|  |
| --- |
| using System;  using System.Collections.Generic;  using System.Linq;  using System.Runtime.InteropServices.ComTypes;  namespace RawData  {  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 input = Console  .ReadLine()  .Split()  .ToList();  string model = input[0];  int engineSpeed = int.Parse(input[1]);  int enginePower = int.Parse(input[2]);  int cargoWeight = int.Parse(input[3]);  string cargoType = input[4];  Engine engine = new Engine(engineSpeed, enginePower);  Cargo cargo = new Cargo(cargoWeight, cargoType);  Car car = new Car(model, engine, cargo);  carsList.Add(car);  }  string command = Console.ReadLine();  switch (command)  {  case "fragile":  carsList = carsList  .Where(x => x.Cargo.CargoType == "fragile" && x.Cargo.CargoWeight < 1000)  .ToList();  Console.WriteLine(string.Join(Environment.NewLine, carsList));  break;  case "flamable":  carsList = carsList  .Where(x => x.Cargo.CargoType == "flamable" && x.Engine.EnginePower > 250)  .ToList();  Console.WriteLine(string.Join(Environment.NewLine, carsList));  break;  }  }  }  class Car  {  public string Model { get; set; }  public Engine Engine { get; set; }  public Cargo Cargo { get; set; }  public Car(string model, Engine engine, Cargo cargo)  {  this.Model = model;  this.Engine = engine;  this.Cargo = cargo;  }  public override string ToString()  {  return $"{Model}";  }  }  class Engine  {  public int EngineSpeed { get; set; }  public int EnginePower { get; set; }  public Engine(int engineSpeed, int enginePower)  {  this.EngineSpeed = engineSpeed;  this.EnginePower = enginePower;  }  }  class Cargo  {  public int CargoWeight { get; set; }  public string CargoType { get; set; }  public Cargo(int cargoWeight, string cargoType)  {  this.CargoWeight = cargoWeight;  this.CargoType = cargoType;  }  }  } |