

Question - 1

Java_PRA2_Vehicle_min_Search

Create a class Vehicle with below attributes:

number - int

name - String

price - double

Write getters, setters and parameterized constructor in the above mentioned attribute sequence as required.

Create class Solution with main method

Implement two static methods - findVehicleWithMinimumPrice and searchVehicleByName in Solution class.

findVehicleWithMinimumPrice

Create a static method findVehicleWithMinimumPrice in the Solution class. This method will take array of Vehicle objects and returns the Vehicle object having the minimum Price if found else return null if not found.

searchVehicleByName

Create a static method searchVehicleByName in the Solution class. This method will take array of Vehicle objects and Name as input and returns the Vehicle object having the mentioned Name if found else return null if not found.

These methods should be called from the main method.

Write code to perform the following tasks:

1. Take necessary input variable and call findVehicleWithMinimumPrice. For this method - The main method should print the Vehicle object with minimum of mentioned attribute as it is if the returned value is not null, or it should print "No Vehicle found with mentioned attribute".

2. Take necessary input variable and call searchVehicleByName. For this method - The main method should print the Vehicle object details as it is, if the returned value is not null, or it should print "No Vehicle found with mentioned attribute".

The above mentioned static methods should be called from the main method. Also write the code for accepting the inputs and printing the outputs. Don't use any static test or formatting for printing the result. Just invoke the method and print the result

Note :

All String comparison needs to be case in-sensitive

You can use/refer the below given sample input and output to verify your solution.

Sample Input (below) description:

The 1st input taken in the main section is the number of Vehicle objects to be added to the list of Vehicle.

The next set of inputs are number,name,price for each Vehicle object taken one after other and is repeated for number of Vehicle objects given in the first line of input.

The last line of inputs will be the arguments which needs to be passed as parameter to the methods.

Consider below sample input and output to test your code:

Sample Input 1 - public

4

3111

bus

1000.0

4112

crane

1800.0

5114

Tractor

2400.0

6115

Scooter

2700.0

Scooter

Sample Output 1

number-3111

name-bus

price-1000.0
number-6115
name-Scooter
price-2700.0

Sample Input 2 - public

4
1109
Train
2000.0
5104
taxi
2800.0
4112
bike
5600.0
8107
Scooter
3300.0
bike

Sample Output 2

number-1109
name-Train
price-2000.0
number-4112
name-bike
price-5600.0