**National University of Computer & Emerging Sciences**

**Karachi Campus**



**Project Report**

**CS217-Object Oriented Programming**

**Department: BS(CS)**

**Section: J**

**[Car Management System]**

**Group Members:**

**19K-0330 Ziyaan Ali**

**19K-1331 Ali Hasnain Ladhani**

**19K-0209 Syed Muhammad Saad**

* **Motivation:**

I am grateful to few sources that supported us throughout the project to overcome all the problems and complete the project, for which I thank Sir Danish, Miss Nida and Stack Overflow.

* **Introduction:**

We are developing a program in which we would be handling all the necessary details about cars in a store / garage / showroom from the phase of data to be entered by the owner till the buying process of the costumer and would maintain it at a particular place.

So, mainly there are 2 different setups/mode.

1. Owner
2. Customer.

Owner can access all day and fill all the information about the car from buying to selling phase. In our customer mode customer can view all the cars available by their kind of specifications they provide, and while selling phase owner will be marking or tracking /filling all the necessary information he must have to maintain a record of the car and customer too in a legal manner.

* **Methodology of OOP:**

1. **Encapsulation (data Hiding) + Inheritance:**

The use of Accessibility modes that are public, private and protected ensures the role of encapsulation. In Our project these features are used as we are making classes and Our classes , the member data s and functions like entering the data of customer while buying can only be filled by the owner and owner can view anything he want , so here it means that all the data of a car is inherited and is managed by the owner of the store and he is the only accesses of everything and while calling anything directly from the main body costumer cannot access or change any data without owner’s permission which means data has been hidden or in safe mode.

1. **Polymorphism:**  
   The use of function, operator overloading and binding is included our program. In our program we are using hierarchical as well as multiple inheritance.

* **Pseudo code:**

1. Class Car, class tyres, class brand, and class model are pure abstract class. We made them pure abstract class because we don’t want the object of any of these.
2. Then there is class customer which is multiple inherited with all above classes.
3. We made a multiple inheritance between classes to provide a common interface and to access the features of car by customer and then single inherited with owner to control the whole execution.
4. By multiple inheritance customer has all functionality of all above classes explained and also their functionality.
5. The authority to use and access the code is for owner, so that’s why owner class is single inherited with customer class. Then it also has all the method and data of inherited classes.
6. There is also a rule of polymorphism between classes, in car, tyres, brand, and model class all functions are pure virtual. Then after inheritance they all are define in other classes (function overriding).
7. Owner has the authority to change any data (Data hiding/abstraction).
8. This program is structured to manage the data for cars.
9. It has functionality to add the record in previous record.
10. Owner can change any record or update any record.
11. If customer wants to search any record/data owner help to show these data.
12. If customer wants to buy any car, so owner has authority to sell his car. And automatically it deletes the record from unsold car and save into another separate file of sold cars

* **Problem:**

As seen nowadays due to increase in crime rate, cars are snatched or stole, so to inquire about that to whom it belongs it may lack the management or proper data handling.

* **Class Diagram:**

|  |
| --- |
| CAR |
| \*type char |
| \*color char |
| \*fueltype char |
| \*cn char |
| \*registrationnumber char |
| \*taxprice float |
| \*totalprice float |
| +Scarname() virtual void |
| +assigncolor() virtual void |
| +assignnoofgear() virtual void |
| +assignfueltype() virtual void |
| +assigntype() virtual void |
| +assignprice() virtual void |
| +assigntax() virtual void |
| +assignregno() virtual void |
| +Gcarname() virtual inline void |
| +displaycolor()virtual inline void |
| +displaynoofgear()virtual inline void |
| +displayfueltype()virtual inline void |
| +displaytype()virtual inline void |
| +displayprice()virtual inline void |
| +displaytax() virtual inline void |
| +displayregno() virtual inline void |

|  |
| --- |
| -private |
| +public |
| \*protected |

|  |
| --- |
| TYRES |
| \*nooftyres int |
| \*warranty float |
| \*company char |
| \*assignnot() virtual void |
| \*assignwarranty() virtual void |
| \*assigncompany() virtual void |
| +displaynot() virtual inline void |
| +displaywarranty() virtual inline void |
| +displaycompany() virtual inline void |

|  |
| --- |
| BRAND |
| \*name char |
| \*lrange float |
| \*rrange float |
| \*yearsince int |
| \*assignbrand() virtual void |
| \*assignrange() virtual void |
| \*assignyearsince() virtual void |
| \*displaybrand() virtual inline void |
| \*displayrange() virtual inline void |
| \*displayyearsince() virtual inline void |

|  |
| --- |
| MODEL |
| \*year int |
| \*enginenumber char |
| \*assignyear() virtual void |
| \*assignengno() virtual void |
| +displayyear() virtual void |
| +displayengno() virtual void |

|  |
| --- |
| OWNER |
| \*ownername char |
| \*carstatus char |
| +noc int |
| +owner() |
| +assignowname() void |
| +assignnoc() void |
| +assigncs() void |
| +displaycs() void |
| +changetprice() void |
| +changeall() void |
| +changebychoice() void |
| +displayoall() void |
| +displaybyochoice() void |
| +ownerdisplay() void |
| +assigner() void |
| + E\_Time() void |
| + write\_data() void |
| +readall\_data() void |
| + sell() void |
| + read\_CarDetail() void |
| +read\_BrandDetail() void |
| +read\_ModelDetail() void |
| +read\_TyreDetail() void |
| +updateall() void |

|  |
| --- |
| CUSTOMER |
| +cname char |
| +address char |
| +licensen char |
| +pno char |
| +cnic char |
| +design() void |
| +assign () void |
| +assigcname() void |
| +assignaddress() void |
| +assignlicensen() void |
| +assignpno() void |
| +assigncnic() void |
| +displaymydetails() inline void |
| +displayall() void |
| +displaybychoice() void |
| +readall\_data() virtual void |
| +read\_CarDetail() virtual void |
| +read\_BrandDetail() virtual void |
| +read\_ModelDetail() virtual void |
| +read\_TyreDetail() virtual void |
| +CAR CLASS FUNCTIONS |
| +BRAND CLASS FUNCTIONS |
| +TYRES CLASS FUNCTIONS |
| +MODEL CLASS FUNCTIONS |

* **Relationship:**
* **Conclusion:**

In conclusion, our program is secured, user friendly and easy to use by any one, in which the owner has its own interface which cannot be accessed by the customers and customer can perform their respective tasks as a customer without having any technical issues or further guidance. Therefore, beside learning new technologies and concept in future we are going to upgrade it further.