|  |
| --- |
| using System;  using System.Collections.Generic;  using System.Linq;  namespace SpeedRacing  {  class Program  {  static void Main(string[] args)  {  int carsCount = int.Parse(Console.ReadLine());  var carsList = new List<Car>();  for (int i = 0; i < carsCount; i++)  {  var carsInfo = Console  .ReadLine()  .Split()  .ToList();  string carModel = carsInfo[0];  double carFuelAmount = double.Parse(carsInfo[1]);  double carFuelConsumptionPerKm = double.Parse(carsInfo[2]);  double startingKm = 0.0;  Car car = new Car(carModel, carFuelAmount, carFuelConsumptionPerKm, startingKm);  carsList.Add(car);  }  string input;  while ((input = Console.ReadLine()) != "End")  {  var commands = input  .Split()  .ToList();  string carModel = commands[1];  double amountOfKm = double.Parse(commands[2]);  foreach (var item in carsList)  {  if (item.Model == carModel)  {  double currentCarTotalFuelConsumption = amountOfKm \* item.FuelConsumptionPerKm;  if (currentCarTotalFuelConsumption <= item.FuelAmount)  {  item.FuelAmount -= currentCarTotalFuelConsumption;  item.TraveledDistance += amountOfKm;  }  else  {  Console.WriteLine("Insufficient fuel for the drive");  }  }  }  }  foreach (var item in carsList)  {  Console.WriteLine($"{item.Model} {item.FuelAmount:f2} {item.TraveledDistance}");  }  }  }  class Car  {  public string Model { get; set; }  public double FuelAmount { get; set; }  public double FuelConsumptionPerKm { get; set; }  public double TraveledDistance { get; set; }  public Car(string model, double fuelAmount, double fuelConsumptionPerKm, double traveledDistance)  {  this.Model = model;  this.FuelAmount = fuelAmount;  this.FuelConsumptionPerKm = fuelConsumptionPerKm;  this.TraveledDistance = traveledDistance;  }  public void CalculatingDoesCarCanMakeThatRide(string model, double fuelAmount, double fuelConsumptionPerKm, double traveledDistance)  {  this.Model = model;  this.FuelAmount = fuelAmount;  this.FuelConsumptionPerKm = fuelConsumptionPerKm;  this.TraveledDistance = traveledDistance;  }  }  } |