



/\*

\* TestCar Class

\* Contains the Main Method

\* Creates Car, Engine, and Wheel objects and assigns values

\* Performs 3 trips, each with different fuel levels

\* The Car file has a printState method that is ran in this class.

\* it prints out currents levels of the variables as well as totals

\*/

public class TestCar {

public static void main(String[] args){

// create objects and add to car object

Car car = new Car("X7");

Engine engine = new Engine("DX7", 43);

car.add(engine);

Wheel wheel = new Wheel("Wichelin15", 15);

car.add(wheel);

// the setFuel method sets the amount of fuel in the engine

// the drive method "drives" the car aka performs all the calculations

// printState method prints out all the information about the trip

// trip 1

car.setFuel(100);

car.drive();

car.printState();

// trip 2

car.setFuel(80);

car.drive();

car.printState();

// trip 4, testing negative number

car.setFuel(-20);

car.drive();

car.printState();

// trip 5, testing for float number

car.setFuel(56.1F);

car.drive();

car.printState();

// trip 6, testing with irrational number

car.setFuel((float)Math.PI);

car.drive();

car.printState();

}

}

-------------------------------------------------------------------------------------------------------------

/\*\*

\* Car class

\* Car class will have an engine object (which will be then connected to the wheel object)

\* Support to add Engine object and add wheel object to the engine (abstraction)

\* accessor/mutator methods are called getters / setters

\* Will have a printState() method that will print out the desired information about the car and trip

\*/

public class Car{

// declare and initilize all variables we need for class Car

private String name ="";

private float distance = 0;

private float totalKm = 0;

private Engine engine;

// constructor

public Car(String name)

{

this.name = name;

}

// add method has ability to add both the engine and engine.engine.wheel

public void add(Engine engine){

this.engine = engine;

}

// uses the method add() inside of engine class to add the wheel to the engine from the Car class

public void add(Wheel wheel){

engine.add(wheel);

}

// setters

// sets the fuel of the engine by running engine's method setFuel()

// if it detects that we entered a negative fuel level, it will treat it as a mistake and reverse it

// -x = |x|

public void setFuel(float floatLevel){

if(floatLevel<0){

System.out.println("Negative fuel!, Absoluting value.");

fuelLevel \*= -1;

}

engine.setFuel(floatLevel);

}

// getters

public float getFuel(){

return engine.getFuel();

}

// run the car method

// calculates distance based on circumference of the weel \* engine.wheel turns per litre \* litres in engine

// calculates the total distance travelled by summing up all previous distances travelled

public void drive(){

distance = engine.getWheel().getCircumference() \* engine.getTpl() \* engine.getFuel();

totalKm += distance;

engine.incrementTurns(engine.getTpl()\*engine.getFuel());

engine.useFuel(engine.getInitialFuelLevel());

}

// print method

// prints out the info about the car, engine, and engine.wheel.

// prints out engine.wheel circumference, distance travelled on trip, total distanced travelled and the remaining fuel after trip.

public void printState(){

System.out.println("Current fuel: " + engine.getInitialFuelLevel());

System.out.printf("Confiuration: Car Body %s\n", name);

System.out.println("Engine name: " + engine.getName());

System.out.println("Engine turns per litre: " + engine.getTpl());

System.out.println("Engine's total turn count: " + engine.getTurns());

System.out.println("Wheel name: " + engine.getWheel().getName());

System.out.printf("Wheel radius: %.2f\n", engine.getWheel().getRadius());

System.out.printf("Wheel circumference (distance travelled per turn): %.2f\n", engine.getWheel().getCircumference());

System.out.printf("Distance this trip: %.2f\n", distance);

System.out.printf("total km travelled: %.2f\n", totalKm);

System.out.printf("Current fuel status: %.2f\n\n\n", getFuel());

}

}

----------------------------------------------------------------------------------------------------------------------

/\*

\* Engine class

\* Creates wheel object

\* setters and getters for private variables

\*/

public class Engine{

// initilize variables

private String name = "";

private int tpl = 0; // turns per litre (engine efficiceny)

private int totalNumTurns = 0; // total number of engine turns

private float fuelLevel = 0;

private float initialFuelLevel = 0; // keep track of how much fuel we had at the beginning

private Wheel wheel; // creating an undeclared variable of the wheel class later to be added

// Constructor

public Engine(String name, int tpl)

{

this.name = name;

this.tpl = tpl;

}

// ####################

// #### setters ######

// ####################

public void setFuel(float fuelLevel){

this.fuelLevel = fuelLevel;

initialFuelLevel = fuelLevel;

}

// decreases fuelLevel by the amount

public void useFuel(float amount){

this.fuelLevel -= amount;

}

// The wheel class is the parameter for this method

public void add(Wheel wheel){

this.wheel = wheel;

}

// ####################

// #### getters ######

// ####################

public float getFuel(){

return fuelLevel;

}

public float getInitialFuelLevel(){

return initialFuelLevel;

}

public int getTurns(){

return totalNumTurns;

}

public int getTpl(){

return tpl;

}

public String getName(){

return name;

}

public Wheel getWheel(){

return wheel;

}

// increments the amount of turns of the engine

public void incrementTurns(float turns){

totalNumTurns += (int)turns;

}

}

------------------------------------------------------------------------------------------------------------------------

/\*

\* Wheel class

\* calculates the circumference of the wheel

\* simple getters for external classes

\*/

public class Wheel {

// initilize and declare variables

private String name = "";

private float radius = 0;

private float circumference = 0;

// constructor, takes in name and radius

public Wheel(String name, float radius){

this.name = name;

this.radius = radius;

circumference = (float)(this.radius \* 2 \* Math.PI); // circumference of the wheel

}

// getters

public float getCircumference(){

return circumference;

}

public float getRadius(){

return radius;

}

public String getName(){

return name;

}

}