**Fish on the Hunt coding from greenfoot:**

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

/\*\*

\* Write a description of class MyWorld here.

\*

\* @author (your name)

\* @version (a version number or a date)

\*/

public class MyWorld extends World

{

/\*\*

\* Constructor for objects of class MyWorld.

\*

\*/

public MyWorld()

{

// Create a new world with 600x400 cells with a cell size of 1x1 pixels.

super(600, 600, 1);

prepare();

}

/\*\*

\* Prepare the world for the start of the program.

\* That is: create the initial objects and add them to the world.

\*/

private void prepare()

{

BrickWall brickWall = new BrickWall();

addObject(brickWall,24,574);

BrickWall brickWall2 = new BrickWall();

addObject(brickWall2,25,524);

removeObject(brickWall);

BrickWall brickWall3 = new BrickWall();

addObject(brickWall3,24,468);

brickWall3.setLocation(36,477);

brickWall3.setLocation(43,486);

brickWall3.setLocation(31,481);

brickWall2.setLocation(90,544);

brickWall3.setLocation(90,544);

brickWall2.setLocation(165,478);

brickWall3.setLocation(96,121);

brickWall2.setLocation(232,228);

BrickWall brickWall4 = new BrickWall();

addObject(brickWall4,398,129);

BrickWall brickWall5 = new BrickWall();

addObject(brickWall5,475,339);

BrickWall brickWall6 = new BrickWall();

addObject(brickWall6,470,503);

BrickWall brickWall7 = new BrickWall();

addObject(brickWall7,240,484);

BrickWall brickWall8 = new BrickWall();

addObject(brickWall8,138,337);

BrickWall brickWall9 = new BrickWall();

addObject(brickWall9,92,491);

BrickWall brickWall10 = new BrickWall();

addObject(brickWall10,362,339);

BrickWall brickWall11 = new BrickWall();

addObject(brickWall11,573,500);

BrickWall brickWall12 = new BrickWall();

addObject(brickWall12,332,564);

BrickWall brickWall13 = new BrickWall();

addObject(brickWall13,243,364);

BrickWall brickWall14 = new BrickWall();

addObject(brickWall14,478,204);

BrickWall brickWall15 = new BrickWall();

addObject(brickWall15,238,81);

BrickWall brickWall16 = new BrickWall();

addObject(brickWall16,480,66);

brickWall9.setLocation(38,479);

brickWall8.setLocation(27,427);

brickWall9.setLocation(28,475);

brickWall3.setLocation(30,344);

brickWall13.setLocation(98,360);

brickWall8.setLocation(39,404);

brickWall3.setLocation(37,361);

brickWall3.setLocation(38,364);

brickWall13.setLocation(92,354);

removeObject(brickWall7);

removeObject(brickWall12);

removeObject(brickWall6);

brickWall11.setLocation(579,487);

removeObject(brickWall11);

brickWall10.setLocation(344,333);

removeObject(brickWall10);

removeObject(brickWall5);

removeObject(brickWall8);

removeObject(brickWall9);

removeObject(brickWall3);

removeObject(brickWall13);

brickWall2.setLocation(220,236);

brickWall2.setLocation(220,236);

removeObject(brickWall2);

removeObject(brickWall15);

removeObject(brickWall4);

removeObject(brickWall14);

removeObject(brickWall16);

BrickWall brickWall17 = new BrickWall();

addObject(brickWall17,112,247);

brickWall17.setLocation(84,577);

BrickWall brickWall18 = new BrickWall();

addObject(brickWall18,84,526);

BrickWall brickWall19 = new BrickWall();

addObject(brickWall19,133,527);

BrickWall brickWall20 = new BrickWall();

addObject(brickWall20,182,527);

BrickWall brickWall21 = new BrickWall();

addObject(brickWall21,180,476);

BrickWall brickWall22 = new BrickWall();

addObject(brickWall22,179,424);

BrickWall brickWall23 = new BrickWall();

addObject(brickWall23,229,423);

BrickWall brickWall24 = new BrickWall();

addObject(brickWall24,23,288);

BrickWall brickWall25 = new BrickWall();

addObject(brickWall25,72,288);

BrickWall brickWall26 = new BrickWall();

addObject(brickWall26,68,340);

brickWall26.setLocation(73,339);

BrickWall brickWall27 = new BrickWall();

addObject(brickWall27,72,390);

BrickWall brickWall28 = new BrickWall();

addObject(brickWall28,24,191);

brickWall28.setLocation(28,185);

brickWall28.setLocation(29,188);

BrickWall brickWall29 = new BrickWall();

addObject(brickWall29,75,180);

BrickWall brickWall30 = new BrickWall();

addObject(brickWall30,125,180);

brickWall28.setLocation(15,168);

brickWall28.setLocation(32,174);

brickWall28.setLocation(32,171);

brickWall29.setLocation(73,160);

brickWall30.setLocation(112,159);

brickWall29.setLocation(68,145);

brickWall30.setLocation(354,565);

brickWall29.setLocation(356,508);

brickWall28.setLocation(343,463);

brickWall28.setLocation(347,482);

brickWall28.setLocation(184,274);

brickWall29.setLocation(240,300);

brickWall30.setLocation(292,301);

brickWall28.setLocation(191,284);

brickWall29.setLocation(241,297);

brickWall29.setLocation(252,279);

brickWall30.setLocation(301,292);

brickWall30.setLocation(567,555);

brickWall30.setLocation(500,575);

brickWall29.setLocation(488,504);

brickWall29.setLocation(507,522);

removeObject(brickWall29);

removeObject(brickWall28);

BrickWall brickWall31 = new BrickWall();

addObject(brickWall31,499,522);

BrickWall brickWall32 = new BrickWall();

addObject(brickWall32,498,470);

BrickWall brickWall33 = new BrickWall();

addObject(brickWall33,444,472);

brickWall33.setLocation(444,472);

brickWall33.setLocation(447,470);

BrickWall brickWall34 = new BrickWall();

addObject(brickWall34,395,471);

BrickWall brickWall35 = new BrickWall();

addObject(brickWall35,393,419);

BrickWall brickWall36 = new BrickWall();

addObject(brickWall36,392,368);

BrickWall brickWall37 = new BrickWall();

addObject(brickWall37,389,315);

BrickWall brickWall38 = new BrickWall();

addObject(brickWall38,438,316);

BrickWall brickWall39 = new BrickWall();

addObject(brickWall39,488,316);

BrickWall brickWall40 = new BrickWall();

addObject(brickWall40,26,155);

BrickWall brickWall41 = new BrickWall();

addObject(brickWall41,77,153);

BrickWall brickWall42 = new BrickWall();

addObject(brickWall42,128,154);

BrickWall brickWall43 = new BrickWall();

addObject(brickWall43,180,154);

BrickWall brickWall44 = new BrickWall();

addObject(brickWall44,229,155);

BrickWall brickWall45 = new BrickWall();

addObject(brickWall45,228,103);

BrickWall brickWall46 = new BrickWall();

addObject(brickWall46,219,283);

BrickWall brickWall47 = new BrickWall();

addObject(brickWall47,270,284);

BrickWall brickWall48 = new BrickWall();

addObject(brickWall48,340,265);

brickWall47.setLocation(269,269);

brickWall47.setLocation(278,300);

removeObject(brickWall47);

removeObject(brickWall46);

BrickWall brickWall49 = new BrickWall();

addObject(brickWall49,286,265);

BrickWall brickWall50 = new BrickWall();

addObject(brickWall50,389,264);

BrickWall brickWall51 = new BrickWall();

addObject(brickWall51,390,213);

BrickWall brickWall52 = new BrickWall();

addObject(brickWall52,391,162);

BrickWall brickWall53 = new BrickWall();

addObject(brickWall53,441,163);

BrickWall brickWall54 = new BrickWall();

addObject(brickWall54,492,163);

BrickWall brickWall55 = new BrickWall();

addObject(brickWall55,490,111);

FishPlayer fishPlayer = new FishPlayer();

addObject(fishPlayer,41,466);

Fishfood fishfood = new Fishfood();

addObject(fishfood,556,49);

Fishfood fishfood2 = new Fishfood();

addObject(fishfood2,64,88);

Fishfood fishfood3 = new Fishfood();

addObject(fishfood3,450,203);

removeObject(fishfood3);

Fishfood fishfood4 = new Fishfood();

addObject(fishfood4,570,537);

Fishfood fishfood5 = new Fishfood();

addObject(fishfood5,56,226);

Fishfood fishfood6 = new Fishfood();

addObject(fishfood6,397,551);

Fishfood fishfood7 = new Fishfood();

addObject(fishfood7,465,386);

brickWall40.setLocation(46,145);

Paranas paranas = new Paranas();

addObject(paranas,46,145);

brickWall24.setLocation(28,294);

Paranas paranas2 = new Paranas();

addObject(paranas2,28,294);

Paranas paranas3 = new Paranas();

addObject(paranas3,30,554);

brickWall40.setLocation(24,233);

brickWall40.setLocation(11,151);

brickWall40.setLocation(13,156);

brickWall24.setLocation(148,284);

brickWall24.setLocation(19,288);

paranas.setLocation(307,139);

paranas3.setLocation(282,540);

brickWall24.setLocation(181,292);

brickWall40.setLocation(44,162);

paranas2.setLocation(329,345);

brickWall24.setLocation(11,292);

paranas2.setLocation(220,353);

brickWall40.setLocation(15,155);

}

}

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

/\*\*

\* Write a description of class BrickWall here.

\*

\* @author (your name)

\* @version (a version number or a date)

\*/

public class BrickWall extends Actor

{

/\*\*

\* Act - do whatever the BrickWall wants to do. This method is called whenever

\* the 'Act' or 'Run' button gets pressed in the environment.

\*/

public void act()

{

// Add your action code here.

}

}

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

/\*\*

\* Write a description of class Fishfood here.

\*

\* @author (your name)

\* @version (a version number or a date)

\*/

public class Fishfood extends Actor

{

/\*\*

\* Act - do whatever the Fishfood wants to do. This method is called whenever

\* the 'Act' or 'Run' button gets pressed in the environment.

\*/

public void act()

{

// Add your action code here.

}

public Fishfood()

{

getImage().scale(getImage().getWidth()/2, getImage().getHeight()/2);

}

}

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

/\*\*

\* Write a description of class enemy here.

\*

\* @author (your name)

\* @version (a version number or a date)

\*/

public class enemy extends Actor

{

/\*\*

\* Act - do whatever the enemy wants to do. This method is called whenever

\* the 'Act' or 'Run' button gets pressed in the environment.

\*/

public void act()

{

// Add your action code here.

}

public void wrapAtEdge()

{

int x = getX();

int y = getY();

int worldWidth = getWorld().getWidth() - 1;

int worldHeight = getWorld().getHeight() - 1;

if (x >= worldWidth)

{

setLocation(1 ,y);

}

if (x <= 0)

{

setLocation(worldWidth - 1 , y);

}

if (y >= worldHeight)

{

setLocation(x, 1);

}

if (y <= 0)

{

setLocation(x , worldHeight - 1);

}

}

}

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

/\*\*

\* Write a description of class Paranas here.

\*

\* @author (your name)

\* @version (a version number or a date)

\*/

public class Paranas extends enemy

{

/\*\*

\* Act - do whatever the Paranas wants to do. This method is called whenever

\* the 'Act' or 'Run' button gets pressed in the environment.

\*/

public void act()

{

move(5);

wrapAtEdge();

}

}

import greenfoot.\*; // (World, Actor, GreenfootImage, Greenfoot and MouseInfo)

/\*\*

\* Write a description of class mover here.

\*

\* @author (your name)

\* @version (a version number or a date)

\*/

public class mover extends Actor

{

/\*\*

\* Act - do whatever the mover wants to do. This method is called whenever

\* the 'Act' or 'Run' button gets pressed in the environment.

\*/

public void act()

{

// Add your action code here.

}

public void moveandTurn()

{

if(Greenfoot.isKeyDown("right"))

{

setRotation(0);

move(4);

}

if(Greenfoot.isKeyDown("left"))

{

setRotation(180);

move(4);

}

if(Greenfoot.isKeyDown("up"))

{

setRotation(270);

move(4);

}

if(Greenfoot.isKeyDown("down"))

{

setRotation(90);

move(4);

}

}

public void slideAround()

{

int x = getX();

int y = getY();

if(Greenfoot.isKeyDown("right"))

{

setLocation(x + 4, y);

if(hitwalls())

{

setLocation(x - 4, y);

}

}

if(Greenfoot.isKeyDown("left"))

{

setLocation(x - 4, y);

if(hitwalls())

{

setLocation(x + 4, y);

}

}

if(Greenfoot.isKeyDown("up"))

{

setLocation(x, y - 4);

if(hitwalls())

{

setLocation(x , y + 4);

}

}

if(Greenfoot.isKeyDown("down"))

{

setLocation(x , y + 4);

if(hitwalls())

{

setLocation(x , y - 4);

}

}

}

public boolean hitwalls()

{

if(isTouching(BrickWall.class))

{

return true;

}

else

{

return false;

}

}

public void collectitem()

{

if (isTouching(Fishfood.class))

{

removeTouching(Fishfood.class);

}

}

public boolean hitEnemy()

{

if (isTouching(enemy.class))

{

return true;

}

else

{

return false;

}

}

}

/\*\*

\* Write a description of class FishPlayer here.

\*

\* @author (your name)

\* @version (a version number or a date)

\*/

public class FishPlayer extends mover

{

/\*\*

\* Act - do whatever the FishPlayer wants to do. This method is called whenever

\* the 'Act' or 'Run' button gets pressed in the environment.

\*/

//Create our Constructor method

public FishPlayer()

{

getImage().scale(getImage().getWidth()/2, getImage().getHeight()/2);

}

public void act()

{

slideAround();

collectitem();

mainPlayerHit();

}

public void mainPlayerHit()

{

if(hitEnemy())

{

setLocation(41,466);

}

}

}