```
/*
```

- To change this license header, choose License Headers in Project Properties.
- To change this template file, choose Tools | Templates
- and open the template in the editor.

```
*/
package trees;

import java.io.IOException;
import java.net.URL;
import java.util.List;
import java.util.ResourceBundle;
import javafx.fxml.FXML;
import javafx.fxml.FXMLLoader;
import javafx.fxml.Initializable;
import javafx.scene.Parent;
import javafx.scene.layout.AnchorPane;
import javafx.scene.paint.Color;
import javafx.scene.shape.Rectangle;
import javafx.scene.shape.Line;
import java.util.List;

/**
*
```

- @author zeckzer

```
*/
public class TreesPaneController
implements Initializable {

    @FXML
    private AnchorPane rootPane;

    private double coordAx = 250;
    private double coordBx = 250;
    private double coordCx = 200;
    private double coordDx = 300;

    private double coordAy = 500;
    private double coordBy = 400;
    private double coordCy = 350;
    private double coordDy = 350;

    /**
    *
```

  - @param rootPane the parent element

```
    */
    public void setPane(AnchorPane rootPane) {
    this.rootPane = rootPane;
    }

    /**
    *
```

  - @return parent element

```
    */
```

public Parent getRoot() {
return rootPane;
}

/**

- @return the object loaded
  */
  public static TreesPaneController getInstance() {
  FXMLLoader loader = new FXMLLoader();
  try {
  // Load root layout from fxml file.
  loader.setLocation(TreesPaneController.class.getResource("TreesPane.fxml"));
  AnchorPane rootPane = (AnchorPane) loader.load();
  TreesPaneController treesPaneController = loader.getController();
  treesPaneController.setPane(rootPane);
  treesPaneController.draw ();
  return treesPaneController;
  } catch (IOException e) {
  e.printStackTrace();
  return null;
  }
  }

@Override
public void initialize(URL url, ResourceBundle rb) {

}

public void draw () {
rootPane.getChildren().clear();

```
double x = 200.0;
double y = 50.0;
double width = 100.0;

for (int i = 0; i <600; i+=10) {
    Line line1 = new Line(i, 0, i, 600);
    line1.setStroke(Color.LIGHTGRAY);
    Line line2 = new Line(0, i, 600, i);
    line2.setStroke(Color.LIGHTGRAY);
    rootPane.getChildren().add(line2);
    rootPane.getChildren().add(line1);
}

TreeGenerator treeGen = new TreeGenerator(
    new Point(coordAx, coordAy),
    new Point(coordBx, coordBy),
    new Point(coordCx, coordCy),
    new Point(coordDx, coordDy),
    400.0, 400.0,
    10
);
List<TreeLine> lines = treeGen.generateTree();

System.out.println(lines);
float r = 1.0f;
float g = 1.0f;
float b = 1.0f;

for (TreeLine tline: lines) {
    Line line = new Line(tline.getA().getX(), tline.getA().getY(), tline.getB().getX(), tline.getB().getY());
    r *= 0.99;
    g *= 0.99;
```

```
        b *= 0.99;
        Color color = new Color(r, g, b);
        line.setStroke(color.getColor());

       rootPane.getChildren().add(line);


    }

    // Rectangle leftRectangle = createRectangle(x - 1.2 * width, y,
    //                                           width, leftSize,
    //                                           leftColor);
    // rootPane.getChildren().add(leftRectangle);

    // Rectangle rightRectangle = createRectangle(x + 1.2 * width, y,
    //                                            width, rightSize,
    //                                            rightColor);
    // rootPane.getChildren().add(rightRectangle);
```

}

/**

- @param x
- @param y
- @param w idth
- @param height
- @param color
- @return

*/
private Rectangle draw Tree(List lines) {
// Rectangle rectangle = new Rectangle(x, y, w idth, height);
// rectangle.setFill(color);
// rectangle.setStroke(Color.BLACK);
// return rectangle;
return null;
}

public double getAx() {
return coordAx;
}

public void setAx(double coordAx) {
this.coordAx = coordAx;
this.draw ();
}

public double getBx() {
return coordBx;
}

public void setBx(double coordBx) {
this.coordBx = coordBx;
this.draw ();
}

public double getCx() {
return coordCx;
}

```
    public void setCx(double coordCx) {
    this.coordCx = coordCx;
    this.draw();
    }

    public double getDx() {
    return coordDx;
    }

    public void setDx(double coordDx) {
    this.coordDx = coordDx;
    this.draw();
    }

    public double getAy() {
    return coordAy;
    }

    public void setAy(double coordAy) {
    this.coordAy = coordAy;
    this.draw();
    }

    public double getBy() {
    return coordBy;
    }

    public void setBy(double coordBy) {
    this.coordBy = coordBy;
    this.draw();
    }

    public double getCy() {
    return coordCy;
    }

    public void setCy(double coordCy) {
    this.coordCy = coordCy;
    this.draw();
    }

    public double getDy() {
    return coordDy;
    }

    public void setDy(double coordDy) {
    this.coordDy = coordDy;
    this.draw();
    }

}
```