Module 1 Assignment, Results Document

## Program Listing:

/\* *Edgar Rosales*

*\* 29 May 2024*

*\* CSD405-J318 Intermediate Java Programming*

*\**

*\* Module 1 Assignment*

*\* Write a program with a class titled Fan. This class is to contain:*

*\* 1.Four constants named STOPPED, SLOW, MEDIUM, and FAST. The constants are to hold the values of 0, 1, 2, and 3 respectively.*

*\* 2.A private field named speed that holds one of the constant values with the default being STOPPED.*

*\* 3.A private Boolean field titled on that specifies whether the fan is on or off.*

*\* 4.A private field named radius that holds the radius of the fan with a default value of 6.*

*\* 5.A String field that holds the color, with the default being white.*

*\* 6.Setter and getter methods for all mutable fields.*

*\* 7.A no-argument constructor that sets all fields with a default value.*

*\* 8.A constructor taking arguments and setting values.*

*\* 9.Write a toString() method that returns a description of the Fans state.*

*\* 10.Write test code that creates two instances of the Fan class, one using the default constructor and the other using the argument constructor.*

*\* Write code that displays the functionality of the Fan class methods.*

\*/

*public* *class* Fan {

*static* *final* int STOPPED = 0;

*static* *final* int SLOW = 1;

*static* *final* int MEDIUM = 2;

*static* *final* int FAST = 3;

*static* int numberOfFans = 0;

*private* boolean isOn;

*private* int speed;

*private* int radius;

*private* String color;

    // *Default constructor.*

*public* Fan() {

        this.*isOn* = false;

        this.*speed* = STOPPED;

        this.*radius* = 6;

        this.*color* = "white";

        numberOfFans++;

    }

    // *Constructor.*

*public* Fan(boolean isOn, int speed, int radius, String color) {

        this.*isOn* = isOn;

        this.*speed* = speed;

        this.*radius* = radius;

        this.*color* = color;

        numberOfFans++;

    }

    // *Setters and Getters.*

*public* void setOn(boolean isOn) {

        this.*isOn* = isOn;

    }

*public* void setSpeed(int speed) {

        this.*speed* = speed;

    }

*public* void setRadius(int radius) {

        this.*radius* = radius;

    }

*public* void setColor(String color) {

        this.*color* = color;

    }

*public* boolean isOn() {

        return isOn;

    }

*public* int getSpeed() {

        return speed;

    }

*public* int getRadius() {

        return radius;

    }

*public* String getColor() {

        return color;

    }

    // *Get a count of the number of fans created.*

*public* *static* int getNumberOfFans() {

        return numberOfFans;

    }

    // *Methods for increasing and decreasing speed.*

*public* void increaseSpeed() {

        if (speed < FAST) {

            speed++;

        }

    }

*public* void decreaseSpeed() {

        if (speed > STOPPED) {

            speed--;

        }

    }

    // *Methods for increasing and decreasing radius.*

*public* void increaseRadius() {

        if (radius < 6) {

            radius++;

        }

    }

*public* void decreaseRadius() {

        if (radius > 180) {

            radius--;

        }

    }

    // *Method for toggling the fan power.*

*public* void toggleFan() {

        isOn = !isOn;

        speed = (isOn) ? SLOW : STOPPED;

        radius = (isOn)? 6 : 0;

        color = (isOn)? "green" : "red";

    }

    @Override

*public* String toString() {

        return "[ Powered: " + isOn + ", speed= " + speed + ", radius= " + radius

                + ", color= " + color + " ]";

    }

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

        // *Initialize Fan objects*

        Fan fan1 = new Fan();

        Fan fan2 = new Fan(true, MEDIUM, 8, "blue");

        System.*out*.println("Fans after initialization and count on how many are created.");

        System.*out*.println("Fan 1 Initial Status: " + fan1);

        System.*out*.println("Fan 2 Initial Status: " + fan2);

        System.*out*.println("Number of Fans: " + Fan.getNumberOfFans());

        //*Test the setter methods*

        fan1.setOn(true);

        fan1.setSpeed(SLOW);

        fan1.setRadius(10);

        fan1.setColor("blue");

        fan2.setSpeed(FAST);

        fan2.setColor("green");

        System.*out*.println("\nFan 1 after using setter: " + fan1);

        System.*out*.println("Fan 2 after using setter: " + fan2);

        //*Test fan controls*

        fan1.increaseSpeed();

        fan2.decreaseSpeed();

        System.*out*.println("\nFan 1 after control modification: " + fan1);

        System.*out*.println("Fan 2 after control modification: " + fan2);

        // *Toggle power fans to off*

        fan1.toggleFan();

        fan2.toggleFan();

        System.*out*.println("\nFan 1 after toggle power: " + fan1);

        System.*out*.println("Fan 2 after toggle power: " + fan2);

        // *Toggle them on again.*

        fan1.toggleFan();

        fan2.toggleFan();

        System.*out*.println("\nFan 1 after toggle power: " + fan1);

        System.*out*.println("Fan 2 after toggle power: " + fan2);

    }

}

## GitHub Link:

<https://github.com/erosales48/csd-405/blob/main/Module1-Assignment/Fan.java>

## Example 1:

A screen shot of a computer

Description automatically generated