CS 1073

FR03A

Assignment #5

Ethan A. McCarthy

3573807

# Section 1

/\*\*

\* @author Ethan McCarthy 3573807

\*/

public class Window{

    private String address;

    private double height;

    private double size;

    private boolean insulation;

    /\*\*

     \* constructor

     \* @param addressIn address

     \* @param heightIn height

     \* @param sizeIn size

     \* @param insulationIn if the insulation is special or not

     \*/

    public Window(String addressIn, double heightIn, double sizeIn, boolean insulationIn){

        address = addressIn;

        height = heightIn;

        size =  sizeIn;

        insulation = insulationIn;

    }

    /\*\*

     \*  getter method to get address

     \* @return the address

     \*/

    public String getAddress(){

        return address;

    }

    /\*\*

     \* returns height

     \* @return height

     \*/

    public double getHeight(){

        return height;

    }

    /\*\*

     \* returns area

     \* @return size

    \*/

    public double getSize(){

        return size;

    }

    /\*\*

     \* returns wether or not the window has special insulation

     \* @return insulation

     \*/

    public boolean getInsulation(){

        return insulation;

    }

    /\*\*

     \* method to get the total cost

     \* @param materialRate how much per cm^2

     \* @return the total cost

     \*/

    public double installationCost(double materialRate){

        double cost = 100 + (size \* materialRate);

        if(insulation == true){

            cost += 25;

        }

        return cost;

    }

}

/\*\*

\* @author Ethan McCarthy 3573807

\*/

public class HopperWindow extends Window{

        private boolean inwards;

        /\*\*

         \* constructor

         \* @param addressIn address from superclass

         \* @param heightIn height from superclass

         \* @param sizeIn size from superclass

         \* @param insulationIn insulation from superclass

         \* @param stainedSizeIn stained glass area

         \*/

        public HopperWindow(String addressIn, double heightIn, double sizeIn, boolean insulationIn, boolean inwardsIn){

            super (addressIn, heightIn, sizeIn, insulationIn);

            inwards = inwardsIn;

        }

        /\*\*

         \* returns if the hopper window open out or in

         \* @return if its inward or not

         \*/

        public boolean inwardsStatus(){

            return inwards;

        }

        /\*\*

         \* method to get the total cost

         \* @param materialRate how much per cm^2

         \* @return the total cost

         \*/

        public double installationCost(double materialRate){

            double cost = super.installationCost(materialRate);

            if(inwards == true){

                cost += 125;

            }

            else{

                cost += 75;

            }

            return cost;

        }

    }

/\*\*

\* @author Ethan McCarthy 3573807

\*/

public class StainedGlassWindow extends Window{

    private double stainedSize;

    /\*\*

     \* constructor

     \* @param addressIn address from superclass

     \* @param heightIn height from superclass

     \* @param sizeIn size from superclass

     \* @param insulationIn insulation from superclass

     \* @param stainedSizeIn stained glass area

     \*/

    public StainedGlassWindow(String addressIn, double heightIn, double sizeIn, boolean insulationIn, double stainedSizeIn){

        super (addressIn, heightIn, sizeIn, insulationIn);

        stainedSize = stainedSizeIn;

    }

    /\*\*

     \* method to retreive the area of the stained part

     \* @return stained area

     \*/

    public double getStainedSize(){

        return stainedSize;

    }

    /\*\*

     \* method to get the total cost

     \* @param materialRate how much per cm^2

     \* @return the total cost

     \*/

    public double installationCost(double materialRate){

        double cost = super.installationCost(materialRate);

        if(stainedSize > (super.getSize()/2)){

            cost += super.getSize() \* 0.11;

        }

        else{

            cost += super.getSize()\*0.14;

        }

        return cost;

    }

}

# Section 2

/\*\*

 \* @author Ethan McCarthy 3573807

 \*/

 import java.text.NumberFormat;

 public class WindowInstallationCalculator{

    public static void main(String[] args){

        NumberFormat formatter = NumberFormat.getNumberInstance();

        formatter.setMaximumFractionDigits(2);

        formatter.setMinimumFractionDigits(2);

        Window window1 = new Window("99 Rose Point Rd", 1.5, 3325, true);

        StainedGlassWindow window2 = new StainedGlassWindow("965 Major Mack Dr W", 1.75, 5234, false, 1842);

        HopperWindow window3 = new HopperWindow("180 Hughson St", 4.5, 5054, true, true);

        StainedGlassWindow window4 = new StainedGlassWindow("925 W Georgia St", 2, 4235, true, 3733);

        StainedGlassWindow window5 = new StainedGlassWindow("2550 Rue Beaugrand", 1.78, 2950, false, 1475);

        HopperWindow window6 = new HopperWindow("970 Southdown Rd", 6, 3680, false, false);

        Window window7 = new Window("297 Bernard Ave", 1.8, 5287, false);

        HopperWindow window8 = new HopperWindow("21453 Lake Rd", 9.5, 8400, false, true);

        System.out.println("Address & Window height:\tWindow Size:\tInstallation Cost:");

        System.out.println("========================\t============\t==================");

        System.out.println(window1.getAddress() + "(" + window1.getHeight() + ")\t\t" + window1.getSize() + "cm^2\t" + "$" + formatter.format(window1.installationCost(0.05)));

        System.out.println(window2.getAddress() + "(" + window2.getHeight() + ")\t" + window2.getSize() + "cm^2\t" + "$" + formatter.format(window2.installationCost(0.07)));

        System.out.println(window3.getAddress() + "(" + window3.getHeight() + ")\t\t" + window3.getSize() + "cm^2\t" + "$" + formatter.format(window3.installationCost(0.07)));

        System.out.println(window4.getAddress() + "(" + window4.getHeight() + ")\t\t" + window4.getSize() + "cm^2\t" + "$" + formatter.format(window4.installationCost(0.07)));

        System.out.println(window5.getAddress() + "(" + window5.getHeight() + ")\t" + window5.getSize() + "cm^2\t" + "$" + formatter.format(window5.installationCost(0.14)));

        System.out.println(window6.getAddress() + "(" + window6.getHeight() + ")\t\t" + window6.getSize() + "cm^2\t" + "$" + formatter.format(window6.installationCost(0.12)));

        System.out.println(window7.getAddress() + "(" + window7.getHeight() + ")\t\t" + window7.getSize() + "cm^2\t" + "$" + formatter.format(window7.installationCost(0.04)));

        System.out.println(window8.getAddress() + "(" + window8.getHeight() + ")\t\t" + window8.getSize() + "cm^2\t" + "$" + formatter.format(window8.installationCost(0.04)));

    }

 }

# Section 3

Text

Description automatically generated