

1.

The screenshot shows a database query editor with a SQL query and its results. The query is designed to find records in the 'shopping_trends' table where the 'size' is NULL and the 'purchase_amount' is greater than 50. The results table displays the following data:

#	CUSTOMER_ID	SIZE	PURCHASE_AMOUNT	ITEM_PURCHASED
1	11	null	74.0	Handbag
2	15	null	54.0	Jeans
3	22	null	88.0	Shirt
4	32	null	54.0	Blouse

2. List the total number of purchases grouped by Season, treating NULL values as 'Unknown Season'

The screenshot shows a database query editor with a SQL query and its results. The query is designed to count the total number of purchases grouped by season, treating NULL values as 'Unknown Season'. The results table displays the following data:

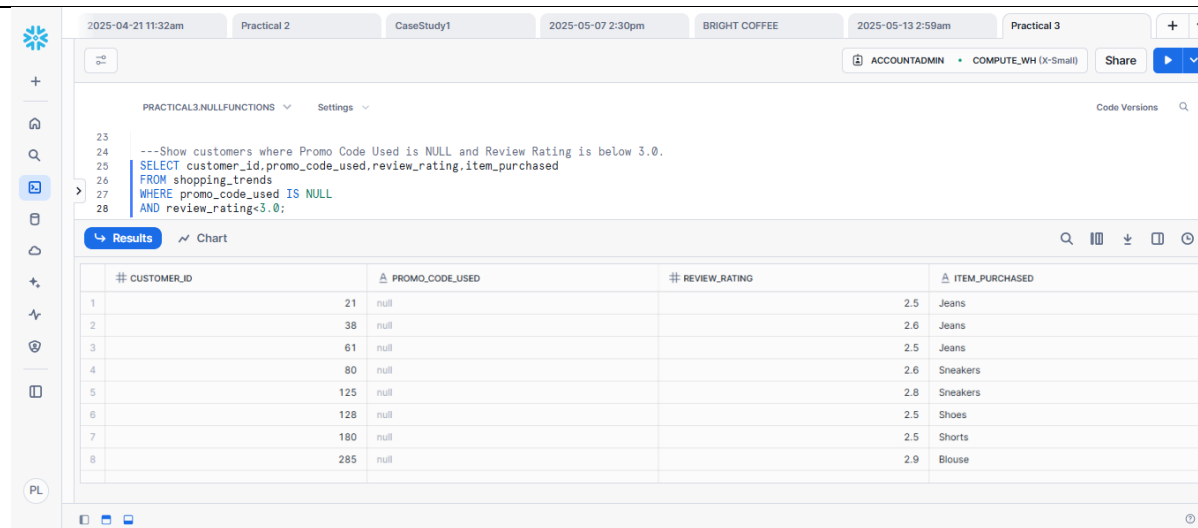
#	SEASON	TOTAL_NUMBER_OF_PURCHASES
1	Winter	71
2	Spring	66
3	Unknown season	26
4	Summer	58
5	Fall	50

3. Count how many customers used each Payment Method, treating NULLs as 'Not Provided'.

The screenshot shows a database query editor with a SQL query and its results. The query is designed to count the number of customers who used each payment method, treating NULLs as 'Not Provided'. The results table displays the following data:

#	CUSTOMER_COUNT	PAYMENT_METHOD
1	51	PayPal
2	42	Debit Card
3	30	Not Provided
4	38	Bank Transfer
5	53	Venmo
6	44	Credit Card
7	42	Cash

4. Show customers where Promo Code Used is NULL and Review Rating is below 3.0.



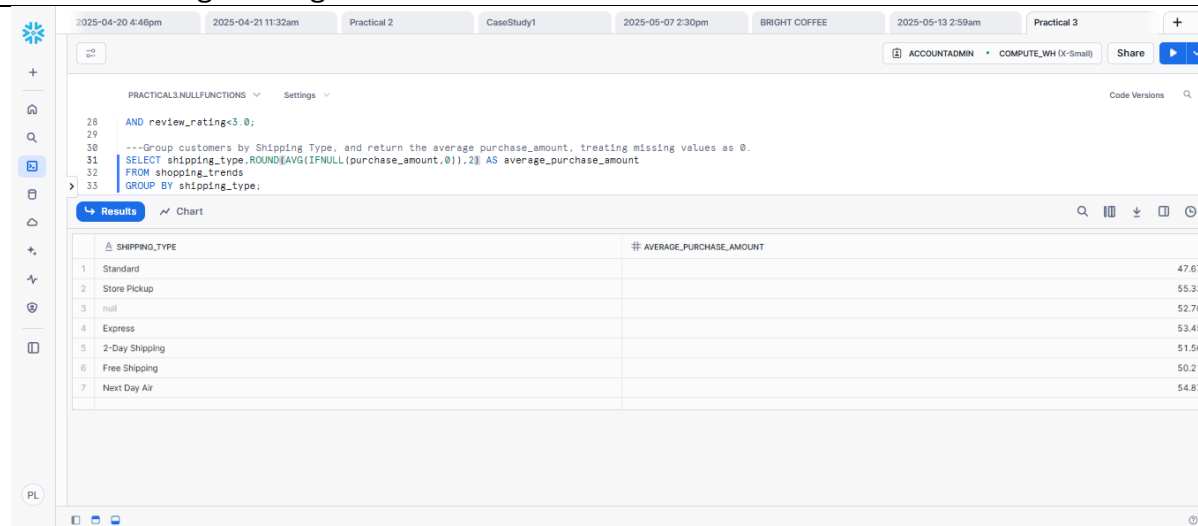
The screenshot shows a SQL IDE interface with a query editor on the left and a results table on the right. The query is as follows:

```
23 ---Show customers where Promo Code Used is NULL and Review Rating is below 3.0.
24 SELECT customer_id,promo_code_used,review_rating,item_purchased
25 FROM shopping_trends
26 WHERE promo_code_used IS NULL
27 AND review_rating<3.0;
28
```

The results table displays the following data:

#	CUSTOMER_ID	PROMO_CODE_USED	REVIEW_RATING	ITEM_PURCHASED
1	21	null	2.5	Jeans
2	38	null	2.6	Jeans
3	61	null	2.5	Jeans
4	80	null	2.6	Sneakers
5	125	null	2.8	Sneakers
6	128	null	2.5	Shoes
7	180	null	2.5	Shorts
8	285	null	2.9	Blouse

5. Group customers by Shipping Type, and return the average purchase_amount, treating missing values as 0.



The screenshot shows a SQL IDE interface with a query editor on the left and a results table on the right. The query is as follows:

```
28 AND review_rating<3.0;
29
30 ---Group customers by Shipping Type, and return the average purchase_amount, treating missing values as 0.
31 SELECT shipping_type,ROUND(AVG(IFNULL(purchase_amount,0)),2) AS average_purchase_amount
32 FROM shopping_trends
33 GROUP BY shipping_type;
```

The results table displays the following data:

#	SHIPPING_TYPE	AVERAGE_PURCHASE_AMOUNT
1	Standard	47.6
2	Store Pickup	55.3
3	null	52.7
4	Express	53.4
5	2-Day Shipping	51.5
6	Free Shipping	50.2
7	Next Day Air	54.8

6. Display the number of purchases per Location only for those with more than 5 purchases and no NULL Payment Method

2025-04-20 4:46pm 2025-04-21 11:32am Practical 2 CaseStudy1 2025-05-07 2:30pm BRIGHT COFFEE 2025-05-13 2:59am Practical 3

ACCOUNTADMIN COMPUTE_WH (X-Small) Share

PRACTICAL3.NULLFUNCTIONS Settings Code Versions

```

34
35 ---Display the number of purchases per Location only for those with more than 5 purchases and no NULL Payment Method
36 SELECT Location,COUNT(item_purchased) AS Total_Purchases
37 FROM shopping_trends
38 WHERE (SELECT COUNT(item_purchased) FROM shopping_trends) >5
39 AND payment_method IS NOT NULL
40 GROUP BY Location;
41

```

Results Chart

LOCATION	TOTAL_PURCHASES
1 Maine	36
2 Rhode Island	21
3 null	20
4 Florida	3
5 Kentucky	26
6 New York	24
7 Massachusetts	3
8 Texas	2
9 Oregon	24

7. Create a column Spender Category that classifies customers using CASE: 'High' if amount > 80, 'Medium' if BETWEEN 50 AND 80, 'Low' otherwise. Replace NULLs in purchase_amount with 0.

2025-04-20 4:46pm 2025-04-21 11:32am Practical 2 CaseStudy1 2025-05-07 2:30pm BRIGHT COFFEE 2025-05-13 2:59am Practical 3

ACCOUNTADMIN COMPUTE_WH (X-Small) Share

PRACTICAL3.NULLFUNCTIONS Settings Code Versions

```

42 ---Create a column Spender Category that classifies customers using CASE: 'High' if amount > 80, 'Medium' if BETWEEN 50 AND 80, 'Low' otherwise. Replace NULLs in purchase_amount
43 with 0.
44 SELECT customer_id,
45 IFNULL(purchase_amount,0) AS purchase_amount,
46 CASE
47 WHEN purchase_amount > 80 THEN 'High'
48 WHEN purchase_amount BETWEEN 50 AND 80 THEN 'Medium'
49 ELSE 'Low'
50 END AS Spender_category
51 FROM shopping_trends;
52

```

Results Chart

CUSTOMER_ID	PURCHASE_AMOUNT	SPENDER_CATEGORY
1	20.0	Low
2	21.0	Low
3	27.0	Low
4	45.0	Low
5	80.0	Medium
6	82.0	High
7	50.0	Medium
8	29.0	Low
9	100.0	High

8. Find customers who have no Previous Purchases value but whose Color is not NULL

2025-04-20 4:46pm 2025-04-21 11:32am Practical 2 CaseStudy1 2025-05-07 2:30pm BRIGHT COFFEE 2025-05-13 2:59am Practical 3 ACCOUNTADMIN COMPUTE_WH (X-Small) Share

PRACTICAL3.NULLFUNCTIONS Settings Code Versions

```

51 FROM shopping_trends;
52 ---Find customers who have no Previous Purchases value but whose Color is not NULL
53 SELECT customer_id,color,previous_purchases
54 FROM shopping_trends
55 WHERE previous_purchases IS NULL
56 AND color IS NOT NULL;

```

Results Chart

#	CUSTOMER_ID	COLOR	PREVIOUS_PURCHASES
1	8	Green	null
2	21	Yellow	null
3	25	White	null
4	37	Maroon	null
5	40	Gray	null
6	43	Black	null
7	44	Green	null
8	70	White	null
9	73	Maroon	null
10	75	Pink	null
11	83	Black	null
12	85	Yellow	null

9. Group records by Frequency of Purchases and show the total amount spent per group, treating NULL frequencies as 'Unknown'.

2025-04-20 4:46pm 2025-04-21 11:32am Practical 2 CaseStudy1 2025-05-07 2:30pm BRIGHT COFFEE 2025-05-13 2:59am Practical 3 ACCOUNTADMIN COMPUTE_WH (X-Small) Share

PRACTICAL3.NULLFUNCTIONS Settings Code Versions

```

57 FROM shopping_trends;
58 ---Group records by Frequency of Purchases and show the total amount spent per group, treating NULL frequencies as 'Unknown'.
59 SELECT IFNULL(frequency_of_purchases,'Unknown') AS frequency_of_purchases,
60 SUM(purchase_amount) AS total_amount_spent
61 FROM shopping_trends
62 GROUP BY frequency_of_purchases;

```

Results Chart

#	FREQUENCY_OF_PURCHASES	TOTAL_AMOUNT_SPENT
1	Every 3 Months	1749.0
2	Weekly	2184.0
3	Bi-Weekly	2099.0
4	Monthly	1780.0
5	Annually	1765.0
6	Unknown	1518.0
7	Quarterly	2541.0
8	Fortnightly	2033.0

10.

2025-04-20 4:46pm 2025-04-21 11:32am Practical 2 CaseStudy1 2025-05-07 2:30pm BRIGHT COFFEE

PRACTICAL3.NULLFUNCTIONS Settings

```

62 GROUP BY frequency_of_purchases;
63
64 ---Display a list of all Category values with the number of times each was purchased, excluding rows where Category is NULL
65 SELECT category,COUNT(previous_purchases) AS number_of_times_each_was_purchased
66 FROM shopping_trends
67 WHERE category IS NOT NULL
68 GROUP BY category;

```

Results Chart

	△ CATEGORY	# NUMBER_OF_TIMES_EACH_WAS_PURCHASED
1	Footwear	65
2	Outerwear	51
3	Clothing	53
4	Accessories	66

PL

11. Return the top 5 Locations with the highest total purchase_amount, replacing NULLs in amount with 0.

2025-04-20 4:46pm 2025-04-21 11:32am Practical 2 CaseStudy1 2025-05-07 2:30pm BRIGHT COFFEE 2025-05-13 2:59am Practical 3

ACCOUNTADMIN COMPUTE_WH (X-Small) Share

PRACTICAL3.NULLFUNCTIONS Settings Code Versions

```

69
70 ---Return the top 5 Locations with the highest total purchase_amount, replacing NULLs in amount with 0.
71 SELECT Location,SUM(IFNULL(purchase_amount,0)) AS total_purchase_amount
72 FROM shopping_trends
73 GROUP BY Location
74 ORDER BY total_purchase_amount DESC
75 LIMIT 5;

```

Results Chart

	△ LOCATION	# TOTAL_PURCHASE_AMOUNT
1	Maine	2294
2	Florida	1980
3	Massachusetts	1899
4	Rhode Island	1876
5	Kentucky	1798

PL

12. Group customers by Gender and Size and count how many entries have a NULL Color.

7:00pm 2025-04-20 4:46pm 2025-04-21 11:32am Practical 2 CaseStudy1 2025-05-07 2:30pm BRIGHT COFFEE 2025-05-13 2:59am Practical 3 ACCOUNTADMIN COMPUTE_WH (X-Small) Share

PRACTICAL3.NULLFUNCTIONS Settings Code Versions

```

76
77 ---Group customers by Gender and Size, and count how many entries have a NULL Color.
78 SELECT gender,size, COUNT(customer_id) AS NULL_Color_count
79 FROM shopping_trends
80 WHERE color IS NULL
81 GROUP BY gender,size;

```

Results Chart

	GENDER	SIZE	# NULL_COLOR_COUNT
1	Male	M	7
2	Male	null	6
3	Male	S	5
4	Male	XL	5
5	Male	L	6

Query Details

Query duration 1.3s

Rows 5

Query ID 01bc5ca7-0000-fb41-0-

Show more

GENDER

Male 5

SIZE

80% filled 20% null

NULL_COLOR_COUNT

#

13. Identify all Item Purchased where more than 3 purchases had NULL Shipping Type

7:00pm 2025-04-20 4:46pm 2025-04-21 11:32am Practical 2 CaseStudy1 2025-05-07 2:30pm BRIGHT COFFEE 2025-05-13 2:59am Practical 3 ACCOUNTADMIN COMPUTE_WH (X-Small) Share

PRACTICAL3.NULLFUNCTIONS Settings Code Versions

```

82
83 ---Identify all Item Purchased where more than 3 purchases had NULL Shipping Type
84 SELECT item_purchased,shipping_type
85 FROM shopping_trends
86 WHERE shipping_type IS NULL;
87

```

Results Chart

	ITEM_PURCHASED	SHIPPING_TYPE
1	null	null
2	Jeans	null
3	Blouse	null
4	Blouse	null
5	Shoes	null
6	Handbag	null
7	Shirt	null
8	Shirt	null
9	Shorts	null
10	Shirt	null
11	Shorts	null
12	Blouse	null
13	Sandals	null
14	Coat	null

Query Details

Query duration 136ms

Rows 22

Query ID 01bc5cfa-0000-fb45-0-

Show more

ITEM_PURCHASED

Shirt 5

Shoes 4

Blouse 3

+ 5 more

SHIPPING_TYPE

0% filled 100% null

14. Show a count of how many customers per Payment Method have NULL Review Rating

7:00pm 2025-04-20 4:46pm 2025-04-21 11:32am Practical 2 CaseStudy1 2025-05-07 2:30pm BRIGHT COFFEE 2025-05-13 2:59am Practical 3 ACCOUNTADMIN COMPUTE_WH (X-Small) Share

PRACTICAL3.NULLFUNCTIONS Settings Code Versions

```

90
91 ---Show a count of how many customers per Payment Method have NULL Review Rating
92 SELECT payment_method,COUNT(customer_id) AS Missing_Review_Rating_Count
93 FROM shopping_trends
94 WHERE review_rating IS NULL
95 GROUP BY payment_method;

```

Results Chart

	PAYMENT_METHOD	MISSING_REVIEW_RATING_COUNT
1	Credit Card	
2	Cash	
3	Bank Transfer	
4	Debit Card	
5	Venmo	
6	PayPal	
7	null	

15. Group by Category and return the average Review Rating, replacing NULLs with 0, and filter only where average is greater than 3.5.



The screenshot shows a SQL IDE interface with a query editor and a results pane. The query is as follows:

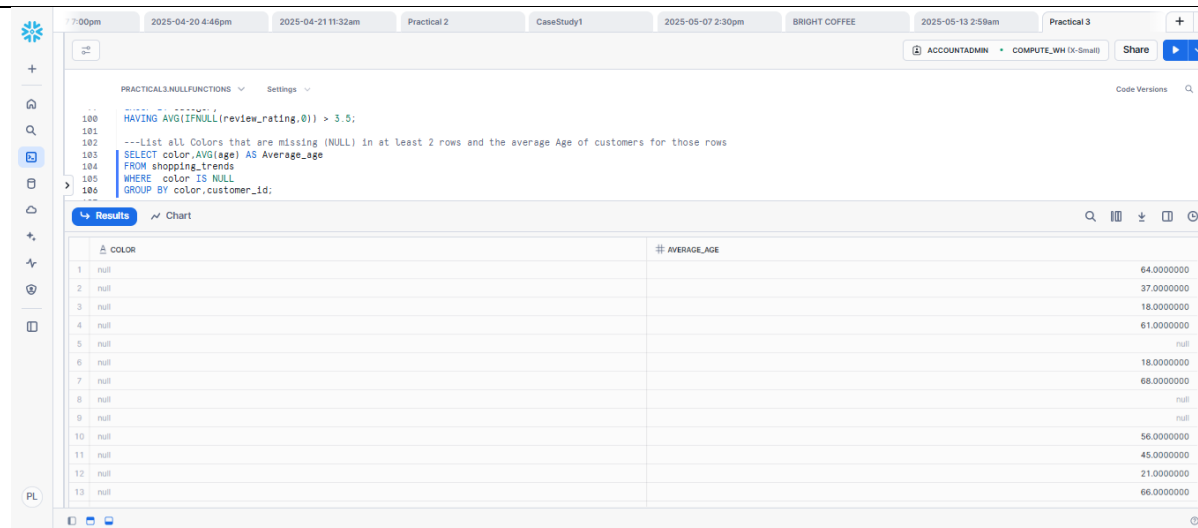
```

94 ---Group by Category and return the average Review Rating, replacing NULLs with 0, and filter only where average is greater than 3.5.
95 SELECT category,AVG(IFNULL(review_rating,0)) AS Average_review_rating
96 FROM shopping_trends
97 GROUP BY category
98 HAVING AVG(IFNULL(review_rating,0)) > 3.5;
99
100 ---List all Colors that are missing (NULL) in at least 2 rows and the average Age of customers for those rows
101 SELECT color,AVG(age) AS Average_age

```

The results pane shows a table with two columns: CATEGORY and AVERAGE_REVIEW_RATING. The message "Query produced no results" is displayed in the center of the table.

16. List all Colors that are missing (NULL) in at least 2 rows and the average Age of customers for those rows



The screenshot shows a SQL IDE interface with a query editor and a results pane. The query is as follows:

```

100 HAVING AVG(IFNULL(review_rating,0)) > 3.5;
101
102 ---List all Colors that are missing (NULL) in at least 2 rows and the average Age of customers for those rows
103 SELECT color,AVG(age) AS Average_age
104 FROM shopping_trends
105 WHERE color IS NULL
106 GROUP BY color,customer_id;

```

The results pane shows a table with two columns: COLOR and AVERAGE_AGE. The table contains 13 rows of data:

	COLOR	AVERAGE_AGE
1	null	64.00000000
2	null	37.00000000
3	null	18.00000000
4	null	61.00000000
5	null	null
6	null	18.00000000
7	null	68.00000000
8	null	null
9	null	null
10	null	56.00000000
11	null	45.00000000
12	null	21.00000000
13	null	66.00000000

17. Use CASE to create a column Delivery Speed: 'Fast' if Shipping Type is 'Express' or 'Next Day Air', 'Slow' if 'Standard', 'Other' for all else including NULL. Then count how many customers fall into each category.

20. Show customers who have a NULL Shipping Type but made a purchase in the range of 30 to 70 USD.

100pm2025-04-20 4:46pm2025-04-21 11:32amPractical 2CaseStudy12025-05-07 2:30pmBRIGHT COFFEE2025-05-13 2:59amPractical 3+

ACCOUNTADMINCOMPUTE_WH (X-Small)Share

PRACTICAL3.NULLFUNCTIONSSettings

Code Versions

```
---
151  -----
152  HAVING AVG(review_rating)>4.0;
153
154  ---Show customers who have a NULL Shipping Type but made a purchase in the range of 30 to 70 USD.
155  SELECT customer_id,shipping_type,purchase_amount,item_purchased
156  FROM shopping_trends
157  WHERE shipping_type IS NULL
158  AND purchase_amount BETWEEN 30 AND 70;
```

ResultsChart

#	CUSTOMER_ID	SHIPPING_TYPE	PURCHASE_AMOUNT	ITEM_PURCHASED
1	15	null	54.0	Jeans
2	105	null	43.0	Shirt
3	141	null	37.0	Shorts
4	196	null	66.0	Coat
5	213	null	36.0	Shirt
6	235	null	38.0	Sandals
7	293	null	35.0	null

PL