

PMR: This program was difficult for me. I had to do online research to figure out what to do and ended with a simple program for cars. The online notes were little to no help to me, and most of my knowledge came from online.

/\*\*

\* Project title: 08.03 Default constructors.

\*

\* Purpose of Project: Calculates the gas mileage on an example car.

\*

\* @version 11/24/2019

\*

\* @author Anika Jallipalli

\*\*/

public class CarV3 {

private String carType;

private double costOfTrip;

private double milesPerGallon;

private double gallonsPerMile;

private double gallonsUsed;

private double pricePerGallon;

private int endMiles;

private int startMiles;

//Default constructor

public CarV3() {}

public CarV3(String type, int endMi, int startMi, double galUsed, double pricePerGal, double costOfGals, double milesPerGal, double galsPerMile) {

pricePerGallon = pricePerGal;

costOfTrip = costOfGals;

milesPerGallon = milesPerGal;

gallonsPerMile = galsPerMile;

carType = type;

endMiles = endMi;

startMiles = startMi;

gallonsUsed = galUsed;

}

public int calcDistance() {

return endMiles - startMiles;

}

public double calcMPG() {

return ((double)calcDistance()) / gallonsUsed;

}

public double calcGPM() {

return gallonsUsed / calcDistance();

}

public double calcCost() {

double cost = (gallonsUsed \* pricePerGallon);

return cost;

}

public static void main(String[] args) {

double gals;

int sMiles;

int eMiles;

int distance;

double mpg;

String car;

double price;

double cost;

gals = 5.5;

sMiles = 110000;

eMiles = 110100;

price = 2.11;

car = "2012 Hyundai Sonata";

CarV3 car1 = new CarV3(car, eMiles, sMiles, gals, price, gals \*price, ((eMiles - sMiles)/gals), gals/price);

cost = car1.calcCost();

distance = car1.calcDistance();

mpg = car1.calcMPG();

System.out.println(" Gas Mileage Calculations ");

System.out.println(" Type of Car Start Miles End Miles Distance Gallons Price Cost Miles/Gal Gal/Mile");

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

System.out.printf("%3s %10d %11d %11d %11.1f %7.1f %8.1f %8.1f %10.2f\n", car1.carType, car1.startMiles, car1.endMiles, car1.calcDistance(), car1.gallonsUsed, car1.pricePerGallon, car1.calcCost(), car1.calcMPG(), car1.calcGPM());

}

}