**SQL project-CAR DATASET**

**By-Phanindra Bhushan Chaturvedi & Ankit Yadav**

STEP 1. Created a database

create database project\_car\_dataset\_analysis;

use project\_car\_dataset\_analysis;

Step2.Created a view to Do Union of all 5 brands table

create view vw\_brands as

select 'audi' as brand\_name, \* from audi

union all

select 'bmw' as brand\_name, \* from bmw

union all

select 'merc' as brand\_name,\* from merc

union all

select 'cclass' as brand\_name,ID,model\_Id,year,price,mileage,'' as tax,'' as mpg,engineSize,transmission\_id,fuel\_id from cclass

union all

select 'hyndai' as brand\_name,\* from hyndai;

Now our view vw\_brands was created

Step3-

1.Create an analysis to find the income class of UK citizens based on the price of Cars

select case when price <= 10000 then 'Lower Income'

when price > 10000 and price <=20000 then 'Lower Middle Income'

when price > 20000 and price <=30000 then 'Upper Middle Income'

when price > 30000 then 'High Income'

end income\_group, count(\*) as Total\_Count

from vw\_brands

group by case when price <= 10000 then 'Lower Income'

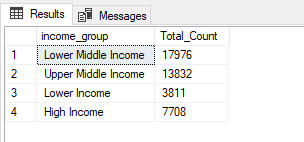
when price > 10000 and price <=20000 then 'Lower Middle Income'

when price > 20000 and price <=30000 then 'Upper Middle Income'

when price > 30000 then 'High Income'

end

output-



--#2.Categorize the cars on the basis of their price

select case when price <= 10000 then 'Mini Compact'

when price > 10000 and price <=20000 then 'Sub Compact'

when price > 20000 and price <=30000 then 'Compact'

when price > 30000 then 'Luxary'

end Type\_of\_Car, count(\*) as Total\_Count

from vw\_brands

group by case when price <= 10000 then 'Mini Compact'

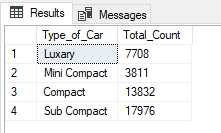
when price > 10000 and price <=20000 then 'Sub Compact'

when price > 20000 and price <=30000 then 'Compact'

when price > 30000 then 'Luxary'

end

Output—



---2.a. price changes across the years and identifies

---------the categories which have seen a significant jump in their price

select year,count(case when price <= 10000 then 1 end ) as price\_below\_10000,

count(case when price > 10000 and price <= 20000 then 1 end ) as price\_10000\_20000,

count(case when price < 20000 and price <= 30000 then 1 end ) as price\_20000\_30000,

count(case when price > 30000 then 1 end ) as price\_above\_30000

from vw\_brands

group by year

order by year

Output-



---2.b Variation in Sale of car with year

with temp as

(

select year,count(\*) as Counts from vw\_brands

group by year

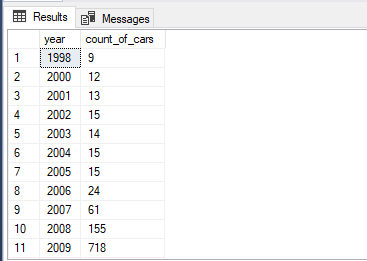
)

select year, sum(counts) over(order by year ROWS BETWEEN CURRENT ROW AND 4 FOLLOWING) as

count\_of\_cars

from temp

order by year;



3. Find relationship between fuel efficiency & price of car/sales of car/fuel type/, etc.

---considering mpg as fuel efficiency

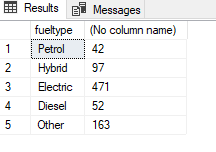
---#fuel Type Vs Efficiency

select fueltype,Round(avg(mpg),0) from vw\_brands

left join fueltype on vw\_brands.fuel\_ID=fueltype.fuel\_ID

group by fueltype

Output-



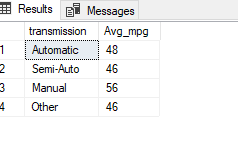
---efficiency Vs transmission

select transmission,Round(avg(mpg) ,0)as Avg\_mpg from vw\_brands

left join transmission on vw\_brands.transmission\_ID=transmission.ID

group by transmission

output-



----efficiency vs engine size

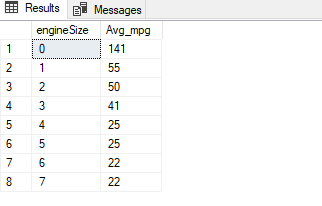
select \* from vw\_brands;

select distinct Round(engineSize,0) as engineSize,Round(avg(mpg) ,0)as Avg\_mpg from vw\_brands

group by Round(engineSize,0)

order by 1

output-



**4.Rank across all the models based on their total sales, average price, average mileage, average engine size, etc.**

---Total sales

select dense\_rank() over (order by count(\*) desc) as Rank\_Sales,a.brand\_name,b.model\_name,

avg(price) as Avg\_Price,avg(mileage) as Avg\_Mileage,avg(engineSize) as Avg\_engineSize

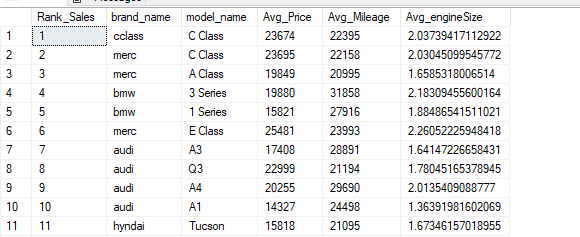
from

(select \* from vw\_brands) as a

left join models as b on a.model\_ID=b.model\_ID

group by brand\_name,model\_name

Output-



----Avg Price

select dense\_rank() over (order by avg(price) desc) as Rank\_Price,a.brand\_name,b.model\_name,

avg(price) as Avg\_Price,avg(mileage) as Avg\_Mileage,avg(engineSize) as Avg\_engineSize

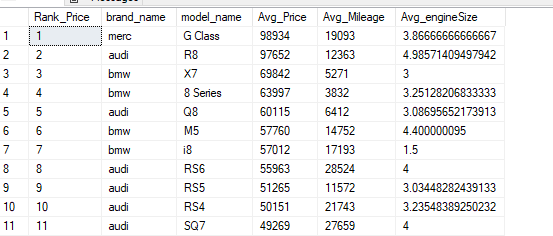
from

(select \* from vw\_brands) as a

left join models as b on a.model\_ID=b.model\_ID

group by brand\_name,model\_name

Output-



---Avg mileage

select dense\_rank() over (order by avg(mileage) desc) as Rank\_Mileage,a.brand\_name,b.model\_name,

avg(price) as Avg\_Price,avg(mileage) as Avg\_Mileage,avg(engineSize) as Avg\_engineSize

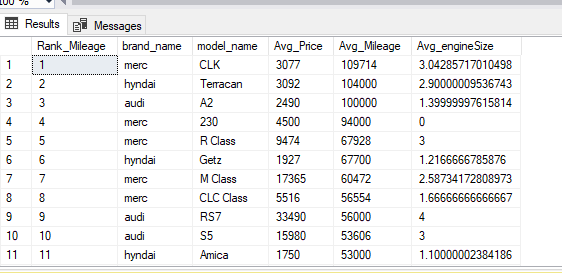
from

(select \* from vw\_brands) as a

left join models as b on a.model\_ID=b.model\_ID

group by brand\_name,model\_name

Output-



---Avg engine size

select dense\_rank() over (order by avg(engineSize) desc) as Rank\_EngineSize,a.brand\_name,b.model\_name,

avg(price) as Avg\_Price,avg(mileage) as Avg\_Mileage,avg(engineSize) as Avg\_engineSize

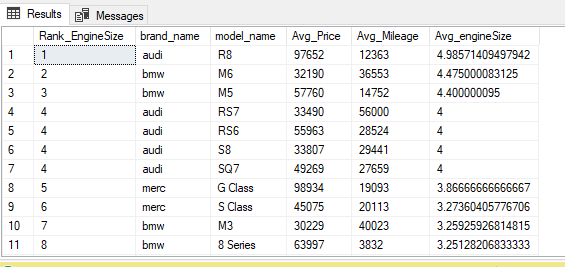
from

(select \* from vw\_brands) as a

left join models as b on a.model\_ID=b.model\_ID

group by brand\_name,model\_name;

Output-



----now filter the top 5 basis their sales.

select top 5 dense\_rank() over (order by count(\*) desc) as Rank\_Sales,a.brand\_name,b.model\_name,

avg(price) as Avg\_Price,avg(mileage) as Avg\_Mileage,avg(engineSize) as Avg\_engineSize

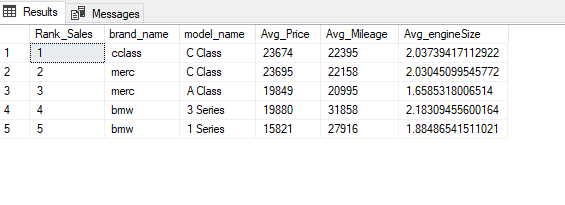
from

(select \* from vw\_brands) as a

left join models as b on a.model\_ID=b.model\_ID

group by brand\_name,model\_name;

output-

****