

Automobile Inventory and Stock Control System

Toyota Motor Corporation

ANALYSIS

1.1.a Introduction:

Automobile industry is one of the world's most important economic sectors by revenue. The industry covers a vast range of companies and organizations involved in the design, development, and manufacture, marketing and selling of motor vehicles. The Automotive industry has been a growing field in Pakistan for a long time, however not as much established to be part of the prominent list of the top automotive industries. The Pakistani Automobile industry produces around 120,000 to 140,000 vehicles per year.

Despite significant production volumes, transfer of technology and localization of vehicle components remains low, and only a few car models are assembled in the country and so the customers have a very small variety of vehicles to choose from. The lack of competition in the local auto industry due to the presence of just three assemblers -and only one small car assembler- has resulted in slow technological advancement of the industry. Small cars produced by Pak Suzuki, the country's largest auto assembler are globally retired models making use of obsolete technology and not offering any safety features.

Currently some of the major world automakers have set up assembly plants or are in joint ventures with local companies. These include **Toyota**, **General** Honda and Nissan Motors. The total contribution of Auto industry to GDP in 2007 was 2.8% which is likely to increase up to 5.6% in the next 8 years. Auto sector presently, contributes 16% to the manufacturing sector which is predicted to increase 25% in the next 7 years. Many cars in the country have dual fuel options and run on CNG to increase affordability.

Pakistan's Ministry of Industries claims that the country produced its first vehicle in 1953, at the National Motors Limited. This was the Bedford truck, after which buses, light trucks and cars were assembled in the same plant. The industry was highly regulated until the early 1990s. After deregulation major Japanese manufacturers entered in the market thereby creating some competition in this sector. Assemblers of HI-NO Trucks, Suzuki Cars, Mazda Trucks, Toyota and Honda in particular, entered once deregulation was introduced.

Auto Sector remains the second largest payer of indirect taxes after the Petroleum Sector. In Pakistan's context there are 10 cars in 1,000 persons which is one of the lowest in the emerging economies which itself speaks of high potential of growth in the auto sector and more so in the car production. Rising per capita income with changing demographic distribution and an anticipated influx of 30 to 40 million young people in the economically active workforce in the next few years provides a stimulus to the industry to expand and grow

1.1. b.i Background

The company was founded by Kiichiro Toyoda in 1937 as a spinoff from his father's company Toyota Industries to create automobiles. Three years earlier, in 1934, while still a department of Toyota Industries, it created its first product, the Type A engine, and, in 1936, its first passenger car, the Toyota AA. Toyota Motor Corporation group companies are Toyota (including the Scion brand), Lexus, Daihatsu, and Hino Motors, along with several "nonautomotive" companies. TMC is part of the Toyota Group, one of the largest conglomerates in the world.

Toyota Indus Pakistan is the Pakistan division of Toyota Motor Corporation, the multinational automaker headquartered in Toyota, Aichi, Japan. In 2010, Toyota employed 325,905 people worldwide, and was the third-largest automobile manufacturer in 2011 by production behind General Motors and Volkswagen Group. Toyota is the eleventh-largest company in the world by revenue. In July 2012, the company reported it had manufactured its 200-millionth vehicle.

As of 2009, Toyota officially lists approximately 70 different models sold under its namesake brand, including sedans, coupes, vans, trucks, hybrids, and crossovers. Many of these models are produced as passenger sedans, which range from the subcompact Toyota Yaris, to compact Corolla, to mid-size Camry, and full-size Avalon. Vans include the Previa/Estima, Sienna, and Hiace. Several small cars, such as the Vitz and its variants are also being manufactured.

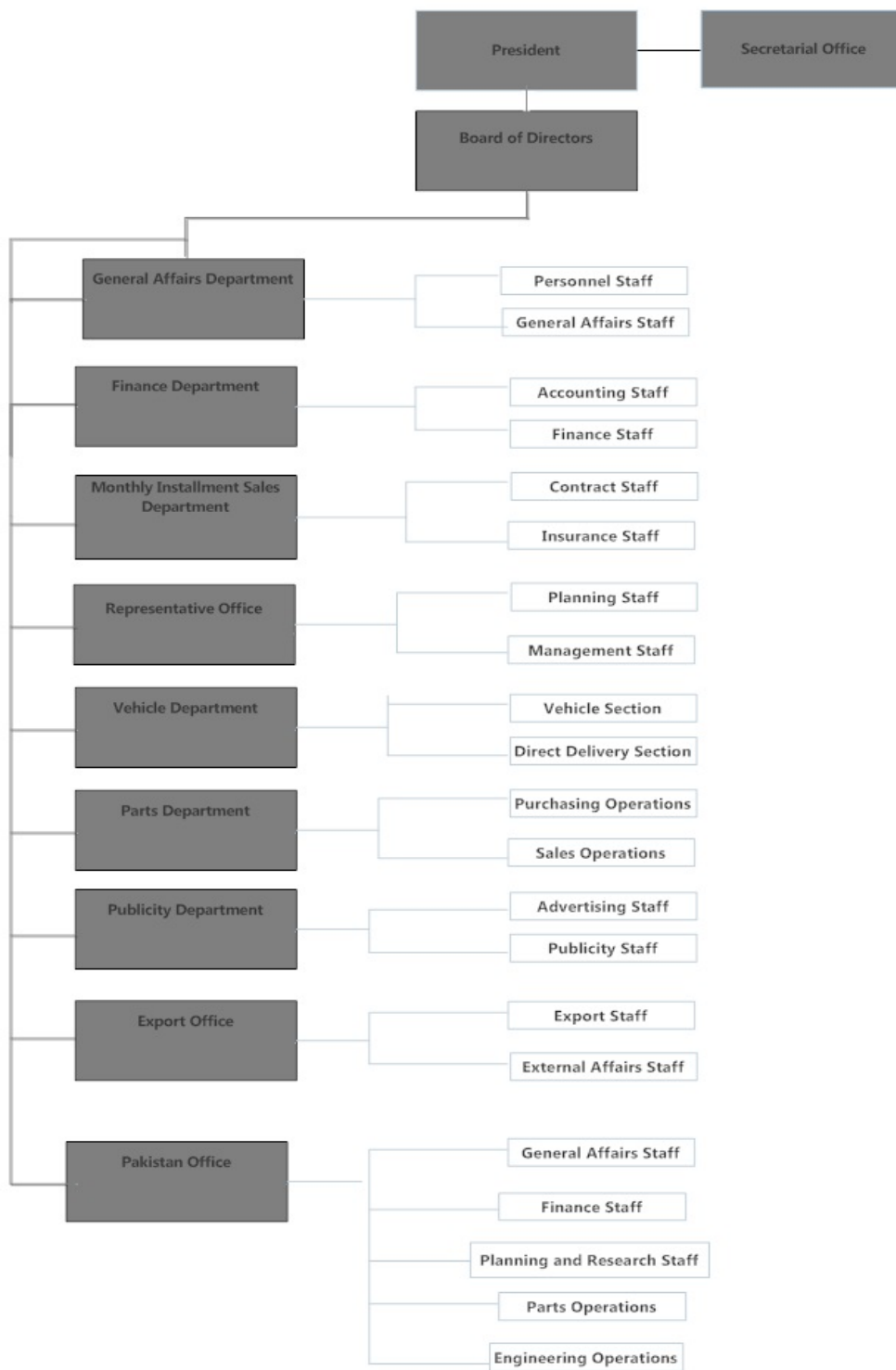
In 2011, the Toyota Group (including Daihatsu, Hino and Chinese joint ventures) fell to place three with 8,050,181 units produced globally. It regained its top rank with 9,909,440 units produced globally in 2012. On May 8, 2013, Toyota announced plans to produce 10.1 million units in fiscal 2013, which, if achieved, would make it the first auto manufacturer to cross the 10 million unit threshold.

On May 8, 2013, Toyota Motor Corporation announced its financial results for the fiscal year ended March 31, 2013. Net revenues totaled 22.0 trillion yen (US\$ 216.7 billion), Operating income was 1.32 trillion yen (US\$13 billion), net income 962.1 billion yen (US\$9.47 billion).

Toyota has been involved in many global motorsports series. They also represent their Lexus brand in other sports car racing categories. Toyota also makes engines and other auto parts for other Japanese motorsports including formula Nippon, Super GT, formula 3 and formula Toyota series. Toyota also runs a driver development programme known as TDP (Toyota Young Drivers Program) which they made for funding and educating future Japanese motorsports talent. Toyota Motorsport GmbH, with headquarters in Cologne, Germany has been responsible for Toyota's major motorsports development including Formula One, the World Rally Championship, the Le Mans Series and most recently the FIA World Endurance Championship. Toyota enjoyed success in all these motorsports categories.

Currently Toyota Indus Pakistan operates multiple factories in cities of the country, which include Karachi, Islamabad, Hyderabad, Lahore, Multan, Faisalabad and Peshawar.

1.1.b.ii Organization Chart



1.1.b.iii Functional Areas:

The company works in following areas:

- Manufacturing: Process of making products. (including brakes, engine, suspension etc)
- Sales: Selling of finished cars.

1.1.c Nature of Problem:

The main problem with the present system is that it is totally manual due to which the company faces many problems and sometimes it also gets late in submitting any kind of report or investigation. Each and every work is done manually except for printing of forms from Printing press.

Secondly it is also getting difficult for the company to manage data stored in different files. It also has tested its employees who are young, active & very responsibly carry on with their designated jobs.

Next, the data can never be organized in any manner as there are no links between two files so single record is to be written several times to maintain separate files, for example a vehicle's record on paper could not be attached to three files at the same time. Also, details of one car need to be seen in one place, which includes details about the engine, safety features of the car, interior features and exterior features. This is not possible without making a software interface.

The system is also becoming time consuming with new entries of vehicles. Sometimes the record manager also gets confused over the type of managing of files done by him (i.e. He has to update the data of a particular model of a vehicle and instead updates the wrong one.)

The method of processing is also very old causing it to be very slow and full of errors which creates problem for the financial staff. Important work is also done by estimation due to non-availability of calculation stationeries.

There is no surcharge on late fees as no proper fees invoice is available. This adds to the misery of the company that it is unable to calculate its monthly income on time.

Due to the data existing only on paper, it is extremely difficult to search and find a particular record.(For example finding the wheels used in a car, and having to search through papers to find the car which has the wheels and then looking for the wheels' record itself).

Rapidly expanding data makes it a tedious task to make sure that it is not lost. Physical files can be easily misplaced and a lot of time is wasted in finding them.

And at last the current system is insecure. Any person, or intruder or even unauthorized company employees can access data and can edit it with ease.

1.2 OBJECTIVES:

1.2.a. General Business Terms

1. To design a computerized system, that is visually appealing and user friendly for the user. It must also be able to manage all the records without confusing the user.
2. The program must link several files that provide collective information of a single relevant idea in an organized manner.
3. The designed program must be fast, based upon modern techniques along with data validations and checks, in order to eliminate errors.
4. Proper data reports must be printed along with information about the buyer, what parts he buys and how much stock was used. System should also allow the ability to reorder stock as needed and view a monthly report of how much stock was given and the financial transactions that will follow. This will give a new revolutionized look to the working of company with style and ease.
5. Security is also given priority keeping in view the threats faced by the present system.
6. Searching of records without having to go through manual records and wasting a lot of time.

1.2.b. Computer related terms:

1. To design a system using Microsoft Access for creating a data base and using Microsoft Visual Basic to create forms and graphical interface to enhance the system (by creating MDI and SDI forms.)
2. Linking database tables to each other and to forms to upgrade any record in any table easily without repeating the process several times.
3. Using validation checks to protect the records against any typing mistake.
4. Creating Data Report Forms to allow printing of monthly reports regarding stock used, stock reordered and how much stock has to be re-ordered.
5. Designing Login Forms to secure data. Upon providing the correct Username and Password, the user will be granted access to the information and records.
6. Easy searching of files to find particular records.

1.2. c. Specific Objectives:

1. To devise a system with a secured Username and Password.
 2. To be able to access different forms from a menu.
 3. To be able to add, edit, delete and save Buyer Records by clicking a button.
 4. To be able to add, edit, delete and save Vehicle Records by clicking a buttons only.
 5. To be able to add, edit, delete and save Dimensions Records by clicking button.
 6. To be able to add, edit, delete and save Engine Records by clicking a button.
 7. To be able to add, edit, delete and save Exterior Details Records by clicking a button.
 8. To be able to add, edit, delete and save Interior Details Records by clicking a button.
 9. To be able to add, edit, delete and save Order Records by clicking a button.
 10. To be able to add, edit, delete and save Performance Records by clicking a button.
 11. To be able to add, edit, delete and save Safety Details Records by clicking a button.
-
12. To be able to print a record with a click of a button.
 13. To be able to print records according to a specific category.
-
14. To be able to search for a Vehicle by Name, Model Code, Make, Year, Variant, Transmission Type, Class, Doors
 15. To be able to search for a Buyer by buyer ID, Title, City, Bank Name.
 16. To be able to search for Safety details by Safety ID and Vehicle ID.
 17. To be able to search for Performance details by PID (Performance ID) and Vehicle ID
 18. To be able to search for Order Details by OID(Order ID), Order Date and Buyer ID
 19. To be able to search for Interior Details by IID (Interior ID) and Safety ID
 20. To be able to search for Exterior Details by exterior ID and Dimensions ID.
 21. To be able to search for Engine Details by Engine ID and Vehicle ID.
 22. To be able to search for Dimensions Details by Dimensions ID and Vehicle ID.
-
23. To be able to browse through all records methodically
 24. To be able to print bills for a client on the basis of Purchased vehicle
 25. To be able to computerize all manual records stored on paper in the past
 26. To be able to make regular backups of any update being performed on the system.
 27. To be able to have validation checks on numeric data.
 28. To be able to have validation checks on the amount of characters allowed when entering information in a field.
 29. To check whether a unique code is already in use or not.
 30. To be able to reduce human errors as much as possible, for example, transcription, transposition and random errors
 31. To be able to cancel actions being formed by clicking a button.
 32. To be able to navigation from one form to another.

1.3 Description of Existing System:

1.3. a.i Data entry in existing system:

The existing system requires the data to be manually input from hard copies (written documents). This makes it hard to manage and edit. A form is given out to the buyer. The buyer fills it and sends it by post or delivers it to the appropriate office (For parts he delivers it to the Head office).

1.3. a.ii Order Form Details:

This form contains relevant information about the buyer, which includes buyer name/title, city, address, contact person, phone number, email, bank account details etc. Also, a waiver of liability is also signed to ensure that whatever order has been made is final and can't be changed and order can't be refunded. This waiver ensures that no misunderstanding takes place in the future. It also reminds both parties of their respective duties. Following this, 3 copies of the order are made. One of these is sent to the Head office to maintain a record. Second one is sent to the accounts office for financial records and the third is given to the buyer.

1.3. a.iii Order Processing:

After this the order is forwarded to the respective offices for fulfillment. In the current system, fulfillment of orders is linked to the amount of production of the specific parts for each car in sectors of "Engine", "Performance", "Exterior", "Interior" and "Safety". . However service is affected whenever any of the machines become faulty and efficiency is greatly reduced. The company tries to overcome this flaw by carrying out monthly check-ups of all machines.

1.3. a.iv Payment system:

Regarding payment of order, every buyer is reminded on the 1st day of each month to submit payment within 5 working days. Upon failure, an official notice is sent on the 7th day of the particular month. After two months of unpaid service the company files a lawsuit to claim the money. On the other hand, every buyer is provided with a payment card after submitting the payment of the order and it is returned to the company along with the payment for a new order. This card is kept as a record by the buyer that to ensure that they have fulfilled the payment for a particular order.

At the end of every fiscal year the total for each order is calculated. Each order is checked for fulfillment of payment and after this the records are updated for the year.

The following pages show the way data was manually being input in different parts of the company.

1.3.b Data Input requirement:

Buyer:

- Name of buyer (Only if buyer does not belong to a company)
- Buyer's company
- Address.
- City
- Phone number.
- Office phone number
- Email address
- Bank account number
- Bank branch

Vehicle Details:

- Quantity of engines of a specific specification required.
- Quantity of exterior packages of a specific specification required.
- Quantity of interior packages of a specific specification required.
- Quantity of performance feature packages of a specific specification required.
- Quantity of safety feature packages of a specific specification required.

Payment:

- Monthly payment
- Payment of specific company
- January/July
- February/August
- March/September
- April/October
- May/November
- June/December

1.3.c Data specifications:

Buyer:

- Name of buyer (Only if buyer does not belong to a company)
- Buyer's company
- Address.
- City
- Phone number.
- Office phone number
- Email address
- Bank account number
- Bank branch

Vehicle Details:

- Engine (Specific packages and their quantity)
- Exterior (Specific packages and their quantity)
- Interior (Specific packages and their quantity)
- Performance (Specific packages and their quantity)
- Safety Features (Specific packages and their quantity)

Payment:

- Monthly payment,
- Date of receiving,
- January/July,
- February/August,
- March/September,
- April/October,
- May/November,
- June/December.

1.3.d Data Processing Requirements:

In this company data processing is done in two areas:

1. Payment collection
2. Order being received and fulfilled.

When payment by a specific buyer is made, a slip is prepared and handed over to the buyer. Next, a copy of that particular slip is forwarded to records manager who updates the payment logs. This file is kept safe until the next transaction and money is deposited in local bank account of company.

Secondly the information of the order is transferred to the respective department. Following this, payment calculation is done and any arrears or surcharges are received from the buyer on the day of delivery.

These processes require some data about the buyer and company, which is as follows:

1. Personal data of buyer,
2. Order form must be present,
3. Payment history of the buyer,
4. Due date of payment,
5. Payment amount,
6. Last payment fulfillment date,
7. Bank account in the name of company,
8. Bank deposit slips.

1.3. e Output Requirement:

Buyers

- Buyer by personal data
- Buyer by city
- Buyer by his organization/company
- Buyer by Title

Vehicles

- Vehicles by Engine (Specific packages and their quantity)
- Vehicles by Exterior(Specific packages and their quantity)
- Vehicles by Interior (Specific packages and their quantity)
- Vehicles by Performance (Specific packages and their quantity)
- Vehicles by Safety Features (Specific packages and their quantity)

Orders

- Orders by Buyer
- Orders by Vehicle
- Orders by Order Number
- Orders by Order Date
- Orders by Bank Name

1.4 EVALUATION OF EXISTING SYSTEM

1.4 a. Advantages of Existing System

1. Organized working of system is possible.
2. Required documents are easily accessible.
3. Proper management is possible.
4. Payment logs are proof of payments made.
5. More employment opportunities.
6. No major fault present.

1.4 b. Disadvantages of Existing solution

1. Decreased profit.
2. Less reliable.
3. Records and data are often lost.
4. They require more paper work.
5. Amount spent on stationery is increasing.
6. Slow order processing fulfillment.
7. Less time saving.
8. It is Difficult to edit old records.
9. It is Difficult to keep all records updated.
10. Searching a record is a tedious task.
11. Many calculations are confusing.

1.4 b. Suggested Improvements:

1. Increase employment.
2. Hiring a third party to organize the management.
3. Using Microsoft Excel for data manipulation.
4. Using Microsoft Access for keeping records.
5. Purchasing ready-made software
6. Designing a computerized system by hiring services for System Analysis.

1.5 DESCRIPTION OF OTHER POSSIBLE SOLUTIONS:

- The first possibility was to increase employment. By increasing employment, company would have more people to solve big problems faced by company, e.g. the debit-credit calculations by company sometimes become too complex for a single person to solve, and so two to three people can easily manage the problem. At times the work load increases too much that it takes more than a week to complete it, so increasing the number of employees could reduce that time.
- Secondly, the company can hire a third party to organize the management system. The company would take support from other companies to solve its management crises, e.g. it could rent out its management side to some multi-national company and could receive monthly shares in the profit.
- Next option given was to use basic computer programs like Microsoft Excel or Microsoft Access for data manipulation or data base maintenance.
- Another option was of purchasing ready-made software from market. It was a good option for the company as it was less expensive and could be immediately implemented.
- Finally the suggestion of designing a computerized system by hiring services for System Analysis was given. This system might be a little expensive for the company but will be most profitable in the long run, e.g. preparing organized managing software that could meet company's requirement without implementing any of the above listed proposals.

DESIGN

TOYOTA SOUTHERN MOTORS | PAKISTAN

2.1 ACTION PLAN

Description of Gantt chart

A total of 32 weeks were taken to complete the project which started from end of May 2013 and continued till the end of February 2014. The description of the problem was completed in the first 2 weeks, alongside which the plan of action was also designed. Then objectives were setup and the description of the existing solution was started which took a week to be completed. Then evaluation of the existing solution was started which completed in the following week, followed by the evaluation of other possible solutions which took 10 days to complete. The whole of this stage took almost a month to be completed.

The Design stage was started at the end of June 2013 and this took another 4 weeks to finalize. First of all, an overall plan was formulated which took 10 days, followed by the System flowchart which took a total of 18 days. Hardware and Software requirements took almost 2 months to be completed as appropriate Hardware and Software combinations had to be found. After this was finalized, the working of the new system was then described.

The Implementation stage started from mid-August and continued till October. The method of solution related to the problem took almost 2 weeks and then the accurate method of solution was shaped which took almost 2 weeks. The Programming of the software took around 20 days in total.

Then the Testing stage was started on the 9th of October and continued till the end of the year. The test strategy took almost 2 weeks to prepare, followed by tests using normal, abnormal and extreme data. These results took almost a month.

The technical documentation was then formulated at the start of the year. This was completed in 16 days followed by User documentation taking more than a week. The program tree and file structure were finished in February.

The evaluation stage began in February and took 1 month to complete. The System development was completed in a week.

2.2 System Flowchart

2.3 DATA STRUCTURE

BUYER: TABLE

FIELD NAME	FIELD TYPE	FIELD SIZE	SAMPLE DATA
BID	Auto Number	3	1
TITLE	Text	20	Danish Motors
CITY	Text	13	Karachi
ADDRESS	Alphanumeric	25	B-26, Sultan Mirza Road
CONTACT PERSON	Text	27	Jokhio
CONTACT NUMBER	Alphanumeric	15	021-34569874
EMAIL	Alphanumeric	25	jokhio@cardeals.com
BANK NAME	Text	23	Standard Chartered
BANK BRANCH	Text	25	Hill Park Branch
ACCOUNT NUMBER	Number	13	1264-456-9023

Primary Key: BID

DIMENSIONS: TABLE

FIELD NAME	FIELD TYPE	FIELD SIZE	SAMPLE DATA
DID	Auto Number	3	1
EXTERIOR	Alphanumeric	18	4540 X 1230 X 2390
INTERIOR	Alphanumeric	18	1809 X 1203 X 1230
WHEEL BASE	Number	4	2600
MIN TURN RADIUS	Alphanumeric	4	5.3 M
VID	Number	3	1

Primary Key:DID

ENGINE: TABLE

FIELD NAME	FIELD TYPE	FIELD SIZE	SAMPLE DATA
EID	Auto Number	3	1
TYPE	Alphanumeric	17	2 NZ – FE
CC	Number	4	1299
FUEL SYSTEM	Alphanumeric	30	EFI VTI
MAX OUTPUT	Number	14	63 (84) / 6000
MAX TORQUE	Number	15	121 / 4400
CYCLES	Alphanumeric	18	4 cyl. in-line type
VALVE MECHANISM	Alphanumeric	28	16-Valve Dohc With Dual Vvti
VID	Number	3	1
PRICE	Number	7	120000
QUANTITY	Number	3	2
RE-ORDER LEVEL	Number	2	6

Primary Key: EID

EXTERIOR: TABLE

FIELD NAME	FIELD TYPE	FIELD SIZE	SAMPLE DATA
extID	Auto Number	3	1
DID	Number	3	1
ANTENNA	Text	32	Glass Embedded Print Antenna
DOOR MIRRORS	Alphanumeric	26	Body Colored (Electric)
FRONT HEAD LAMP	Alphanumeric	32	Halogen 4 Bulbs
MUD GUARD	Alphanumeric	20	Black (RR)
REAR GARNISH	Text	22	Body Colored
REAR LAMPS	Text	25	Led Type Stop Lamp
IMPACT BARS	Logical (YES/NO)	3	No
SIDE SKIRTS	Text	16	Extended Type
WIND SHIELD	Text	25	Green Laminated
WIND SWIPERS	Text	15	Intermittent
WHEELS CAPS	Alphanumeric	32	Full cap 15"
FRONT GRILL	Text	15	Chrome
DOOR HANDLES	Text	13	Chrome
QUANTITY	Number	3	10
PRICE	Number	7	120000
RE-ORDER LEVEL	Number	2	6

Primary Key: extID

INTERIOR: TABLE

FIELD NAME	FIELD TYPE	FIELD SIZE	SAMPLE DATA
IID	Auto Number	3	1
SID	Number	3	1
CD PLAYER/ FM/ MP3	Text	32	1 Din Audio
CENTRAL CONSOLE BOX	Text	24	With Lid
CENTRAL DOOR LOCKING	Logical (YES/NO)	3	No, Yes
DIGITAL CLOCK	Logical (YES/NO)	3	No, Yes
FRONT REST	Logical (YES/NO)	3	No, Yes
GEAR LEVER	Text	33	Standard
INTERIOR COLOR SCHEME	Text	25	Grege
POWER WINDOWS	Logical (YES/NO)	3	No, Yes
SEATS	Text	33	Fabric with Leather
SPEAKERS	Alphanumeric	1	4
SPEEDOMETER	Text	37	Standard
SUNVISOR	Alphanumeric	32	Driver + Passenger(with vanity mirror)
TACHOMETER	Logical (YES/NO)	3	No, Yes
WOODGRAIN	Logical (YES/NO)	3	No, Yes
QUANTITY	Number	3	10
PRICE	Number	7	120000
RE-ORDER LEVEL	Number	2	6

Primary Key: IID

ORDER: TABLE

FIELD NAME	FIELD TYPE	FIELD SIZE	SAMPLE DATA
OID	Auto Number	3	1
ORDER NUMBER	Alphanumeric	11	2013090001
ORDER DATE	Date/Time	11	9/9/2013
BID	Number	3	4
VID	Number	3	18
extID	Number	3	8
IID	Number	3	7
PID	Number	3	1
EID	Number	3	8
SID	Number	3	1
QUANTITY	Number	3	6
PRICE	Number	7	120000
ORDER COST	Number	8	Rs. 63,34,326

Primary Key: OID

PERFORMANCE: TABLE

FIELD NAME	FIELD TYPE	FIELD SIZE	SAMPLE DATA
PID	Auto Number	3	1
TRANSMISSION	Alphanumeric	18	6 Speed Automatic
FRONT BRAKES	Text	15	Ventilated Disc
REAR BRAKES	Text	15	Solid Disc
FRONT SUSPENSION	Text	16	Macpherson Strut
STABILIZER BAR	Text	10	Front/Rear
STEERING SYSTEM	Text	15	Electric Power
WHEEL	Alphanumeric	17	195/ 65 R 19 Steel
TANK CAPACITY	Number	2	76
REAR SUSPENSION	Text	16	Torsion Beam
EID	Number	3	1
QUANTITY	Number	3	6
PRICE	Number	7	120000
RE-ORDER LEVEL	Number	2	6

Primary Key: PID

SAFETY: TABLE

FIELD NAME	FIELD TYPE	FIELD SIZE	SAMPLE DATA
SID	Auto Number	3	1
KEY WARNING	Logical (YES/NO)	3	No, Yes
LIGHTS WARNING	Logical (YES/NO)	3	No, Yes
REVERSE GEAR WARNING	Logical (YES/NO)	3	No, Yes
REAR SEAT BELT	Alphanumeric	31	3 POINT ELR & CENTER PASSENGER: NON- RETRACTABLE
FRONT SEAT BELT	Text	25	DRIVER + PASSENGER: 3 POINT ELR
VID	Number	3	2
AIRBAGS	Text	17	DRIVER + PASSENGER
COLLISION SAFETY	Logical (YES/NO)	3	No, Yes
STABILITY CONTROL	Logical (YES/NO)	3	No, Yes
QUANTITY	Number	3	6
PRICE	Number	7	120000
RE-ORDER LEVEL	Number	2	6

Primary Key: SID

VEHICLE: TABLE

FIELD NAME	FIELD TYPE	FIELD SIZE	SAMPLE DATA
VID	Auto Number	3	1
VEHICLE NAME	Text	20	Land Cruiser Prado
CLASS	Text	13	SUVS And 4WD
MAKE	Text	6	Toyota
VARIANT	Text	20	200 VX A/T SW- PETROL
YEAR	Year	4	2012
TRANSMISSION TYPE	Text	15	Manual
DOORS	Alphanumeric	1	4
WHEELS	Number	1	4
MODEL CODE	Alphanumeric	14	TH4X2M-2010
extID	Number	3	1
SID	Number	3	1

Primary Key: VID

Screen Designs

Main Menu



Vehicle Info

Vehicle Name:-

Model Code:-

Vehicle Class:-

Variant:-

Year:-

Transmission Type:-

Number of doors:-

Number of wheels:-

Safety Features

Key warning system:-

Lights warning system:-

Reverse gear warning:-

Front seat belt:-

Rear seat belt:-

Performance

Transmission:-

Front brakes:-

Rear brakes:-

Front suspension:-

Rear suspension:-

Stabilizer bar:-

Steering system:-

Wheel:-

Tank capacity:-

Interior

Air conditioning:-

Heater:-

CD Player/AM-FM/MP3:-

Console box:-

Door lock:-

Digital Clock:-

Foot rest:-

Gear lever:-

Interior Color Scheme:-

Power windows:-

Seats:-

Speaker System:-

Speedometer:-

Tachometer:-

Wood grain finish:-

Sunvisor:-

Exterior

Antenna:-

Door mirrors:-

Front head lamp:-

Mud guard:-

Rear Garnish:-

Rear Lamps:-

Impact Bars:-

Side skirts:-

Windshield:-

Wheel caps:-

Engine Specifications

Engine type:-

CC:-

Fuel system:-

Max output:-

Max torque:-

DimensionsExterior:-Interior:-Wheel base:-Minimum Turn Radius:-

2.4 Pseudo Codes

MDITTOYOTA

START

Initialize Variables and Environment

Do While True

Begin

Display Menu Options

1: Orders Information

2: Buyers Information

3: Vehicle Information

4: Engine Information

5: Performance Information

6: Exterior Information

7: Safety Information

8: Interior Information

9: Dimensions Information

10: Query/Search: Buyer, Vehicle, Order

11: Reports

12: Backup Database

End(Display of menu options)

Input Choice

Do Case

Case Choice=1

Do Orders Information

Case Choice=2

Do Buyers Information

Case Choice=3

Do Vehicle Information

Case Choice=4

Do Engine Information

Case Choice=5

Do Performance Information

Case Choice=6

Do Exterior Information

Case Choice=7

Do Safety Information

Case Choice=8

Do Interior Information

Case Choice=9

Do Dimensions Information

Case Choice=10

Do Query/Search: Buyer, Vehicle, Order

Case Choice=11

Do Reports

Case Choice=12

Do Backup Database

Close all files

Clear all variables

END

Endcase

Enddo

ORDERS INFORMATION:

ADD Routine**START**

Initialize Variables and Environment

Use Order

Input Vehicle ID

Search for Order ID in Vehicle table

IF Found

Output Order ID already exists'

ELSE

Input 'Order Date, Quantity, Vehicle, Features: Engine, Exterior, Interior, Performance, Safety, Buyer ID, Buyer Name'

Validate the data

If any error then output "Error Message"

ENDIF

Write 'Order Date, Quantity, Vehicle, Features: Engine, Exterior, Interior, Performance, Safety, Buyer ID, Buyer Name'

ToOrders table

ENDIF

SAVE Routine**START**

Initialize variable and environment

Use Order

Do While True

Input 'Order Date, Quantity, Vehicle, Features: Engine, Exterior, Interior, Performance, Safety, Buyer ID, Buyer Name'

 Search for OID in Order table

IF Not Found

 Validate all fields

IF Fields not validated

 Input data in all fields again

ELSE

 Save Record.

END IF

ELSE

 Display message 'Record already exists.'

 Input data in all fields again

ENDIF

ENDDO

EDIT Routine**START**

Initialize variable and environment

Use Order

Do While True**Input**Order ID

Search for Order ID in Vehicle table

IF Not Found

Output 'Order ID does not exist'

ELSE

Display 'Order Date, Quantity, Vehicle, Features: Engine, Exterior, Interior, Performance, Safety, Buyer ID, Buyer Name'

Update fields

ENDIF**ENDDO**

SEARCHROUTINE**START**

Initialize variable and environment

Use Order

Do While True

Input Order ID

 Search for Order ID in Order table

IF Not Found

 Output 'Order ID does not exist.'

ELSE

 Display 'Order Date, Quantity, Vehicle, Features: Engine, Exterior, Interior, Performance, Safety, Buyer ID, Buyer Name'

 Close all files

 Clear all variables

 Return to main menu

ENDIF

ENDDO

DELETE Routine**START**

Initialize variable and environment

Use Order

Do While True**Input** Order ID

Search for Order ID in Order table

IF Not Found

Output 'Order ID does not exist.'

ELSE

Display 'Order Date, Quantity, Vehicle, Features: Engine, Exterior, Interior, Performance, Safety, Buyer ID, Buyer Name'

INPUT 'Do you want to Delete this record'**IF** answer is 'Yes'

Delete record

Write Changes to Order table

ENDIF**ENDIF**

Close all files

Clear all variables

Return to main menu

ENDIF**ENDDO****BUYER INFORMATION:**

ADD Routine**START**

Initialize Variables and Environment

Use Buyer

Input 'Title, City, Address, Contact Person, Number, Email, Bank Name, Bank Branch, Account Number'

Search for BID in Buyer table

IF Found

Output 'BID already exists.'

ELSE

Input 'Title, City, Address, Contact Person, Number, Email, Bank Name, Bank Branch, Account Number'

Validate the data

If any error then output "Error Message"

ENDIF

Write 'Title, City, Address, Contact Person, Number, Email, Bank Name, Bank Branch, Account Number'
to Order table

SAVE Routine**START**

Initialize variable and environment

Use Buyer

Do While True

Input 'Title, City, Address, Contact Person, Number, Email, Bank Name, Bank Branch, Account Number'

 Search for BID in Buyer table

IF Not Found

 Validate all fields

IF Fields not validated

 Input data in all fields again

ELSE

 Save Record.

END IF

ELSE

 Display message 'Record already exists.'

 Input data in all fields again

ENDIF

ENDDO

EDIT Routine**START**

Initialize variable and environment

Use Buyer

Do While True

Input Title, City, Address, Contact Person, Number, Email, Bank Name, Bank Branch, Account Number

 Search for BID in Buyer table

IF Not Found

 Output 'BID does not exist'

ELSE

 Display 'Title, City, Address, Contact Person, Number, Email, Bank Name, Bank Branch, Account Number'

 Update fields

 Write Changes to Buyer table

ENDIF

ENDDO

Search Routine**START**

Initialize variable and environment

Use Buyer

Do While True

Input BID

 Search for BID in Buyer table

IF Not Found

 Output 'Buyer ID does not exist.'

ELSE

 Display 'Title, City, Address, Contact Person, Number, Email, Bank Name, Bank Branch,
Account Number'

 Close all files

 Clear all variables

 Return to main menu

ENDDO

DELETE Routine

RT

Initialize variable and environment

Use Buyer

Do While True

Input Order ID

Search for Order ID in Order table

IF Not Found

Output 'Order ID does not exist.'

ELSE

Display 'Title, City, Address, Contact Person, Number, Email, Bank Name, Bank Branch, Account Number'

INPUT 'Do you want to Delete this record' **IF** answer is 'Yes'

Delete record

Write Changes to Buyer table

ENDIF **ENDIF**

Close all files

Clear all variables

Return to main menu

ENDIF

ENDDO

VEHICLE INFORMATION:

ADD Routine**START**

Initialize Variables and Environment

Use Vehicle

Input 'Model Code, Name, Make, Year, Variant, Transmission Type, Class and Doors'

Search for VID in Vehicle table

IF Found

Output 'VID already exists.'

ELSE

Input 'Model Code, Name, Make, Year, Variant, Transmission Type, Class and Doors'

Validate the data

If any error then output "Error Message"

ENDIF

Write 'Model Code, Name, Make, Year, Variant, Transmission Type, Class and Doors'

To Vehicle table

SAVE Routine**START**

Initialize variable and environment

Use Vehicle

Do While True

Input 'Model Code, Name, Make, Year, Variant, Transmission Type, Class and Doors'

 Search for VID in Vehicle table

IF Not Found

 Validate all fields

IF Fields not validated

 Input data in all fields again

ELSE

 Save Record.

END IF

ELSE

 Display message 'Record already exists.'

 Input data in all fields again

ENDIF

ENDDO

EDIT Routine**START**

Initialize variable and environment

Use Vehicle

Do While True

Input 'Model Code, Name, Make, Year, Variant, Transmission Type, Class and Doors'

 Search for VID in Vehicle table

IF Not Found

 Output 'VID does not exist'

ELSE

 Display 'Model Code, Name, Make, Year, Variant, Transmission Type, Class and Doors'

 Update fields

 Write Changes to Vehicle table

ENDIF

ENDDO

Search Routine**START**

Initialize variable and environment

Use Vehicle

Do While True

Input BID

 Search for VID in Vehicle table

IF Not Found

 Output 'Vehicle ID does not exist.'

ELSE

 Display 'Model Code, Name, Make, Year, Variant, Transmission Type, Class and Doors'

 Close all files

 Clear all variables

 Return to main menu

ENDDO

DELETE Routine**START**

Initialize variable and environment

Use Vehicle

Do While True

Input Vehicle ID

 Search for Vehicle ID in Vehicle table

IF Not Found

 Output 'Vehicle ID does not exist.'

ELSE

 Display 'Model Code, Name, Make, Year, Variant, Transmission Type, Class and Doors'

INPUT 'Do you want to Delete this record'

IF answer is 'Yes'

 Delete record

 Write Changes to Vehicle table

ENDIF

ENDIF

 Close all files

 Clear all variables

 Return to main menu

ENDIF

ENDDO

ENGINE INFORMATION:

ADD Routine**START**

Initialize Variables and Environment

Use Engine

Input 'VID, Type, CC, Fuel System, Output, Max Torque, Cycles, Valve Mechanism, Quantity in Stock and Price'

Search for EID in Engine table

IF Found

Output 'EID already exists.'

ELSE

Input 'VID, Type, CC, Fuel System, Output, Max Torque, Cycles, Valve Mechanism, Quantity in Stock and Price'

Validate the data

If any error then output "Error Message"

ENDIF

Write 'VID, Type, CC, Fuel System, Output, Max Torque, Cycles, Valve Mechanism, Quantity in Stock and Price'

To Engine table

SAVE Routine**START**

Initialize variable and environment

Use Engine

Do While True

Input ' VID, Type, CC, Fuel System, Output, Max Torque, Cycles, Valve Mechanism, Quantity in Stock and Price'

Search for EID in Engine table

IF Not Found

Validate all fields

IF Fields not validated

Input data in all fields again

ELSE

Save Record.

END IF

ELSE

Display message 'Record already exists.'

Input data in all fields again

ENDIF

ENDDO

EDIT Routine**START**

Initialize variable and environment

Use Engine

Do While True

Input 'VID, Type, CC, Fuel System, Output, Max Torque, Cycles, Valve Mechanism, Quantity in Stock and Price'

Search for EID in Engine table

IF Not Found

Output 'EID does not exist'

ELSE

Display 'VID, Type, CC, Fuel System, Output, Max Torque, Cycles, Valve Mechanism, Quantity in Stock and Price'

Update fields

Write Changes to Engine table

ENDIF**ENDDO**

Search Routine**START**

Initialize variable and environment

Use Engine

Do While True

Input BID

 Search for EID in Engine table

IF Not Found

 Output 'Engine ID does not exist.'

ELSE

 Display 'VID, Type, CC, Fuel System, Output, Max Torque, Cycles, Valve Mechanism,
Quantity in Stock and Price'

 Close all files

 Clear all variables

 Return to main menu

ENDDO

DELETE Routine**START**

Initialize variable and environment

Use Engine

Do While True

Input Engine ID

 Search for Engine ID in Engine table

IF Not Found

 Output 'Engine ID does not exist.'

ELSE

 Display 'VID, Type, CC, Fuel System, Output, Max Torque, Cycles, Valve Mechanism, Quantity in Stock and Price'

INPUT 'Do you want to Delete this record'

IF answer is 'Yes'

 Delete record

 Write Changes to Engine table

ENDIF

ENDIF

 Close all files

 Clear all variables

 Return to main menu

ENDIF

ENDDO

PERFORMANCE INFORMATION:

ADD Routine**START**

Initialize Variables and Environment

Use Performance

Input 'VID, Type, CC, Fuel System, Output, Max Torque, Cycles, Valve Mechanism, Quantity in Stock and Price'

Search for PID in Performance table

IF Found

Output 'PID already exists.'

ELSE

Input 'EID, Quantity in Stock, Price, Transmission, Front Brakes, Rear Brakes, Front Suspension, Stabilizer Bar, Steering System, Wheel, Tank Capacity and Rear Suspension'

Validate the data

If any error then output "Error Message"

ENDIFWrite 'EID, Quantity in Stock, Price, Transmission, Front Brakes, Rear Brakes, Front Suspension, Stabilizer Bar, Steering System, Wheel, Tank Capacity and Rear Suspension'
to Performance table

SAVE Routine**START**

Initialize variable and environment

Use Performance

Do While True

Input 'EID, Quantity in Stock, Price, Transmission, Front Brakes, Rear Brakes, Front Suspension, Stabilizer Bar, Steering System, Wheel, Tank Capacity and Rear Suspension'

Search for PID in Performance table

IF Not Found

Validate all fields

IF Fields not validated

Input data in all fields again

ELSE

Save Record.

END IF**ELSE**

Display message 'Record already exists.'

Input data in all fields again

ENDIF**ENDDO**

EDIT Routine**START**

Initialize variable and environment

Use Performance

Do While True

Input 'EID, Quantity in Stock, Price, Transmission, Front Brakes, Rear Brakes, Front Suspension, Stabilizer Bar, Steering System, Wheel, Tank Capacity and Rear Suspension'

 Search for PID in Performance table

IF Not Found

 Output 'PIDdoes not exist'

ELSE

 Display 'EID, Quantity in Stock, Price, Transmission, Front Brakes, Rear Brakes, Front Suspension, Stabilizer Bar, Steering System, Wheel, Tank Capacity and Rear Suspension'

 Update fields

 Write Changes to Performance table

ENDIF

ENDDO

Search Routine**START**

Initialize variable and environment

Use Performance

Do While True

Input PID

 Search for PID in Performance table

IF Not Found

 Output 'Performance ID does not exist.'

ELSE

 Display 'EID, Quantity in Stock, Price, Transmission, Front Brakes, Rear Brakes, Front Suspension, Stabilizer Bar, Steering System, Wheel, Tank Capacity and Rear Suspension'

 Close all files

 Clear all variables

 Return to main menu

ENDDO

DELETE Routine**START**

Initialize variable and environment

Use Performance

Do While True

Input Performance ID

 Search for Performance ID in Performance table

IF Not Found

 Output 'Performance ID does not exist.'

ELSE

 Display 'EID, Quantity in Stock, Price, Transmission, Front Brakes, Rear Brakes, Front Suspension, Stabilizer Bar, Steering System, Wheel, Tank Capacity and Rear Suspension'

INPUT 'Do you want to delete this record'

IF answer is 'Yes'

 Delete record

 Write Changes to Performance table

ENDIF

ENDIF

 Close all files

 Clear all variables

 Return to main menu

ENDIF

ENDDO

EXTERIOR INFORMATION:

ADD Routine**START**

Initialize Variables and Environment

Use Exterior

Input 'VID, Type, CC, Fuel System, Output, Max Torque, Cycles, Valve Mechanism, Quantity in Stock and Price'

Search for extID in Exterior table

IF Found

Output 'extID already exists.'

ELSE

Input 'DID, Door Mirrors, Front Head Lamp, Rear Lamps, Windshield, Quantity in Stock, Price, Antenna, Mud Guard, Rear Garnish, Impact Bars, Side Skirts, Wind Swipers, Wheel Caps and Doors Handles'

Validate the data

If any error then output "Error Message"

ENDIF

Write 'DID, Door Mirrors, Front Head Lamp, Rear Lamps, Windshield, Quantity in Stock, Price, Antenna, Mud Guard, Rear Garnish, Impact Bars, Side Skirts, Wind Swipers, Wheel Caps and Doors Handles'

To Exterior table

SAVE Routine**START**

Initialize variable and environment

Use Exterior

Do While True

Input 'DID, Door Mirrors, Front Head Lamp, Rear Lamps, Windshield, Quantity in Stock, Price, Antenna, Mud Guard, Rear Garnish, Impact Bars, Side Skirts, Wind Swipers, Wheel Caps and Doors Handles'

Search for extID in Exterior table

IF Not Found

Validate all fields

IF Fields not validated

Input data in all fields again

ELSE

Save Record.

END IF

ELSE

Display message 'Record already exists.'

Input data in all fields again

ENDIF

ENDDO

EDIT Routine**START**

Initialize variable and environment

Use Exterior

Do While True

Input 'DID, Door Mirrors, Front Head Lamp, Rear Lamps, Windshield, Quantity in Stock, Price, Antenna, Mud Guard, Rear Garnish, Impact Bars, Side Skirts, Wind Swipers, Wheel Caps and Doors Handles'

 Search for extID in Exterior table

IF Not Found

 Output 'extID does not exist'

ELSE

 Display 'DID, Door Mirrors, Front Head Lamp, Rear Lamps, Windshield, Quantity in Stock, Price, Antenna, Mud Guard, Rear Garnish, Impact Bars, Side Skirts, Wind Swipers, Wheel Caps and Doors Handles'

 Update fields

 Write Changes to Exterior table

ENDIF

ENDDO

Search Routine**START**

Initialize variable and environment

Use Exterior

Do While True

Input extID

 Search for extID in Exterior table

IF Not Found

 Output 'Exterior ID does not exist.'

ELSE

 Display 'DID, Door Mirrors, Front Head Lamp, Rear Lamps, Windshield, Quantity in Stock, Price, Antenna, Mud Guard, Rear Garnish, Impact Bars, Side Skirts, Wind Swipers, Wheel Caps and Doors Handles'

 Close all files

 Clear all variables

 Return to main menu

ENDDO

DELETE Routine**START**

Initialize variable and environment

Use Exterior

Do While True

Input Exterior ID

 Search for Exterior ID in Exterior table

IF Not Found

 Output 'Exterior ID does not exist.'

ELSE

 Display 'DID, Door Mirrors, Front Head Lamp, Rear Lamps, Windshield, Quantity in Stock, Price, Antenna, Mud Guard, Rear Garnish, Impact Bars, Side Skirts, Wind Swipers, Wheel Caps and Doors Handles'

INPUT 'Do you want to Delete this record'

IF answer is 'Yes'

 Delete record

 Write Changes to Exterior table

ENDIF

ENDIF

 Close all files

 Clear all variables

 Return to main menu

ENDIF

ENDDO

INTERIOR INFORMATION:

ADD Routine**START**

Initialize Variables and Environment

Use Interior

Input 'SID, CDplayer, Digital Clock, Power Windows, Speakers, Woodgrain, Tachometer, Quantity in Stock, Price, AC, Heater, Central Console Box, Central Door Locking, Gear Lever, Interior Colour Scheme, Seats, Speedometer, Sunvisor'

Search for IID in Interior table

IF Found

Output 'IID already exists.'

ELSE

Input 'SID, CDplayer, Digital Clock, Power Windows, Speakers, Woodgrain, Tachometer, Quantity in Stock, Price, AC, Heater, Central Console Box, Central Door Locking, Gear Lever, Interior Colour Scheme, Seats, Speedometer, Sunvisor'

Validate the data

If any error then output "Error Message"

ENDIF

Write 'SID, CDplayer, Digital Clock, Power Windows, Speakers, Woodgrain, Tachometer, Quantity in Stock, Price, AC, Heater, Central Console Box, Central Door Locking, Gear Lever, Interior Colour Scheme, Seats, Speedometer, Sunvisor'

ToInterior table

SAVE Routine**START**

Initialize variable and environment

Use Interior

Do While True

Input 'SID, CDplayer, Digital Clock, Power Windows, Speakers, Woodgrain, Tachometer, Quantity in Stock, Price, AC, Heater, Central Console Box, Central Door Locking, Gear Lever, Interior Colour Scheme, Seats, Speedometer, Sunvisor'

 Search for IID in Interior table

IF Not Found

 Validate all fields

IF Fields not validated

 Input data in all fields again

ELSE

 Save Record.

END IF

ELSE

 Display message 'Record already exists.'

 Input data in all fields again

ENDIF

ENDDO

EDIT Routine**START**

Initialize variable and environment

Use Interior

Do While True

Input 'SID, CDplayer, Digital Clock, Power Windows, Speakers, Woodgrain, Tachometer, Quantity in Stock, Price, AC, Heater, Central Console Box, Central Door Locking, Gear Lever, Interior Colour Scheme, Seats, Speedometer, Sunvisor'

 Search for IID in Interior table

IF Not Found

 Output 'IID does not exist'

ELSE

 Display 'SID, CDplayer, Digital Clock, Power Windows, Speakers, Woodgrain, Tachometer, Quantity in Stock, Price, AC, Heater, Central Console Box, Central Door Locking, Gear Lever, Interior Colour Scheme, Seats, Speedometer, Sunvisor'

 Update fields

 Write Changes to Interior table

ENDIF

ENDDO

Search Routine**START**

Initialize variable and environment

Use Interior

Do While True

Input IID

 Search for IID in Interior table

IF Not Found

 Output 'Interior ID does not exist.'

ELSE

 Display 'SID, CDplayer, Digital Clock, Power Windows, Speakers, Woodgrain, Tachometer, Quantity in Stock, Price, AC, Heater, Central Console Box, Central Door Locking, Gear Lever, Interior Colour Scheme, Seats, Speedometer, Sunvisor'

 Close all files

 Clear all variables

 Return to main menu

ENDDO

DELETE Routine**START**

Initialize variable and environment

Use Interior

Do While True

Input Interior ID

 Search for Interior ID in Exterior table

IF Not Found

 Output 'Interior ID does not exist.'

ELSE

 Display 'SID, CDplayer, Digital Clock, Power Windows, Speakers, Woodgrain, Tachometer, Quantity in Stock, Price, AC, Heater, Central Console Box, Central Door Locking, Gear Lever, Interior Colour Scheme, Seats, Speedometer, Sunvisor'

INPUT 'Do you want to Delete this record'

IF answer is 'Yes'

 Delete record

 Write Changes to Interior table

ENDIF

ENDIF

 Close all files

 Clear all variables

 Return to main menu

ENDIF

ENDDO

SAFETY INFORMATION:

ADD Routine**START**

Initialize Variables and Environment

Use Safety

Input 'VID, Quantity in Stock, Price, Key Warning, Light Warning, Reverse Gear Warning, Front Seat Belt, Rear Seat Belt, Collision Safety, Airbags and Stability Control'

Search for SID in Safety table

IF Found

Output 'SID already exists.'

ELSE

Input 'VID, Quantity in Stock, Price, Key Warning, Light Warning, Reverse Gear Warning, Front Seat Belt, Rear Seat Belt, Collision Safety, Airbags and Stability Control'

Validate the data

If any error then output "Error Message"

ENDIF

Write 'VID, Quantity in Stock, Price, Key Warning, Light Warning, Reverse Gear Warning, Front Seat Belt, Rear Seat Belt, Collision Safety, Airbags and Stability Control'

To Safety table

SAVE Routine**START**

Initialize variable and environment

Use Safety

Do While True

Input 'VID, Quantity in Stock, Price, Key Warning, Light Warning, Reverse Gear Warning, Front Seat Belt, Rear Seat Belt, Collision Safety, Airbags and Stability Control'

 Search for SID in Safety table

IF Not Found

 Validate all fields

IF Fields not validated

 Input data in all fields again

ELSE

 Save Record.

END IF

ELSE

 Display message 'Record already exists.'

 Input data in all fields again

ENDIF

ENDDO

EDIT Routine**START**

Initialize variable and environment

Use Safety

Do While True

Input 'VID, Quantity in Stock, Price, Key Warning, Light Warning, Reverse Gear Warning, Front Seat Belt, Rear Seat Belt, Collision Safety, Airbags and Stability Control'

 Search for SID in Safety table

IF Not Found

 Output 'SID does not exist'

ELSE

 Display 'VID, Quantity in Stock, Price, Key Warning, Light Warning, Reverse Gear Warning, Front Seat Belt, Rear Seat Belt, Collision Safety, Airbags and Stability Control'

 Update fields

 Write Changes to Safety table

ENDIF

ENDDO

Search Routine**START**

Initialize variable and environment

Use Safety

Do While True

Input SID

 Search for SID in Safety table

IF Not Found

 Output 'Safety ID does not exist.'

ELSE

 Display 'VID, Quantity in Stock, Price, Key Warning, Light Warning, Reverse Gear Warning, Front Seat Belt, Rear Seat Belt, Collision Safety, Airbags and Stability Control'

 Close all files

 Clear all variables

 Return to main menu

ENDDO

DELETE Routine**START**

Initialize variable and environment

Use Safety

Do While True

Input Safety ID

 Search for Safety ID in Safety table

IF Not Found

 Output 'Safety ID does not exist.'

ELSE

 Display 'VID, Quantity in Stock, Price, Key Warning, Light Warning, Reverse Gear Warning, Front Seat Belt, Rear Seat Belt, Collision Safety, Airbags and Stability Control'

INPUT 'Do you want to Delete this record'

IF answer is 'Yes'

 Delete record

 Write Changes to Safety table

ENDIF

ENDIF

 Close all files

 Clear all variables

 Return to main menu

ENDIF

ENDDO

DIMENSIONS INFORMATION:

ADD Routine**START**

Initialize Variables and Environment

Use Dimensions

Input 'VID, Exterior, Interior, Wheel Base and Minimum Turning Radius'

Search for DID in Dimensions table

IF Found

Output 'DID already exists.'

ELSE

Input 'VID, Exterior, Interior, Wheel Base and Minimum Turning Radius'

Validate the data

If any error then output "Error Message"

ENDIF

Write 'VID, Exterior, Interior, Wheel Base and Minimum Turning Radius'

To Dimensions table

SAVE Routine**START**

Initialize variable and environment

Use Dimensions

Do While True

Input ' VID, Exterior, Interior, Wheel Base and Minimum Turning Radius'

 Search for DID in Dimensions table

IF Not Found

 Validate all fields

IF Fields not validated

 Input data in all fields again

ELSE

 Save Record.

END IF

ELSE

 Display message 'Record already exist'

 Input data in all fields again

ENDIF

ENDDO

EDIT Routine**START**

Initialize variable and environment

Use Dimensions

Do While True

Input 'VID, Exterior, Interior, Wheel Base and Minimum Turning Radius'

 Search for DID in Dimensions table

IF Not Found

 Output 'DID does not exist'

ELSE

 Display 'VID, Exterior, Interior, Wheel Base and Minimum Turning Radius'

 Update fields

 Write Changes to Dimensions table

ENDIF

ENDDO

Search Routine**START**

Initialize variable and environment

Use Dimensions

Do While True

Input DID

 Search for DID in Dimensions table

IF Not Found

 Output 'Dimensions ID does not exist.'

ELSE

 Display 'VID, Exterior, Interior, Wheel Base and Minimum Turning Radius'

 Close all files

 Clear all variables

 Return to main menu

ENDDO

DELETE Routine**START**

Initialize variable and environment

Use Dimensions

Do While True

Input Dimensions ID

 Search for Dimensions ID in Dimensions table

IF Not Found

 Output 'Dimensions ID does not exist.'

ELSE

 Display 'VID, Exterior, Interior, Wheel Base and Minimum Turning Radius'

INPUT 'Do you want to Delete this record'

IF answer is 'Yes'

 Delete record

 Write Changes to Dimensions table

ENDIF

ENDIF

 Close all files

 Clear all variables

 Return to main menu

ENDIF

ENDDO

2.5 Hardware Requirements

Name of Hardware:	Central Processing Unit (CPU)
Specification:	Intel i3 or above
Reason 1:	For smooth running of program
Reason 2:	For multitasking
Name of Hardware:	Motherboard
Specification:	Intel/Asus WS series
Reason 1:	Essential for the basic functioning of a computer
Reason 2:	Motherboards from reliable companies ensure a smooth running of the system
Name of Hardware:	RAM
Specification:	4 GB DDR3
Reason 1:	To ensure program runs smoothly as database size grows
Reason 2:	Larger RAM size supports processor in working with different programs and instructions
Name of Hardware:	Hard Disk
Specification:	200 GB
Reason 1:	To ensure data is stored easily even when database size increases
Reason 2:	Enough space for support softwares to be installed
Name of Hardware:	Mouse
Specification:	Laser
Reason 1:	Allows accurate movement of cursor due to laser technology
Reason 2:	Important for making on screen selections
Name of Hardware:	Keyboard
Specification:	Anti-RSI
Reason 1:	To prevent injuries (Anti-RSI)
Reason 2:	To type data into software
Name of Hardware:	CD ROM
Specification:	ANY
Reason 1:	To install programs
Reason 2:	To make backups (if needed)
Name of Hardware:	Monitor
Specification:	LCD
Reason 1:	For comfort (if computer used for extended periods.)
Reason 2:	Important for interacting with computer

Name of Hardware:	Printer
Specification:	Laser
Reason 1:	To print copies of documents
Reason 2:	To print invoices.

2.6 Software Requirements

Name of Software:	Microsoft Windows
Specification:	XP/Windows Vista/ Windows 7
Reason 1:	To support different softwares
Reason 2:	Essential for carrying out basic functions

Name of Software:	Microsoft Office
Specification:	MS Access [Database Management System]
Reason 1:	Allows interaction between data and software
Reason 2:	Allows viewing of data in a Sub Datasheet

Name of Software:	Microsoft Visual Basic
Specification:	Version 6.0
Reason 1:	Can easily work with softwares such as MS Access
Reason 2:	Program can be tested in real time and debugged Accordingly

```

Attribute VB_Name = "frmBuyer"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Dim ad As Integer
Dim ed As Integer

```

```

Private Sub cmdSearchBID_Click()
    sbid = InputBox("Enter Buyer's ID to search the record", "Search")

    If sbid = 0 Then Exit Sub

    DE1.rsBuyer.MoveFirst
    DE1.rsBuyer.Find ("bid =" & sbid)

    If DE1.rsBuyer.EOF Then
        MsgBox ("Record not found!")
        DE1.rsBuyer.MoveFirst
    End If

End Sub

```

```

Private Sub cmdSearchBname_Click()
    stitle = InputBox("Enter Buyer's Title to search the record", "Search")

    If stitle = " " Then Exit Sub

    DE1.rsBuyer.MoveFirst
    DE1.rsBuyer.Find ("title =" & stitle & " ")

    If DE1.rsBuyer.EOF Then
        MsgBox ("Record not found!")
        DE1.rsBuyer.MoveFirst
    End If

End Sub

```

```

Private Sub Form_Load()
    lblDate.Caption = Format(Date, "long date")
End Sub

```

```

Private Sub TB1_ButtonClick(ByVal Button As MSComctlLib.Button)
    Select Case Button.Index
    Case 1          'First
        DE1.rsBuyer.MoveFirst
    Case 2          'Previous
        If DE1.rsBuyer.BOF Then
            MsgBox ("Already on the first record!")
            DE1.rsBuyer.MoveFirst
        Else
            DE1.rsBuyer.MovePrevious
        End If
    Case 3          'Next
        If DE1.rsBuyer.EOF Then
            MsgBox ("Already on the last record!")
            DE1.rsBuyer.MoveLast
        Else
            DE1.rsBuyer.MoveNext
        End If
    Case 4          'Last
        DE1.rsBuyer.MoveLast
    Case 6          'New
        ad = 1
1      'DE1.rsBuyer.MoveLast
    End Select
End Sub

```

```

1      'nbid = DE1.rsBuyer.Fields(0).Value + 1
      DE1.rsBuyer.AddNew
      'txtBID = nbid

      Frame1.Enabled = True
      Frame2.Enabled = True
      Frame3.Enabled = True
      Frame4.Enabled = True
      TB1.Buttons(8).Enabled = True

      For btn = 1 To 4
      TB1.Buttons(btn).Enabled = False
      Next

      Case 7          'Edit
          ed = 1
          Frame1.Enabled = True
          Frame2.Enabled = True
          Frame3.Enabled = True
          Frame4.Enabled = True

          TB1.Buttons(8).Enabled = True
          For btn = 1 To 4
          TB1.Buttons(btn).Enabled = False
          Next

      Case 8          'Save
          If ed = 0 And ad = 0 Then Exit Sub

          If txttitle = "" Then
              MsgBox ("Title field cannot be left empty!")
              txttitle.SetFocus
              Exit Sub
          ElseIf IsNumeric(txttitle) Then
              MsgBox ("Invalid data type!")
              txttitle.SetFocus
              Exit Sub
          End If

          If txtcity = "" Then
              MsgBox ("City field cannot be left empty!")
              txtcity.SetFocus
              Exit Sub
          ElseIf IsNumeric(txtcity) Then
              MsgBox ("Invalid data type!")
              txtcity.SetFocus
              Exit Sub
          End If

          If txtadd = "" Then
              MsgBox ("Address field cannot be left empty!")
              txtadd.SetFocus
              Exit Sub
          ElseIf IsNumeric(txtadd) Then
              MsgBox ("Invalid data type!")
              txtadd.SetFocus
              Exit Sub
          End If

          If txtcperson = "" Then
              MsgBox ("Contact Person field cannot be left empty!")
              txtcperson.SetFocus
              Exit Sub
          ElseIf IsNumeric(txtcperson) Then
              MsgBox ("Invalid data type!")
              txtcperson.SetFocus
              Exit Sub
          End If

          If txtcnumber = "" Then
              MsgBox ("Contact Number field cannot be left empty!")
              txtcnumber.SetFocus

```

```

1  2  Exit Sub
    ElseIf IsNumeric(txtcnumber) Then
        MsgBox ("Invalid data type, please enter in the correct format.")
        txtcnumber.SetFocus
        Exit Sub
    End If

    If txtemail = "" Then
        MsgBox ("Email field cannot be left empty!")
        txtemail.SetFocus
        Exit Sub
    End If

    If txtbnkname = "" Then
        MsgBox ("Bank Name field cannot be left empty!")
        txtbnkname.SetFocus
        Exit Sub
    ElseIf IsNumeric(txtbnkname) Then
        MsgBox ("Invalid data type!")
        txtbnkname.SetFocus
        Exit Sub
    End If

    If txtbnkbranch = "" Then
        MsgBox ("Bank Branch field cannot be left empty!")
        txtbnkbranch.SetFocus
        Exit Sub
    ElseIf IsNumeric(txtbnkbranch) Then
        MsgBox ("Invalid data type!")
        txtbnkbranch.SetFocus
        Exit Sub
    End If

    If txtaccno = "" Then
        MsgBox ("Account Number field cannot be left empty!")
        txtaccno.SetFocus
        Exit Sub
    ElseIf IsNumeric(txtaccno) Then
        MsgBox ("Invalid data type!")
        txtaccno.SetFocus
        Exit Sub
    End If

    DE1.rsBuyer.Update
    ad = 0
    ed = 0
    TB1.Buttons(8).Enabled = False

    Frame1.Enabled = False
    Frame2.Enabled = False
    Frame3.Enabled = False
    Frame4.Enabled = False
    For btn = 1 To 4
        TB1.Buttons(btn).Enabled = True
    Next

    Case 9      'Delete
        d = MsgBox("Are you sure you want to delete this record?", vbYesNo, "Delete
        confirmation")
        If d = vbNo Then Exit Sub
        DE1.rsBuyer.Delete
        DE1.rsBuyer.MoveFirst

    Case 11     'Print
        DE1.rsBuyer.Filter = ("bid = " & txtBID)
        rptBuyer.Show

End Select
End Sub

```

```
Private Sub tmrTime_Timer()
```

```
lblTime.Caption = Time
End Sub
```

```
Private Sub txtaccno_KeyPress(KeyAscii As Integer) 'Validation for numbers only.
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txtadd_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txtbnkbranch_KeyPress(KeyAscii As Integer) 'Validation for numbers only.
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txtbnkname_KeyPress(KeyAscii As Integer) 'Validation for numbers only.
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txtcity_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txtcnumber_KeyPress(KeyAscii As Integer) 'Validation for numbers only.
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txtcperson_KeyPress(KeyAscii As Integer) 'Validation for
```

alphabets only.

```
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii  
= 32 Or KeyAscii = 8 Then  
Else  
    MsgBox ("Invalid character entered, please re-enter data!")  
    'SendKeys "{Home}+{End}"  
    'SendKeys "{Delete}"  
End If  
End Sub
```

Private Sub txtemail_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.

```
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii  
= 32 Or KeyAscii = 8 Then  
Else  
    MsgBox ("Invalid character entered, please re-enter data!")  
    'SendKeys "{Home}+{End}"  
    'SendKeys "{Delete}"  
End If  
End Sub
```

Private Sub txttitle_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.

```
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii  
= 32 Or KeyAscii = 8 Then  
Else  
    MsgBox ("Invalid character entered, please re-enter data!")  
    'SendKeys "{Home}+{End}"  
    'SendKeys "{Delete}"  
End If  
End Sub
```

```
Attribute VB_Name = "frmBuyerQuery"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
```

Private Sub Combol_Click()

```
    Text1.SetFocus
```

End Sub

Private Sub Command1_Click()

```
    'For f = 0 To dgSafety.Columns.Count - 1
    'dgSafety.Columns(f).Caption = UCase(dgSafety.Columns(f).Caption)
    'Next
    DE1.rsBuyer.Filter = "title <> NULL"
```

End Sub

Private Sub Form_Load()

```
    lblDate.Caption = Format(Date, "long date")
```

End Sub

Private Sub Text1_KeyPress(KeyAscii As Integer)

```
    If KeyAscii = 13 Then
        Select Case Combol.Text
            Case "Title"
                DE1.rsBuyer.Filter = "title='" & Text1.Text & "'"
            Case "City"
                DE1.rsBuyer.Filter = "city='" & Text1.Text & "'"
            Case "Address"
                DE1.rsBuyer.Filter = "add ='" & Text1.Text & "'"
            Case "Contact Person"
                DE1.rsBuyer.Filter = "cperson ='" & Text1.Text & "'"
            Case "Bank Name"
                DE1.rsBuyer.Filter = "bnkname ='" & Text1.Text & "'"
            Case "Bank Branch"
                DE1.rsBuyer.Filter = "bnkbranch ='" & Text1.Text & "'"
        End Select
        Label1.Caption = "TOTAL RECORDS FOUND: " & DE1.rsBuyer.RecordCount
```

```
    End If
```

End Sub

Private Sub tmrTime_Timer()

```
    lblTime.Caption = Time
```

End Sub

```

Attribute VB_Name = "frmDimensions"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Dim ad As Integer
Dim ed As Integer

```

Private Sub Form_Load()

```
lblDate.Caption = Format(Date, "long date")
```

End Sub

Private Sub TB1_ButtonClick(ByVal Button As MSComctlLib.Button)

```
Select Case Button.Index
```

Case 1 'First

```
DE1.rsDimensions.MoveFirst
```

Case 2 'Previous

```
If DE1.rsDimensions.BOF Then
```

```
MsgBox ("Already on the first record!")
```

```
DE1.rsDimensions.MoveFirst
```

```
Else
```

```
DE1.rsDimensions.MovePrevious
```

```
End If
```

Case 3 'Next

```
If DE1.rsDimensions.EOF Then
```

```
MsgBox ("Already on the last record!")
```

```
DE1.rsDimensions.MoveLast
```

```
Else
```

```
DE1.rsDimensions.MoveNext
```

```
End If
```

Case 4 'Last

```
DE1.rsDimensions.MoveLast
```

Case 6 'New

```
ad = 1
```

```
DE1.rsDimensions.MoveLast
```

```
nDID = DE1.rsDimensions.Fields(0).Value + 1
```

```
DE1.rsDimensions.AddNew
```

```
txtDID = nDID
```

```
Frame1.Enabled = True
```

```
Frame4.Enabled = True
```

```
TB1.Buttons(8).Enabled = True
```

```
For btn = 1 To 4
```

```
TB1.Buttons(btn).Enabled = False
```

```
Next
```

Case 7 'Edit

```
ed = 1
```

```
Frame1.Enabled = True
```

```
Frame4.Enabled = True
```

```
TB1.Buttons(8).Enabled = True
```

```
For btn = 1 To 4
```

```
TB1.Buttons(btn).Enabled = False
```

```
Next
```

Case 8 'Save

```
If ed = 0 And ad = 0 Then Exit Sub
```

```
If txttext = "" Then
```

```
MsgBox ("Exterior field cannot be left empty!")
```

```
txttext.SetFocus
```

```
Exit Sub
```

```
1 2 End If
```

```

1
    If txtint = "" Then
        MsgBox ("Interior field cannot be left empty!")
        txtint.SetFocus
        Exit Sub
    End If

    If txtwheelbase = "" Then
        MsgBox ("Wheel Base field cannot be left empty!")
        txtwheelbase.SetFocus
        Exit Sub
    End If

    If txtminradius = "" Then
        MsgBox ("Minimum Turn Radius field cannot be left empty!")
        txtminradius.SetFocus
        Exit Sub
    End If

    DE1.rsDimensions.Update
    ad = 0
    ed = 0
    TB1.Buttons(8).Enabled = False

    Frame1.Enabled = False
    Frame4.Enabled = False

    For btn = 1 To 4
        TB1.Buttons(btn).Enabled = True
    Next

    Case 9          'Delete
        d = MsgBox("Are you sure you want to delete this record?", vbYesNo, "Delete confirmation")
        If d = vbNo Then Exit Sub
        DE1.rsDimensions.Delete
        DE1.rsDimensions.MoveFirst

    Case 11         'Print
        DE1.rsDimensions.Filter = ("did = " & txtDID)
        rptDimensions.Show

End Select
End Sub

```

```

Private Sub tmrTime_Timer()
    lblTime.Caption = Time

```

```
End Sub
```

```

Private Sub txtminradius_KeyPress(KeyAscii As Integer) 'Validation for numbers only.

```

```

    If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Or KeyAscii = 77 Then
    Else
        MsgBox ("Invalid character entered, please re-enter data!")
        SendKeys "{Home}+{End}"
        SendKeys "{Delete}"
    End If
End Sub

```

```

Private Sub txtVID_KeyPress(KeyAscii As Integer) 'Validation for numbers only

```

```

    If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Or KeyAscii = 77 Then
    Else
        MsgBox ("Invalid character entered, please re-enter data!")
        SendKeys "{Home}+{End}"
        SendKeys "{Delete}"
    End If

```

```
1 End If
```

```
End Sub
```

```
Private Sub txtwheelbase_KeyPress(KeyAscii As Integer) 'Validation for numbers only.
```



```
    If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
```

```
    Else
```

```
        MsgBox ("Invalid character entered, please re-enter data!")
```

```
        SendKeys "{Home}+{End}"
```

```
        SendKeys "{Delete}"
```

```
    End If
```

```
End Sub
```

```

Attribute VB_Name = "frmEngine"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Dim ad As Integer
Dim ed As Integer

```

Private Sub Form_Load()

```
    lblDate.Caption = Format(Date, "long date")
```

End Sub

Private Sub TB1_ButtonClick(ByVal Button As MSComctlLib.Button)

```

Select Case Button.Index

Case 1      'First
    DE1.rsEngine.MoveFirst

Case 2      'Previous
    If DE1.rsEngine.BOF Then
        MsgBox ("Already on the first record!")
        DE1.rsEngine.MoveFirst
    Else
        DE1.rsEngine.MovePrevious
    End If

Case 3      'Next
    If DE1.rsEngine.EOF Then
        MsgBox ("Already on the last record!")
        DE1.rsEngine.MoveLast
    Else
        DE1.rsEngine.MoveNext
    End If

Case 4      'Last
    DE1.rsEngine.MoveLast

Case 6      'New
    ad = 1
    DE1.rsEngine.MoveLast
    neid = DE1.rsEngine.Fields(0).Value + 1
    DE1.rsEngine.AddNew
    txtEID = neid

    Frame1.Enabled = True
    Frame2.Enabled = True
    Frame3.Enabled = True

    TB1.Buttons(8).Enabled = True
    For btn = 1 To 4
        TB1.Buttons(btn).Enabled = False
    Next

Case 7      'Edit
    ed = 1
    Frame1.Enabled = True
    Frame2.Enabled = True
    Frame3.Enabled = True

    TB1.Buttons(8).Enabled = True
    For btn = 1 To 4
        TB1.Buttons(btn).Enabled = False
    Next

Case 8      'Save
    If ed = 0 And ad = 0 Then Exit Sub

    If txttype = "" Then
        MsgBox ("Type field cannot be left empty!")
        txttype.SetFocus
    
```

```

1 2 Exit Sub
   End If

   If txtcc = "" Then
       MsgBox ("CC field cannot be left empty!")
       txtcc.SetFocus
       Exit Sub
   End If

   If txtfuelsystem = "" Then
       MsgBox ("Fuel System field cannot be left empty!")
       txtfuelsystem.SetFocus
       Exit Sub
   End If

   If txtmaxoutput = "" Then
       MsgBox ("Max output field cannot be left empty!")
       txtmaxoutput.SetFocus
       Exit Sub
   End If

   If txtmaxtorque = "" Then
       MsgBox ("Max torque field cannot be left empty!")
       txtmaxtorque.SetFocus
       Exit Sub
   End If

   If txtprice = "" Then
       MsgBox ("Price field cannot be left empty!")
       txtprice.SetFocus
       Exit Sub
   End If

   DE1.rsEngine.Update
   ad = 0
   ed = 0
   TB1.Buttons(8).Enabled = False

   Frame1.Enabled = False
   Frame2.Enabled = False
   Frame3.Enabled = False
   For btn = 1 To 4
       TB1.Buttons(btn).Enabled = True
   Next

Case 9      'Delete
    d = MsgBox("Are you sure you want to delete this record?", vbYesNo, "Delete
confirmation!")
    If d = vbNo Then Exit Sub
    DE1.rsEngine.Delete
    DE1.rsEngine.MoveFirst

Case 11     'Print
    DE1.rsEngine.Filter = ("eid = " & txtEID)
    rptEngine.Show

End Select
End Sub

```

```

Private Sub tmrTime_Timer()
    lblTime.Caption = Time

```

```
End Sub
```

```

Private Sub txtcc_KeyPress(KeyAscii As Integer) 'Validation for numbers only.
    If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
    Else
1  MsgBox ("Invalid character entered, please re-enter data!")

```

```

1   SendKeys "{Home}+{End}"
    SendKeys "{Delete}"
End If

```

End Sub

Private Sub txtfuelsystem_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.

```

If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii
= 32 Or KeyAscii = 8 Or KeyAscii = 45 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    SendKeys "{Home}+{End}"
    SendKeys "{Delete}"
End If

```

End Sub

Private Sub txtmaxoutput_KeyPress(KeyAscii As Integer) 'Validation for numbers only.

```

If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Or KeyAscii = 32 Or KeyAscii = 47
Or KeyAscii = 40 Or KeyAscii = 41 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If

```

End Sub

Private Sub txtmaxtorque_KeyPress(KeyAscii As Integer) 'Validation for numbers only.

```

If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    SendKeys "{Home}+{End}"
    SendKeys "{Delete}"
End If

```

End Sub

Private Sub txtprice_KeyPress(KeyAscii As Integer) 'Validation for numbers only.

```

If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    SendKeys "{Home}+{End}"
    SendKeys "{Delete}"
End If

```

End Sub

Private Sub txtqty_KeyPress(KeyAscii As Integer) 'Validation for numbers only

```

If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If

```

End Sub

Private Sub txtVID_KeyPress(KeyAscii As Integer) 'Validation for numbers only

```

If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
1   'SendKeys "{Delete}"

```

```
1 End If  
End Sub
```

```

Attribute VB_Name = "frmExterior"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Dim ad As Integer
Dim ed As Integer

```

Private Sub Form_Load()

```
    lblDate.Caption = Format(Date, "long date")
```

End Sub

Private Sub TB1_ButtonClick(ByVal Button As MSComctlLib.Button)

```

Select Case Button.Index

Case 1      'First
    DE1.rsExterior.MoveFirst

Case 2      'Previous
    If DE1.rsExterior.BOF Then
        MsgBox ("Already on the first record!")
        DE1.rsExterior.MoveFirst
    Else
        DE1.rsExterior.MovePrevious
    End If

Case 3      'Next
    If DE1.rsExterior.EOF Then
        MsgBox ("Already on the last record!")
        DE1.rsExterior.MoveLast
    Else
        DE1.rsExterior.MoveNext
    End If

Case 4      'Last
    DE1.rsExterior.MoveLast

Case 6      'New
    ad = 1
    DE1.rsExterior.MoveLast
    nextID = DE1.rsExterior.Fields(0).Value + 1
    DE1.rsExterior.AddNew
    txttextID = nextID

    Frame1.Enabled = True
    Frame2.Enabled = True
    Frame3.Enabled = True
    FrameID.Enabled = True

    TB1.Buttons(8).Enabled = True

    For btn = 1 To 4
        TB1.Buttons(btn).Enabled = False
    Next

Case 7      'Edit
    ed = 1
    Frame1.Enabled = True
    Frame2.Enabled = True
    Frame3.Enabled = True
    FrameID.Enabled = True
    TB1.Buttons(8).Enabled = True

    For btn = 1 To 4
        TB1.Buttons(btn).Enabled = False
    Next

Case 8      'Save
    If ed = 0 And ad = 0 Then Exit Sub

1  2-If txtDID = "" Then

```

```
1 2 MsgBox ("DID field cannot be left empty!")
   txtDID.SetFocus
   Exit Sub
End If

If txtdoormirrors = "" Then
    MsgBox ("Door Mirrors field cannot be left empty!")
    txtdoormirrors.SetFocus
    Exit Sub
End If

If txtFheadlamp = "" Then
    MsgBox ("Front Head Lamp field cannot be left empty!")
    txtFheadlamp.SetFocus
    Exit Sub
End If

If txtrearlamps = "" Then
    MsgBox ("Rear Lamps field cannot be left empty!")
    txtrearlamps.SetFocus
    Exit Sub
End If

If txtwindshield = "" Then
    MsgBox ("Wind Shield field cannot be left empty!")
    txtwindshield.SetFocus
    Exit Sub
End If

If txtantenna = "" Then
    MsgBox ("Antenna field cannot be left empty!")
    txtantenna.SetFocus
    Exit Sub
End If

If txtmudguard = "" Then
    MsgBox ("Mud guard field cannot be left empty!")
    txtmudguard.SetFocus
    Exit Sub
End If

If txtreargarnish = "" Then
    MsgBox ("Rear Garnish field cannot be left empty!")
    txtreargarnish.SetFocus
    Exit Sub
End If

If txtimpactbars = "" Then
    MsgBox ("Impact Bars field cannot be left empty!")
    txtimpactbars.SetFocus
    Exit Sub
End If

If txtsideskirts = "" Then
    MsgBox ("Side Skirts field cannot be left empty!")
    txtsideskirts.SetFocus
    Exit Sub
End If

If txtswipers = "" Then
    MsgBox ("Wind Swipers field cannot be left empty!")
    txtswipers.SetFocus
    Exit Sub
```

```
1 2
```

```

1 2└End If

┌If txtwheelcaps = "" Then
│   MsgBox ("Wheel caps field cannot be left empty!")
│   txtwheelcaps.SetFocus
│   Exit Sub
└End If

┌If txtldhandles = "" Then
│   MsgBox ("Door Handles field cannot be left empty!")
│   txtldhandles.SetFocus
│   Exit Sub
└End If

DE1.rsExterior.Update
ad = 0
ed = 0
TB1.Buttons(8).Enabled = False

Frame1.Enabled = False
Frame2.Enabled = False
Frame3.Enabled = False
FrameID.Enabled = False

┌For btn = 1 To 4
│   TB1.Buttons(btn).Enabled = True
└Next

┌Case 9      'Delete
│   d = MsgBox("Are you sure you want to delete this record?", vbYesNo, "Delete
│   confirmation!")
│   If d = vbNo Then Exit Sub
│   DE1.rsExterior.Delete
│   DE1.rsExterior.MoveFirst
└Case 11     'Print
│   DE1.rsExterior.Filter = ("extid = " & txttextID)
│   rptExterior.Show
└End Select
End Sub

```

```

Private Sub tmrTime_Timer()
    lblTime.Caption = Time
End Sub

```

```

Private Sub txtantenna_KeyPress(KeyAscii As Integer) 'Validation for
alphabets only.
┌If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii
= 32 Or KeyAscii = 8 Or KeyAscii = 45 Then
└Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
└End If
End Sub

```

```

Private Sub txtldhandles_KeyPress(KeyAscii As Integer) 'Validation for
alphabets only.
┌If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii
= 32 Or KeyAscii = 8 Then
└Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
└End If
End Sub

```

```

Private Sub txtimpactbars_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txtmudguard_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Or KeyAscii = 40 Or KeyAscii = 41 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txttprice_KeyPress(KeyAscii As Integer) 'Validation for numbers only.
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txtqty_KeyPress(KeyAscii As Integer) 'Validation for numbers only.
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txtreargarnish_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Or KeyAscii = 40 Or KeyAscii = 41 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txtrearlamps_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txtsideskirts_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.

```

```
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

Private Sub txtwheelcaps_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.

```
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Or KeyAscii = 34 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

Private Sub txtwindshield_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.

```
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Or KeyAscii = 47 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

Private Sub txtwsipers_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.

```
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```

Attribute VB_Name = "frmInterior"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Dim ad As Integer
Dim ed As Integer

```

Private Sub Form_Load()

```
    lblDate.Caption = Format(Date, "long date")
```

End Sub

Private Sub TB1_ButtonClick(ByVal Button As MSComctlLib.Button)

```

Select Case Button.Index

Case 1      'First
    DE1.rsInterior.MoveFirst

Case 2      'Previous
    If DE1.rsInterior.BOF Then
        MsgBox ("Already on the first record!")
        DE1.rsInterior.MoveFirst
    Else
        DE1.rsInterior.MovePrevious
    End If

Case 3      'Next
    If DE1.rsInterior.EOF Then
        MsgBox ("Already on the last record!")
        DE1.rsInterior.MoveLast
    Else
        DE1.rsInterior.MoveNext
    End If

Case 4      'Last
    DE1.rsInterior.MoveLast

Case 6      'New
    ad = 1
    DE1.rsInterior.MoveLast
    nIID = DE1.rsInterior.Fields(0).Value + 1
    DE1.rsInterior.AddNew
    txtIID = nIID

    Frame1.Enabled = True
    Frame2.Enabled = True
    Frame3.Enabled = True
    Frame4.Enabled = True
    TB1.Buttons(8).Enabled = True

    For btn = 1 To 4
        TB1.Buttons(btn).Enabled = False
    Next

Case 7      'Edit
    ed = 1
    Frame1.Enabled = True
    Frame2.Enabled = True
    Frame3.Enabled = True
    Frame4.Enabled = True

    TB1.Buttons(8).Enabled = True

    For btn = 1 To 4
        TB1.Buttons(btn).Enabled = False
    Next

Case 8      'Save
    If ed = 0 And ad = 0 Then Exit Sub

```

1

```
1  If txtoptions = "" Then
    MsgBox ("CD player/ AM- FM field  cannot be left empty!")
    txtoptions.SetFocus
    Exit Sub
End If

If txtdigclock = "" Then
    MsgBox ("Digital Clock field cannot be left empty!")
    txtdigclock.SetFocus
    Exit Sub
End If

If txtpowerwindows = "" Then
    MsgBox ("Power Windows field cannot be left empty!")
    txtpowerwindows.SetFocus
    Exit Sub
End If

If txtspeakers = "" Then
    MsgBox ("Speakers field cannot be left empty!")
    txtspeakers.SetFocus
    Exit Sub
End If

If txtwoodgrain = "" Then
    MsgBox ("Wood Grain field cannot be left empty!")
    txtwoodgrain.SetFocus
    Exit Sub
End If

If txttachometer = "" Then
    MsgBox ("Tachometer field cannot be left empty!")
    txttachometer.SetFocus
    Exit Sub
End If

If txtac = "" Then
    MsgBox ("AC field cannot be left empty!")
    txtac.SetFocus
    Exit Sub
End If

If txtheater = "" Then
    MsgBox ("Heater field cannot be left empty!")
    txtheater.SetFocus
    Exit Sub
End If

If txtgearlever = "" Then
    MsgBox ("Gear Lever field cannot be left empty!")
    txtgearlever.SetFocus
    Exit Sub
End If

If txttintcolorscheme = "" Then
    MsgBox ("Interior Color scheme field cannot be left empty!")
    txttintcolorscheme.SetFocus
    Exit Sub
End If

If txtseats = "" Then
    MsgBox ("Seats field cannot be left empty!")
    txtseats.SetFocus
1  2
```

```

1  2  Exit Sub

    End If

    If txtspeedometer = "" Then
        MsgBox ("Speedometer field cannot be left empty!")
        txtspeedometer.SetFocus
        Exit Sub
    End If

    If txtsunvisor = "" Then
        MsgBox ("Sunvisor field cannot be left empty!")
        txtsunvisor.SetFocus
        Exit Sub
    End If

    If txtSID = "" Then
        MsgBox ("Safety ID field cannot be left empty!")
        txtSID.SetFocus
        Exit Sub
    End If

    If txtCconsolebox = "" Then
        MsgBox ("Central Console Box field cannot be left empty!")
        txtCconsolebox.SetFocus
        Exit Sub
    End If

    DE1.rsInterior.Update
    ad = 0
    ed = 0
    TB1.Buttons(8).Enabled = False

    Frame1.Enabled = False
    Frame2.Enabled = False
    Frame3.Enabled = False
    Frame4.Enabled = False

    For btn = 1 To 4
        TB1.Buttons(btn).Enabled = True
    Next

    Case 9      'Delete
        d = MsgBox("Are you sure you want to delete this record?", vbYesNo, "Delete
        confirmation!")
        If d = vbNo Then Exit Sub
        DE1.rsInterior.Delete
        DE1.rsInterior.MoveFirst

    Case 11     'Print
        DE1.rsInterior.Filter = ("iid = " & txtIID)
        rptInterior.Show

    End Select
End Sub

```

```

Private Sub tmrTime_Timer()
    lblTime.Caption = Time

```

```
End Sub
```

```

Private Sub txtac_KeyPress(KeyAscii As Integer) 'Validation for alphabets
only.
    If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii
    = 32 Or KeyAscii = 8 Then
    Else
        MsgBox ("Invalid character entered, please re-enter data!")
1  'SendKeys "{Home}+{End}"

```



```

1      'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txtCdoorlocking_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txtdigclock_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txtgearlever_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Or KeyAscii = 40 Or KeyAscii = 41 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txttheater_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txtintcolorscheme_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txtpowerwindows_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
1

```

```
1└End If
End Sub
```

```
Private Sub txtprice_KeyPress(KeyAscii As Integer) 'Validation for numbers only.
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txtqty_KeyPress(KeyAscii As Integer) 'Validation for numbers only
.
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txtseats_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txtspeakers_KeyPress(KeyAscii As Integer) 'Validation for numbers only.
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txtspeedometer_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txtsunvisor_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Or KeyAscii = 40 Or KeyAscii = 41 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txttachometer_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txtwoodgrain_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Attribute VB_Name = "frmOrder"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
```

```
Dim ad As Integer
Dim ed As Integer
```

```
Public Sub buyerSearch()
    If ed = 1 Or ad = 1 Or txtBID = "" Then Exit Sub
    DE1.rsBuyer.MoveFirst
    DE1.rsBuyer.Find ("bid =" & txtBID)
```

```
End Sub
```

```
Private Sub chkFeatures_Click()
```

```
End Sub
```

```
Private Sub chkVehicle_Click() 'toggles visibility of dgVehicle
```

```
    If dgVehicle.Visible = False Then
        dgVehicle.Visible = True
    Else
        dgVehicle.Visible = False
    End If
```

```
End Sub
```

```
Private Sub cmdSearchBID_Click()
```

```
    sbid = InputBox("Enter Buyer's ID to search the record", "Search")
```

```
    If sbid = 0 Then Exit Sub
```

```
    DE1.rsBuyer.MoveFirst
    DE1.rsBuyer.Find ("bid =" & sbid)
```

```
    If DE1.rsBuyer.EOF Then
        MsgBox ("Record not found!")
        DE1.rsBuyer.MoveFirst
    End If
```

```
End Sub
```

```
Private Sub cmdSearchBname_Click()
```

```
    stitle = InputBox("Enter Buyer's Title to search the record", "Search")
```

```
    If stitle = " " Then Exit Sub
```

```
    DE1.rsBuyer.MoveFirst
    DE1.rsBuyer.Find ("title =" & stitle & "")
```

```
    If DE1.rsBuyer.EOF Then
        MsgBox ("Record not found!")
        DE1.rsBuyer.MoveFirst
    End If
```

```
End Sub
```

```
Public Sub collapse()
```

```
    Me.Height = 6525
    blackbar.Top = 5640
    redbar.Top = 6000
```

```
End Sub
```

```
Private Sub Combo1_Click()
```

```
    If ad = 0 And ed = 0 Then Exit Sub
    DE1.rsVehicle.Filter = "vname =" & Combo1.Text & ""
```

```
dgVehicle.Visible = True
```

```
End Sub
```

```
Private Sub Combo2_Click()
```

```
    If ad = 0 And ed = 0 Then Exit Sub
    If Combo1.Text = "" Then
        MsgBox ("Select Vehicle First")
        Exit Sub
    End If
    Dim dg As Control
    For Each dg In Controls
        If TypeOf dg Is DataGridView Then
            dg.Visible = False
        End If
    Next

    dgVehicle.Visible = True
    If Combo2.Text = "Performance" Then
        DE1.rsPerformance.Filter = ("qty >=" & "1")
        dgPerformance.Visible = True

    ElseIf Combo2.Text = "Engine" Then
        DE1.rsEngine.Filter = ("qty >=" & "1")
        dgEngine.Visible = True

    ElseIf Combo2.Text = "Interior" Then
        DE1.rsInterior.Filter = ("qty >=" & "1")
        dgInterior.Visible = True

    ElseIf Combo2.Text = "Exterior" Then
        DE1.rsExterior.Filter = ("qty >=" & "1")
        dgExterior.Visible = True

    ElseIf Combo2.Text = "Safety" Then
        DE1.rsSafety.Filter = ("qty >=" & "1")
        dgSafety.Visible = True
    End If
```

```
    Call extend
```

```
End Sub
```

```
Private Sub dgEngine_Click()
```

```
    txtEID = DE1.rsEngine.Fields(0).Value
    Check1(0).Value = 1
    'Call collapse
```

```
End Sub
```

```
Private Sub dgexterior_click()
```

```
    txttextID = DE1.rsExterior.Fields(0).Value
    Check1(1).Value = 1
    'Call collapse
```

```
End Sub
```

```
Private Sub dgInterior_click()
```

```
    txtIID = DE1.rsInterior.Fields(0).Value
    Check1(2).Value = 1
    'Call collapse
```

```
End Sub
```

```
Private Sub dgperformance_click()
```

```
    txtPID = DE1.rsPerformance.Fields(0).Value
    Check1(3).Value = 1
    'Call collapse
```

```
End Sub
```

```
Private Sub dgSafety_click()
```

```

txtSID = DE1.rsSafety.Fields(0).Value
Check1(4).Value = 1
'Call collapse
End Sub

```

```

Private Sub dgVehicle_Click()
txtVID = DE1.rsVehicle.Fields(0).Value
'dgVehicle.Visible = False
End Sub

```

```

Public Sub extend()
Me.Height = 10230
blackbar.Top = 9360
redbar.Top = 9720
End Sub

```

```

Private Sub Form_Activate()                                'adds all the distinct vehicle names
to the combo box
While ADOV.Recordset.EOF = False
    Combo1.AddItem (ADOV.Recordset.Fields("vname").Value)
    ADOV.Recordset.MoveNext
Wend
Call relation
End Sub

```

```

Private Sub Form_Load()                                    'applies distinct filter of vehicle
name to the ADODC table
TB1.Enabled = True
lblDate = Format(Date, "long date")

ADOV.RecordSource = "Select distinct vname from vehicle"
ADOV.Refresh

'capitalizes all the captions of datagrids

For s = 0 To dgSafety.Columns.Count - 1
    dgSafety.Columns(s).Caption = UCase(dgSafety.Columns(s).Caption)
Next

For p = 0 To dgSafety.Columns.Count - 1
    dgSafety.Columns(f).Caption = UCase(dgSafety.Columns(f).Caption)
Next

For i = 0 To dgSafety.Columns.Count - 1
    dgSafety.Columns(f).Caption = UCase(dgSafety.Columns(f).Caption)
Next
Dim dg As Control
For Each dg In Controls
    If TypeOf dg Is DataGrid Then
        dg.Visible = False
    End If
Next

End Sub

```

```

Public Function relation()
If txtOID = "" Then Exit Function
With DE1

    .rsEngine.Find ("eid =") & txtEID
    .rsExterior.Find ("extid =") & txttextID
    .rsInterior.Find ("iid =") & txtIID
    .rsPerformance.Find ("pid =") & txtPID
    .rsSafety.Find ("sid =") & txtSID
    .rsVehicle.Filter = ("vname <> Null")
    .rsVehicle.Find ("vid =" & txtVID)
    Combo1.Text = .rsVehicle.Fields("vname").Value
End With
End Function

```

```
Private Sub TB1_ButtonClick(ByVal Button As MSComctlLib.Button)
```

```
    Select Case Button.Index
```

```
    Case 1          'First
```

```
        DE1.rsOrder.MoveFirst
        Call buyerSearch
        Call relation
```

```
    Case 2          'Previous
```

```
        If DE1.rsOrder.BOF Then
            MsgBox ("Already on the first record!")
            DE1.rsOrder.MoveFirst
        Else
            DE1.rsOrder.MovePrevious
        End If
        Call buyerSearch
        Call relation
```

```
    Case 3          'Next
```

```
        If DE1.rsOrder.EOF Then
            MsgBox ("Already on the last record!")
            DE1.rsOrder.MoveLast
        Else
            DE1.rsOrder.MoveNext
        End If
        Call buyerSearch
        Call relation
```

```
    Case 4          'Last
```

```
        DE1.rsOrder.MoveLast
        Call buyerSearch
        Call relation
```

```
    Case 6          'New
```

```
        ad = 1
        DTPicker1.Value = Date
        DE1.rsOrder.MoveLast
        noid = DE1.rsOrder.Fields(0).Value + 1
        DE1.rsOrder.MoveLast
        DE1.rsOrder.AddNew
        txtOID = noid
        txtqty = 1
        txtBID = txtBIDB

        txtono = Format(Date, "yymmdd") & "-" & txtOID
        For btn = 1 To 4
            TB1.Buttons(btn).Enabled = False
        Next
        'DE1.rsOrder.Fields ("odate" = DTPicker1.Value)
        'Dim ctla As Control
        'For Each ctla In Controls
        'If TypeOf ctla Is TextBox Then
        '    ctla.Locked = False
        'End If
        'Next
        Frame1.Enabled = True

        framBuyer.Enabled = True
        TB1.Buttons(8).Enabled = True
```

```
    Case 7          'Edit
```

```
        ed = 1
        Dim ctle As Control
        For Each ctle In Controls
            If TypeOf ctle Is TextBox Then
                ctle.Locked = False
            End If
        Next
```

```
        For btn = 1 To 4
```

```
            TB1.Buttons(btn).Enabled = False
```

```
1
```

```
2
```

```

1 2-Next

    TB1.Buttons(8).Enabled = True
    Frame1.Enabled = True
    framBuyer.Enabled = True

    For chk = 0 To 4
        Check1(chk).Value = 1
    Next

    Call relation

Case 8      'Save/Update

    For chk = 0 To 4
        If Check1(chk).Value = 0 Then
            MsgBox ("Select " & Check1(chk).Caption & " features to complete and save the order!")
            Exit Sub
        End If
    Next

    With DE1
        Dim ocost As Long
        Dim ocostqty As Long

        ocost = Val(.rsEngine.Fields("price").Value) + Val(.rsExterior.Fields("price").Value) + Val(.rsInterior.Fields("price").Value) + Val(.rsPerformance.Fields("price").Value)
        ocostqty = ocost * Val(txtqty)
        txtordercost = Format(ocostqty, "#,##")
        ' .rsOrder.Fields("ordercost").Value = ocostqty
        .rsOrder.Fields("odate").Value = DTPicker1.Value
        .rsEngine.Fields("qty").Value = .rsEngine.Fields("qty").Value - 1
        .rsExterior.Fields("qty").Value = .rsExterior.Fields("qty").Value - 1
        .rsInterior.Fields("qty").Value = .rsInterior.Fields("qty").Value - 1
        .rsPerformance.Fields("qty").Value = .rsPerformance.Fields("qty").Value - 1
        .rsSafety.Fields("qty").Value = .rsSafety.Fields("qty").Value - 1
    End With

    Dim ctls As Control
    For Each ctls In Controls
        If TypeOf ctls Is TextBox Then
            ctls.Locked = True
        End If
    Next
    TB1.Buttons(8).Enabled = False
    Frame1.Enabled = False
    framBuyer.Enabled = False
    ad = 0
    ed = 0
    DE1.rsOrder.Update
    'txtOID = DE1.rsOrder.Fields("oid").Value
    For btn = 1 To 4
        TB1.Buttons(btn).Enabled = True
    Next

    DE1.rsCommand1.Filter = ("oid = " & txtOID)
    rptOrder.Show

Case 9      'Delete

    d = MsgBox("Are you sure you want to delete this record?", vbYesNo, "Delete Confirmation")
    If d = vbYes Then
        DE1.rsOrder.Delete
        DE1.rsOrder.MoveFirst
    End If

Case 11     'Print
1  DE1.rsCommand1.Filter = ("oid = " & txtOID)

```

```
1| rptOrder.Show
|End Select
End Sub
```

```
Private Sub tmrTime_Timer()
    lblTime = Time
End Sub
```

```
Private Sub txtBIDB_Change()
    If txtBIDB = "" Then Exit Sub
    txtBID = txtBIDB
End Sub
```

```
Private Sub txtqty_KeyPress(KeyAscii As Integer) 'Validation for numbers only
.
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Private Sub txttitle_KeyPress(KeyAscii As Integer) 'Validation for alphabets
only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii
= 32 Or KeyAscii = 8 Or KeyAscii = 34 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub
```

```
Attribute VB_Name = "frmOrderQuery"  
Attribute VB_GlobalNameSpace = False  
Attribute VB_Creatable = False  
Attribute VB_PredeclaredId = True  
Attribute VB_Exposed = False
```

```
Private Sub Command1_Click()  
    DE1.rsCommand2.Filter = "oid <> NULL"  
End Sub
```

```
Private Sub Form_Load()  
    lblDate.Caption = Format(Date, "long date")  
End Sub
```

```
Private Sub Text1_KeyPress(KeyAscii As Integer)  
    If KeyAscii = 13 Then  
        Select Case Combo1.Text  
            Case "Buyer"  
                DE1.rsCommand2.Filter = ("title like '%" & Text1 & "%'")  
            Case "Vehicle"  
                DE1.rsCommand2.Filter = ("vname like '%" & Text1 & "%'")  
            Case "Order No"  
                DE1.rsCommand2.Filter = ("ono =" & Text1 & "")  
            Case "Order Date"  
                DE1.rsCommand2.Filter = ("odatetitle =" & Text1 & "#")  
            Case "Bank Name"  
                DE1.rsCommand2.Filter = ("bnkname like '%" & Text1 & "%'")  
        End Select  
    End If  
End Sub
```

```
Private Sub tmrTime_Timer()  
    lblTime.Caption = Time  
End Sub
```

```

Attribute VB_Name = "frmPerformance"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Dim npid As Integer
Dim ad As Integer
Dim ed As Integer

```

```

Private Sub cmdReset_Click()
    If ad = 0 And ed = 0 Then Exit Sub

    Dim rs As Control
    For Each rs In Controls
        If TypeOf rs Is TextBox Then
            rs.Text = ""
        End If
    Next
    txtPID = npid

```

End Sub

```

Private Sub Command1_Click()

```

End Sub

```

Private Sub Command1_MouseMove(Button As Integer, Shift As Integer, X As
Single, Y As Single)

```

End Sub

```

Private Sub Form_Load()
    lblDate.Caption = Format(Date, "long date")

```

End Sub

```

Private Sub TB1_ButtonClick(ByVal Button As MSComctlLib.Button)

```

```

    Select Case Button.Index
    Case 1      'First
        DE1.rsPerformance.MoveFirst
    Case 2      'Previous
        If DE1.rsPerformance.BOF Then
            MsgBox ("Already on the first record!")
            DE1.rsPerformance.MoveFirst
        Else
            DE1.rsPerformance.MovePrevious
        End If
    Case 3      'Next
        If DE1.rsPerformance.EOF Then
            MsgBox ("Already on the last record!")
            DE1.rsPerformance.MoveLast
        Else
            DE1.rsPerformance.MoveNext
        End If
    Case 4      'Last
        DE1.rsPerformance.MoveLast
    Case 6      'New
        ad = 1
        DE1.rsPerformance.MoveLast
        npid = DE1.rsPerformance.Fields(0).Value + 1
        DE1.rsPerformance.AddNew
        txtPID = npid
1      Frame1.Enabled = True

```

```

1      Frame2.Enabled = True
      Frame3.Enabled = True

      TB1.Buttons(8).Enabled = True

      For btn = 1 To 4
      TB1.Buttons(btn).Enabled = False
      Next

Case 7      'Edit
      ed = 1
      Frame1.Enabled = True
      Frame2.Enabled = True
      Frame3.Enabled = True

      TB1.Buttons(8).Enabled = True

      For btn = 1 To 4
      TB1.Buttons(btn).Enabled = False
      Next

Case 8      'Save
      If ed = 0 And ad = 0 Then Exit Sub

      If txttransmission = "" Then
      MsgBox ("Transmission field cannot be left empty!")
      txttransmission.SetFocus
      Exit Sub
      End If

      If txtfbrakes = "" Then
      MsgBox ("Front brakes field cannot be left empty!")
      txtfbrakes.SetFocus
      Exit Sub
      End If

      If txtrbrakes = "" Then
      MsgBox ("Rear Brakes field cannot be left empty!")
      txtrbrakes.SetFocus
      Exit Sub
      End If

      If txtfsuspension = "" Then
      MsgBox ("Front Suspension field cannot be left empty!")
      txtfsuspension.SetFocus
      Exit Sub
      End If

      If txtstablizerbar = "" Then
      MsgBox ("Stabilizer Bar field cannot be left empty!")
      txtstablizerbar.SetFocus
      Exit Sub
      End If

      If txtsteeringsystem = "" Then
      MsgBox ("Steering System field cannot be left empty!")
      txtsteeringsystem.SetFocus
      Exit Sub
      End If

      If txtwheel = "" Then
      MsgBox ("Wheel field cannot be left empty!")
      txtwheel.SetFocus
      Exit Sub
      End If

1 2 If txttankcapacity = "" Then

```

```

1  2  MsgBox ("Tank Capacity field cannot be left empty!")
    txttankcapacity.SetFocus
    Exit Sub

End If

If txtsuspension = "" Then
    MsgBox ("Rear Suspension field cannot be left empty!")
    txtsuspension.SetFocus
    Exit Sub
End If

If txtEID = "" Then
    MsgBox ("Engine ID field cannot be left empty!")
    txtEID.SetFocus
    Exit Sub
End If

DE1.rsPerformance.Update
ad = 0
ed = 0
TB1.Buttons(8).Enabled = False

Frame1.Enabled = False
Frame2.Enabled = False
Frame3.Enabled = False
For btn = 1 To 4
    TB1.Buttons(btn).Enabled = True
Next

Case 9      'Delete
    d = MsgBox("Are you sure you want to delete this record?", vbYesNo, "Delete
confirmation!")
    If d = vbNo Then Exit Sub
    DE1.rsPerformance.Delete
    DE1.rsPerformance.MoveFirst

Case 11     'Print
    DE1.rsPerformance.Filter = ("pid =" & txtPID)
    rptPerformance.Show
End Select
End Sub

```

```

Private Sub tmrTime_Timer()
    lblTime.Caption = Time

```

```
End Sub
```

```

Private Sub txtfsuspension_KeyPress(KeyAscii As Integer) 'Validation for
alphabets only.
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii
= 32 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txtprice_KeyPress(KeyAscii As Integer) 'Validation for numbers
only.
If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
Else
    MsgBox ("Invalid character entered, please re-enter data!")
    'SendKeys "{Home}+{End}"
    'SendKeys "{Delete}"
End If
End Sub

```

```

Private Sub txtqty_KeyPress(KeyAscii As Integer) 'Validation for numbers only
    .
    If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
    Else
        MsgBox ("Invalid character entered, please re-enter data!")
        'SendKeys "{Home}+{End}"
        'SendKeys "{Delete}"
    End If
End Sub

```

```

Private Sub txtrsuspension_KeyPress(KeyAscii As Integer) 'Validation for
alphabets only.
    If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii
    = 32 Or KeyAscii = 8 Then
    Else
        MsgBox ("Invalid character entered, please re-enter data!")
        'SendKeys "{Home}+{End}"
        'SendKeys "{Delete}"
    End If
End Sub

```

```

Private Sub txtstablizerbar_KeyPress(KeyAscii As Integer) 'Validation for
alphabets only.
    If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii
    = 32 Or KeyAscii = 8 Or KeyAscii = 47 Then
    Else
        MsgBox ("Invalid character entered, please re-enter data!")
        'SendKeys "{Home}+{End}"
        'SendKeys "{Delete}"
    End If
End Sub

```

```

Private Sub txtsteeringsystem_KeyPress(KeyAscii As Integer) 'Validation for
alphabets only.
    If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii
    = 32 Or KeyAscii = 8 Then
    Else
        MsgBox ("Invalid character entered, please re-enter data!")
        'SendKeys "{Home}+{End}"
        'SendKeys "{Delete}"
    End If
End Sub

```

```

Private Sub txttankcapacity_KeyPress(KeyAscii As Integer) 'Validation for
numbers only.
    If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
    Else
        MsgBox ("Invalid character entered, please re-enter data!")
        'SendKeys "{Home}+{End}"
        'SendKeys "{Delete}"
    End If
End Sub

```

```

Attribute VB_Name = "frmSafety"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Dim ad As Integer
Dim ed As Integer

```

Private Sub Form_Load()

```
    lblDate.Caption = Format(Date, "long date")
```

End Sub

Private Sub TB1_ButtonClick(ByVal Button As MSComctlLib.Button)

```
    Select Case Button.Index
```

Case 1 'First

```
        DE1.rsSafety.MoveFirst
```

Case 2 'Previous

```
        If DE1.rsSafety.BOF Then
```

```
            MsgBox ("Already on the first record!")
```

```
            DE1.rsSafety.MoveFirst
```

```
        Else
```

```
            DE1.rsSafety.MovePrevious
```

```
        End If
```

Case 3 'Next

```
        If DE1.rsSafety.EOF Then
```

```
            MsgBox ("Already on the last record!")
```

```
            DE1.rsSafety.MoveLast
```

```
        Else
```

```
            DE1.rsSafety.MoveNext
```

```
        End If
```

Case 4 'Last

```
        DE1.rsSafety.MoveLast
```

Case 6 'New

```
        ad = 1
```

```
        DE1.rsSafety.MoveLast
```

```
        nSID = DE1.rsSafety.Fields(0).Value + 1
```

```
        DE1.rsSafety.AddNew
```

```
        txtSID = nSID
```

```
        Frame1.Enabled = True
```

```
        Frame2.Enabled = True
```

```
        Frame3.Enabled = True
```

```
        TB1.Buttons(8).Enabled = True
```

```
        For btn = 1 To 4
```

```
            TB1.Buttons(btn).Enabled = False
```

```
        Next
```

Case 7 'Edit

```
        ed = 1
```

```
        Dim ctle As Control
```

```
        For Each ctle In Controls
```

```
            If TypeOf ctle Is TextBox Then
```

```
                ctle.Locked = False
```

```
            End If
```

```
        Next
```

```
        TB1.Buttons(8).Enabled = True
```

```
        For btn = 1 To 4
```

```
            TB1.Buttons(btn).Enabled = False
```

```
        Next
```

```
        Frame1.Enabled = True
```

```
        Frame2.Enabled = True
```

1

```

1      Frame3.Enabled = True

Case 8      'Save
    If ed = 0 And ad = 0 Then Exit Sub

    If txtkeywarn = "" Then
        MsgBox ("Key Warning field cannot be left empty!")
        txtkeywarn.SetFocus
        Exit Sub
    End If

    If txtlightswarn = "" Then
        MsgBox ("Lights Warning field cannot be left empty!")
        txtlightswarn.SetFocus
        Exit Sub
    End If

    If txtreversegearwarn = "" Then
        MsgBox ("Reverse Gear Warning field cannot be left empty!")
        txtreversegearwarn.SetFocus
        Exit Sub
    End If

    If txtfseatbelt = "" Then
        MsgBox ("Front Seat Belt field cannot be left empty!")
        txtfseatbelt.SetFocus
        Exit Sub
    End If

    If txtrseatbelt = "" Then
        MsgBox ("Rear Seat Belt field cannot be left empty!")
        txtrseatbelt.SetFocus
        Exit Sub
    End If

    If txtairbags = "" Then
        MsgBox ("Air Bags field cannot be left empty!")
        txtairbags.SetFocus
        Exit Sub
    End If

    DE1.rsSafety.Update
    ad = 0
    ed = 0
    TB1.Buttons(8).Enabled = False

    Frame1.Enabled = False
    Frame2.Enabled = False
    Frame3.Enabled = False

    For btn = 1 To 4
        TB1.Buttons(btn).Enabled = True
    Next

Case 9      'Delete
    d = MsgBox("Are you sure you want to delete this record?", vbYesNo, "Delete
confirmation!")
    If d = vbNo Then Exit Sub
    DE1.rsSafety.Delete
    DE1.rsSafety.MoveFirst

Case 11     'Print
    DE1.rsSafety.Filter = ("sid =" & txtSID)
    rptSafety.Show

End Select
End Sub

```

```
Private Sub tmrTime_Timer()
```

```
    lblTime.Caption = Time
```

```
End Sub
```

```
Private Sub txtcollisionsafety_KeyPress(KeyAscii As Integer) 'Validation for alphabets only. ↴
```

```
    If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then ↴
```

```
    Else
```

```
        MsgBox ("Invalid character entered, please re-enter data!")
```

```
        SendKeys "{Home}+{End}"
```

```
        SendKeys "{Delete}"
```

```
    End If
```

```
End Sub
```

```
Private Sub txtkeywarn_KeyPress(KeyAscii As Integer) 'Validation for alphabets only. ↴
```

```
    If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then ↴
```

```
    Else
```

```
        MsgBox ("Invalid character entered, please re-enter data!")
```

```
        SendKeys "{Home}+{End}"
```

```
        SendKeys "{Delete}"
```

```
    End If
```

```
End Sub
```

```
Private Sub txtlightswarn_KeyPress(KeyAscii As Integer) 'Validation for alphabets only. ↴
```

```
    If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then ↴
```

```
    Else
```

```
        MsgBox ("Invalid character entered, please re-enter data!")
```

```
        SendKeys "{Home}+{End}"
```

```
        SendKeys "{Delete}"
```

```
    End If
```

```
End Sub
```

```
Private Sub txtprice_KeyPress(KeyAscii As Integer) 'Validation for numbers only. ↴
```

```
    If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
```

```
    Else
```

```
        MsgBox ("Invalid character entered, please re-enter data!")
```

```
        SendKeys "{Home}+{End}"
```

```
        SendKeys "{Delete}"
```

```
    End If
```

```
End Sub
```

```
Private Sub txtqty_KeyPress(KeyAscii As Integer) 'Validation for numbers only ↴
```

```
    *
```

```
    If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
```

```
    Else
```

```
        MsgBox ("Invalid character entered, please re-enter data!")
```

```
        SendKeys "{Home}+{End}"
```

```
        SendKeys "{Delete}"
```

```
    End If
```

```
End Sub
```

```
Private Sub txtreversegearwarn_KeyPress(KeyAscii As Integer) 'Validation for alphabets only. ↴
```

```
    If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then ↴
```

```
    Else
```

```
        MsgBox ("Invalid character entered, please re-enter data!")
```

```
        SendKeys "{Home}+{End}"
```

```
        SendKeys "{Delete}"
```

```
1 End If
```

End Sub

Private Sub txtstabilitycontrol_KeyPress(KeyAscii As Integer) 'Validation for alphabets only. ↵

If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Or KeyAscii = 8 Then ↵

Else

MsgBox ("Invalid character entered, please re-enter data!")

SendKeys "{Home}+{End}"

SendKeys "{Delete}"

End If

End Sub

```
Attribute VB_Name = "frmSplash"  
Attribute VB_GlobalNameSpace = False  
Attribute VB_Creatable = False  
Attribute VB_PredeclaredId = True  
Attribute VB_Exposed = False
```

```
Private Sub cmdCancel_Click()  
    End  
End Sub
```

```
Private Sub cmdOK_Click()  
    If txtUserName.Text = "TALHA" And txtPassword = "talha" Then  
        Unload Me  
        mdiToyota.Show  
    Else  
        MsgBox ("Invalid Username or Password!"), vbCritical  
        txtPassword.SetFocus  
        SendKeys "{Home}+{End}"  
    End If  
End Sub
```

```

Attribute VB_Name = "frmVehicle"
Attribute VB_GlobalNameSpace = False
Attribute VB_Creatable = False
Attribute VB_PredeclaredId = True
Attribute VB_Exposed = False
Dim ad As Integer
Dim ed As Integer

```

```

Private Sub Command1_Click()

```

```

    'DE1.rsVehicle.MoveNext

```

```

    MSHFlexGrid1.Refresh

```

```

End Sub

```

```

Private Sub Form_Load()

```

```

    lblDate.Caption = Format(Date, "long date")

```

```

End Sub

```

```

Private Sub TB1_ButtonClick(ByVal Button As MSComctlLib.Button)

```

```

    Select Case Button.Index

```

```

    Case 1          'First

```

```

        DE1.rsVehicle.MoveFirst

```

```

    Case 2          'Previous

```

```

        'If DE1.rsVehicle.BOF Then

```

```

            If txtVID = "" Then

```

```

                MsgBox ("Already on the first record!")

```

```

                DE1.rsVehicle.MoveFirst

```

```

            Else

```

```

                DE1.rsVehicle.MovePrevious

```

```

            End If

```

```

    Case 3          'Next

```

```

        'If DE1.rsVehicle.EOF Then

```

```

            If txtVID = "" Then

```

```

                MsgBox ("Already on the last record!")

```

```

                DE1.rsVehicle.MoveLast

```

```

            Else

```

```

                DE1.rsVehicle.MoveNext

```

```

            End If

```

```

    Case 4          'Last

```

```

        DE1.rsVehicle.MoveLast

```

```

    Case 6          'New

```

```

        ad = 1

```

```

        DE1.rsVehicle.MoveLast

```

```

        nvid = DE1.rsVehicle.Fields(0).Value + 1

```

```

        DE1.rsVehicle.AddNew

```

```

        txtVID = nvid

```

```

        Dim ctla As Control

```

```

        For Each ctla In Controls

```

```

            If TypeOf ctla Is TextBox Then

```

```

                ctla.Locked = False

```

```

            End If

```

```

        Next

```

```

        TB1.Buttons(8).Enabled = True

```

```

        For btn = 1 To 4

```

```

            TB1.Buttons(btn).Enabled = False

```

```

        Next

```

```

    Case 7          'Edit

```

```

        ed = 1

```

```

        Dim ctle As Control

```

```

        For Each ctle In Controls

```

```

            If TypeOf ctle Is TextBox Then

```

```

                ctle.Locked = False

```

```

            End If

```

```

1 2 3 End If

```

```
1 2 Next
    TB1.Buttons(8).Enabled = True

    For btn = 1 To 4
        TB1.Buttons(btn).Enabled = False
    Next

Case 8      'Save
    If ad = 0 And ed = 0 Then Exit Sub

    If txtvname = "" Then
        MsgBox ("Vehicle Name field cannot be left empty!")
        txtvname.SetFocus
        Exit Sub
    ElseIf IsNumeric(txtvname) Then
        MsgBox ("Invalid data type!")
        txtvname.SetFocus
        Exit Sub
    End If

    If txtclass = "" Then
        MsgBox ("Class cannot be left empty!")
        txtclass.SetFocus
        Exit Sub
    ElseIf IsNumeric(txtclass) Then
        MsgBox ("Invalid data type!")
        txtclass.SetFocus
        Exit Sub
    End If

    If txtmake = "" Then
        MsgBox ("Make field cannot be left empty!")
        txtmake.SetFocus
        Exit Sub
    ElseIf IsNumeric(txtmake) Then
        MsgBox ("Invalid data type!")
        txtmake.SetFocus
        Exit Sub
    End If

    If txtvariant = "" Then
        MsgBox ("Variant field cannot be left empty!")
        txtvariant.SetFocus
        Exit Sub
    ElseIf IsNumeric(txtvariant) Then
        MsgBox ("Invalid data type!")
        txtvariant.SetFocus
        Exit Sub
    End If

    If txtyear = "" Then
        MsgBox ("Year field cannot be left empty!")
        txtyear.SetFocus
        Exit Sub
    ElseIf Not IsNumeric(txtyear) Then
        MsgBox ("Invalid data type!")
        txtyear.SetFocus
        Exit Sub
    End If

    If txtttype = "" Then
        MsgBox ("Transmission Type field cannot be left empty!")
        txtttype.SetFocus
        Exit Sub
    ElseIf IsNumeric(txtttype) Then
        MsgBox ("Invalid data type!")
        txtttype.SetFocus
        Exit Sub
    End If

    If txtmcode = "" Then
1 2 MsgBox ("Model Code field cannot be left empty!")
```

```

1  2      txtmcode.SetFocus
      Exit Sub
    ElseIf IsNumeric(txtmcode) Then
      MsgBox ("Invalid data type!")
      txtmcode.SetFocus
      Exit Sub
    End If

    DE1.rsVehicle.Update
    ad = 0
    ed = 0
    TB1.Buttons(8).Enabled = False

    For btn = 1 To 4
      TB1.Buttons(btn).Enabled = True
    Next

    Case 9      'Delete
      d = MsgBox("Are you sure you want to delete this record?", vbYesNo, "Delete Confirmation")
      If d = vbYes Then
        DE1.rsVehicle.Delete
        DE1.rsVehicle.MoveFirst
      End If

    Case 11     'Print
      DE1.rsVehicle.Filter = ("vid =" & txtVID)
      rptVehicle.Show
    End Select
End Sub

```

```

Private Sub tmrTime_Timer()
    lblTime.Caption = Time
End Sub

```

```

Private Sub txtclass_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
    If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Then
    Else
      MsgBox ("Invalid character entered, please re-enter data!")
      'SendKeys "{Home}+{End}"
      'SendKeys "{Delete}"
    End If
End Sub

```

```

Private Sub txtdoors_KeyPress(KeyAscii As Integer) 'Validation for numbers only.
    If KeyAscii >= 48 And KeyAscii <= 57 Or KeyAscii = 8 Then
    Else
      MsgBox ("Invalid character entered, please re-enter data!")
      'SendKeys "{Home}+{End}"
      'SendKeys "{Delete}"
    End If
End Sub

```

```

Private Sub txtmake_KeyPress(KeyAscii As Integer) 'Validation for alphabets only.
    If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii = 32 Then
    Else
      MsgBox ("Invalid character entered, please re-enter data!")
      SendKeys "{Home}+{End}"
      SendKeys "{Delete}"
    End If
End Sub

```

```

Private Sub txtvname_KeyPress(KeyAscii As Integer) 'Validation for alphabets

```

only.

```
If KeyAscii >= 65 And KeyAscii <= 90 Or KeyAscii >= 97 And KeyAscii <= 122 Or KeyAscii  
= 32 Then  
Else  
    MsgBox ("Invalid character entered, please re-enter data!")  
    SendKeys "{Home}+{End}"  
    SendKeys "{Delete}"  
End If
```

End Sub

Private Sub txtyear_KeyPress(KeyAscii As Integer) 'Validation for numbers
only.

```
If KeyAscii >= 48 And KeyAscii <= 57 Then  
Else  
    MsgBox ("Invalid character entered, please re-enter data!")  
    SendKeys "{Home}+{End}"  
    SendKeys "{Delete}"  
End If
```

End Sub

Private Sub txtyear_LostFocus() 'Validation for length of data item.

```
If Len(txtyear) > 4 Then  
    MsgBox ("Invalid data length!")  
    txtyear = ""  
    txtyear.SetFocus  
End If
```

End Sub

```
Attribute VB_Name = "frmVehicleQuery"  
Attribute VB_GlobalNameSpace = False  
Attribute VB_Creatable = False  
Attribute VB_PredeclaredId = True  
Attribute VB_Exposed = False
```

Private Sub Combo1_Click()

```
    Text1.SetFocus
```

End Sub

Private Sub Command1_Click()

```
    'For f = 0 To dgSafety.Columns.Count - 1  
    'dgSafety.Columns(f).Caption = UCase(dgSafety.Columns(f).Caption)  
    'Next  
    DE1.rsVehicle.Filter = "vname <> NULL"
```

End Sub

Private Sub Form_Load()

```
    lblDate.Caption = Format(Date, "long date")
```

End Sub

Private Sub Text1_KeyPress(KeyAscii As Integer)

```
    If KeyAscii = 13 Then  
        Select Case Combo1.Text  
            Case "Vehicle Name"  
                DE1.rsVehicle.Filter = "vname ='" & Text1.Text & "'"   
            Case "Class"  
                DE1.rsVehicle.Filter = "class ='" & Text1.Text & "'"   
            Case "Make"  
                DE1.rsVehicle.Filter = "make ='" & Text1.Text & "'"   
            Case "Variant"  
                DE1.rsVehicle.Filter = "variant ='" & Text1.Text & "'"   
            Case "Year"  
                DE1.rsVehicle.Filter = "year ='" & Text1.Text & "'"   
            Case "Transmission"  
                DE1.rsVehicle.Filter = "ttype ='" & Text1.Text & "'"   
            Case "Color"  
                DE1.rsVehicle.Filter = "color ='" & Text1.Text & "'"   
            Case "Doors"  
                DE1.rsVehicle.Filter = "doors ='" & Text1.Text & "'"   
            Case "Wheels"  
                DE1.rsVehicle.Filter = "wheels ='" & Text1.Text & "'"   
        End Select  
        Label1.Caption = "TOTAL RECORDS FOUND: " & DE1.rsVehicle.RecordCount  
    End If
```

End Sub

Private Sub tmrTime_Timer()

```
    lblTime.Caption = Time
```

End Sub

```
Attribute VB_Name = "mdiToyota"  
Attribute VB_GlobalNameSpace = False  
Attribute VB_Creatable = False  
Attribute VB_PredeclaredId = True  
Attribute VB_Exposed = False
```

```
Private Sub mDBuyer_Click()  
    frmBuyer.Show  
End Sub
```

```
Private Sub mDDimensions_Click()  
    frmDimensions.Show  
End Sub
```

```
Private Sub mDEngine_Click()  
    frmEngine.Show  
End Sub
```

```
Private Sub mDExterior_Click()  
    frmExterior.Show  
End Sub
```

```
Private Sub mDInterior_Click()  
    frmInterior.Show  
End Sub
```

```
Private Sub mDOrder_Click()  
    frmOrder.Show  
End Sub
```

```
Private Sub mDPerformance_Click()  
    frmPerformance.Show  
End Sub
```

```
Private Sub mDSafety_Click()  
    frmSafety.Show  
End Sub
```

```
Private Sub mDVehicle_Click()  
    frmVehicle.Show  
End Sub
```

```
Private Sub mEAbout_Click()  
    frmAbout.Show  
End Sub
```

```
Private Sub mEEnd_Click()  
    End  
End Sub
```

```
Private Sub mELock_Click()  
    frmSplash.Show vbModal  
End Sub
```

```
Private Sub mQBuyer_Click()  
    frmBuyerQuery.Show  
End Sub
```

```
Private Sub mQOrder_Click()  
    frmOrderQuery.Show  
End Sub
```

```
Private Sub mQVehicle_Click()
```

```
frmVehicleQuery.Show
```

```
End Sub
```

```
Private Sub mRBuyer_Click()
```

```
On Error Resume Next
```

```
Unload frmBuyer
```

```
bprint = InputBox("Please enter the Buyer ID to print the record")
```

```
If bprint = "" Then Exit Sub
```

```
DE1.rsBuyer.Open
```

```
DE1.rsBuyer.MoveFirst
```

```
DE1.rsBuyer.Find ("bid =" & bprint)
```

```
If DE1.rsBuyer.EOF Then
```

```
MsgBox ("This Buyer's record does not exist")
```

```
Else
```

```
DE1.rsBuyer.Filter = ("bid =" & bprint)
```

```
rptBuyer.Show
```

```
End If
```

```
End Sub
```

```
Private Sub mRorderbydate_Click()
```

```
DE1.rsCommand3.Open
```

```
orddate = InputBox("Enter Date to produce Orders Report" & vbCrLf & "Format  
mm/dd/yyyy")
```

```
If orddate = "" Then Exit Sub
```

```
DE1.rsCommand3.Find ("odate =#" & orddate & "#")
```

```
If DE1.rsCommand3.EOF Then
```

```
MsgBox ("This date's order doesn't exist!")
```

```
Else
```

```
DE1.rsCommand3.Filter = ("odate =#" & orddate & "#")
```

```
rptOrderList.Show
```

```
End If
```

```
End Sub
```

```
Private Sub mROrders_Click()
```

```
Unload frmOrder
```

```
oprint = InputBox("Please enter order ID to print the record")
```

```
If oprint = "" Then Exit Sub
```

```
DE1.rsCommand1.Open
```

```
DE1.rsCommand1.MoveFirst
```

```
DE1.rsCommand1.Find ("oid =" & oprint)
```

```
If DE1.rsCommand1.EOF Then
```

```
MsgBox ("This order does not exist")
```

```
Else
```

```
DE1.rsCommand1.Filter = ("oid =" & oprint)
```

```
rptOrder.Show
```

```
End If
```

```
End Sub
```

```
Private Sub mRreorderengine_Click()
```

```
rptReorderStatusEngine.Show
```

```
End Sub
```

```
Private Sub mRROexterior_Click()
```

```
rptReorderStatusExterior.Show
```

```
End Sub
```

```
Private Sub mRROinterior_Click()
```

```
rptReorderStatusInterior.Show
```

```
End Sub
```

```
Private Sub mRROperformance_Click()
```

```
rptReorderStatusPerformance.Show
```

```
End Sub
```

```
Private Sub mRROsafety_Click()
```

```
rptReorderStatusSafety.Show
```

End Sub

```
Private Sub mRVehicle_Click()
    On Error Resume Next
    Unload frmVehicle
    vprint = InputBox("Please enter the Vehicle ID to print the record")
    If vprint = "" Then Exit Sub
    DE1.rsVehicle.Open
    DE1.rsVehicle.MoveFirst
    DE1.rsVehicle.Find ("vid =" & vprint)
    If DE1.rsVehicle.EOF Then
        MsgBox ("This Vehicle's record does not exist")
    Else
        DE1.rsVehicle.Filter = ("vid =" & vprint)
        rptVehicle.Show
    End If
End Sub
```

End Sub

```
Private Sub mVDimensions_Click()
```

End Sub

```
Private Sub mWArrange_Click()
    Me.Arrange vbArrangeIcons
```

End Sub

```
Private Sub mWCascade_Click()
    Me.Arrange vbCascade
```

End Sub

```
Private Sub mWHorizontally_Click()
    Me.Arrange vbTileHorizontal
```

End Sub

```
Private Sub mWVertically_Click()
    Me.Arrange vbTileVertical
```

End Sub

IMPLEMENTATION

TOYOTA SOUTHERN MOTORS | PAKISTAN

3.1. a Method of Solution related to Problem:

Problem 1:

The main problem with the present system is that it is totally manual due to which the company faces many problems and sometimes it also gets late in submitting any kind of report or investigation. Each and every work is done manually except for printing of forms from Printing press.

Solution 1:

The above mentioned problem is solved by creating a database in MS Access and tables with primary keys and foreign keys, where the tables are related on the basis of their relevancy and requirement.

Problem 2:

Secondly it is also getting difficult for the company to manage data stored in different files. It also has tested its employees who are young, active & very responsibly carry on with their designated jobs. Also data can never be organized in any manner as there are no links between two files so single record is to be written several times to maintain separate files,

Solution 2:

This problem is solved by linking database tables to each other and the forms to upgrade any record in any table easily without repeating the process several times.

Problem 3:

The system is also becoming time consuming with new entries of vehicles. Sometimes the record manager also gets confused over the type of managing of files done by him (i.e He has to update the data of a particular model of a vehicle and instead updates the wrong one.)

Solution 3:

Several forms were made to do a specific job in a short period of time.

Problem 4:

The method of processing is also very old making it slow and full of errors and mistakes.

Solution 4:

Each form has fields with appropriate checks on them. This ensures that data is present and integrity is maintained. Also, data that is typed is free of transposition errors.

Problem 5:

There is no surcharge on late fees as no proper fees invoice is available. This adds to the misery of the company that it is unable to calculate its monthly income on time.

Solution 5:

This problem is solved by creating data report forms to allow printing of monthly reports regarding stock used, stock reordered and financial transactions.

Problem 6:

Due to the data existing only on paper, it is extremely difficult to search and find a particular record.(For example finding the wheels used in a car, and having to search through papers to find the car which has the wheels and then looking for the wheels' record itself).

Solution 6:

Each form is given a search button to search for a particular record. If multiple records having the same information need to be found, a query is used to bring up multiple records matching the criteria.

Problem 7:

The current system is insecure. Any person, or intruder or even unauthorized company employees can access data and can edit it with ease.

Solution 7:

A password screen is designed to provide different user access levels to different users.

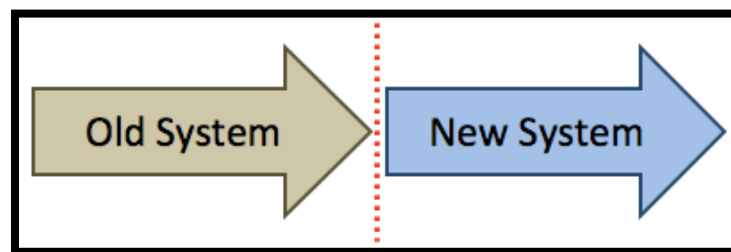
3.1.b Possible Implementation methods:

After the completion of the design stage of the program, we start the implementation stage. A changeover is a process by which the old system is replaced by the newer system. This process can be done in any of the following ways:

1. Direct changeover
2. Parallel changeover
3. Phased changeover

1. Direct Changeover:

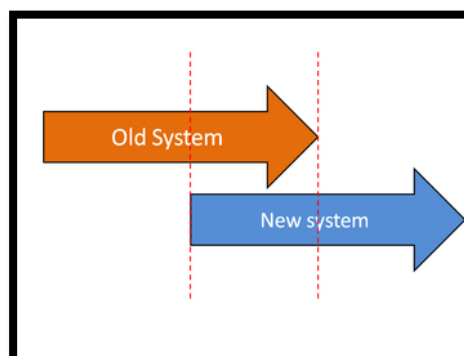
The direct changeover method involves replacing the old system with the newer system. This method for changeover is not preferred for large organizations because the risk of failure might be too high. An additional concern is the fact that the old system is completely replaced by the newer one and so if this fails, then the effects can be devastating. For a company as large as Toyota, if such a method fails, it would result in seizing of transactions and would lead to chaos.



Direct changeover

2. Parallel Changeover:

Another way to change from an existing system to a new one is known as Parallel changeover. During this changeover, a new system and old system run side by side. Both of the systems are run for a trial period. The same data is input and both systems perform the same processes. This is done to compare working and reliability of the newer system as compared to the older one. If the new system is accepted, then it will be replaced by the new one. Many firms with limited economic resources do not prefer this method because it is not cheap and is time consuming. The ability to revert to a backup is the reason why large companies prefer to use this method.

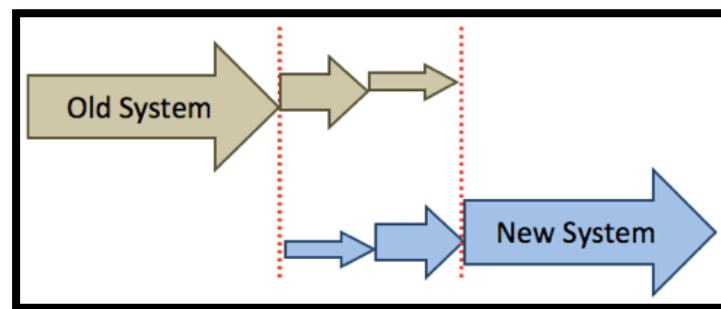


Parallel Running

3. Phased Changeover:

This method of changeover takes place in stages. If Toyota was to use this method, then first, all of the Vehicle data would be compiled into a database and linked with the software interface. After this is suc-

cessfully completed, then the sales system would be computerized. If at any point, then new system doesn't work as it should, it is very easy to revert back to the old system. This system makes sure that the company's operations do not have to shut down. For Toyota, this method is NOT feasible because it takes a lot of time to complete and get the software running.



Phased Changeover

Preferred Method:

The company prefers to use "Parallel changeover" method to transition from the manual system to a computerized system. The reason for this are:

1. The magnitude of the company requires that the system must be up and running at ALL TIMES.
2. Data backups are extremely important and to ensure that customers get the best service, this system is used. No risk can be taken when it comes to customer satisfaction!

Training of Staff:

Training of the staff is very important for a successful transition from the current manual system. For this, a "Training Period" will be set up for employees. Staff will be trained by members of the software development team. In the Pakistan office (**See 1.1.b.ii Organization Chart in Analysis.**), there are a total of 2 departments that need to be trained (excluding planning and research, General Affairs and Engineering), namely:

- Finance Department
- Parts Operations

The Finance department has around 30 employees and the Parts Operations has 45 employees. For this, 6 members from the software development team are chosen to train the employees to ensure that work is done smoothly.

The Finance Department and Parts Operations Department will have 6 members from the software development team who will train the staff for 2 months. In the first month, the staff will ensure that each and every member is familiar with software basics. They will teach them how to add, edit, and delete records as well as completing transactions. Each and every form will be explained to them along with a compiled book (User documentation). The staff will be reading the User Documentation and will request help should they not understand anything. This will be completed in 1 month after which the software development staff team will be taking short tests to assess how well the employees have learned to navigate through the various forms in the software. Hopefully, this will ensure a smooth running of the system.



Staff Training

3.1. c Method Of Solution:

Objective: Designing Login Forms to secure data. Upon providing the correct Username and Password, the user will be granted access to the information and records.

Name: Login Form

Purpose: To provide security.

This is the first form that is loaded once the executable file is executed. The User must provide his Username and Password.

A screenshot of a login window titled "TOYOTA SOUTHERN MOTORS" in red text. The window has a dark background with the Toyota logo in the center. Below the logo, there are two input fields: "User Name:" with the text "TALHA" and "Password:". At the bottom right, there are "OK" and "Cancel" buttons.

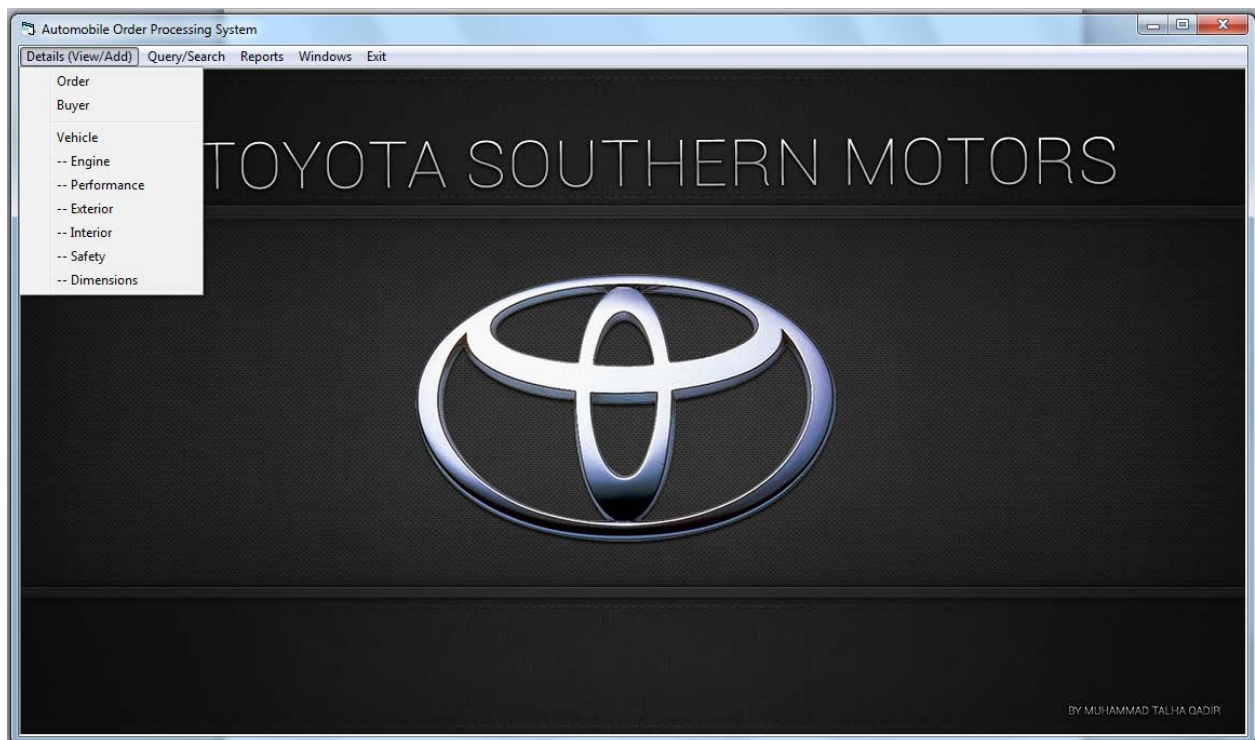
Upon entering the correct Username and Password, the user is brought to the Main menu.



Name: Main Menu

Purpose: Connectivity to all other forms, reports and queries.

Thus another objective of connectivity of forms is completed.



In order to view the Order form, the user has to click on "Details (View/Add)" and then select "Order" from the options in the drop down list. The main form for Order records will be opened as shown below.

Name: Order Details

Purpose: To maintain, move, add, delete, save and edit Order details. Thus the objective of maintaining relevant records is completed.

The screenshot shows a web application window titled "Automobile Order Processing System". The menu bar includes "Details (View/Add)", "Query/Search", "Reports", "Windows", and "Exit". The main content area is titled "Order Details" and features the Toyota logo and the text "TOYOTA SOUTHERN MOTORS". The date and time "Tuesday, February 25, 2014 11:46:07 PM" are displayed in the top right corner. Below the header, there is a navigation bar with buttons: "First", "Previous", "Next", "Last", "New", "Edit", "Save", "Delete", and "Print". A price tag "1,099,743" is visible on the right. The "ORDER DETAILS" section contains the following fields: "Order ID" (6), "Order No" (12321), "Quantity" (1), "BID" (1), "Order Date" (25-Feb-2014), "Select Vehicle" (Corolla), and "Select Features" (Engine, Exterior, Interior, Performance, Safety). The "SELECT BUYER" section includes "Buyer ID" (1) and "Title" (General Motors). The footer text reads "BY MUHAMMAD TALHA QADIR".

Then we move to the report of Order Details:


The screenshot shows the same web application window, but the "Reports" menu is open, displaying a list of options: "Single Order", "Order by Date", "Buyer", "Vehicle", and "Re-order Status". The main content area shows the Toyota logo and the text "TOYOTA SOUTHERN MOTORS". The footer text reads "BY MUHAMMAD TALHA QADIR".

The report of Order will be generated and then can be printed.

Automobile Order Processing System - [Report: Order Details]

Details (View/Add) Query/Search Reports Windows Exit

Zoom 100%

 **TOYOTA SOUTHERN MOTORS**

Order Details Report Saturday, March 01, 2014

Order ID: 6 Order No: 12321 Order Date: Friday, 28 February

SAFETY DETAILS			
SID	1	Front Seat Belt	DRIVER &
Key warning	YES	Rear Seat Belt	3 POINT ELR &
Lights Warning	YES	VID	2
Reverse Gear Warning	YES	Airbags	DRIVER +
Collision Safety			YES
Stability Control			YES
Price			123,904

PERFORMANCE DETAILS			
PID	1	Front	MACPHERSON
Transmission	6 SPEED	Stabilizer Bar	FRONT/REAR
Front Brakes	VENTILATED DISC	Steer system	ELECTRIC
Rear Brakes	VENTILATED DISC	Wheel	195/ 65 R 19
Tank capacity			76
Rear Suspension			TORSION BEAM
EID			13
Price			123,512

INTERIOR DETAILS			
IID	1	Digital Clock	YES
SID	3	Front Rest	YES
AC	YES	Gear Lever	GATE TYPE
Speedometer		MULTI INFO	
Sunvisor		DRIVE +	
Tachometer		YES	
Modesto		YES	

Pages: 1

Next to view the Buyer form, the user has to click on "Details (View/Add)" and then select "Buyer" from the options in the drop down list, as shown below:



Name: Buyer Details

Purpose: To maintain, move, add, delete, save and edit Buyer details. Thus the objective of maintaining relevant records is completed.

Upon clicking "Buyer", the following form is opened:

The screenshot shows the 'Buyer Details' form. At the top, there's a header with the Toyota logo, 'TOYOTA SOUTHERN MOTORS', and the date/time 'Wednesday, February 26, 2014 12:07:17 AM'. Below the header is a toolbar with icons for First, Previous, Next, Last, New, Edit, Save, Delete, and Print. The form is divided into four sections:

- IDENTIFICATION:** Buyer ID (1), Title (General Motors).
- ADDRESS:** City (Islamabad), Address (B-26, Sultan Mirza Road).
- CONTACT INFO:** Email (AsifK@gm.com), Contact Person (Asif Khan), Contact Number (042-30254901).
- BANK INFO:** Bank Name (Allied Bank), Bank Branch (Embassy Road Branch), Account No (9201-540-2049).

In the bottom right corner, it says 'BY MUHAMMAD TALHA QADIR'.

Then we move the report of Buyer Details.

Automobile Order Processing System - [Report: Buyer Details]

Details (View/Add) Query/Search Reports Windows Exit

Zoom 100%

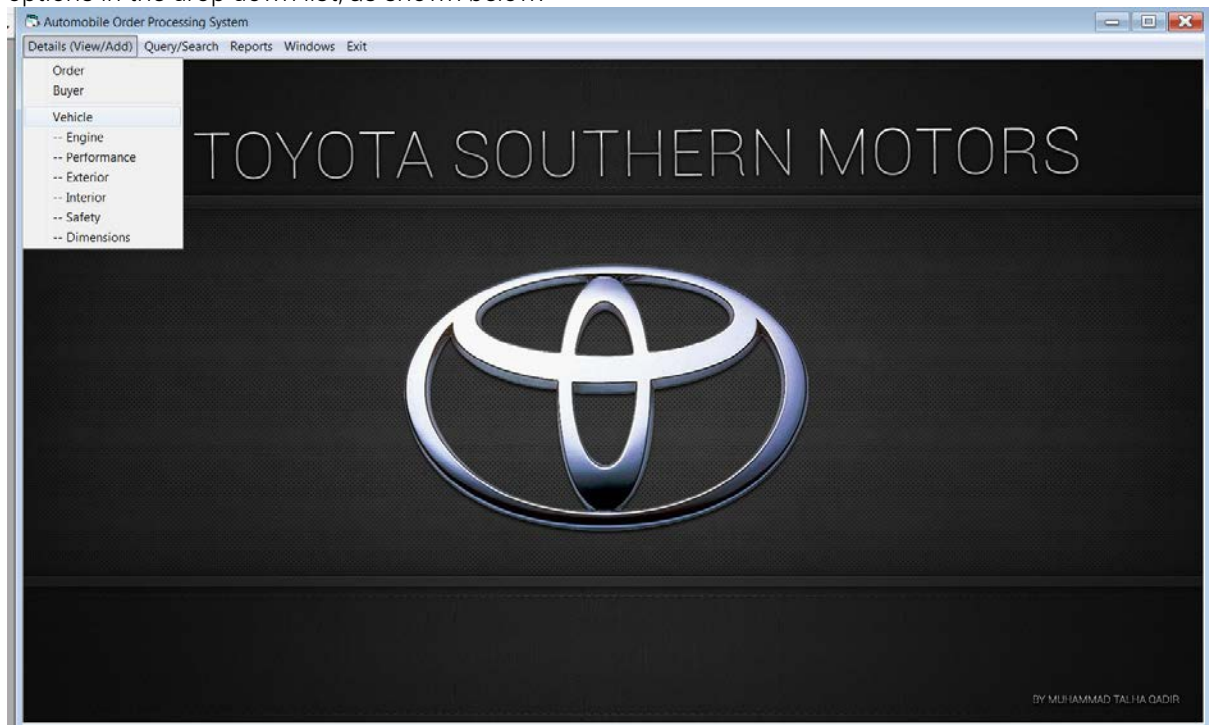
TOYOTA SOUTHERN MOTORS

Buyer Details Report Saturday, March 01, 2014

BID	4
Title	Talha Motors
City	Islamabad
Address	A-2319, Sultan Mirza Road
Contact Person	Muhammad Talha
Contact Number	042-21390410
Email	Talha@tln.com
Bank Name	Standard Chartered Bank
Bank Branch	Khan Road Branch
Account Number	2130-590-2095

Pages: 1

To view the Vehicle form, the user has to click on "Details (View/Add)" and then select "Vehicle" from the options in the drop down list, as shown below:



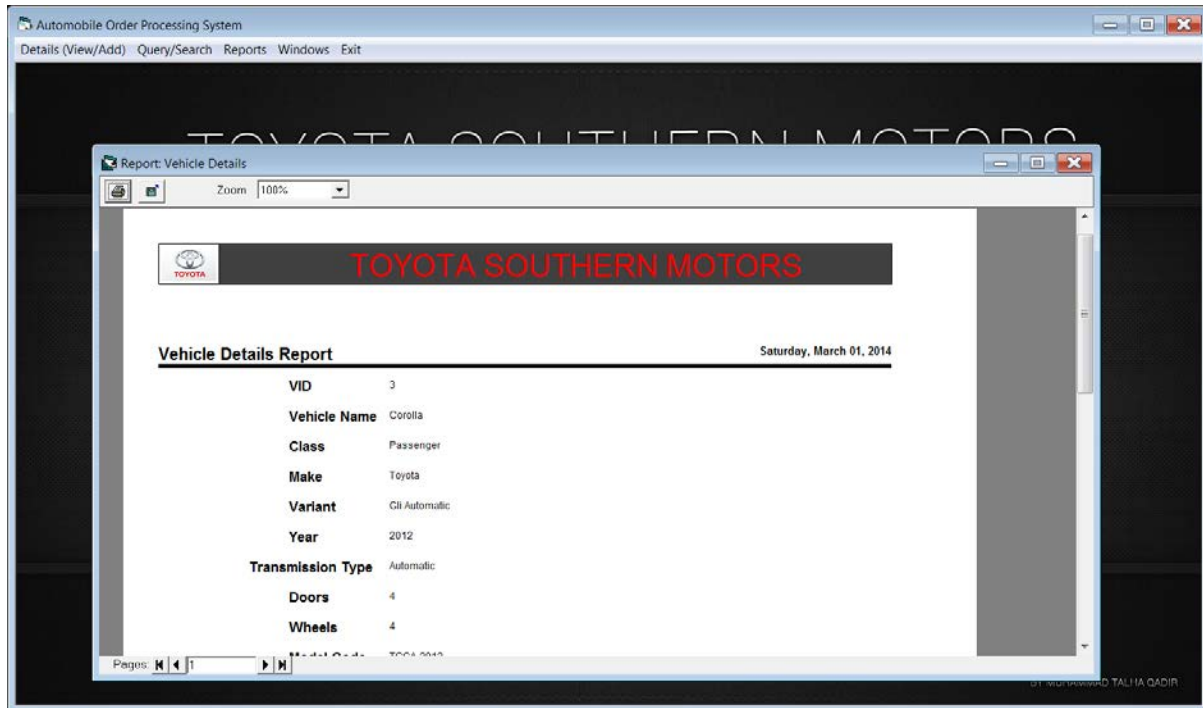
Name: Vehicle Details

Purpose: To maintain, move, add, delete, save and edit Vehicle details. Thus the objective of maintaining relevant records is completed.

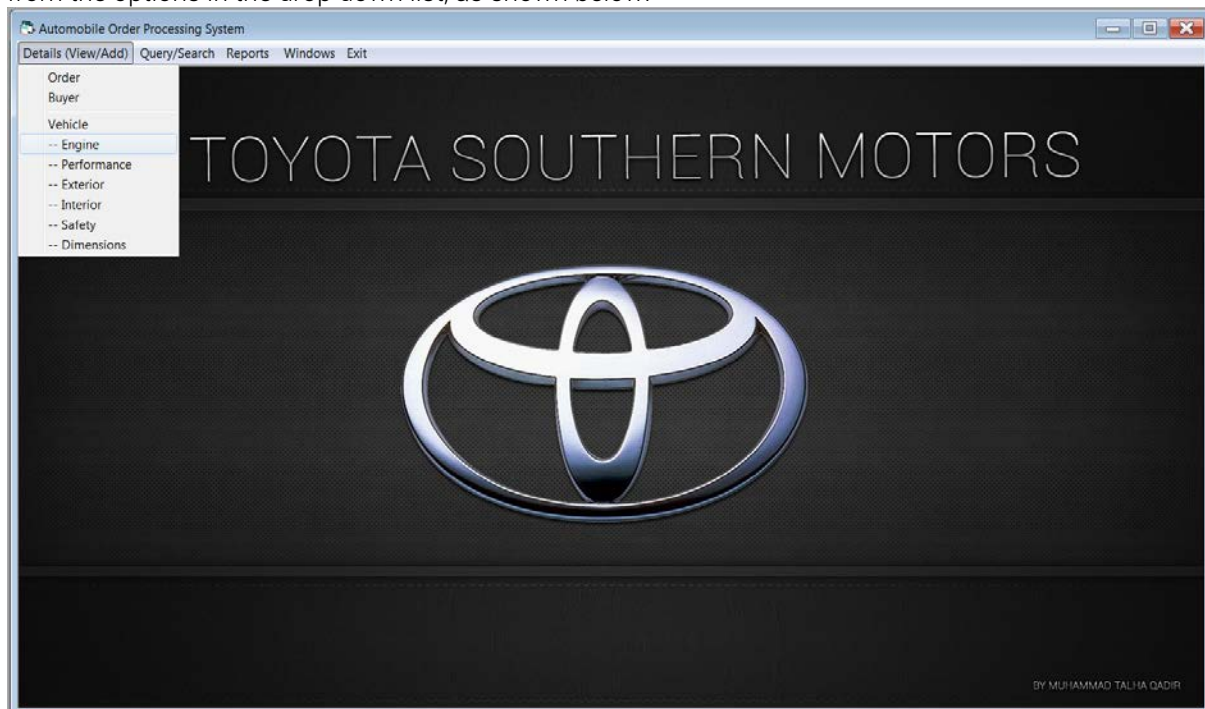
Upon clicking "Vehicle", the following form is opened:

A screenshot of the 'Vehicle Details' form within the 'Automobile Order Processing System'. The form has a title bar 'Vehicle Details' and a menu bar 'Details (View/Add) Query/Search Reports Windows Exit'. The header section includes the Toyota logo, 'TOYOTA SOUTHERN MOTORS', and the date/time 'Wednesday, February 26, 2014 9:17:33 PM'. Below the header is a toolbar with icons for 'First', 'Previous', 'Next', 'Last', 'New', 'Edit', 'Save', 'Delete', and 'Print'. The form is divided into two main sections: 'IDENTIFICATION' and 'SPECIFICATION'. The 'IDENTIFICATION' section contains fields for 'Vehicle ID' (1), 'Model Code' (TCXM-2012), 'Name' (Corolla), 'Make' (Toyota), and 'Year' (2012). The 'SPECIFICATION' section contains fields for 'Variant' (Xli), 'Transmission Type' (Manual), 'Class' (Passenger), and 'Doors' (4). A 'Command1' button is located above the 'SPECIFICATION' section. The bottom right corner has the text 'BY MUHAMMAD TALHA QADIR'.

Then we move to the report of Vehicle Details



Next to view the Engine Details form, the user has to click on "Details (View/Add)" and then select "Engine" from the options in the drop down list, as shown below:



Name: Engine Details

Purpose: To maintain, move, add, delete, save and edit Engine details. Thus the objective of maintaining relevant records is completed.

Upon clicking "Engine", the following form is opened:

Name: Re-Order Status for Engine, Interior, Safety, Performance and Exterior

Purpose: To allow easy Re-ordering of Engine, Interior, Safety, Performance and Exterior parts. Also to maintain a record of how much stock is used for each package.

Re-Order Status report for Engine. The report can be generated and printed.

Automobile Order Processing System

Details (View/Add) Query/Search Reports Windows Exit

Engine Re-order Status

Zoom 100%

TOYOTA SOUTHERN MOTORS

Engine Re-order Status Saturday, March 01, 2014

EID	TYPE	QUANTITY	REORDER LEVEL	REORDER QTY
7	2AR-FE	10	6	4
9	2KD-FTV	10	6	4
13	1-ZR-FE	12	10	2
14	2 NZ-FE	12	5	7
21	2 KD - AFT	5	4	1
25	ARJ-V08	9	7	2
26	ARI-V12	10	8	2

Total Items to be reordered: 7

Pages: 1

Re-Order Status report for Exterior. The report can be generated and printed.

Automobile Order Processing System - [Reorder Status Exterior]

Details (View/Add) Query/Search Reports Windows Exit

Zoom 100%

TOYOTA SOUTHERN MOTORS

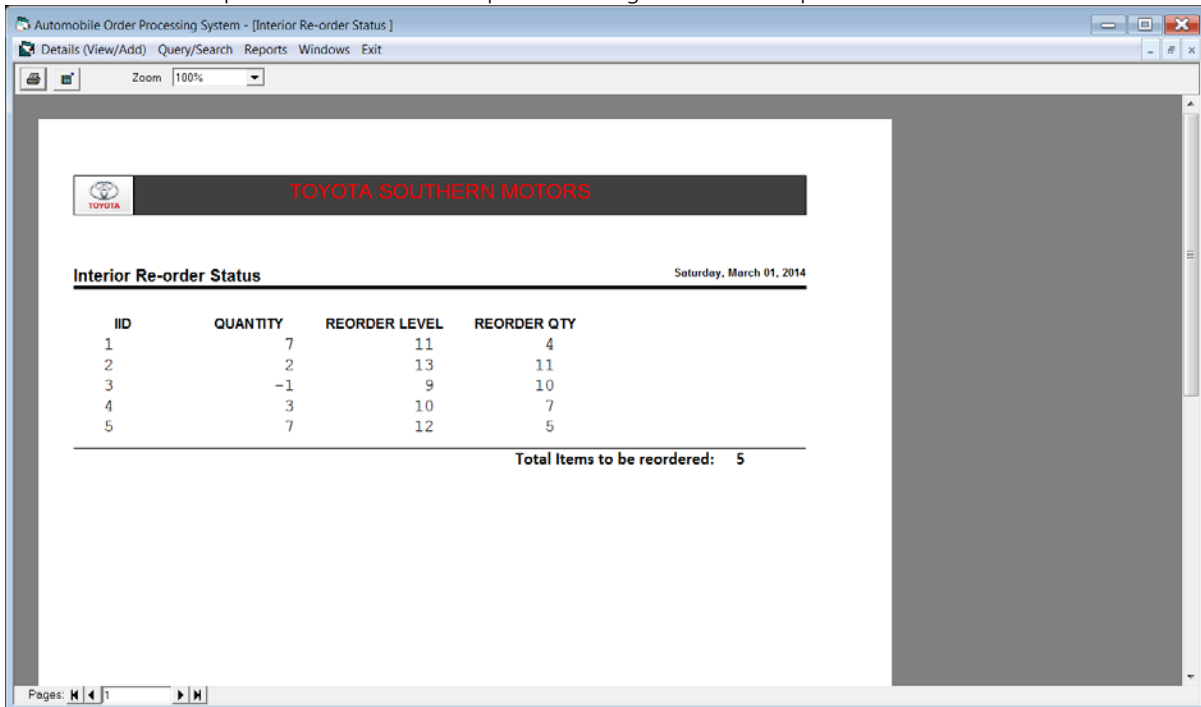
Exterior Re-order Status Saturday, March 01, 2014

extID	QUANTITY	REORDER LEVEL	REORDER QTY
1	5	10	5
2	7	10	3
3	5	15	10
4	5	12	7
5	8	14	6
6	6	9	3
7	3	12	9
8	5	15	10
10	8	9	1
11	4	7	3

Total Items to be reordered: 10

Pages: 1

Re-Order Status report for Interior. The report can be generated and printed.



Automobile Order Processing System - [Interior Re-order Status]

Details (View/Add) Query/Search Reports Windows Exit

Zoom 100%

TOYOTA SOUTHERN MOTORS

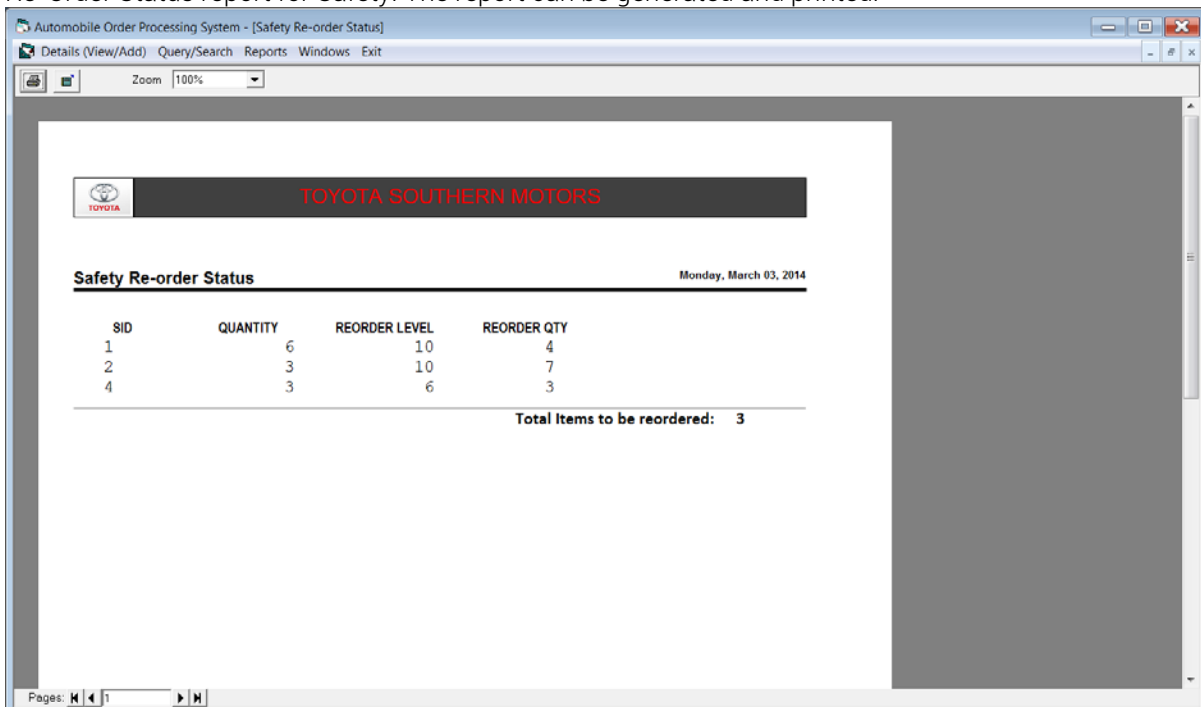
Interior Re-order Status Saturday, March 01, 2014

IID	QUANTITY	REORDER LEVEL	REORDER QTY
1	7	11	4
2	2	13	11
3	-1	9	10
4	3	10	7
5	7	12	5

Total Items to be reordered: 5

Pages: 1

Re-Order Status report for Safety. The report can be generated and printed.



Automobile Order Processing System - [Safety Re-order Status]

Details (View/Add) Query/Search Reports Windows Exit

Zoom 100%

TOYOTA SOUTHERN MOTORS

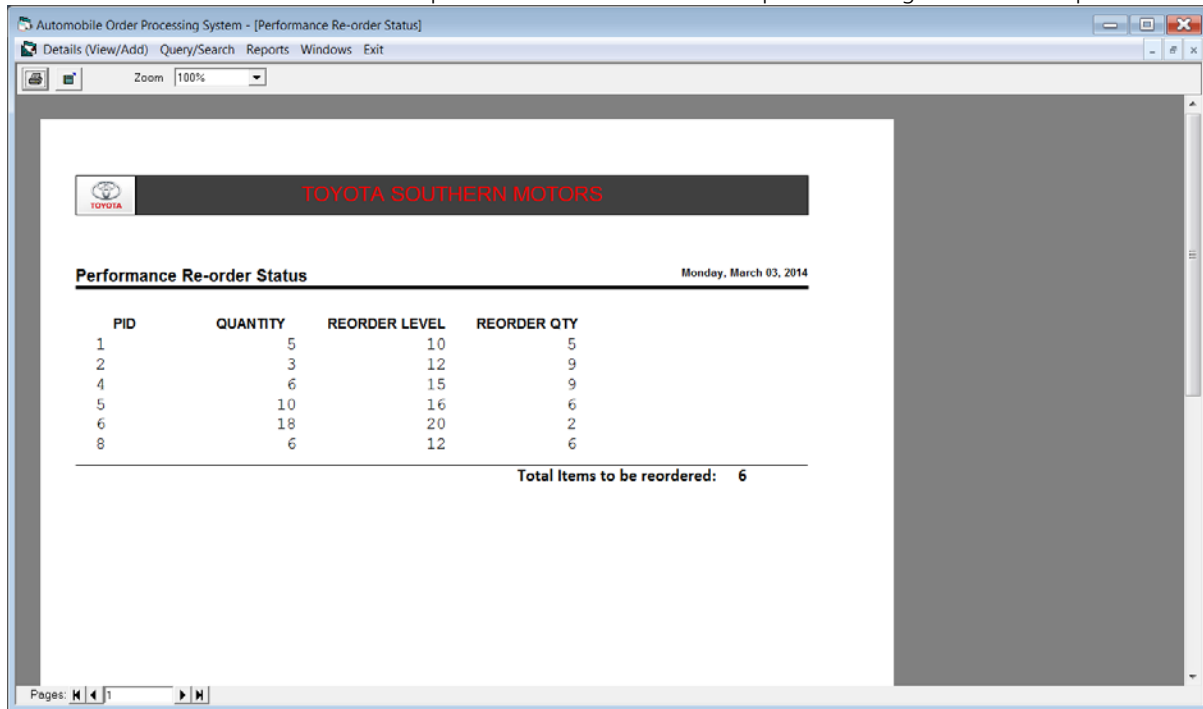
Safety Re-order Status Monday, March 03, 2014

SID	QUANTITY	REORDER LEVEL	REORDER QTY
1	6	10	4
2	3	10	7
4	3	6	3

Total Items to be reordered: 3

Pages: 1


Next we have the Re-Order status report for Performance. The report can be generated and printed.



Automobile Order Processing System - [Performance Re-order Status]

Details (View/Add) Query/Search Reports Windows Exit

Zoom 100%

 **TOYOTA SOUTHERN MOTORS**

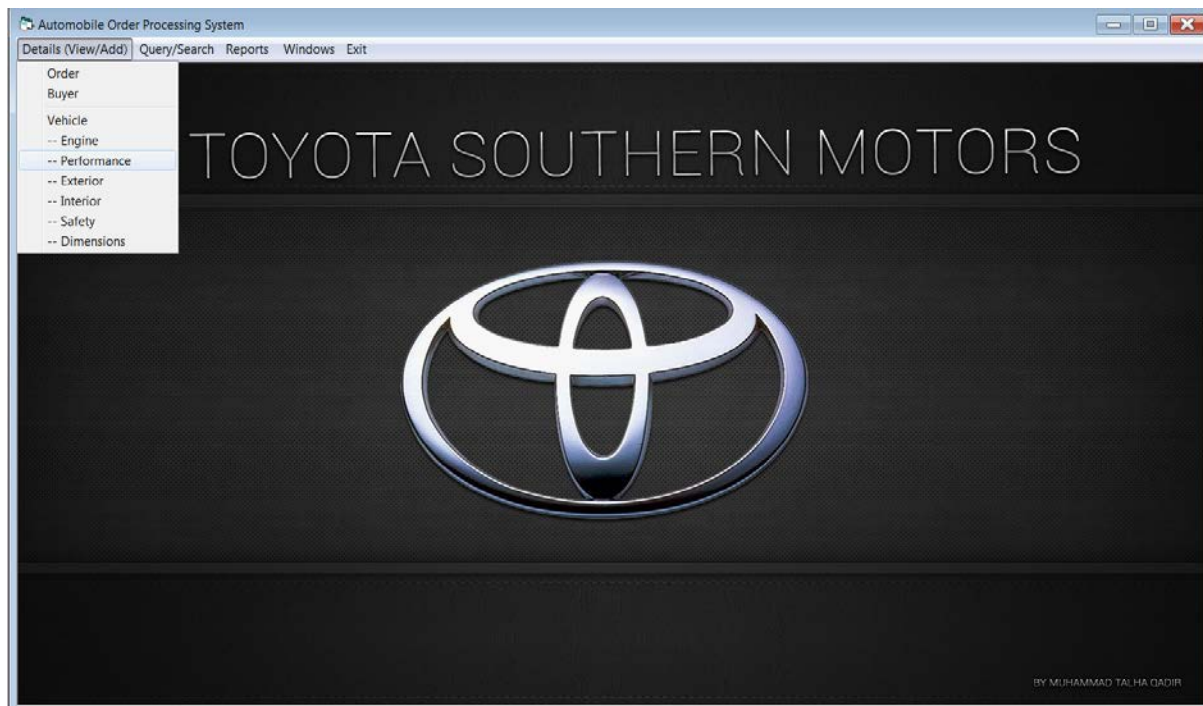
Performance Re-order Status Monday, March 03, 2014

PID	QUANTITY	REORDER LEVEL	REORDER QTY
1	5	10	5
2	3	12	9
4	6	15	9
5	10	16	6
6	18	20	2
8	6	12	6

Total Items to be reordered: 6

Pages: 1

Next to view the Performance Details form, the user has to click on "Details (View/Add)" and then select "Performance" from the options in the drop down list, as shown below:



Name: Performance Details

Purpose: To maintain, move, add, delete, save and edit Performance details. Thus the objective of maintaining relevant records is completed.

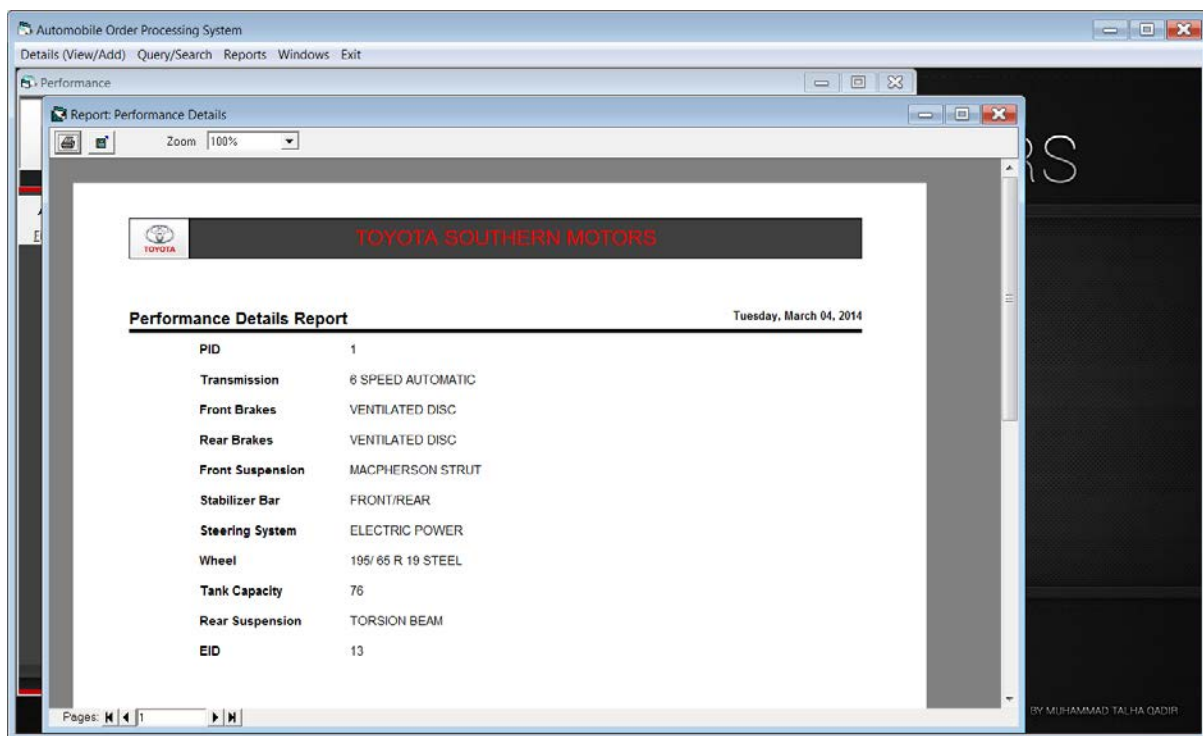
Upon clicking "Performance", the following form is opened:

A screenshot of the "Performance" form within the "Automobile Order Processing System". The form has a header with the Toyota logo, "TOYOTA SOUTHERN MOTORS", and a date/time stamp "Wednesday, February 26, 2014 10:10:58 PM". Below the header is a toolbar with icons for "First", "Previous", "Next", "Last", "New", "Edit", "Save", "Delete", and "Print". The form is divided into three main sections: "ID" with fields for "PID" (value 1) and "EID" (value 13); "INVENTORY" with fields for "Quantity in Stock" (value 6) and "Price" (value 123512); and "ESSENTIALS" which is a table of vehicle specifications.

ESSENTIALS	
Transmission	6 SPEED AUTOMATIC
Front Brakes	VENTILATED DISC
Rear Brakes	VENTILATED DISC
Front Suspension	MACPHERSON STRUT
Stabilizer Bar	FRONT/REAR
Steering System	ELECTRIC POWER
Wheel	195/ 65 R 19 STEEL
Tank Capacity	76
Rear Suspension	TORSION BEAM

The bottom right corner of the form area contains the text "BY MUHAMMAD TALHA QADIR".

Then we move onto report of Performance Details. The report can be generated and printed.



Next to view the Exterior Details form, the user has to click on "Details (View/Add)" and then select "Exterior" from the options in the drop down list, as shown below:



Name: Exterior Details

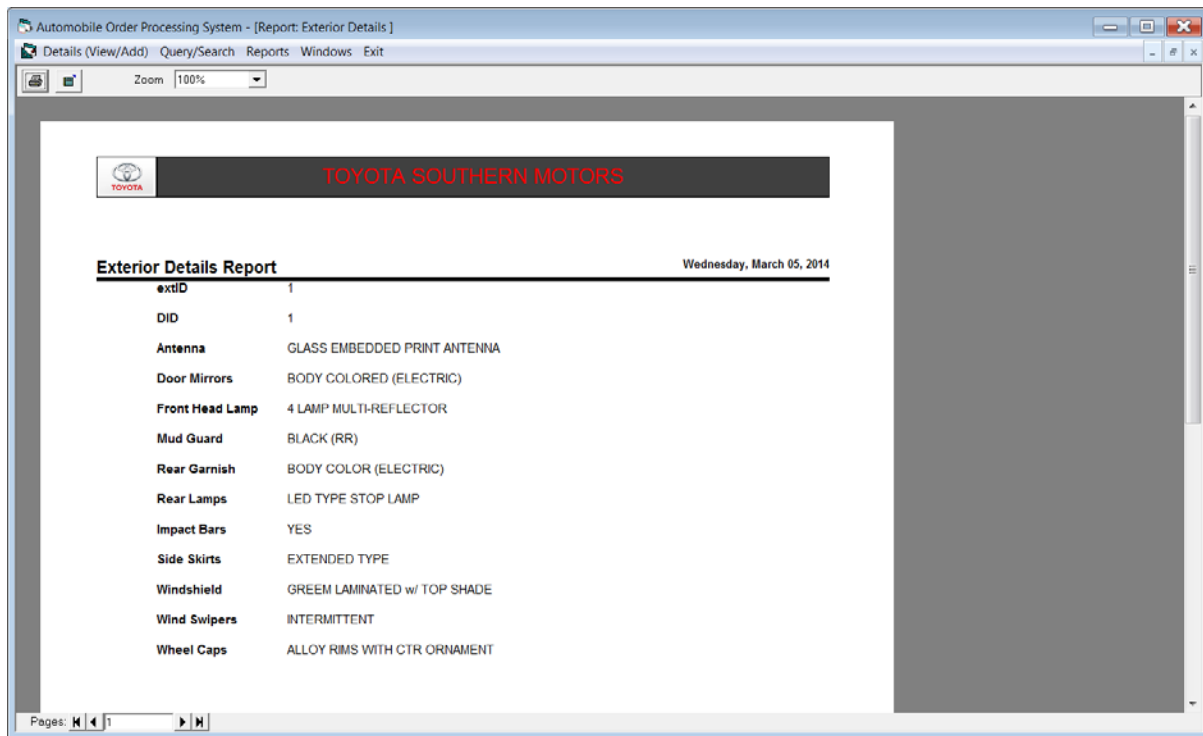
Purpose: To maintain, move, add, delete, save and edit Exterior details. Thus the objective of maintaining relevant records is completed.

Upon clicking "Exterior", the following form is opened:

ID		INVENTORY	
extID	1	Quantity in stock	5
DID	1	Price	145000

ESSENTIALS		ADDITIONAL FEATURES	
Door Mirrors	BODY COLORED (ELECTRIC)	Antenna	GLASS EMBEDDED PRINT ANTENNA
Front Head Lamp	4 LAMP MULTI-REFLECTOR	Mud Guard	BLACK (RR)
Rear Lamps	LED TYPE STOP LAMP	Rear Garnish	BODY COLOR (ELECTRIC)
Windshield	GREEN LAMINATED w/ TOP SHADE	Impact Bars	YES
		Side Skirts	EXTENDED TYPE
		Wind Swipers	INTERMITTENT
		Wheel Caps	ALLOY RIMS WITH CTR ORNAMENT
		Door Handles	CHROME

Then we move onto report of Exterior Details. The report can be generated and printed.



Next to view the Interior Details form, the user has to click on "Details (View/Add)" and then select "Interior" from the options in the drop down list, as shown below:

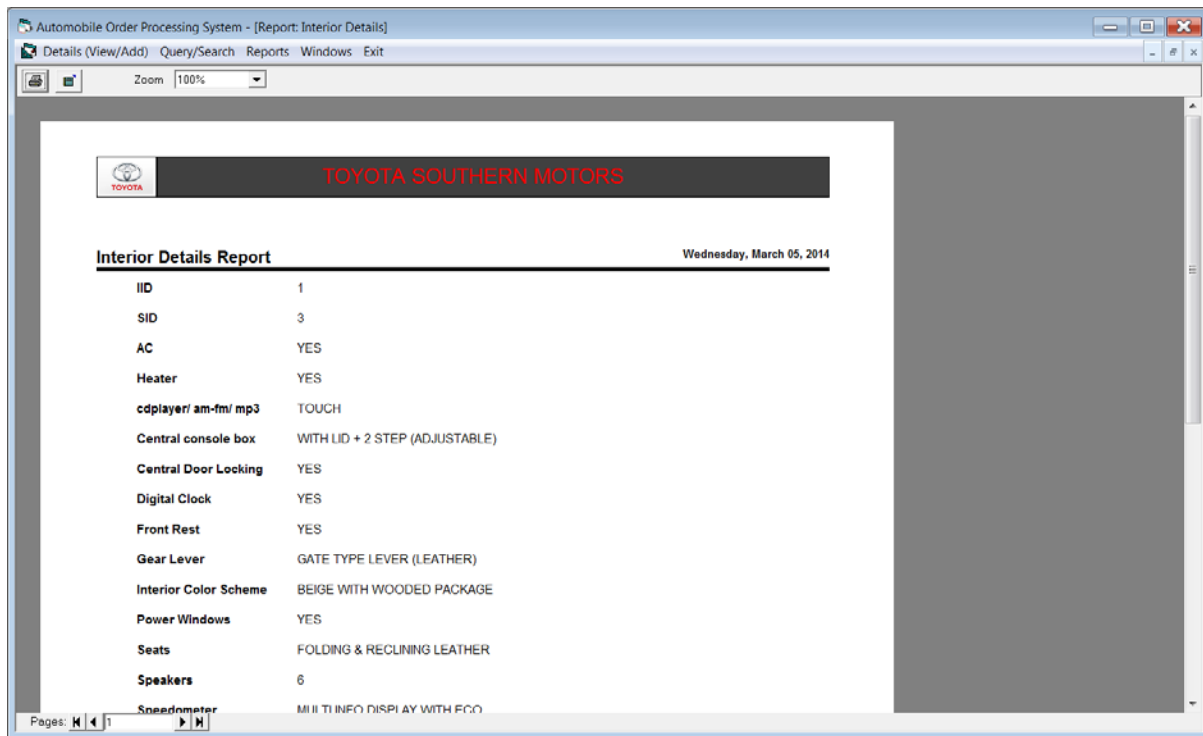


Name: Interior Details

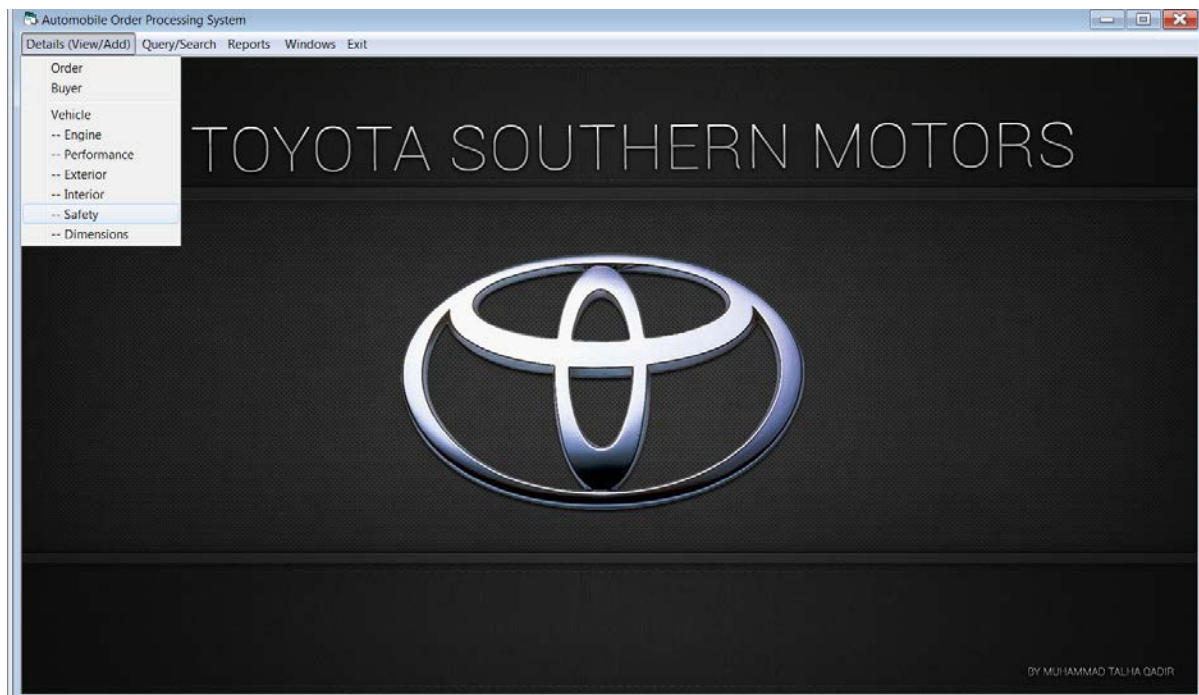
Purpose: To maintain, move, add, delete, save and edit Interior details. Thus the objective of maintaining relevant records is completed.

Upon clicking "Interior", the following form is opened:

Then we move onto report of Interior Details. The report can be generated and printed.



Next to view the Safety Details form, the user has to click on "Details (View/Add)" and then select "Safety" from the options in the drop down list, as shown below:

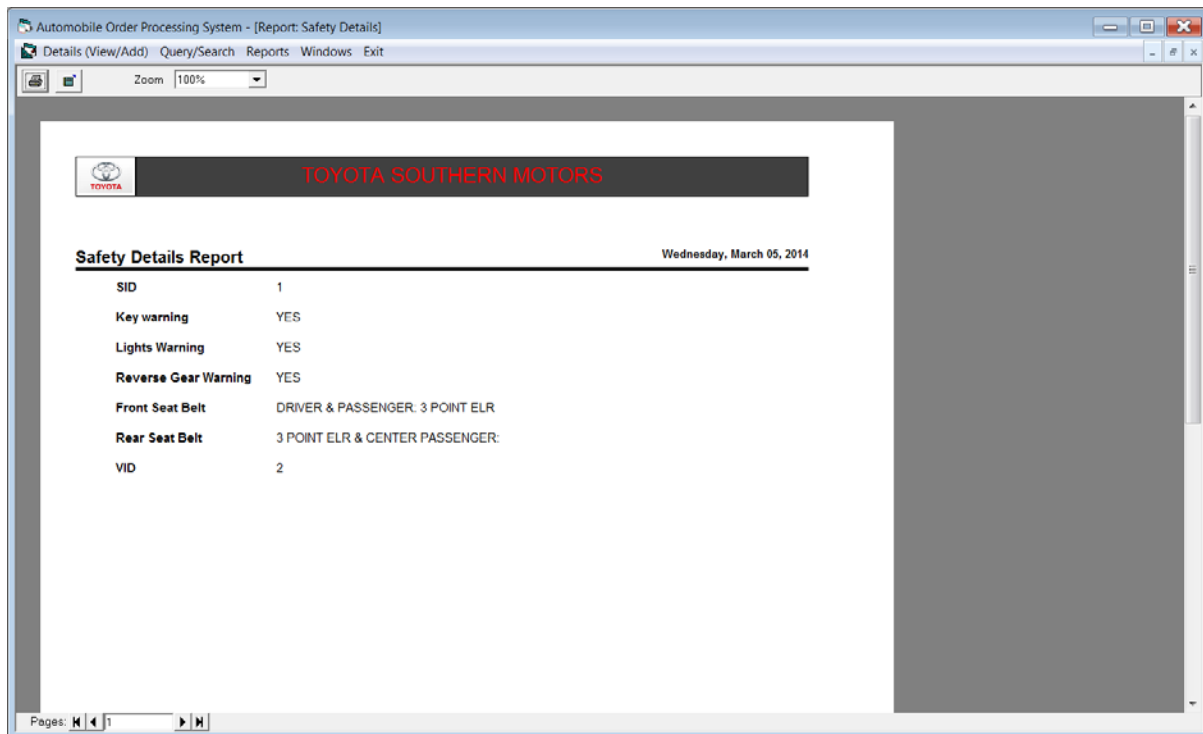


Name: Safety Details

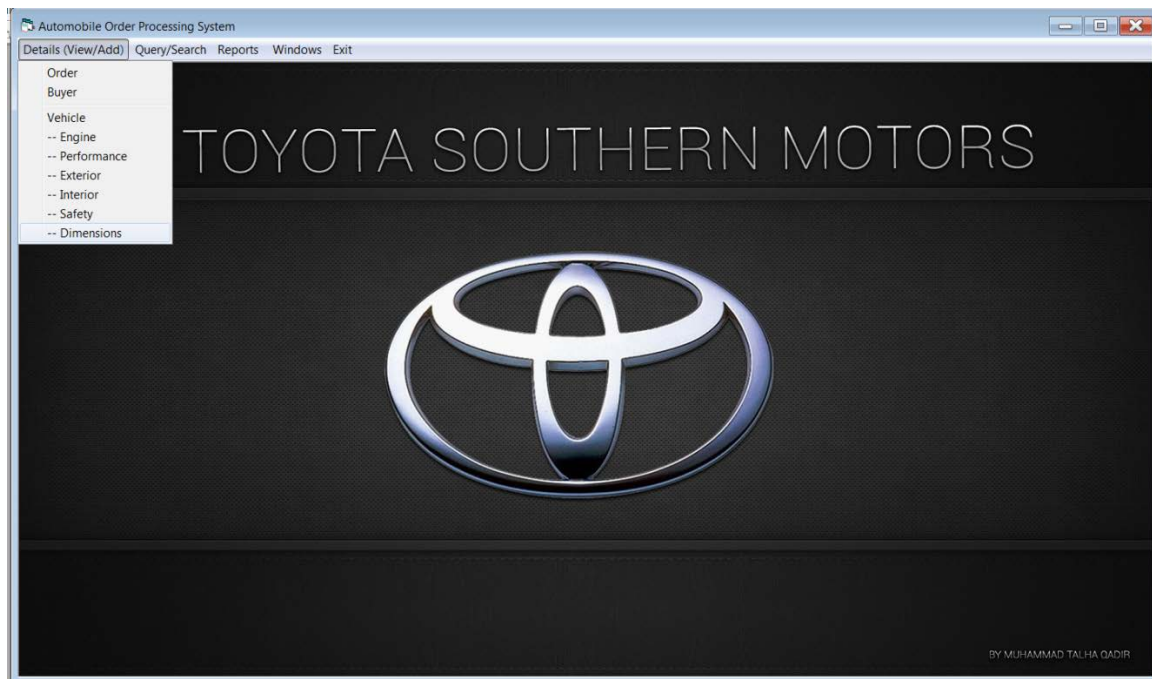
Purpose: To maintain, move, add, delete, save and edit Safety details. Thus the objective of maintaining relevant records is completed.

Upon clicking "Safety", the following form is opened:

Then we move onto report of Safety Details. The report can be generated and printed.



Lastly, to view the Dimensions Details form, the user has to click on "Details (View/Add)" and then select "Dimensions" from the options in the drop down list, as shown below:



Name: Dimensions Details

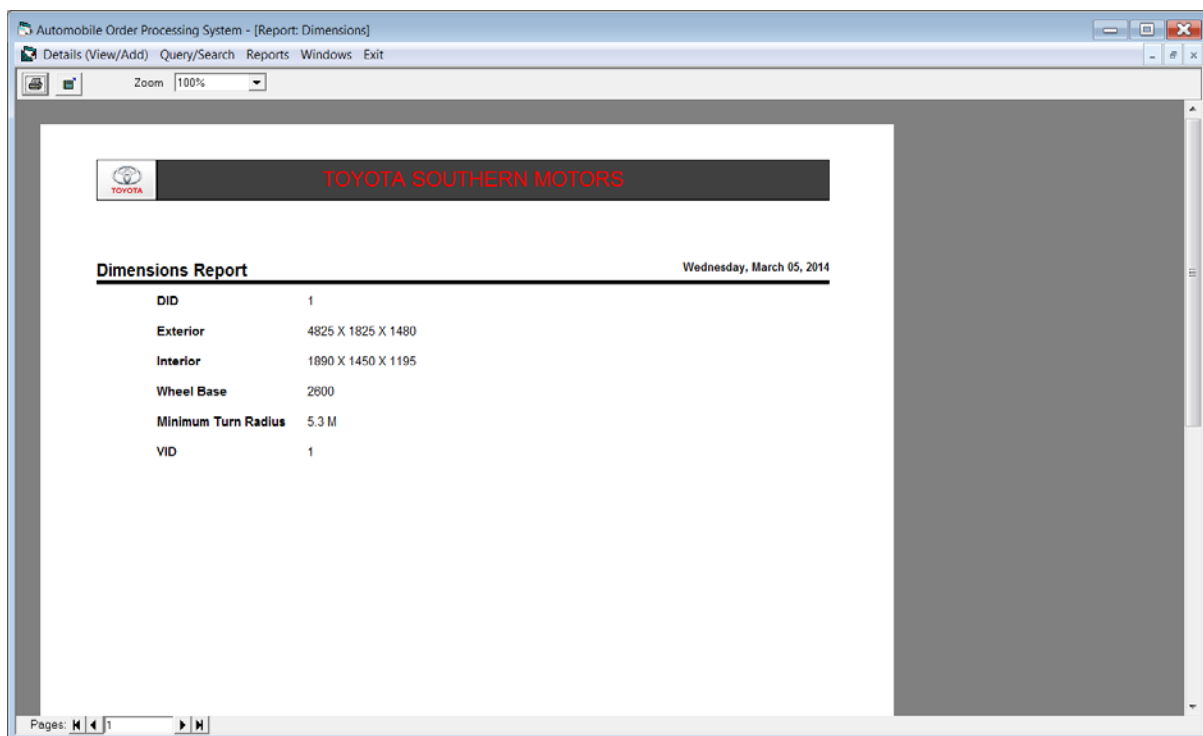
Purpose: To maintain, move, add, delete, save and edit Dimensions details. Thus the objective of maintaining relevant records is completed.

Upon clicking "Dimensions", the following form is opened:

IDENTIFICATION	
DID	1
VID	1
Exterior	4825 X 1825 X 1480
Interior	1890 X 1450 X 1195

WHEEL DETAILS	
Wheel Base	2600
Minimum	5.3 M
Turn Radius	

Then we move onto report of Dimensions Details. The report can be generated and printed.



3.2 .1 Programming Code:

3.2.2 Accurate Method of Solution

ACCURATE METHOD OF SOLUTION

Objective No. 1

Objective in General Business Term: To design a computerized system, that is visually appealing and user friendly for the user. It must also be able to manage all the records without confusing the user.

Objective in Computer Term: To design a system using Microsoft Access for creating a data base and using Microsoft Visual Basic to create forms and graphical interface to enhance the system (by creating MDI and SDI forms.)

Successfully achieved on Page Number: 86 and 87

Objective No. 2

Objective in General Business Term: The program must link several files that provide collective information of a single relevant idea in an organized manner.

Objective in Computer Term: Linking database tables to each other and to forms to upgrade any record in any table easily without repeating the process several times.

Successfully achieved on Page Number: 87 to 94, 98 to 107

Objective No. 3

Objective in General Business Term: The designed program must be fast, based upon modern techniques along with data validations and checks, in order to eliminate errors.

Objective in Computer Term: The "IF" condition should be applied so that the correct data may be entered in the correct field.

Successfully achieved on Page Number: 88, 90, 92, 94, 98, 100, 102, 104 and 106 (All forms where new data is being added.)

Objective No. 4

Objective in General Business Term: Proper data reports must be printed along with information about the buyer, what parts he buys and how much stock was used. System should also allow the ability to reorder stock as needed and view a monthly report of how much stock was given and the financial transactions that will follow. This will give a new revolutionized look to the working of company with style and ease.

Objective in Computer Term: Creating Data Report Forms to allow printing of monthly reports regarding stock used, stock reordered and how much stock has to be re-ordered.

Successfully achieved on Page Number: 88, 89, 91, 93, 95, 96, 97, 99, 101, 103, 105 and 107

Objective No. 5

Objective in General Business Term: Security is also given priority keeping in view the threats faced by the present system.

Objective in Computer Term: Designing Login Forms to secure data. Upon providing the correct Username and Password, the user will be granted access to the information and records.

Successfully achieved on Page Number: 86

Objective No. 6

Objective in General Business Term: Searching of records without having to go through manual records and wasting a lot of time.

Objective in Computer Term: Easy searching of files to find particular records.

Successfully achieved on Page Number: 107-A, 107-B, 107-C, 107-D, 107-E, 107-F and 107-G

3.2.3 Backup Policy

Everyday more than 100 cars are sold by the company, making it essential for the data to be backed up each day. After the standard working hours, which start from 9 AM to 6 PM, the data-base starts backing up and before the next day, it is completely backed up. The sheer size of the data makes it necessary for the company to use a quick backing up medium. The mediums that should be employed for backup are:

1. Cloud storage: Storage on Cloud servers will enable data to be accessed quickly if something goes wrong.
2. Hard drive: Data will also be backed up on a hard drive after every 15 days.