

Mendiso Moyo

## Null Funtions Practical

1. Find all records where Size is missing and the purchase\_amount is greater than 50. Expected Columns: Customer ID, Size, purchase\_amount, Item Purchased

The screenshot shows the Snowflake web interface. The query editor contains the following SQL code:

```
SELECT
  customer_id,
  purchase_amount,
  item_purchased,
  size,
FROM
  shopping_trends
WHERE size IS NULL
AND purchase_amount > 50;
```

The results table shows the following data:

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

Query Details: Query duration 69ms, Rows 22, Query ID 01bc5f6a-0001-d957-0...

1. List the total number of purchases grouped by Season, treating NULL values as 'Unknown Season'. Expected Columns: Season, Total Purchases

The screenshot shows the Snowflake web interface. The query editor contains the following SQL code:

```
SELECT
  COALESCE (season, 'unknown_season') as season,
  COUNT(*) as total_purchases
FROM shopping_trends
GROUP BY COALESCE (season, 'unknown_season');
```

The results table shows the following data:

#	SEASON	TOTAL_PURCHASES
1	Winter	80
2	Spring	73
3	unknown_season	27
4	Summer	65

Query Details: Query duration 66ms, Rows 5, Query ID 01bc5f73-0001-d9d1-0...

2. Count how many customers used each Payment Method, treating NULLs as 'Not Provided'. Expected Columns: Payment Method, Customer Count

```

29 ----3. Count how many customers used each Payment Method, treating NULLs as 'Not Provided'. Expected Columns: Payment Method, Customer Count
30
31 SELECT
32 COALESCE (payment_method, 'not_provided') as payment_method,
33 COUNT (DISTINCT customer_id) as customer_count
34 FROM shopping_trends
35 GROUP BY COALESCE (payment_method, 'not_provided');

```

	PAYMENT_METHOD	# CUSTOMER_COUNT
1	PayPal	51
2	Debit Card	42
3	not_provided	30
4	Bank Transfer	38
5	Credit Card	44

Query Details  
 Query duration 355ms  
 Rows Z  
 Query ID 01bc5f76-0001-d9da-0...

4. Show customers where Promo Code Used is NULL and Review Rating is below 3.0. Expected Columns: Customer ID, Promo Code Used, Review Rating, Item Purchased

```

37 ----4. Show customers where Promo Code Used is NULL and Review Rating is below 3.0. Expected Columns: Customer ID, Promo Code Used, Review Rating, Item Purchased
38
39 SELECT
40 customer_id,
41 promo_code_used,
42 review_rating,
43 item_purchased
44 FROM shopping_trends
45 WHERE promo_code_used IS NULL;
46

```

	# CUSTOMER_ID	PROMO_CODE_USED	# REVIEW_RATING	ITEM_PURCHASED
1	1	null	3.7	Jeans
2	12	null	4.1	Sneakers
3	21	null	2.5	Jeans
4	24	null	4.5	Shorts
5	27	null	null	Blouse
6	33	null	null	null

Query Details  
 Query duration 377ms  
 Rows 40  
 Query ID 01bc5f77-0001-d9bc-0...

5. Group customers by Shipping Type, and return the average purchase\_amount, treating missing values as 0. Expected Columns: Shipping Type, Average purchase\_amount

```

47 ----5. Group customers by Shipping Type, and return the average purchase_amount, treating missing values as 0. Expected Columns: Shipping Type, Average purchase_amount
48
49 SELECT shipping_type,
50 AVG(COALESCE(purchase_amount,0)) as Average_Purchase_Amount,
51 FROM shopping_trends
52 GROUP BY shipping_type;
53

```

	SHIPPING_TYPE	AVERAGE_PURCHASE_AMOUNT
1	Standard	47.6666667
2	Store Pickup	55.3333333
3	null	52.7037037
4	Express	53.4545455
5	2-Day Shipping	51.5576923
6	Free Shipping	50.2142857

Query Details  
 Query duration 91ms  
 Rows Z  
 Query ID 01bc5f79-0001-d957-0...

6. Display the number of purchases per Location only for those with more than 5 purchases and no NULL Payment Method. Expected Columns: Location, Total Purchases.

```

54 ---6. Display the number of purchases per Location only for those with more than 5 purchases and no NULL Payment Method. Expected Columns: Location, Total
Purchases
55
56 SELECT
57 COALESCE (location, 'not_provided') as location,
58 COUNT (*) AS Total_purchases,
59 FROM shopping_trends
60 WHERE payment_method is NULL
61 GROUP BY location
62 HAVING total_purchases> 5;

```

Results		Chart	Query Details
	LOCATION	# TOTAL_PURCHASES	
1	Rhode Island	8	Query duration 363ms

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. Expected Columns: Customer ID, purchase\_amount, Spender Category

```

54 ---6. Display the number of purchases per Location only for those with more than 5 purchases and no NULL Payment Method. Expected Columns: Location, Total
Purchases
55
56 SELECT
57 COALESCE (location, 'not_provided') as location,
58 COUNT (*) AS Total_purchases,
59 FROM shopping_trends
60 WHERE payment_method is NULL
61 GROUP BY location
62 HAVING total_purchases> 5;

```

Results		Chart	Query Details
	LOCATION	# TOTAL_PURCHASES	
1	Rhode Island	8	Query duration 363ms

8. Find customers who have no Previous Purchases value but whose Color is not NULL. Expected Columns: Customer ID, Color, Previous Purchases