1. What are the top 5 most expensive vehicles in the dataset?

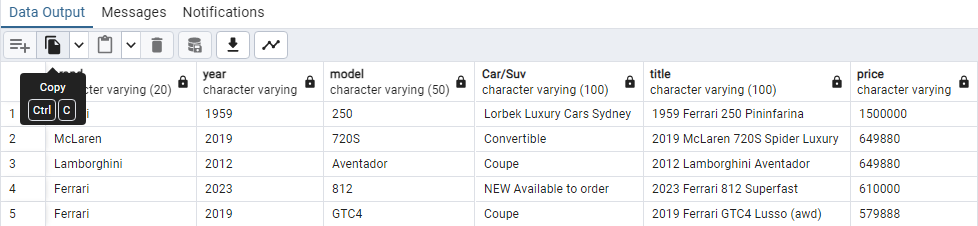
SELECT Brand, Year, Model, "Car/Suv", Title, Price

FROM australian\_vehicle\_prices

WHERE Price IS NOT NULL AND Price ~ '^[0-9]+$'

ORDER BY Price::int DESC

LIMIT 5;



1. How many vehicles are listed in each state?

SELECT

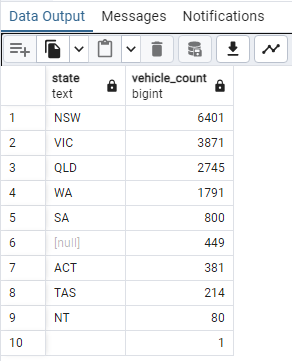
SPLIT\_PART(Location, ', ', 2) AS state,

COUNT(\*) AS vehicle\_count

FROM australian\_vehicle\_prices

GROUP BY state

ORDER BY vehicle\_count DESC;



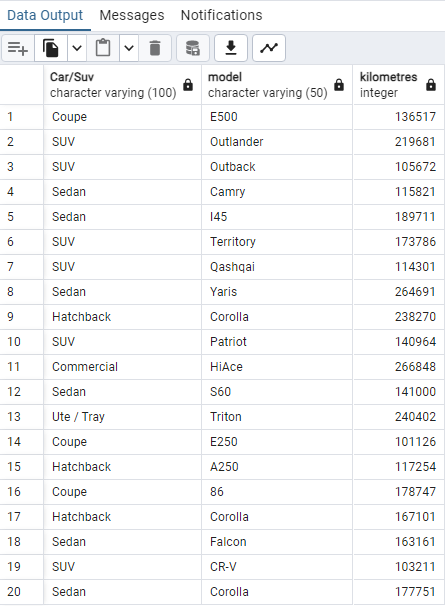
1. List all vehicles with a mileage greater than 100,000 km.

select "Car/Suv", model, kilometres

from australian\_vehicle\_prices

where kilometres >100000

limit 20

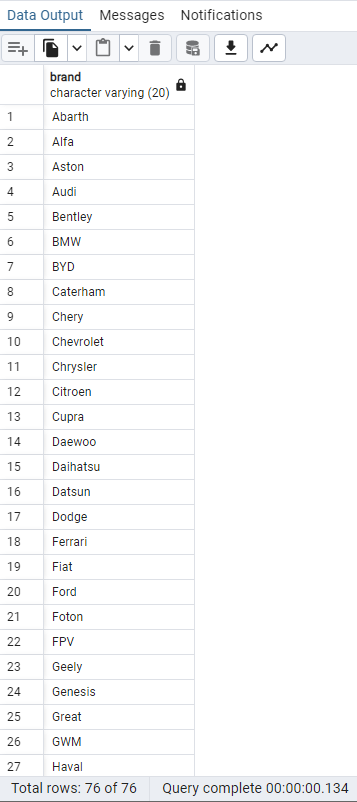


1. What are the distinct brands in the dataset?

select distinct(brand)

from australian\_vehicle\_prices

order by brand;



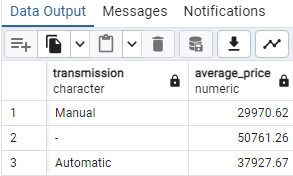
1. Retrieve the average price of vehicles based on their transmission type.

**SELECT transmission,**

**ROUND(AVG(price), 2) AS average\_price**

**FROM australian\_vehicle\_prices**

**GROUP BY transmission;**

****

**Intermediate:**

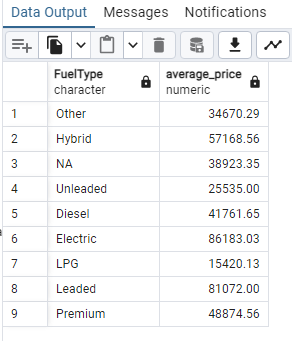
1. Find the average price of vehicles for each fuel type.

SELECT "FuelType",

ROUND(AVG(price), 2) AS average\_price

FROM australian\_vehicle\_prices

GROUP BY "FuelType";



1. List the top 10 most common vehicle models.

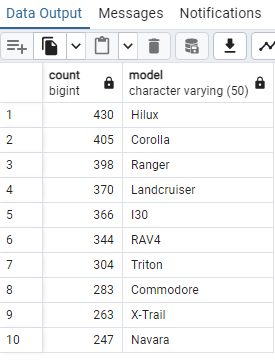
select count(\*), model

FROM australian\_vehicle\_prices

group by model

order by count(\*) desc

limit 10;



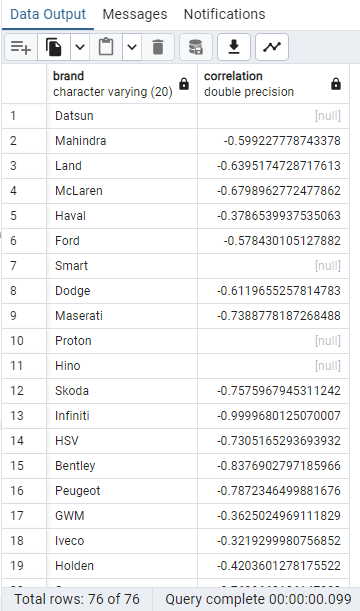
1. What is the correlation between mileage and price for each brand?

SELECT brand,

CORR(CAST(Kilometres AS numeric), CAST(Price AS numeric)) AS correlation

FROM australian\_vehicle\_prices

GROUP BY brand;



1. Show the price distribution for vehicles manufactured in each year.

SELECT Year,

MIN(price) AS min\_price,

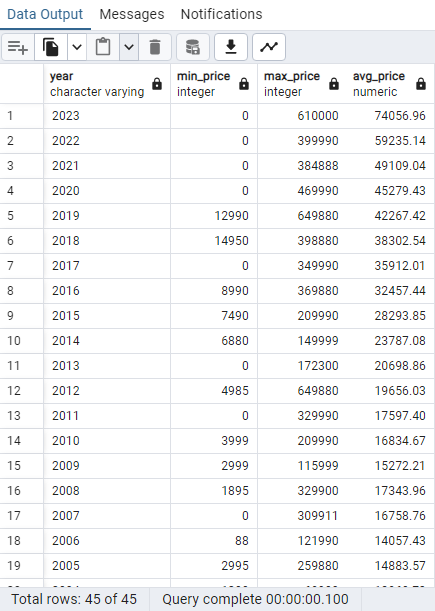
MAX(price) AS max\_price,

ROUND(AVG(price), 2) AS avg\_price

FROM australian\_vehicle\_prices

GROUP BY Year

ORDER BY Year desc;



1. Count the number of vehicles by year and state, and list the top 3 years with the highest counts.

with VehicleCounts AS (

select year,

state,

count(\*) as vehicle\_count

from australian\_vehicle\_prices

group by year, state

)

SELECT Year,

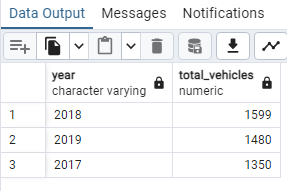
SUM(vehicle\_count) AS total\_vehicles

FROM VehicleCounts

GROUP BY Year

ORDER BY total\_vehicles DESC

LIMIT 3;



**Advanced:**

1. Create a subquery to find the average price of vehicles for each brand and compare it to the overall average price.

select brand,

average\_price,

overall\_avg\_price,

(average\_price - overall\_avg\_price) AS price\_difference

from (

select brand,

round(avg(price),2) as average\_price

from australian\_vehicle\_prices

group by brand

) as brandaverage,

(

select round(avg(price),2) as overall\_avg\_price

FROM australian\_vehicle\_prices

) AS OverallAverage;

OR

WITH BrandAverage AS (

SELECT Make AS brand,

ROUND(AVG(price), 2) AS average\_price

FROM australian\_vehicle\_prices

GROUP BY Make

),

OverallAverage AS (

SELECT ROUND(AVG(price), 2) AS overall\_avg\_price

FROM australian\_vehicle\_prices

)

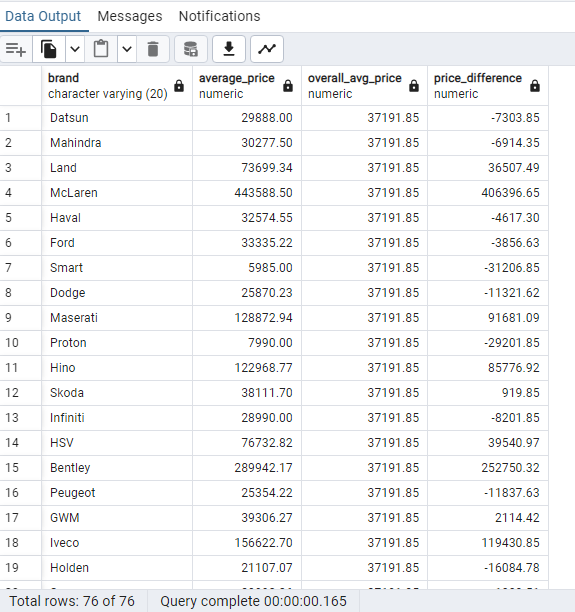
SELECT BrandAverage.brand,

BrandAverage.average\_price,

OverallAverage.overall\_avg\_price,

(BrandAverage.average\_price - OverallAverage.overall\_avg\_price) AS price\_difference

FROM BrandAverage, OverallAverage;



1. Identify any outliers in vehicle prices using a statistical method (e.g., Z-score).

WITH PriceStats AS (

SELECT ROUND(AVG(price), 2) AS mean\_price,

ROUND(STDDEV(price), 2) AS stddev\_price

FROM australian\_vehicle\_prices

),

PriceWithZScore AS (

SELECT \*,

(price - PriceStats.mean\_price) / PriceStats.stddev\_price AS z\_score

FROM australian\_vehicle\_prices

CROSS JOIN PriceStats

)

SELECT PriceStats.mean\_price,

PriceStats.stddev\_price,

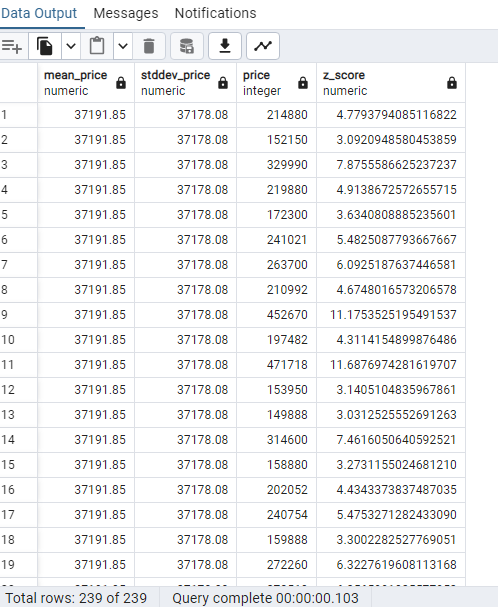
PriceWithZScore.price,

PriceWithZScore.z\_score

FROM PriceWithZScore

CROSS JOIN PriceStats

WHERE ABS(PriceWithZScore.z\_score) > 3;



1. Write a query to calculate the percentage change in vehicle prices year-over-year.

WITH YearlyAverage AS (

SELECT Year,

ROUND(AVG(price), 2) AS avg\_price

FROM australian\_vehicle\_prices

GROUP BY Year

),

PriceChange AS (

SELECT Year,

avg\_price,

LAG(avg\_price) OVER (ORDER BY Year) AS previous\_avg\_price

FROM YearlyAverage

)

SELECT Year AS current\_year,

avg\_price AS current\_avg\_price,

previous\_avg\_price,

ROUND(

CASE

WHEN previous\_avg\_price IS NOT NULL THEN

((avg\_price - previous\_avg\_price) / previous\_avg\_price) \* 100

ELSE

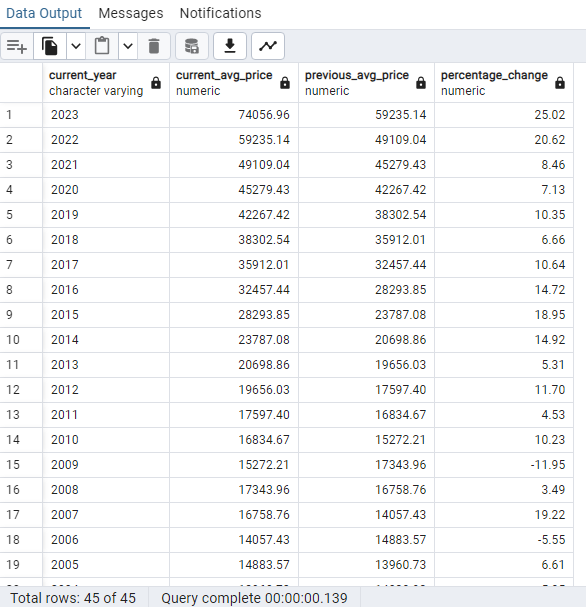
NULL

END, 2

) AS percentage\_change

FROM PriceChange

ORDER BY Year;



1. Use window functions to determine the rank of each vehicle model by price within its fuel type.

WITH RankedVehicles AS (

SELECT Model,

"FuelType",

price,

ROW\_NUMBER() OVER (PARTITION BY "FuelType" ORDER BY price DESC) AS rank

FROM australian\_vehicle\_prices

)

SELECT Model,

"FuelType",

price,

rank

FROM RankedVehicles

WHERE rank <= 10

ORDER BY "FuelType", rank;

