Extra Credit: Random Circles

Xingxuan Zhang

4/20/18

Source Code:

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

// Overlap.java Author: Xingxuan Zhang

//

//Randomly create 20 circles and change their colors if they overlap

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

**import** java.util.Random;

**import** javafx.scene.shape.\*;

**import** javafx.scene.Group;

**import** javafx.application.Application;

**import** javafx.geometry.Pos;

**import** javafx.geometry.Insets;

**import** javafx.scene.Scene;

**import** javafx.scene.paint.Color;

**import** javafx.stage.Stage;

**public** **class** Overlap **extends** Application

{

**public** **void** start(Stage primaryStage)

{

Group root = **new** Group ();

**final** **int** MAXCIRCLE = 20;

**int** circlenum = 0;

//construct four arrays to store circles' characteristics and circles themselves

**int**[] randX = **new** **int** [MAXCIRCLE];

**int**[] randY = **new** **int** [MAXCIRCLE];

**int**[] radius = **new** **int** [MAXCIRCLE];

Circle [] circles = **new** Circle [MAXCIRCLE];

//construct a random number generator

Random gen = **new** Random ();

//for every circle, generate random position and radius

**for** ( circlenum = 0;circlenum<MAXCIRCLE; circlenum++)

{

randX[circlenum] = gen.nextInt(500)+1;

randY[circlenum] = gen.nextInt(500)+1;

radius[circlenum] = gen.nextInt(50)+1;

}

//create 20 circles by using random numbers as characteristics and assign in the circle array

**for** ( circlenum = 0;circlenum<MAXCIRCLE; circlenum++)

{

Circle randcir = **new** Circle (randX[circlenum],randY[circlenum],radius[circlenum]);

randcir.setFill(Color.***BLACK***);

circles[circlenum] = randcir;

root.getChildren().addAll(randcir);

}

**int** difference;

Color Blue=**new** Color(0f,0f,1f,.5f );

//overlap check

**for** (difference = 1;difference<MAXCIRCLE;difference++)

{

**for** (circlenum =0;circlenum<MAXCIRCLE-difference; circlenum++)

{

**double** diferX = Math.*pow*(Math.*abs*(randX[circlenum]-randX[circlenum+difference]), 2);

**double** diferY = Math.*pow*(Math.*abs*(randY[circlenum]-randY[circlenum+difference]), 2);

**double** distance = Math.*sqrt*(diferX+diferY);

**double** sumR = radius[circlenum]+radius[circlenum+difference];

**if** (distance<sumR)

{

circles[circlenum].setFill(Blue);

circles[circlenum+difference].setFill(Blue);

}

}

}

Scene scene = **new** Scene (root, 500, 500);

primaryStage.setTitle("Circle Overlap");

primaryStage.setScene(scene);

primaryStage.show();

}

**public** **static** **void** main(String[] args)

{

*launch*(args);

}

}







