**Program Ten Part One:**

//summary: This program uses classes to create a class named fan. This class holds the data for a fan and can output said

//data for the fan. A fan has a speed, on setting, radius, and color.

//name: Jenna Wolf

//class: Fundamentals of Programming, CS155 - 01

//instructor: Dr. Art Kazmierczak

//date: 10/24/2023

public class Main

{

public static void main(String[] args)

{

fan fanOne = new fan(); //creates a new fan object

fan fanTwo = new fan(); //creates a new fan object

fanOne.setSpeed(fan.FAST); //sets the speed for fan1

fanOne.setOn(true); //sets the on for fan1

fanOne.setRadius(10); //sets the radius for fan1

fanOne.setColor("yellow"); //sets the color for fan1

fanTwo.setSpeed(fan.MEDIUM); //sets the speed for fan2

fanTwo.setOn(false); //sets the on for fan2

fanTwo.setRadius(5); //sets the radius for fan2

fanTwo.setColor("blue"); //sets the color for fan2

System.out.println(fanOne.toString()); //outputs the data for fan1

System.out.println(); //creates a space

System.out.println(fanTwo.toString()); //outputs the data for fan2

}

}

**Fan Class:**

public class fan {

public final static int SLOW = 1; //Constant that holds the SLOW value

public final static int MEDIUM = 2; //Constant that holds the MEDIUM value

public final static int FAST = 3; //Constant that holds the FAST value

private int speed = SLOW; //holds the speed data and defaults it to SLOW

private boolean on = false; //holds the on data and defaults it to false

private double radius = 5; //holds the radius data and defaults it to 5

String color = "blue"; //holds the color data and defaults it to blue

//no arg constuctor

fan(){

}

//returns the speed data

public int getSpeed()

{

return speed;

}

//returns the on data

public boolean getOn()

{

return on;

}

//returns the radius data

public double getRadius()

{

return radius;

}

//returns the color data

public String getColor()

{

return color;

}

//sets the value sent over to speed

public void setSpeed(int temp)

{

speed = temp;

}

//sets the value sent over to on

public void setOn(boolean temp)

{

on = temp;

}

//sets the value sent over to radius

public void setRadius(double temp)

{

radius = temp;

}

//sets the value sent over to color

public void setColor(String temp)

{

color = temp;

}

//outputs the data about the fan

public String toString()

{

if(on == true) //outputs if fan is turned on

return "Fan speed: " + getSpeed() + "\nFan Color: " + getColor() + "\nFan Radius: " + getRadius();

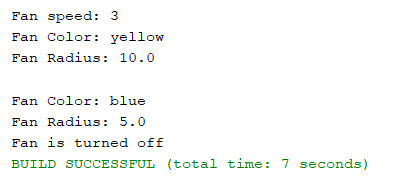
else //outputs if fan is turned off

return "Fan Color: " + getColor() + "\nFan Radius: " + getRadius() + "\nFan is turned off";

}

}

Output:



**Program Ten Part Two:**

//summary: This program uses classes to create a class named triangle, an extension of geometric object. This program takes

//in the side lengths, color, and weather the triangle is filled from the user and displays the sides, area, perimeter,

//color, and weather the triangle is filled.

//name: Jenna Wolf

//class: Fundamentals of Programming, CS155 - 01

//instructor: Dr. Art Kazmierczak

//date: 10/24/2023

import java.util.Scanner; //lets inputs be made

public class Main

{

public static void main(String[] args)

{

Scanner input = new Scanner(System.in); //names the input object

double triSide1, triSide2, triSide3; //holds the data for these objects

String triColor; //holds the triColor data

boolean triFilled; //holds the triFilled data

//takes in the sides from the user

System.out.print("Enter sides of your triangle: ");

triSide1 = input.nextDouble();

triSide2 = input.nextDouble();

triSide3 = input.nextDouble();

//takes in the color from the user

System.out.print("Enter the color of your triangle: ");

triColor = input.next();

//takes in the filled property from the user

System.out.print("indicate weather your triangle is filled of empty (true or false): ");

triFilled = input.nextBoolean();

triangle tri = new triangle(triSide1, triSide2, triSide3); //creates a new triangle object

tri.setColor(triColor); //sets the color of the triangle

tri.setFilled(triFilled); //sets the filled property of the triangle

//outputs all info for the triangle

System.out.println();

System.out.println(tri.toString());

System.out.printf("Area: %.2f", tri.getArea());

System.out.println();

System.out.println("Perimeter: " + tri.getPerimeter());

System.out.println("Color: " + tri.getColor());

if(tri.getFilled())

System.out.println("Triangle is filled");

else

System.out.println("Triangle is not filled");

}

}

**Geometric Object Class:**

import java.util.Date; //lets the Date class be used

public class GeometricObject{

private String color = "white"; //holds the color data and defaults to white

private boolean filled; //holds the filled data

private java.util.Date dateCreated; //holds the dateCreated data

//no args constructor

public GeometricObject(){

dateCreated = new java.util.Date();

}

//constructor that takes in a string and a boolean

public GeometricObject(String temp1, boolean temp2){

color = temp1;

filled = temp2;

dateCreated = new java.util.Date();

}

//returns the color data

public String getColor()

{

return color;

}

//returns the filled data

public boolean getFilled()

{

return filled;

}

//returns the dateCreated data

public Date getDateCreated()

{

return dateCreated;

}

//sets color to the data sent over

public void setColor(String temp)

{

color = temp;

}

//sets filled to the data sent over

public void setFilled(boolean temp)

{

filled = temp;

}

//returns a string with the GeometricObject data

public String toString()

{

return "Created on: " + dateCreated + "\nColor: " + color + "\nFilled: " + filled;

}

}

**Triangle Class**

public class triangle extends GeometricObject{

double side1 = 1; //holds the side1 data and defaults to 1

double side2 = 1; //holds the side2 data and defaults to 1

double side3 = 1; //holds the side3 data and defaults to 1

//no args constructor

public triangle(){

}

//constructor that takes in 3 doubles

public triangle(double temp1, double temp2, double temp3){

side1 = temp1;

side2 = temp2;

side3 = temp3;

}

//returns the info in side1

public double getSide1()

{

return side1;

}

//returns the info in side2

public double getSide2()

{

return side2;

}

//returns the info in side3

public double getSide3()

{

return side3;

}

//sets side1 to the double sent over

public void setSide1(double temp)

{

side1 = temp;

}

//sets side2 to the double sent over

public void setSide2(double temp)

{

side2 = temp;

}

//sets side3 to the double sent over

public void setSide3(double temp)

{

side3 = temp;

}

//gets the area of a triangle

public double getArea()

{

double s = (side1 + side2 + side3) / 2; //finds s (needed for the area equation)

return Math.sqrt(s \* (s - side1) \* (s - side2) \* (s - side3)); //caculates and returns the area

}

//gets the perimeter of a triangle

public double getPerimeter()

{

return (side1 + side2 + side3); //caculates and returns the perimeter

}

//displays the info about a triangle

public String toString()

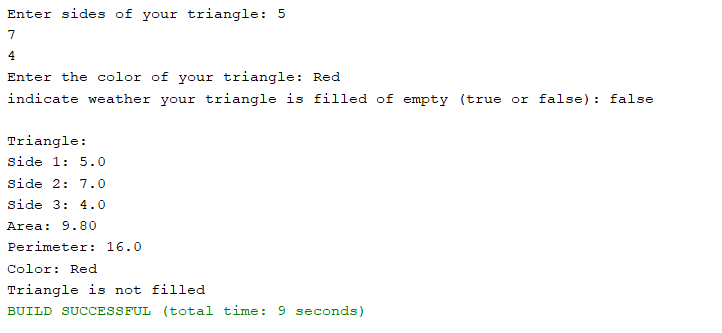
{

return "Triangle:\nSide 1: " + side1 + "\nSide 2: " + side2 + "\nSide 3: " + side3;

}

}

Output:



**Program Ten Part Three:**

//summary: This program uses classes to create many classes that define different people. Each type of person has there

//own specific data, and some are super classes/sub classes of other classes.

//name: Jenna Wolf

//class: Fundamentals of Programming, CS155 - 01

//instructor: Dr. Art Kazmierczak

//date: 10/24/2023

public class Main

{

public static void main(String[] args)

{

//creates and holds data for each class and uses its constructor to set its data

person Mike = new person("Mike", "1313 Farewell Street", "7682232341", "MikeL13.aol.com");

student Liam = new student("Liam", "7612 Hideaway Lane", "6452234342", "Liam.Wart04@gmail.com", student.JUNIOR);

employee Carol = new employee("Carol", "6542 Klark Avenue", "7223431987", "Carolpharris@gmail.com", 17, 45600);

faculty Hannah = new faculty("Hannah", "9231 Orange Lane", "2324467545", "HannahBanana@aol.com", 34, 65000,

"MWF 3:00-4:00", "Teacher");

staff Ian = new staff("Ian", "4432 Gumball Street", "5334236544", "IanCornwall@gmail.com", 10, 39000, "Janitor");

System.out.println(Mike.toString()); //outputs the data for Mike

System.out.println(Liam.toString()); //outputs the data for Liam

System.out.println(Carol.toString()); //outputs the data for Carol

System.out.println(Hannah.toString()); //outputs the data for Hannah

System.out.println(Ian.toString()); //outputs the data for Ian

}

}

**Person Class:**

public class person{

private String name; //holds the name data

private String address; //holds the address data

private String phoneNumber; //holds the phoneNumber data

private String email; //holds the email data

//no args constructor

public person(){

}

//constructor that has 4 String varaibles sent over to it

public person(String temp1, String temp2, String temp3, String temp4){

name = temp1; //name is set to temp1

address = temp2; //address is set to temp2

phoneNumber = temp3; //phoneNumber is set to temp3

email = temp4; //email is set to temp4

}

//returns the name data

public String getName()

{

return name;

}

//returns the address data

public String getAddress()

{

return address;

}

//returns the phoneNumber data

public String getPhoneNumber()

{

return phoneNumber;

}

//returns the email data

public String getEmail()

{

return email;

}

//sets name to the String sent over

public void setName(String temp)

{

name = temp;

}

//sets Address to the String sent over

public void setAddress(String temp)

{

address = temp;

}

//sets phoneNumber to the String sent over

public void setPhoneNumber(String temp)

{

phoneNumber = temp;

}

//sets email to the String sent over

public void setEmail(String temp)

{

email = temp;

}

//Sends back a string with the class name and name data

public String toString()

{

return "person, " + getName();

}

}

**Student Class:**

public class student extends person{

public static int FRESHMAN = 1; //is a constant named FRESHMAN and is set to 1

public static int SOPHOMORE = 2; //is a constant named SOPHOMORE and is set to 2

public static int JUNIOR = 3; //is a constant named JUNIOR and is set to 3

public static int SENIOR = 4; //is a constant named SENIOR and is set to 4

private int grade;

//no args constructor

public student(){

}

//constructor that takes in four strings and 1 ints for the data values

public student(String temp1, String temp2, String temp3, String temp4, int temp5){

super(temp1, temp2, temp3, temp4);

grade = temp5;

}

//sends back the grade data (sends a string depending on grades data)

public String getGrade()

{

//if loop that sends back a string based on the number

if(grade == 1)

return "Freshman";

else if(grade == 2)

return "Sophomore";

else if(grade == 3)

return "Junior";

else if(grade == 4)

return "Senior";

else

return "Error";

}

//sets the grade to the data sent over (as long as its 1-4)

public void setGrade(int temp)

{

//makes sure grade is between 1 and 4

if(temp < 5 && temp > 0)

grade = temp;

}

//Sends back a string with the class name and name data

public String toString()

{

return "student, " + getName();

}

}

**Employee Class:**

public class employee extends person{

private int office; //holds the office data

private double salary; //holds the salary data

private MyDate dateHired = new MyDate(); //holds the dateHired data and is set to a new MyDate

//no args constructor

public employee(){

}

//constructor that takes in four strings and int and a double and sets the values

public employee(String temp1, String temp2, String temp3, String temp4, int temp5, double temp6){

super(temp1, temp2, temp3, temp4);

office = temp5;

salary = temp6;

dateHired = new MyDate();

}

//returns the office data

public int getOffice()

{

return office;

}

//returns the salary data

public double getSalary()

{

return salary;

}

//returns the dateHired data (in string format)

public String getDateHired()

{

return dateHired.getMonth() + "/" + dateHired.getDay() + "/" + dateHired.getYear();

}

//sets office to the data sent over

public void setOffice(int temp)

{

office = temp;

}

//sets salary to the data sent over

public void setSalary(int temp)

{

salary = temp;

}

//sets dateHired to a new MyDate

public void setDateHired()

{

dateHired = new MyDate();

}

//Sends back a string with the class name and name data

public String toString()

{

return "employee, " + getName();

}

}

**Faculty Class:**

public class faculty extends employee{

private String officeHours; //holds the officeHours data

private String rank; //holds the rank data

//no args constructor

public faculty(){

}

//takes in six strings an int and a double and sets the values

public faculty(String temp1, String temp2, String temp3, String temp4,

int temp5, double temp6, String temp7, String temp8){

super(temp1, temp2, temp3, temp4, temp5, temp6);

officeHours = temp7;

rank = temp8;

}

//returns the officeHours data

public String getOfficeHours()

{

return officeHours;

}

//returns the rank data

public String getRank()

{

return rank;

}

//sets officeHours to the String sent over

public void setOfficeHours(String temp)

{

officeHours = temp;

}

//sets rank to the String sent over

public void setRank(String temp)

{

rank = temp;

}

//Sends back a string with the class name and name data

public String toString()

{

return "faculty, " + getName();

}

}

**Staff Class:**

public class staff extends employee{

private String title; //holds the title data

//no arg constructor

public staff(){

}

//takes in five strings an int and a double and sets the values

public staff(String temp1, String temp2, String temp3, String temp4,

int temp5, double temp6, String temp7){

super(temp1, temp2, temp3, temp4, temp5, temp6);

title = temp7;

}

//returns the title data

public String getTitle()

{

return title;

}

//sets title to the string sent over

public void setTitle(String temp)

{

title = temp;

}

//Sends back a string with the class name and name data

public String toString()

{

return "staff, " + getName();

}

}

**My Date Class:**

import java.util.GregorianCalendar; //lets the Gregorian Calendar be used

public class MyDate{

private int year; //holds the year data

private int month; //holds the month data

private int day; //holds the day data

//no arg constructor

public MyDate(){

}

//constructor that takes in a long and uses the setDate method

public MyDate(long temp){

setDate(temp);

}

//constructor that takes in three ints and sets the variables

public MyDate(int temp1, int temp2, int temp3){

year = temp1;

month = temp2;

day = temp3;

}

//returns the year data

public int getYear()

{

return year;

}

//returns the month data

public int getMonth()

{

return month;

}

//returns the day data

public int getDay()

{

return day;

}

//sets the date using the Gregorian Calendar

public void setDate(long temp)

{

GregorianCalendar calander = new GregorianCalendar(); //creaetes a Gregorian Calendar variable

calander.setTimeInMillis(temp); //uses the long to set the Gregorian Calendar

year = calander.get(GregorianCalendar.YEAR); //sets the year

month = calander.get(GregorianCalendar.MONTH); //sets the month

day = calander.get(GregorianCalendar.DAY\_OF\_MONTH); //sets the day

}

}

Output:

