Question - 1
Truck_Min_Capacity

Create a class **Truck** with the below attributes:

- id- int
- ownerName - String
- capacity - int
- rate- int

where id is the id of the truck, ownerName is the name of the truck owner, capacity is the capacity of the truck in tons and rate is the hourly rate of the truck in rupees.

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

Create a class **Solution** with the main method

Implement below two static methods in the Solution class. searchTruckByOwnerName and findTruckWithMinCapacity

1. searchTruckByOwnerName
Create a static method searchTruckByOwnerName in the Solution class. This method will take an array of Truck objects and ownerName as input and returns the Truck object having the mentioned ownerName if found, else return null if not found.

2. findTruckWithMinCapacity
Create a static method findTruckWithMinCapacity in the Solution class. This method will take an array of Truck objects and returns the Truck object having the minimum capacity among all trucks if found, else return null if not found.

These methods should be called from the main method.

Write code in the main method to perform the following tasks:
1. Take the necessary input variable and call searchTruckByOwnerName. For this method -The main method should print the Truck object details as it is if the returned value is not null, or it should print "No Truck found with mentioned attribute".
2. Take the necessary input variable and call findTruckWithMinCapacity. For this method - The main method should print the Truck object details as it is, if the returned value is not null, or it should print "No Truck found with mentioned attribute".

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.

All String comparison needs to be case-insensitive.

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

Sample Input (below) Description:

1. The 1st input taken in the main section is the number of Truck objects to be added to the list of Truck objects.
2. The next set of inputs is id, ownerName, capacity, and rate for each Truck object taken one after the other.
3. The last line of inputs will be the arguments that need to be passed as parameters to the methods.

Consider below sample input and output to test your code:

Sample Input 1

1/2

```
4
101
Ron
5
2500
102
Preetha
4
2000
103
Madhav
6
3000
104
Alex
3
2000
Ron
```

Sample Output 1

```
101
Ron
5
2500
104
Alex
3
2000
```

Sample Input 2

```
4
501
Alex
4
2000
502
Meera
6
3500
503
Prasanth
5
2500
504
Saurabh
3
2000
Ron
```

Sample Output 2

```
No Truck found with mentioned attribute
504
Saurabh
3
2000
```