Practical 3 Advanced SQL NULL Functions – Lefa Makhurubetshi

Question 1

```sql
--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

SELECT CUSTOMER_ID,
       Size,
       purchase_amount,
       Item_Purchased
FROM practical3.shopping.trends
WHERE Size IS NULL AND  purchase_amount > 50;

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

ults (2 minutes ago)

able | Chart | 22 rows | 57ms

CUSTOMER_ID	SIZE	PURCHASE_AMOUNT	ITEM_PURCHASED
11	null	74.0	Handbag
15	null	54.0	Jeans
22	null	88.0	Shirt
32	null	54.0	Blouse
62	null	57.0	Blouse
73	null	65.0	Sandals
91	null	54.0	Shoes
97	null	56.0	Shoes
100	null	55.0	Sneakers
160	null	84.0	Coat

Question 2

```sql
--2. List the total number of purchases grouped by Season, treating NULL values as 'Unknown Season'.
--Expected Columns: Season, Total Purchases

SELECT IFNULL(season, 'Unknown Season') AS season,
       SUM(purchase_amount) AS Total_Purchases
FROM practical3.shopping.trends
GROUP BY  IFNULL(season, 'Unknown Season');
```

Results (just now)

Table | Chart | 5 rows | 38ms

	SEASON	TOTAL_PURCHASES
1	Summer	3861.0
2	Unknown Season	1376.0
3	Winter	3975.0
4	Spring	3887.0
5	Fall	2570.0

Question 3

```sql
--3. Count how many customers used each Payment Method, treating NULLs as 'Not Provided'.
--Expected Columns: Payment Method, Customer Count

SELECT IFNULL(payment_method, 'Not Provided') AS Payment_Method,
       COUNT(customer_id) AS Customer_Count
FROM practical3.shopping.trends
GROUP BY  IFNULL(payment_method, 'Not Provided');
```

Results (2 minutes ago)

Table | Chart — 7 rows — 40ms

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

Question 4

```sql
--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

SELECT Customer_ID,
       Promo_Code_Used,
       Review_Rating,
       Item_Purchased
FROM practical3.shopping.trends
WHERE Promo_Code_Used IS NULL AND Review_Rating < 3;
```

Results (2 minutes ago)

Table | Chart — 8 rows — 45ms

	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

Question 5

```sql
--5. Group customers by Shipping Type, and return the average purchase_amount, treating missing values as 0.
--Expected Columns: Shipping Type, Average purchase_amount

SELECT shipping_type,
    AVG(IFNULL(purchase_amount,0)) AS Average_purchase_amount
FROM practical3.shopping.trends
GROUP BY shipping_type;
```

Results (2 minutes ago)

Table | Chart — 7 rows — 38ms

	SHIPPING_TYPE	AVERAGE_PURCHASE_AMOUNT
1	Free Shipping	50.2142857
2	Store Pickup	55.3333333
3	null	52.7037037
4	Express	53.4545455
5	Standard	47.6666667
6	Next Day Air	54.8666667
7	2-Day Shipping	51.5576923

Question 6

```sql
--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

SELECT
    location,
    SUM(previous_purchases) AS Total_Purchases
FROM practical3.shopping.trends
WHERE payment_method IS NOT NULL
GROUP BY location
HAVING SUM(previous_purchases) > 5;
```

Results (2 minutes ago)

Table | Chart — 9 rows — 39ms

	LOCATION	TOTAL_PURCHASES
1	null	706.0
2	Maine	779.0
3	Oregon	595.0
4	Kentucky	786.0
5	Florida	653.0
6	Massachusetts	792.0
7	Texas	407.0
8	Rhode Island	514.0
9	New York	645.0

Question 7

```sql
--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

SELECT customer_id,
       IFNULL(purchase_amount,0) AS purchase_amount,
  CASE
    WHEN purchase_amount > 80 THEN 'High'
    WHEN purchase_amount BETWEEN 50 AND 80 THEN 'Medium'
    ELSE 'Low'
  END AS Spender_Category
FROM practical3.shopping.trends;
```

Results (2 minutes ago)

Table | Chart — 300 rows — 28ms

	CUSTOMER_ID	PURCHASE_AMOUNT	SPENDER_CATEGORY
1	1	20.0	Low
2	2	21.0	Low
3	3	27.0	Low
4	4	45.0	Low
5	5	80.0	Medium
6	6	82.0	High
7	7	50.0	Medium
8	8	29.0	Low
9	9	100.0	High
10	10	97.0	High

Question 8

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

SELECT customer_id,
       color,
       previous_purchases
FROM practical3.shopping.trends
WHERE color IS NOT NULL AND previous_purchases IS NULL;
```

Results (2 minutes ago)

Table | Chart — 36 rows — 41ms

	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

Question 9

```sql
--9. Group records by Frequency of Purchases and show the total amount spent per group, treating NULL frequencies as 'Unknown'.
--Expected Columns: Frequency of Purchases, Total purchase_amount

SELECT IFNULL(frequency_of_purchases, 'Unknown') AS frequency_of_purchases,
       SUM(purchase_amount) AS Total_Purchases
FROM practical3.shopping.trends
GROUP BY  IFNULL(frequency_of_purchases, 'Unknown');
```

Results (2 minutes ago)

Table | Chart — 8 rows — 38ms

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

Question 10

```sql
--10.Display a list of all Category values with the number of times each was purchased, excluding rows where Categoryis NULL.
--Expected Columns: Category, Total Purchases

SELECT Category,
       SUM(purchase_amount) AS Total_Purchases
FROM practical3.shopping.trends
WHERE Category IS NOT NULL
GROUP BY Category;
```

Results (2 minutes ago)

Table | Chart — 4 rows — 38ms

	CATEGORY	TOTAL_PURCHASES
1	Clothing	3022.0
2	Accessories	4242.0
3	Footwear	3733.0
4	Outerwear	2880.0

Question 11

```sql
--11.Return the top 5 Locations with the highest total purchase_amount, replacing NULLs in amount with 0.
--Expected Columns: Location, Total purchase_amount

SELECT
    location,
    SUM(IFNULL(previous_purchases,0)) AS Total_Purchases
FROM practical3.shopping.trends
GROUP BY location
ORDER BY SUM(IFNULL(previous_purchases,0)) DESC
LIMIT 5;
```

Results (2 minutes ago)

Table | Chart — 5 rows — 26ms

	LOCATION	TOTAL_PURCHASES
1	Maine	924.0
2	Kentucky	912.0
3	Massachusetts	873.0
4	Florida	826.0
5	Rhode Island	743.0

Question 12

```sql
--12.Group customers by Gender and Size, and count how many entries have a NULL Color.
--Expected Columns: Gender, Size, Null Color Count

SELECT gender,
       size,
       COUNT(customer_id) AS NULL_Color_Count
FROM practical3.shopping.trends
WHERE color IS NULL
GROUP BY gender,size;
```

Results (2 minutes ago)

Table | Chart — 5 rows — 32ms

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

Question 13

```
--13.Identify all Item Purchased where more than 3 purchases had NULL Shipping Type.
--Expected Columns: Item Purchased, NULL Shipping Type Count

SELECT ITEM_PURCHASED,
       COUNT(customer_id) AS NULL_Shipping_Type_Count
FROM practical3.shopping.trends
WHERE SHIPPING_TYPE IS NULL
GROUP BY ITEM_PURCHASED
HAVING COUNT(customer_id)  > 3;
 Ctrl+I to generate
```

Results (2 minutes ago)

Table | Chart — 3 rows — 37ms

	ITEM_PURCHASED	NULL_SHIPPING_TYPE_COUNT
1	Shirt	5
2	null	4
3	Shoes	4

Question 14

```
--14.Show a count of how many customers per Payment Method have NULL Review Rating.
--Expected Columns: Payment Method, Missing Review Rating Count

SELECT PAYMENT_METHOD,
       COUNT(customer_id) AS Missing_Review_Rating_Count
FROM practical3.shopping.trends
WHERE REVIEW_RATING IS NULL
GROUP BY PAYMENT_METHOD;
```

Results (2 minutes ago)

Table | Chart — 7 rows — 30ms

	PAYMENT_METHOD	MISSING_REVIEW_RATING_COUNT
1	PayPal	3
2	null	2
3	Credit Card	8
4	Venmo	9
5	Cash	4
6	Bank Transfer	4
7	Debit Card	7

Question 15

```
--15.Group by Category and return the average Review Rating, replacing NULLs with 0, and filter only where average is greater than 3.5.
--Expected Columns: Category, Average Review Rating

SELECT Category,
       AVG(IFNULL(Review_Rating,0)) AS Average_Review_Rating
FROM practical3.shopping.trends
GROUP BY Category
HAVING AVG(IFNULL(Review_Rating,0)) > 3.5;
```

Results (2 minutes ago)

Table | Chart — 0 rows — 31ms

CATEGORY	AVERAGE_REVIEW_RATING

Query produced no results

Question 16

```sql
--16. List all Colors that are missing (NULL) in at least 2 rows and the average Age of customers for those rows.
--Expected Columns: Color, Average Age

SELECT color,
       AVG(Age) AS Average_Age
FROM practical3.shopping.trends
WHERE color IS NULL
GROUP BY color
HAVING COUNT(*) >= 2;
```

Results (2 minutes ago)

Table | Chart — 1 row — 48ms

	COLOR	AVERAGE_AGE
1	null	47.8461538

Question 17

```sql
--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.
--Expected Columns: Delivery Speed, Customer Count

SELECT shipping_type,
       COUNT(customer_id) AS Customer_Count,
   CASE
     WHEN shipping_type IN ('Express','Next Day Air') THEN 'Fast'
     WHEN shipping_type = 'Standard' THEN 'Slow'
     ELSE 'Other'
   END AS Delivery_Speed
FROM practical3.shopping.trends
GROUP BY shipping_type;
```

Results (2 minutes ago)

Table | Chart — 7 rows — 45ms

	SHIPPING_TYPE	CUSTOMER_COUNT	DELIVERY_SPEED
1	Free Shipping	42	Other
2	Store Pickup	45	Other
3	null	27	Other
4	Express	44	Fast
5	Standard	45	Slow
6	Next Day Air	45	Fast
7	2-Day Shipping	52	Other

Question 18

```sql
--18.Find customers whose purchase_amount is NULL and whose Promo Code Used is 'Yes'.
--Expected Columns: Customer ID, purchase_amount, Promo Code Used

SELECT customer_id,
       purchase_amount,
       promo_code_used
FROM practical3.shopping.trends
WHERE purchase_amount IS NULL AND promo_code_used = 'Yes';
```

Results (1 minute ago)

Table | Chart — 20 rows — 49ms

	CUSTOMER_ID	PURCHASE_AMOUNT	PROMO_CODE_USED
1	13	null	TRUE
2	30	null	TRUE
3	78	null	TRUE
4	95	null	TRUE
5	124	null	TRUE
6	129	null	TRUE
7	130	null	TRUE
8	138	null	TRUE
9	153	null	TRUE
10	168	null	TRUE

Question 19

```sql
--19.Group by Location and show the maximum Previous Purchases, replacing NULLs with 0, only where the average rating is above 4.0.
--Expected Columns: Location, Max Previous Purchases, Average Review Rating

SELECT
      location,
      MAX(IFNULL(Previous_Purchases,0)) AS MAX_Previous_Purchases,
      AVG(Review_Rating) AS Average_Review_Rating
FROM practical3.shopping.trends
GROUP BY location
HAVING AVG(Review_Rating) > 4.0;
```

Results (2 minutes ago)

Table | Chart — 0 rows — 35ms

LOCATION	MAX_PREVIOUS_PURCHASES	AVERAGE_REVIEW_RATING

Query produced no results

Question 20

```sql
--20.Show customers who have a NULL Shipping Type but made a purchase in the range of 30 to 70 USD.
--Expected Columns: Customer ID, Shipping Type, purchase_amount, Item Purchased

SELECT customer_id,
      shipping_type,
      purchase_amount,
      item_purchased
FROM practical3.shopping.trends
WHERE shipping_type IS NULL AND purchase_amount BETWEEN 30 AND 70;
```

Results (2 minutes ago)

Table | Chart — 7 rows — 28ms

	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