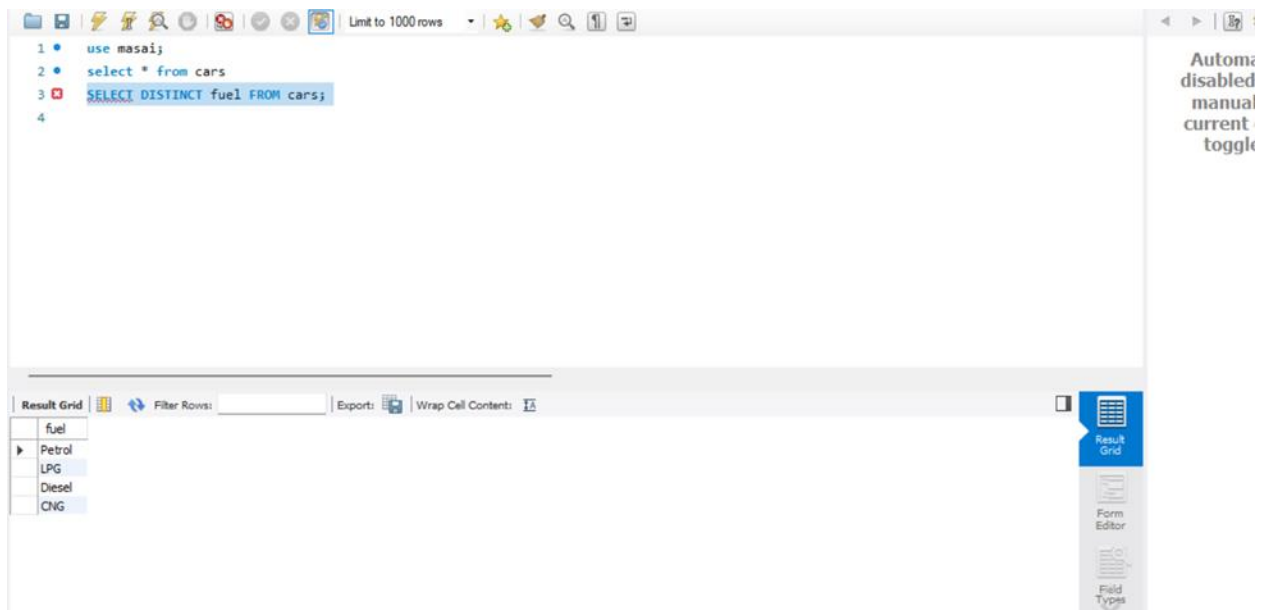


Data Analytics miniSQL Mini-Project

Insight number 1 –fuel types

What are the fuel types present in the table



The screenshot shows a SQL IDE interface. The query editor contains the following SQL code:

```
1 use masai;
2 select * from cars
3 SELECT DISTINCT fuel FROM cars;
4
```

The result grid at the bottom displays the following data:

fuel
Petrol
LPG
Diesel
CNG

On the right side of the IDE, there is a sidebar with the text "Automatically disabled manual current toggle" and a "Result Grid" button. Below the "Result Grid" button are icons for "Form Editor" and "Field Types".

Insight number 2 –maximum cars sold year

Print name and maximum cars sold in the year .sort the result in descending order on the year?

```
4
5 • select name,max(year)as max_cars from cars
6 group by name,year
7 order by year desc;
8
```

name	max_cars
Maruti Eeco CNG 5 Seater AC BSIV	2020
BMW X7 xDrive 30d DPE	2020
Kia Seltos HTX Plus AT D	2020
Toyota Innova Crysta 2.4 ZX AT	2020
Mahindra Bolero Pik-Up FB 1.7T	2020
Maruti Alto 800 LXI CNG	2020

Insight number 3 –revenue

Find the total revenue by Hyundai i20 Asta 1.2 ?

```
8
9 • SELECT SUM(selling_price)as rvenue from cars
10 where name like 'Hyundai i20 Asta 1.2' ;
11
```

rvenue
6390000

Insight number 4 –average selling price

Print name,year,km_driven,fuel type and average selling_price from table and sort the result as descending on km_driven by cars ?

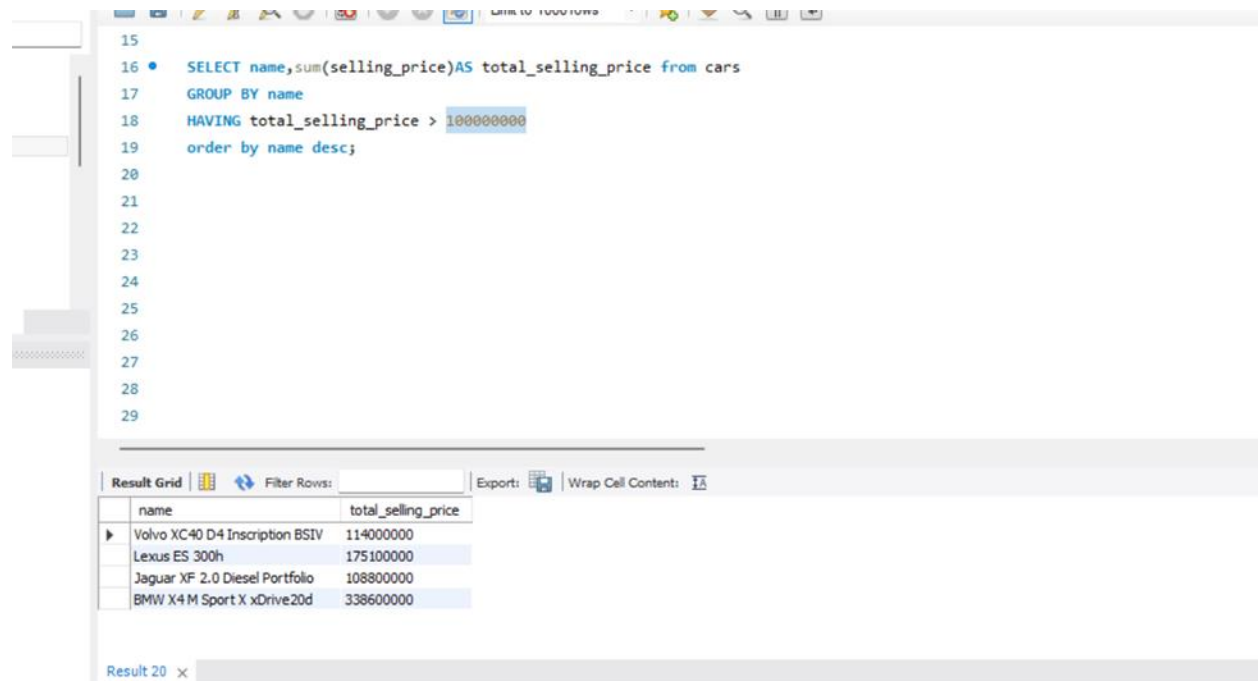
```
1
2 • SELECT name,year,km_driven,fuel,AVG(selling_price) as avg FROM cars
3 GROUP BY name,year,km_driven,fuel
4 ORDER BY km_driven desc;
5
6
7
8
9
10
11
12
```

sult Grid					
Filter Rows:		Export:	Wrap Cell Content:	Fetch rows:	
name	year	km_driven	fuel	avg	
Hyundai i20 Asta 1.2	2007	2360457	Petrol	550000.0000	
Mahindra XUV500 W6 2WD	2012	1500000	Diesel	500000.0000	
Maruti Wagon R LXI Minor	2010	577414	Petrol	194000.0000	
Maruti Wagon R VXI BS IV	2011	500000	Petrol	229999.0000	
Toyota Innova 2.5 EV Diesel PS 7 Seater BSIII	2011	500000	Diesel	350000.0000	
Mahindra Scorpio M2DI	2011	475000	Diesel	400000.0000	

sult 15 x

Insight number 5 – greater than 100000000

Print name and total selling price filter to only which have total selling price greater than 100000000 and order by name desc?



The screenshot shows a SQL query editor with a query that filters cars by total selling price and orders them by name in descending order. The query is as follows:

```
15  
16 • SELECT name, sum(selling_price) AS total_selling_price from cars  
17 GROUP BY name  
18 HAVING total_selling_price > 100000000  
19 order by name desc;  
20  
21  
22  
23  
24  
25  
26  
27  
28  
29
```

Below the query editor, the results are displayed in a table with two columns: 'name' and 'total_selling_price'. The results are filtered to show only cars with a total selling price greater than 100,000,000, ordered by name in descending order.

name	total_selling_price
Volvo XC40 D4 Inscription BSIV	114000000
Lexus ES 300h	175100000
Jaguar XF 2.0 Diesel Portfolio	108800000
BMW X4 M Sport X xDrive20d	338600000

The interface also includes a 'Result Grid' tab, a 'Filter Rows' input field, and an 'Export' button. The status bar at the bottom indicates 'Result 20'.

Insight number 6 – vowels

List the car names which have vowels as both their first and last name as character and order by name of cars by ascending?

```
21 • select * from cars;
22 • select distinct name from cars
23 WHERE (NAME LIKE 'A%' OR NAME LIKE 'E%' OR NAME LIKE 'I%' OR NAME LIKE 'O%'
24 OR NAME LIKE 'U%') AND
25 (name LIKE '%a' or name like '%e' or name like '%i' or name like '%o' or name like '%u')
26 order by name;
27
28
29
30
--
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content: [IA](#)

name
Ambassador Grand 1500 DSZ BSIII
Ashok Leyland Stile LE
Audi A4 1.8 TFSI
Audi A4 2.0 TDI
Audi A6 2.0 TDI
Audi Q5 2.0 TDI
Audi Q5 3.0 TDI Quattro
Audi Q7 3.0 TDI Quattro
Isuzu D-Max V-Cross Z Prestige

Insight number 7 –top 5

Get the name transmission type max_power and minimum selling price from cars table and order by its car name ascending and get top 5 cars ?

27

28 • `SELECT name,transmission,max_power, min(selling_price)AS min from cars`
29 `group by 1,2,3 order by name limit 5;`

30

31

32

33

34

35

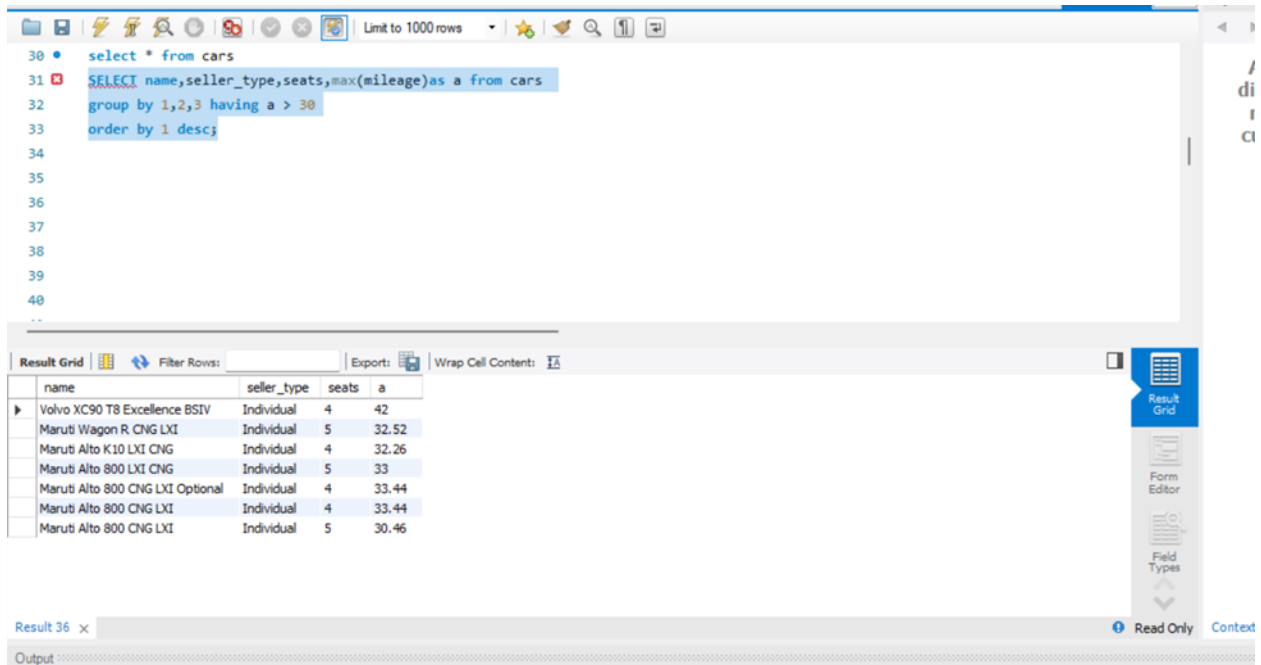
36

37

Result Grid				
Filter Rows:		Export:	Wrap Cell Content:	Fetch rows:
	name	transmission	max_power	min
▶	Ambassador CLASSIC 1500 DSL AC	Manual	35.5	75000
	Ambassador Classic 2000 DSZ AC PS	Manual	52	99000
	Ambassador Grand 1500 DSZ BSIII	Manual	52	122000
	Ambassador Grand 2000 DSZ PW CL	Manual	52	200000
	Ashok Leyland Stile LE	Manual	75	300000

Insight number 8 – maximum mileage

Print the name seller_type seats available in car and find maximum mileage of cars which is having mileage more than 30 and order it by its name descending?



The screenshot shows a SQL query editor with the following query:

```
30 • select * from cars
31 SELECT name,seller_type,seats,max(mileage)as a from cars
32 group by 1,2,3 having a > 30
33 order by 1 desc;
```

The query is executed, and the result grid displays the following data:

name	seller_type	seats	a
Volvo XC90 T8 Excellence BSIV	Individual	4	42
Maruti Wagon R CNG LXI	Individual	5	32.52
Maruti Alto K10 LXI CNG	Individual	4	32.26
Maruti Alto 800 LXI CNG	Individual	5	33
Maruti Alto 800 CNG LXI Optional	Individual	4	33.44
Maruti Alto 800 CNG LXI	Individual	4	33.44
Maruti Alto 800 CNG LXI	Individual	5	30.46

The result grid also includes a 'Filter Rows' section and an 'Export' button. The 'Read Only' status is indicated at the bottom right.

Insight number 9 – name like

print all the details of cars and get the name where “maruti” or
“Hyundai”?

```

35 • select * from cars
36 where name like '%Maruti%' or
37 name like '%Hyundai%'
38
39
40
41
42
43
44

```

name	year	selling_price	km_driven	fuel	seller_type	transmission	owner	mileage	engine [CC]	max_power	seats
Hyundai i20 Asta 1.2	2007	550000	2360457	Petrol	Individual	Manual	Second Owner	18.6	1197	81.83	5
Maruti Wagon R LXI Minor	2010	194000	577414	Petrol	Individual	Manual	Second Owner	18.9	1061	67	5
Maruti Wagon R VXI BS IV	2011	229999	500000	Petrol	Individual	Manual	Second Owner	18.9	998	67.1	5
Maruti Wagon R LXI BS IV	2012	220000	360003	Petrol	Individual	Manual	Second Owner	18.9	998	67.1	5
Hyundai Sonata 2.4 GDI MT	2012	550000	330000	Petrol	Individual	Manual	Second Owner	13.44	2359	198.25	5
Hyundai Sonata 2.4 GDI MT	2012	500000	330000	Petrol	Individual	Manual	Second Owner	13.44	2359	198.25	5
Maruti Ertiga BSIV VXI	2017	700000	227000	Petrol	Individual	Manual	First Owner	17.5	1373	91.1	7
Hyundai i20 1.2 Asta	2011	220000	220000	Petrol	Individual	Manual	Fourth & Above Owner	17	1197	80	5
Maruti 800 EX	2004	70000	220000	Petrol	Individual	Manual	Second Owner	16.1	796	37	4
Hyundai Verna Xxi ABS (Petrol)	2009	340000	214000	Petrol	Individual	Manual	Second Owner	13.9	1599	103.2	5

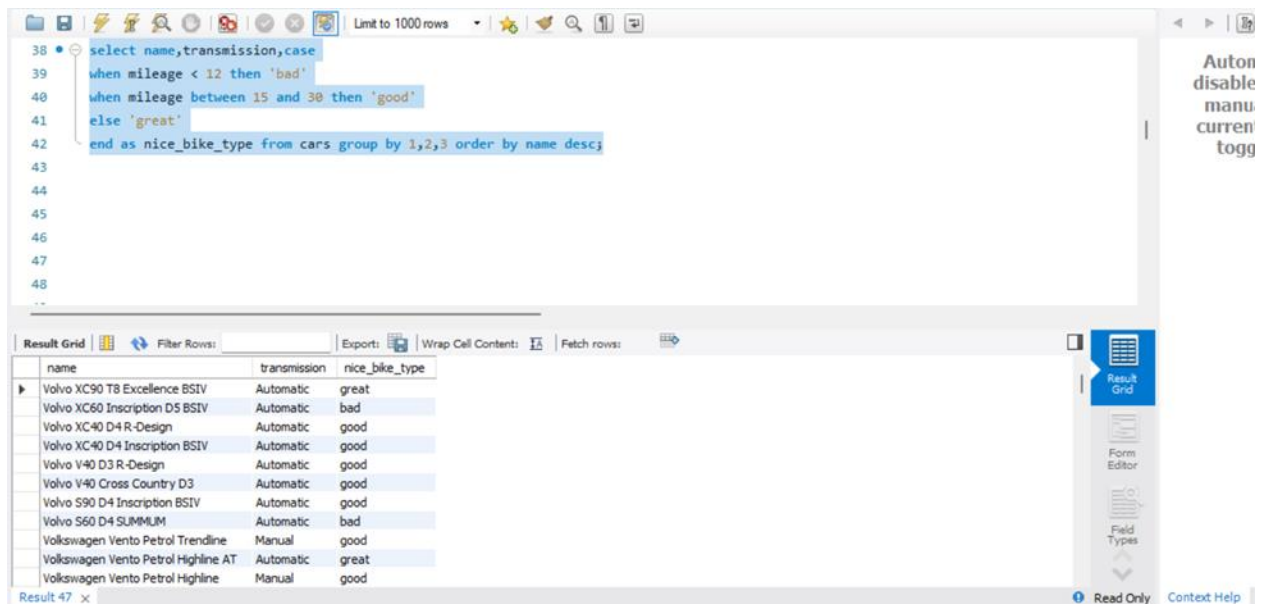
3rs 42 x

Read C

Insight number 10 – mileage classification

Categorize the mileage into 2 classes 'bad' – when mileage is less than 10, 'good' -when mileage is in between 15 to 30 and else 'great'

Print name,transmission,and the new classification .sort the result in descending order of name?



The screenshot shows a SQL query editor with the following code:

```
38 select name,transmission,case
39 when mileage < 12 then 'bad'
40 when mileage between 15 and 30 then 'good'
41 else 'great'
42 end as nice_bike_type from cars group by 1,2,3 order by name desc;
```

Below the editor is a 'Result Grid' showing the output of the query. The grid has three columns: 'name', 'transmission', and 'nice_bike_type'. The results are sorted in descending order of name.

name	transmission	nice_bike_type
Volvo XC90 T8 Excellence BSIV	Automatic	great
Volvo XC60 Inscription D5 BSIV	Automatic	bad
Volvo XC40 D4 R-Design	Automatic	good
Volvo XC40 D4 Inscription BSIV	Automatic	good
Volvo V40 D3 R-Design	Automatic	good
Volvo V40 Cross Country D3	Automatic	good
Volvo S90 D4 Inscription BSIV	Automatic	good
Volvo S60 D4 SUMMUM	Automatic	bad
Volkswagen Vento Petrol Trendline	Manual	good
Volkswagen Vento Petrol Highline AT	Automatic	great
Volkswagen Vento Petrol Highline	Manual	good

At the bottom of the interface, there is a 'Read Only' status and a 'Context Help' link.