Page of

p { margin-bottom: 0.25cm; direction: ltr; line-height: 120%; text-align: left; } **Car Pooling System - Requirement 2**

You are a very active member of a Nature Club in your organization. In one of the meetings, it was discussed to build a car pooling system to help cut down the pollution. Being very active and tech savvy, you wish to contribute towards the development of system. One of the members being an architect has understood the requirement and would be sharing you with smaller requirements.

**Requirement 2:**

The next most important entity of the system is Car.  
  
 a)   Create a Car Class with the following attributes:

|  |  |
| --- | --- |
| **Member Field name** | **Type** |
| id | Long |
| name | String |
| model | String |
| makeYear | Integer |
| company | String |
| comfortLevel | Integer |

b)    Mark all the attributes as private  
  
 c)   Create / Generate appropriate Getters & Setters  
  
 d)   Add a default constructor and a parameterized constructor to take in all attributes.  
  
 e)   One Member can pool one or many cars that he / she owns. To accommodate this, Create a MemberCar class with following attributes

|  |  |
| --- | --- |
| **Member Field name** | **Type** |
| id | Long |
| member | Member |
| car | Car |
| carRegistrationNumber | String |
| carColor | String |

Create / generate appropriate Getters & Setters and constructors.  
  
f)    Add a static findMember method in Member class which takes id as input and returns the Member or null object if the member object is not found.  
  
 g)   Add a static findCar method in Car class which takes id as input and returns the Car or null object if the car object is not found.  
  
 h)   Introduce an Arraylist (carList) in the member class which holds the list of member-car objects. Include appropriate getters and setters. Given a member, Display the number of cars the member owns along with registration number of each car separated by a line.  
  
**Input Output Format:**  
Refer Sample I/O for formatting specifications.  
  
  
**Sample Input and Output:**  
**[All text in bold corresponds to input and the rest corresponds to output.]**  
  
Menu:  
 1) Add a Member  
 2) Add a Car  
 3) Assign Car to Member (Assume valid data is supplied)  
 4) Cars Owned  
 5) Exit  
**1**  
id:  
**6**  
first name:  
**arun**  
last name:  
**kumar**  
email:  
**arun@gmail.com**  
contact number:  
**9089786756**  
license number:  
**TN12ER3423**  
license start date:  
**13-12-1998**  
license expiry date:  
**12-12-2008**  
Menu:  
 1) Add a Member  
 2) Add a Car  
 3) Assign Car to Member (Assume valid data is supplied)  
 4) Cars Owned  
 5) Exit  
**2**  
id:  
**5**  
name:  
**Verna**  
model:  
**CRDI**  
makeYear:  
**2008**  
company:  
**Hyundai**  
comfort level:  
**8**  
Menu:  
 1) Add a Member  
 2) Add a Car  
 3) Assign Car to Member (Assume valid data is supplied)  
 4) Cars Owned  
 5) Exit  
**3**  
member car id  
**12**  
member id  
**3**  
car id  
**5**  
car registration  
**TN38BR9689**  
color  
**White**  
Menu:  
 1) Add a Member  
 2) Add a Car  
 3) Assign Car to Member (Assume valid data is supplied)  
 4) Cars Owned  
 5) Exit  
**3**  
member car id  
**13**  
member id  
**6**  
car id  
**2**  
car registration  
**TN66AB4214**  
color  
**Brown**  
Menu:  
 1) Add a Member  
 2) Add a Car  
 3) Assign Car to Member (Assume valid data is supplied)  
 4) Cars Owned  
 5) Exit  
**4**  
member id  
**6**  
Number of cars : 1  
Registration Numbers :  
TN66AB4214  
Menu:  
 1) Add a Member  
 2) Add a Car  
 3) Assign Car to Member (Assume valid data is supplied)  
 4) Cars Owned  
 5) Exit  
**5**