shows the Greenfoot IDE interface for a project named 'HikingWorld'. The main window displays a 2D game world with a blue sky, brown mountains, and green trees on a brown ground plane. A small hiker character is visible on the ground. The right sidebar shows a class hierarchy for 'World classes' (World, HikingWorld) and 'Actor classes' (Actor, ScrollingActor, HedgeHog, Lemur, Lake, Hiker). The bottom control panel includes buttons for 'Act', 'Run', and 'Reset', a 'Speed' slider, and a 'Compile' button.





Chapter 7





Chapter 8



Play MazeWorld

(a) Button

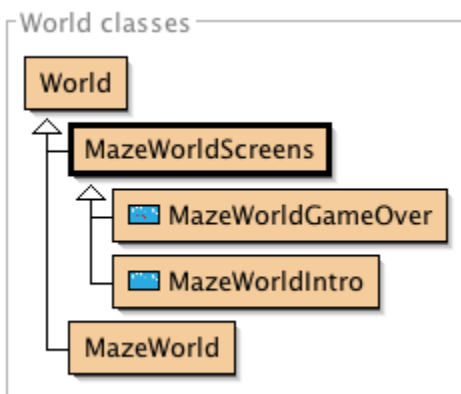
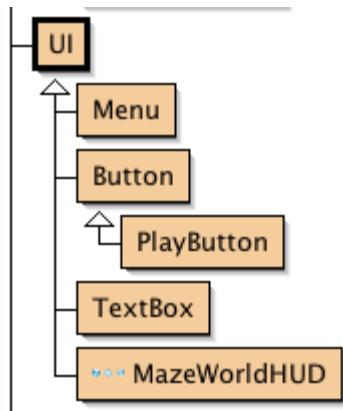
Choose game difficulty...

(b) Menu

Play the game in Easy Mode

(c) TextBox





MAZE WORLD!

Game Over

[Play MazeWorld](#)

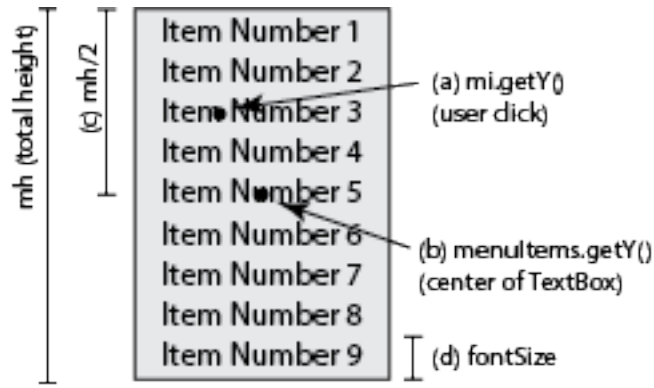
MAZE WORLD!

Play MazeWorld

Choose game difficulty...

Play the game in Easy Mode

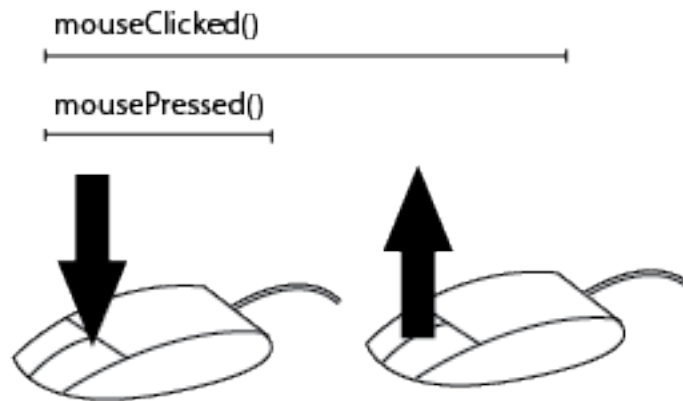




(a)

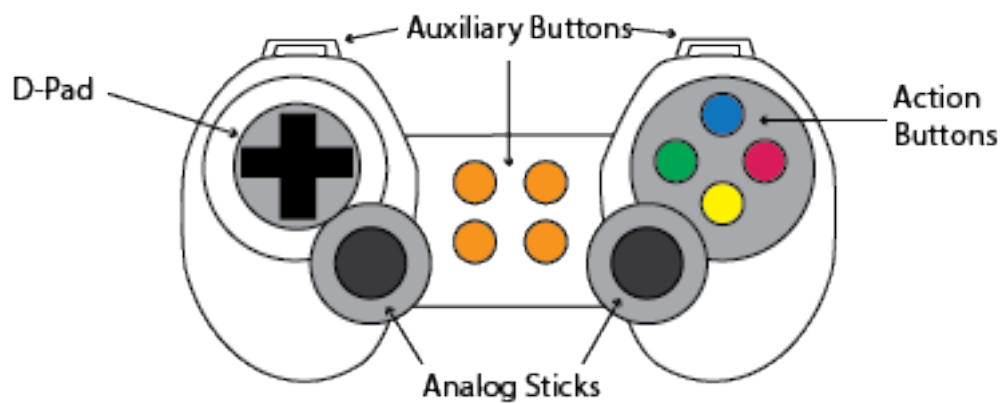


(b)





Chapter 9



Chapter 10




Greenfoot: Share

 Publish  Webpage  Application  Project

Publish the scenario to: Greenfoot Gallery (<http://www.greenfoot.org/>)

Information for display on Greenfoot Gallery

Scenario icon: (scale and move it) 

Title:

One-line description:

Longer description:

Your own page (URL):

Publish source code Lock scenario

Popular tags:

- game
- demo
- simulation
- mouse
- physics
- cabrillo
- uaslp

Additional tags: (one tag per line)

Login Username: Password: [Create account](#)