**Faculty of Technology – Dharmsinh Desai University**

**Knowledge System (IT 714)**

**B.Tech (IT)- SEM 7**

**Title: AUTO MART**

**KS PROJECT**

****

**Overview:**

* Our project will mainly focus on Car dealerships and other aspects of it.
* We basically consider car related data and try to create an expert system which answers all customers queries and searching aspects or other things.
* Our main focus in creating this system is to learn the fundamentals of Prolog.

**Our System would give best result for customer questions like :**

* How many cars are available for any particular brand?
* What is the average and price of any particular cars?
* Search with Best average/price/models etc..
* Bill related queries(Sample Bill).
* Many custom search results , etc….

**Topic Cover in Project:**

* Turbo Prolog features and format
* Using variables in Prolog
* Usage of rules in Prolog
* Input, Output and fail predicates in prolog
* Usage of arithmetic operators in Prolog
* Usage of cut(!),not,fail predicates in prolog
* Usage of recursion in prolog
* Usage of logical , arithmetic ,string operators in Prolog
* Compound object(Basic Implementation)
* Dynamic database(Basic Implementation)

**Used: Turbo Prolog**

**CODE:**

code=2000

domains

model,brand,fuel,variant,type=symbol

price,avg,yearofmfg,totalcar=real

items=item(brand,model,variant,totalcar)

database

dealer(brand,model,variant,fuel,avg,yearofmfg,price)

store(type,items)

predicates

start

company(model,brand)

fueltype(model,fuel)

average(model,avg)

modelvariant(model,variant)

bprice(variant,price)

varianttype(variant,type)

mfgyear(variant,yearofmfg)

rule(integer)

customrule(integer)

bill(model,variant)

preavg(integer,real)

preprice(integer,real)

usecut(yearofmfg,yearofmfg)

dynrule(integer)

comrule(integer)

continue(symbol)

dycontinue(symbol)

cucontinue(symbol)

comcontinue(symbol)

clauses

company(sonet,kia).

company(seltos,kia).

company(carnival,kia).

company(venue,hyundai).

company(verna,hyundai).

company(i20,hyundai).

company(alcazar,hyundai).

company(a4,audi).

company(q8,audi).

fueltype(sonet,petrol).

fueltype(seltos,diesel).

fueltype(carnival,diesel).

fueltype(verna,petrol).

fueltype(venue,diesel).

fueltype(i20,petrol).

fueltype(alcazar,diesel).

fueltype(a4,petrol).

fueltype(q8,petrol).

modelvariant(sonet,"hte 1.2").

modelvariant(sonet,"htk 1.2").

modelvariant(sonet,"htk plus 1.2").

modelvariant(seltos,"hte 1.5 diesel").

modelvariant(seltos,"htk 1.5 diesel").

modelvariant(seltos,"htk plus 1.5 diesel").

modelvariant(seltos,"htk plus 1.5 diesel at").

modelvariant(carnival,"premium 7 str").

modelvariant(carnival,"premium 8 str").

modelvariant(carnival,"prestige 7 str").

modelvariant(verna,"e 1.5 vtvt").

modelvariant(verna,"s plus 1.5 vtvt").

modelvariant(verna,"sx 1.5 vtvt").

modelvariant(venue,"s(o) 1.5 crdi").

modelvariant(venue,"sx 1.5 crdi").

modelvariant(i20,"magna 1.2 mt").

modelvariant(i20,"sportz 1.2 mt").

modelvariant(alcazar,"prestige 7 str 1.5 diesel").

modelvariant(alcazar,"prestige 6 str 1.5 diesel").

modelvariant(a4,"40 tfsi premium plus").

modelvariant(a4,"40 tfsi technology").

modelvariant(q8,"celebration").

modelvariant(q8,"55 tfsi quattro").

varianttype("hte 1.2","base model").

varianttype("htk 1.2","feature base model").

varianttype("htk plus 1.2","top model").

varianttype("hte 1.5 diesel","base model").

varianttype("htk 1.5 diesel","feature base model").

varianttype("htk plus 1.5 diesel","top model").

varianttype("htk plus 1.5 diesel at","fully loaded model").

varianttype("premium 7 str","base model").

varianttype("premium 8 str","feature base model").

varianttype("prestige 7 str","top model").

varianttype("e 1.5 vtvt","base model").

varianttype("s plus 1.5 vtvt","feature base model").

varianttype("sx 1.5 vtvt","top model").

varianttype("s(o) 1.5 crdi","base model").

varianttype("sx 1.5 crdi","top model").

varianttype("magna 1.2 mt","base model").

varianttype("sportz 1.2 mt","top model").

varianttype("prestige 7 str 1.5 diesel","base model").

varianttype("prestige 6 str 1.5 diesel","top model").

varianttype("40 tfsi premium plus","base model").

varianttype("40 tfsi technology","top model").

varianttype("celebration","base model").

varianttype("55 tfsi quattro","top model").

bprice("hte 1.2",687000).

bprice("htk 1.2",787000).

bprice("htk plus 1.2",873000).

bprice("hte 1.5 diesel",1055000).

bprice("htk 1.5 diesel",1199000).

bprice("htk plus 1.5 diesel",1319000).

bprice("htk plus 1.5 diesel at",1415000).

bprice("premium 7 str",2495000).

bprice("premium 8 str",2515000).

bprice("prestige 7 str",2948000).

bprice("e 1.5 vtvt",928000).

bprice("s plus 1.5 vtvt",969000).

bprice("sx 1.5 vtvt",1107000).

bprice("s(o) 1.5 crdi",952000).

bprice("sx 1.5 crdi",1000000).

bprice("magna 1.2 mt",691000).

bprice("sportz 1.2 mt",775000).

bprice("prestige 7 str 1.5 diesel",1653000).

bprice("prestige 6 str 1.5 diesel",1668000).

bprice("40 tfsi premium plus",4319000).

bprice("40 tfsi technology",4761000).

bprice("celebration",9903000).

bprice("55 tfsi quattro",13500000).

mfgyear("hte 1.2",2021).

mfgyear("htk 1.2",2018).

mfgyear("htk plus 1.2",2020).

mfgyear("hte 1.5 diesel",2018).

mfgyear("htk 1.5 diesel",2018).

mfgyear("htk plus 1.5 diesel",2021).

mfgyear("htk plus 1.5 diesel at",2020).

mfgyear("premium 7 str",2021).

mfgyear("premium 8 str",2020).

mfgyear("prestige 7 str",2016).

mfgyear("e 1.5 vtvt",2021).

mfgyear("s plus 1.5 vtvt",2019).

mfgyear("sx 1.5 vtvt",2019).

mfgyear("s(o) 1.5 crdi",2017).

mfgyear("sx 1.5 crdi",2021).

mfgyear("magna 1.2 mt",2015).

mfgyear("sportz 1.2 mt",2020).

mfgyear("prestige 7 str 1.5 diesel",2021).

mfgyear("prestige 6 str 1.5 diesel",2019).

mfgyear("40 tfsi premium plus",2021).

mfgyear("40 tfsi technology",2020).

mfgyear("celebration",2019).

mfgyear("55 tfsi quattro",2021).

average(sonet,18.4).

average(seltos,21).

average(carnival,13.9).

average(venue,23.4).

average(verna,17.7).

average(i20,20.35).

average(alcazar,20.4).

average(a4,17.42).

average(q8,9.8).

start:-

write("\*\*<-OPTIONS->\*\* \n"),

write("1. Choose and Display our custom search result\n"),

write("2. Display Baseavg of perticular car\n"),

write("3. Search Car with variants Based on company\n"),

write("4. Search Car Based on fueltype\n"),

write("5. Sample Bill\n"),

write("6. display cars excluding perticular company\n"),

write("7. Display list of cars Not mfg perticular year \n"),

write("8. Search Car Based on Average\n"),

write("9. Search Car Based on price\n"),

write("10. Search Car Based on car modelvariant type\n"),

write("11. Dynamic Database\n"),

write("12. Compound Object\n"),

write("1000. For Exit Our System\n"),

write("Enter your choice: "),

readint(X),

rule(X),

write("Heyy Do you want to continue with main menu? (y/n) \n"),

readln(Y),

continue(Y).

rule(1):-write("\*<-Welcome special direct ans menu->\*"),nl,

write("1. Display all cars with company\n"),

write("2. Display list of kia cars\n"),

write("3. Display list of hyundai cars\n"),

write("4. Display list of audi cars\n"),

write("5. Display list of petrol cars\n"),

write("6. Display list of diesel cars\n"),

write("7. Display list of best average(18&above) cars\n"),

write("8. Display list of average(18&below) cars\n"),

write("9. Display list of cars with variants in low budgets(below1000000)\n"),

write("10. Display list of cars with variants in high budgets(1000000&above)\n"),

write("11. Display list of base model cars\n"),

write("12. Display list of feature base model cars\n"),

write("13. Display list of top model cars\n"),

write("14. Display list of feature base model cars\n"),

write("15. Display list of cars mfg this year \n"),

write("16. Display list of cars mfg 2020 year \n"),

write("17. Display list of cars mfg 2019 year \n"),

write("1000. Go to main menu \n"),

write("Enter your choice: "),

readint(X),

customrule(X),

write("Heyy Do you want to continue with custom menu? (y/n) \n"),

readln(R),

cucontinue(R).

rule(2):- write("Enter car name: "),readln(Y),upper\_lower(Y,C),average(C,Z),write("Car Baseavg: ",Z),nl.

rule(3):- write("Enter Company name: "),readln(Y),upper\_lower(Y,C),company(Z,C),modelvariant(Z,V),write("Car names: ",Z," Variant: ",V),nl,fail.

rule(3).

rule(4):- write("Enter fuel Type: "),readln(Y),upper\_lower(Y,C),company(Z,B),fueltype(Z,C),write("Company name: ",B," Car name: ",Z),nl,fail.

rule(4).

rule(5):-

write("Enter Car name: "), readln(N),upper\_lower(N,M),

write("Enter Car model variant: "), readln(V),upper\_lower(V,W),

modelvariant(M,W),bill(M,W),fail.

rule(5).

rule(6):- write("Enter Company name which exclude: "),readln(V),upper\_lower(V,Y),company(Z,X),not(X=Y),write("Company:",X," Car:",Z),nl,fail.

rule(6).

rule(7):- write("Enter year which exclude"),readreal(P),mfgyear(V,Y),usecut(Y,P),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V," mfgyear:",Y),nl,fail.

rule(7).

rule(8):- write("Enter Average: "),readreal(Y),

write("Enter 1. for your enter average and above average car models\n"),

write("Enter 2. for your enter average and below average car models\n"),

write("Enter your choice: "),

readint(P),preavg(P,Y),fail.

rule(8).

rule(9):- write("Enter Price: "),readreal(Y),

write("1. for your enter price and above price car models\n"),

write("2. for your enter price and below price car models\n"),

write("Enter your choice: "),

readint(P),preprice(P,Y),fail.

rule(9).

rule(10):-write("Enter Model variant Type:"),readln(Y),upper\_lower(Y,VT),varianttype(V,VT),bprice(V,P),modelvariant(M,V),company(M,C),

write("Company: ",C," Car: ",M," Variant: ",V," Price: ",P),nl,fail.

rule(10).

rule(11):-write("\*<-Dynamic Database List->\*"),nl,

write("1. Enter new car details"),nl,

write("2. Display all the cars"),nl,

write("3. Display specific car details"),nl,

write("4. Delete a old car"),nl,

write("1000. Go to main menu \n"),

write("Enter your choice: "),readint(X),dynrule(X),

write("Heyy Do you want to continue with custom Dynamic menu? (y/n) \n"),

readln(P),

dycontinue(P).

rule(12):-write("\*<-Compound Object->\*"),nl,

write("1. enter cars entery\n"),

write("2. display cars\n"),

write("1000. Go to main menu \n"),

write("Enter your choice: "),

readint(N),comrule(N),

write("Heyy Do you want to continue with compound obj menu? (y/n) \n"),

readln(Q),

comcontinue(Q).

rule(1000).

continue(y):- start.

customrule(1):- company(X,Y),write("Company name: ",Y," Car name: ",X),nl,fail.

customrule(1).

customrule(2):- company(Y,kia),write(Y),nl,fail.

customrule(2).

customrule(3):- company(Y,hyundai),write(Y),nl,fail.

customrule(3).

customrule(4):- company(Y,audi),write(Y),nl,fail.

customrule(4).

customrule(5):- fueltype(Y,petrol),write(Y),nl,fail.

customrule(5).

customrule(6):- fueltype(Y,diesel),write(Y),nl,fail.

customrule(6).

customrule(7):- average(Y,A),A>=18,write("Car: ",Y," bavg: ",A),nl,fail.

customrule(7).

customrule(8):- average(Y,A),A<18,write("Car: ",Y," bavg: ",A),nl,fail.

customrule(8).

customrule(9):- bprice(V,P),P<1000000,modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V," Price: ",P),nl,fail.

customrule(9).

customrule(10):- bprice(V,P),P>=1000000,modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V," Price: ",P),nl,fail.

customrule(10).

customrule(11):- varianttype(V,"base model"),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

customrule(11).

customrule(12):- varianttype(V,"feature base model"),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

customrule(12).

customrule(13):- varianttype(V,"top model"),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

customrule(13).

customrule(14):- varianttype(V,"fully loaded model"),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

customrule(14).

customrule(15):- mfgyear(V,2021),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

customrule(15).

customrule(16):- mfgyear(V,2020),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

customrule(16).

customrule(17):- mfgyear(V,2019),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

customrule(17).

customrule(1000):-start.

cucontinue(y):-rule(1).

cucontinue(n):-start.

usecut(MFG,Input):-MFG=Input,!,fail.

usecut(\_,\_).

bill(M,W):-

write("----Final Sample Bill----"), nl,

write("company name: "),company(M,C),write(C), nl,

write("Car name: "),write(M), nl,

write("Car model variant: "),write(W), nl,

write("Fuel type: "), fueltype(M,F),write(F),nl,

write("Base price without GST: "),bprice(W,P),write(P),nl,

GST=(P\*0.28),

CASH=(0.01\*P),

write("GST: "), write(GST), nl,

write("CASH Discount 1%: "), write(CASH), nl,

FPCASH=P-CASH+GST,

FP=P+GST,

FPON=(P+GST-7500),

write("Final Price Without any discount: "), write(FP), nl,

write("Final Price With cash discount(1%): "), write(FPCASH), nl,

write("Final Price With OnlinePayment(Flat 7500 off) discount: "), write(FPON), nl,fail.

preavg(1,Y):-average(M,AVG),AVG>=Y,company(M,C),fueltype(M,F),write("Company: ",C," Car: ",M," fuel: ",F," Avg: ",AVG), nl,fail.

preavg(2,Y):-average(M,AVG),AVG<=Y,company(M,C),fueltype(M,F),write("Company: ",C," Car: ",M," fuel: ",F," Avg: ",AVG), nl,fail.

preprice(1,Y):-bprice(V,P),P>=Y,modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V," Price: ",P), nl,fail.

preprice(2,Y):-bprice(V,P),P<=Y,modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V," Price: ",P), nl,fail.

dynrule(1):-write("Enter Company name:"),readln(B),

write("Ente car name:"),readln(N),

write("Enter variant:"),readln(C),

write("Enter fuel:"),readln(A),

write("Enter avg:"),readreal(S),

write("Enter yearofmfg:"),readreal(M),

write("Enter price:"),readreal(P),

assertz(dealer(B,N,C,A,S,M,P)),

write("\nRecord save Successfully"),

save("cars.txt"),nl.

dynrule(2):-dealer(B,X,Y,Z,S,M,P),

write("Company:",B," Car:",X," variant:",Y," fuel:",Z," avg:",S," mfgyear:",M," price:",P),nl,fail.

dynrule(2).

dynrule(3):-write("Enter Car name:"),readln(X),dealer(B,X,Y,Z,S,M,P),

write("Company:",B," Car:",X," variant:",Y," fuel:",Z," avg:",S," mfgyear:",M," price:",P),nl.

dynrule(4):-write("Enter the car name:"),readln(P),nl,

write("enter model name:"),readln(Q),nl,

retract(dealer(\_,P,Q,\_,\_,\_,\_)),

write("\nRecord Deleted Successfully"),save("cars.txt"),nl,fail.

dynrule(4).

dynrule(1000):-start.

dycontinue(y):-rule(11).

dycontinue(n):-start.

comrule(1):-write("Enter Car type:"),readln(Type),

write("Enter Company name:"),readln(Brand),

write("Enter car name:"),readln(Model),

write("Enter variant:"),readln(Variant),

write("Enter totalcar:"),readreal(Totalcar),

assertz(store(Type,item(Brand,Model,Variant,Totalcar))),

write("addedsuccessfully"),nl.

comrule(2):-store(Type,item(Brand,Model,Variant,Totalcar)),

write("Type:",Type," Company:",Brand," Car:",Model," variant:",Variant," Totalcar:",Totalcar),nl,fail.

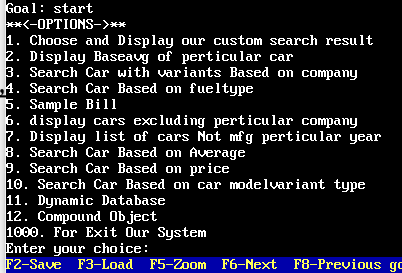
comrule(2).

comrule(1000):-start.

comcontinue(y):- rule(12).

comcontinue(n):- start.

**INPUT/ OUTPUT:**

****

***Basic Direct rule Query, Use Variable,Input output (Rule 1,2,3,4)***

write("\*\*<-OPTIONS->\*\* \n"),

write("1. **Choose and Display our custom search result**\n"),

rule(1):-write("\*<-Welcome special direct ans menu->\*"),nl,

write("1. Display all cars with company\n"),

write("2. Display list of kia cars\n"),

write("3. Display list of hyundai cars\n"),

write("4. Display list of audi cars\n"),

write("5. Display list of petrol cars\n"),

write("6. Display list of diesel cars\n"),

write("7. Display list of best average(18&above) cars\n"),

write("8. Display list of average(18&below) cars\n"),

write("9. Display list of cars with variants in low budgets(below1000000)\n"),

write("10. Display list of cars with variants in high budgets(1000000&above)\n"),

write("11. Display list of base model cars\n"),

write("12. Display list of feature base model cars\n"),

write("13. Display list of top model cars\n"),

write("14. Display list of feature base model cars\n"),

write("15. Display list of cars mfg this year \n"),

write("16. Display list of cars mfg 2020 year \n"),

write("17. Display list of cars mfg 2019 year \n"),

write("1000. Go to main menu \n"),

write("Enter your choice: "),

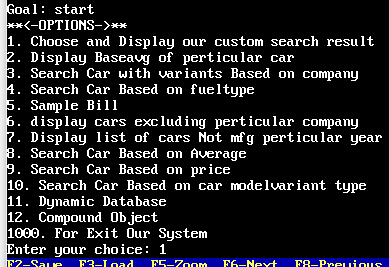
readint(X),

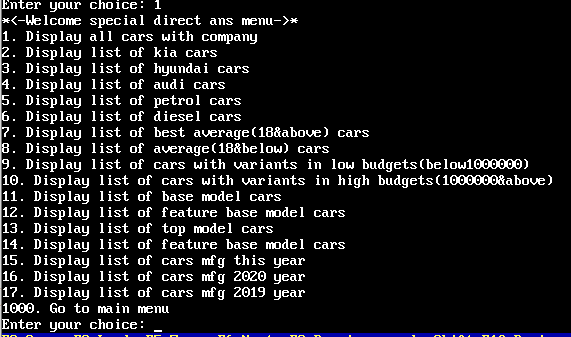
customrule(X),

write("Heyy Do you want to continue with custom menu? (y/n) \n"),

readln(R),

cucontinue(R).

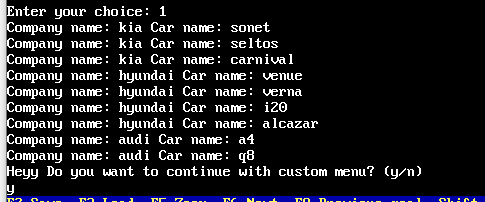




**1.1. Display all cars with company**

customrule(1):- company(X,Y),write("Company name: ",Y," Car name: ",X),nl,fail.

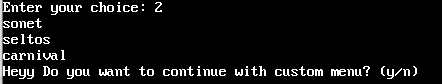
customrule(1).



**1.2. Display list of kia cars**

customrule(2):- company(Y,kia),write(Y),nl,fail.

customrule(2).



**1.3. Display list of hyundai car**

customrule(3):- company(Y,hyundai),write(Y),nl,fail.

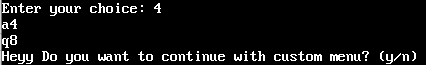
customrule(3).



**1.4. Display list of audi cars**

customrule(4):- company(Y,audi),write(Y),nl,fail.

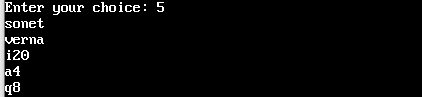
customrule(4).



**1.5. Display list of petrol cars**

customrule(5):- fueltype(Y,petrol),write(Y),nl,fail.

customrule(5).



**1.6. Display list of diesel cars**

customrule(6):- fueltype(Y,diesel),write(Y),nl,fail.

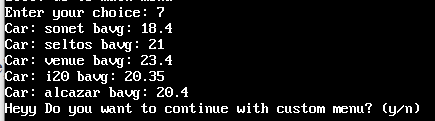
customrule(6).



**1.7. Display list of best average(18&above) cars**

customrule(7):- average(Y,A),A>=18,write("Car: ",Y," bavg: ",A),nl,fail.

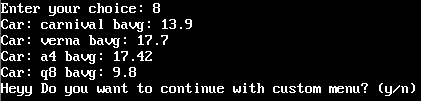
customrule(7).

****

**1.8. Display list of average(18&below) cars**

customrule(8):- average(Y,A),A<18,write("Car: ",Y," bavg: ",A),nl,fail.

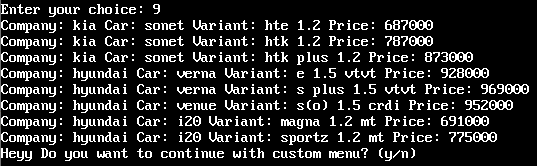
customrule(8).



**1.9. Display list of cars with variants in low budgets(below1000000)**

customrule(9):- bprice(V,P),P<1000000,modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V," Price: ",P),nl,fail.

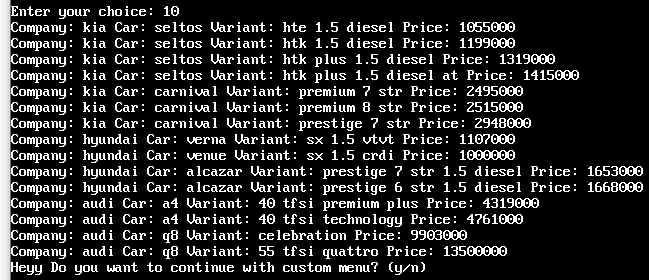
customrule(9).



**1.10. Display list of cars with variants in high budgets(1000000&above)**

customrule(10):- bprice(V,P),P>=1000000,modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V," Price: ",P),nl,fail.

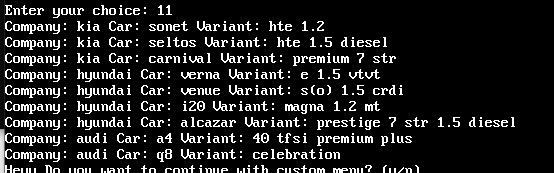
customrule(10).



**1.11. Display list of base model cars**

customrule(11):- varianttype(V,"base model"),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

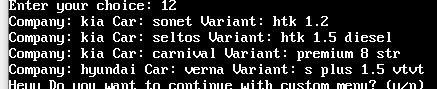
customrule(11).



**1.12. Display list of feature base model cars**

customrule(12):- varianttype(V,"feature base model"),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

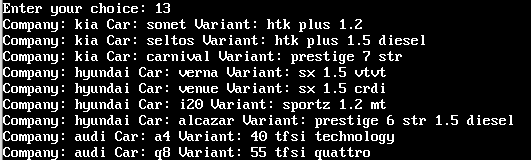
customrule(12).



**1.13. Display list of top model cars**

customrule(13):- varianttype(V,"top model"),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

customrule(13).



**1.14. Display list of feature base model cars**

customrule(14):- varianttype(V,"fully loaded model"),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

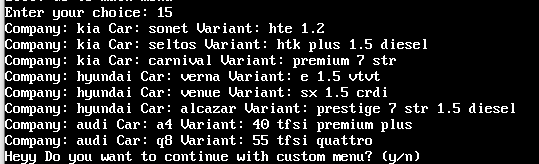
customrule(14).



**1.15. Display list of cars mfg this year**

customrule(15):- mfgyear(V,2021),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

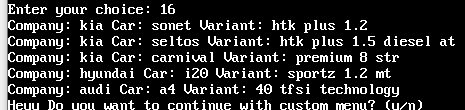
customrule(15).



**1.16. Display list of cars mfg 2020 year**

customrule(16):- mfgyear(V,2020),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

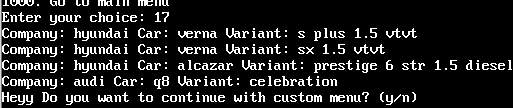
customrule(16).



**1.17. Display list of cars mfg 2019 year**

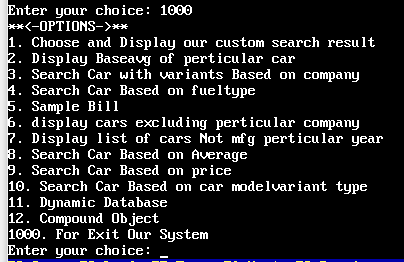
customrule(17):- mfgyear(V,2019),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V),nl,fail.

customrule(17).



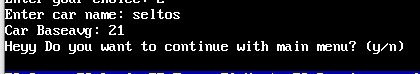
**1000. Go to main menu**

**customrule(1000):-start.**



**2. Display Baseavg of particular car**

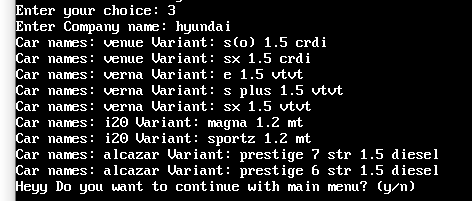
rule(2):- write("Enter car name: "),readln(Y),upper\_lower(Y,C),average(C,Z),write("Car Baseavg: ",Z),nl.



**3. Search Car with variants Based on company**

rule(3):- write("Enter Company name: "),readln(Y),upper\_lower(Y,C),company(Z,C),modelvariant(Z,V),write("Car names: ",Z," Variant: ",V),nl,fail.

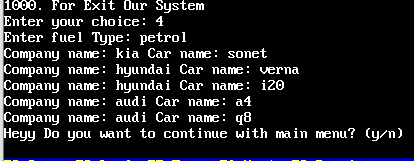
rule(3).



**4. Search Car Based on fuel type**

rule(4):- write("Enter fuel Type: "),readln(Y),upper\_lower(Y,C),company(Z,B),fueltype(Z,C),write("Company name: ",B," Car name: ",Z),nl,fail.

rule(4).



***Using arithmetic operators(Rule 5)***

***Cut,Not,Fail(Rule 6,7)***

***Rule(10)***

**5. Sample Bill**

rule(5):-

write("Enter Car name: "), readln(N),upper\_lower(N,M),

write("Enter Car model variant: "), readln(V),upper\_lower(V,W),

modelvariant(M,W),bill(M,W),fail.

rule(5).

bill(M,W):-

write("----Final Sample Bill----"), nl,

write("company name: "),company(M,C),write(C), nl,

write("Car name: "),write(M), nl,

write("Car model variant: "),write(W), nl,

write("Fuel type: "), fueltype(M,F),write(F),nl,

write("Base price without GST: "),bprice(W,P),write(P),nl,

GST=(P\*0.28),

CASH=(0.01\*P),

write("GST: "), write(GST), nl,

write("CASH Discount 1%: "), write(CASH), nl,

FPCASH=P-CASH+GST,

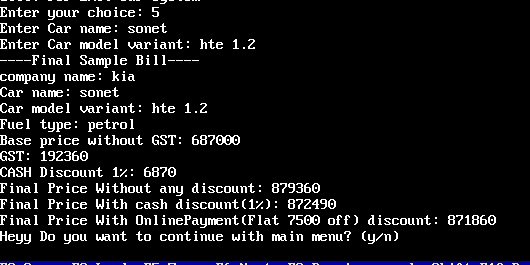
FP=P+GST,

FPON=(P+GST-7500),

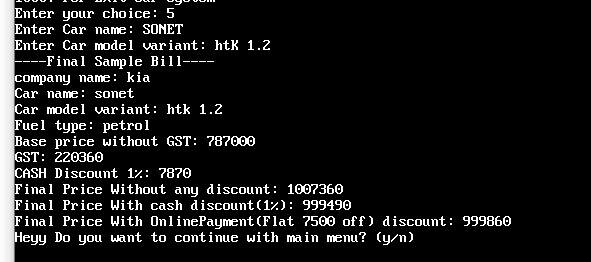
write("Final Price Without any discount: "), write(FP), nl,

write("Final Price With cash discount(1%): "), write(FPCASH), nl,

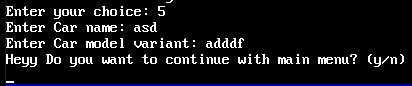
write("Final Price With OnlinePayment(Flat 7500 off) discount: "), write(FPON), nl,fail.



User Can enter Any upper lower String allow best result



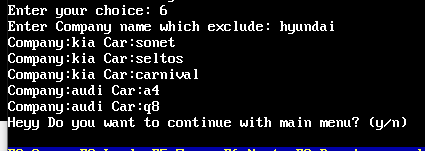
Wrong/do not match input



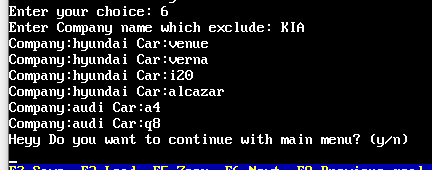
**6. display cars excluding perticular company**

rule(6):- write("Enter Company name which exclude: "),readln(V),upper\_lower(V,Y),company(Z,X),not(X=Y),write("Company:",X," Car:",Z),nl,fail.

rule(6).



User Can enter Any upper lower String allow best result



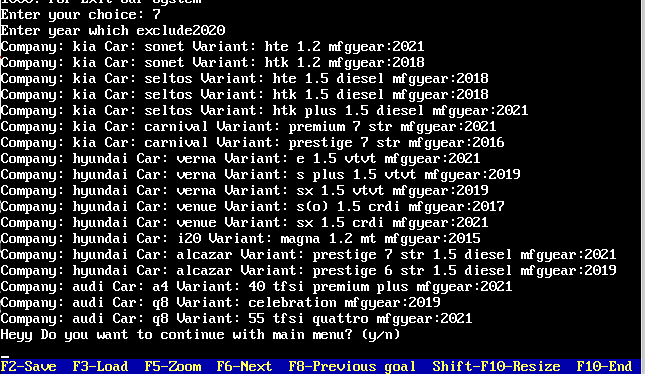
**7. Display list of cars Not mfg perticular year**

rule(7):- write("Enter year which exclude"),readreal(P),mfgyear(V,Y),usecut(Y,P),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V," mfgyear:",Y),nl,fail.

rule(7).

usecut(MFG,Input):-MFG=Input,!,fail.

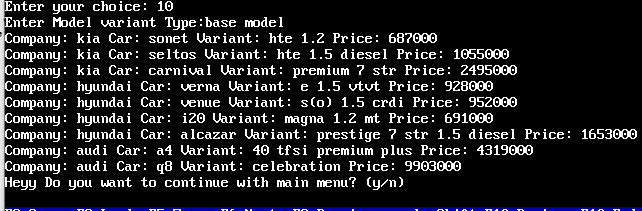
usecut(\_,\_).



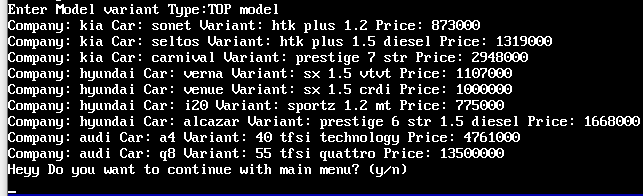
**10. Search Car Based on car modelvariant type**

rule(10):-write("Enter Model variant Type:"),readln(Y),upper\_lower(Y,VT),varianttype(V,VT),bprice(V,P),modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V," Price: ",P),nl,fail.

rule(10).



User Can enter Any upper lower String allow best result



***Logical and arithmetic operators Rule(8,9)***

***Dynamic database(Rule 11)***

***Compound Object(Rule 12)***

**8. Search Car Based on Average**

rule(8):- write("Enter Average: "),readreal(Y),

write("Enter 1. for your enter average and above average car models\n"),

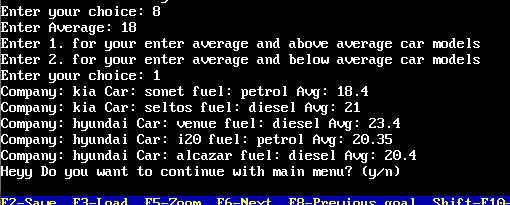
write("Enter 2. for your enter average and below average car models\n"),

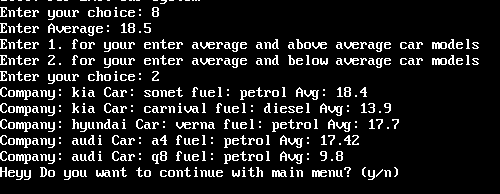
write("Enter your choice: "),readint(P),preavg(P,Y),fail.

rule(8).

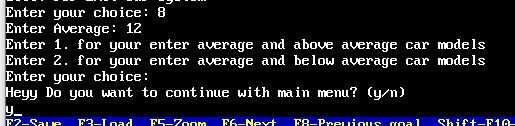
preavg(1,Y):-average(M,AVG),AVG>=Y,company(M,C),fueltype(M,F),write("Company: ",C," Car: ",M," fuel: ",F," Avg: ",AVG), nl,fail.

preavg(2,Y):-average(M,AVG),AVG<=Y,company(M,C),fueltype(M,F),write("Company: ",C," Car: ",M," fuel: ",F," Avg: ",AVG), nl,fail.





Wrong/do not match input



**9. Search Car Based on price**

rule(9):- write("Enter Price: "),readreal(Y),

write("1. for your enter price and above price car models\n"),

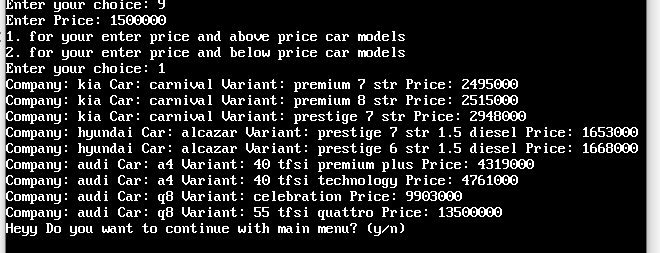
write("2. for your enter price and below price car models\n"),

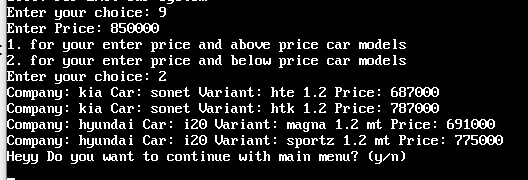
write("Enter your choice: "),readint(P),preprice(P,Y),fail.

rule(9).

preprice(1,Y):-bprice(V,P),P>=Y,modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V," Price: ",P), nl,fail.

preprice(2,Y):-bprice(V,P),P<=Y,modelvariant(M,V),company(M,C),write("Company: ",C," Car: ",M," Variant: ",V," Price: ",P), nl,fail.





**11. Dynamic Database**

rule(11):-write("\*<-Dynamic Database List->\*"),nl,

write("1. Enter new car details"),nl,

write("2. Display all the cars"),nl,

write("3. Display specific car details"),nl,

write("4. Delete a old car"),nl,

write("1000. Go to main menu \n"),

write("Enter your choice: "),readint(X),dynrule(X),

write("Heyy Do you want to continue with custom Dynamic menu? (y/n) \n"),

readln(P),

dycontinue(P).

dynrule(1):-write("Enter Company name:"),readln(B),

write("Ente car name:"),readln(N),

write("Enter variant:"),readln(C),

write("Enter fuel:"),readln(A),

write("Enter avg:"),readreal(S),

write("Enter yearofmfg:"),readreal(M),

write("Enter price:"),readreal(P),

asserta(dealer(B,N,C,A,S,M,P)),

write("\nRecord save Successfully"),

save("cars.txt"),nl.

dynrule(2):-dealer(B,X,Y,Z,S,M,P),

write("Company:",B," Car:",X," variant:",Y," fuel:",Z," avg:",S," mfgyear:",M," price:",P),nl,fail.

dynrule(2).

dynrule(3):-write("Enter Car name:"),readln(X),dealer(B,X,Y,Z,S,M,P),

write("Company:",B," Car:",X," variant:",Y," fuel:",Z," avg:",S," mfgyear:",M," price:",P),nl.

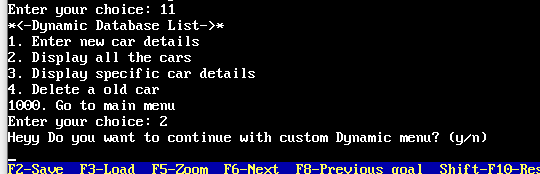
dynrule(4):-write("Enter the car to be deleted:"),readln(P),retract(dealer(\_,P,\_,\_,\_,\_,\_)),

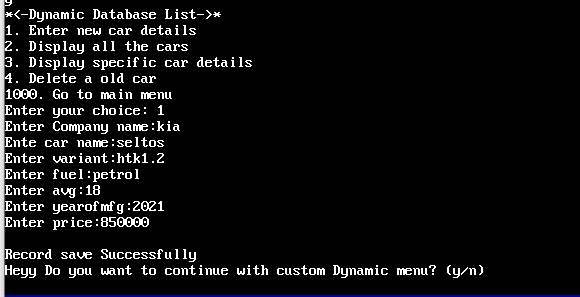
write("\nRecord Deleted Successfully"),nl.

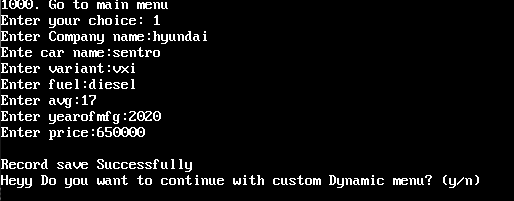
dynrule(1000):-start.

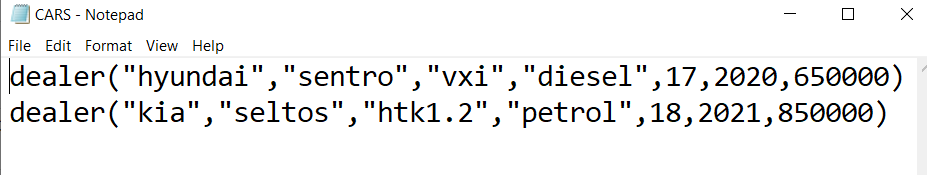
dycontinue(y):-rule(11).

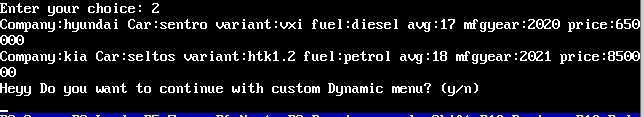
dycontinue(n):-start.

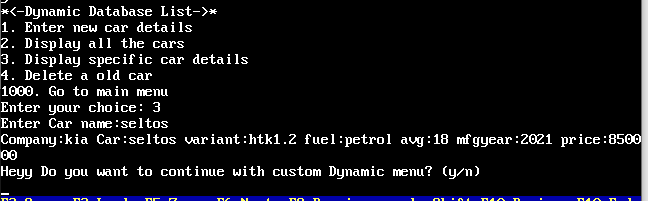


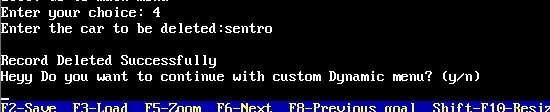


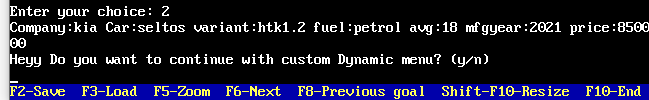












**12. Compound Object**

rule(12):-write("\*<-Compound Object->\*"),nl,

write("1. enter cars entery\n"),

write("2. display cars\n"),

write("1000. Go to main menu \n"),

write("Enter your choice: "),

readint(N),comrule(N),

write("Heyy Do you want to continue with compound obj menu? (y/n) \n"),

readln(Q),

comcontinue(Q).

comrule(1):-write("Enter Car type:"),readln(Type),

write("Enter Company name:"),readln(Brand),

write("Enter car name:"),readln(Model),

write("Enter variant:"),readln(Variant),

write("Enter totalcar:"),readreal(Totalcar),

assertz(store(Type,item(Brand,Model,Variant,Totalcar))),

write("addedsuccessfully"),nl.

comrule(2):-store(Type,item(Brand,Model,Variant,Totalcar)),

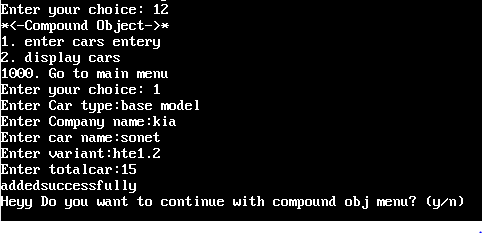
write("Type:",Type," Company:",Brand," Car:",Model," variant:",Variant," Totalcar:",Totalcar),nl,fail.

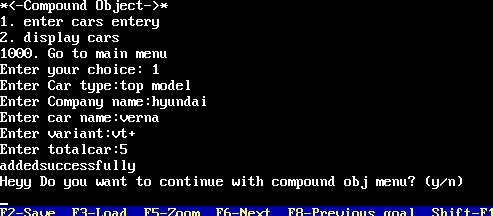
comrule(2).

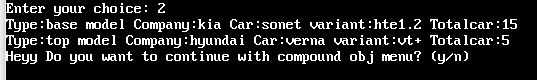
comrule(1000):-start.

comcontinue(y):- rule(12).

comcontinue(n):- start.







**1000. For Exit Our System**

rule(1000).

continue(y):- start.

