Practical 3

Question 1

```
LIMIT 50;

--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.shoppingtrends
WHERE size IS NULL AND purchase_amount > 50;
```

Results (just now)

Table | Chart — 22 rows — 54ms

	CUSTOMER_ID	SIZE	PURCHASE_AMOUNT	ITEM_PURCHASED
	11 – 296	All values are null	54 – 100	Shoes 18.2%, Handbag 13.6%, +8 more
1	11	null	74.0	Handbag
2	15	null	54.0	Jeans
3	22	null	88.0	Shirt
4	32	null	54.0	Blouse
5	62	null	57.0	Blouse
6	73	null	65.0	Sandals
7	91	null	54.0	Shoes

Question 2

```
--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,
       COUNT(customer_id) AS TotalPurchases
FROM practical3.shopping.shoppingtrends
GROUP BY season;
```

Results (just now)

Table | Chart — 5 rows — 35ms

	SEASON	TOTALPURCHASES
1	Summer	65
2	Winter	80
3	Unknown Season	27
4	Fall	55
5	Spring	73

Question 3

```
--3. Count how many customers used each Payment Method, treating NULLs as 'Not Provided'.
--Expected Columns: Payment Method, Customer Count
-- To Check dataset
SELECT DISTINCT IFNULL(payment_method,'Not Provided') AS payment_method,
                COUNT(payment_method) AS Customer_count
FROM practical3.shopping.shoppingtrends
GROUP BY payment_method;
```

Results (just now)

Table | Chart — 7 rows — 26ms

	PAYMENT_METHOD	CUSTOMER_COUNT
1	Venmo	53
2	Bank Transfer	38
3	Credit Card	44
4	Debit Card	42
5	PayPal	51
6	Not Provided	0
7	Cash	42

Question 4

```
28      --4. Show customers where Promo Code Used is NULL and Review Rating is below 3.0.
29      --Expected Columns: Customer ID, Promo Code Used, Review Rating, Item Purchased
30      SELECT customer_id, promo_code_used,review_rating,item_purchased
31      FROM practical3.shopping.shoppingtrends
32      WHERE promo_code_used IS NULL AND review_rating < 3.0;
33      ----------------------------------------------------------------------------------------------------------------------------
```

Results (4 minutes ago)

Table | Chart — 4 rows — 32ms

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

Question 5

```
33      ----------------------------------------------------------------------------------------------------------------------------
34      --5. Group customers by Shipping Type, and return the average purchase_amount, treating missing values as 0.
35      --Expected Columns: Shipping Type, Average purchase_amount
36      SELECT IFNULL(shipping_type,'0') AS shipping_type,
37             AVG(purchase_amount) AS avearage_purchase_amount
38      FROM practical3.shopping.shoppingtrends
39      GROUP BY shipping_type;
40      ----------------------------------------------------------------------------------------------------------------------------
```

Results (4 minutes ago)

Table | Chart — 4 rows — 32ms

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

Question 6

```
41      --6. Display the number of purchases per Location only for those with more than 5 purchases and no NULL Payment Method.
42      --Expected Columns: Location, Total Purchases
43      SELECT location,
44             COUNT(customer_id) AS TotalPurchases
45      FROM practical3.shopping.shoppingtrends
46      WHERE payment_method IS NOT NULL
47      GROUP BY location
48      HAVING TotalPurchases > 5;
49      ----------------------------------------------------------------------------------------------------------------------------
```

Results (3 minutes ago)

Table | Chart — 4 rows — 32ms

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

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.shoppingtrends;
```

Results (2 minutes ago)

Table | Chart — 4 rows — 32ms

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

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.shoppingtrends
WHERE previous_purchases IS NULL AND color IS NOT NULL;
```

Results (1 minute ago)

Table | Chart — 4 rows — 32ms

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

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_purchase_amount
FROM practical3.shopping.shoppingtrends
GROUP BY frequency_of_purchases;
```

Results (just now)

Table | Chart — 4 rows — 32ms

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

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_purchase_amount
FROM practical3.shopping.shoppingtrends
WHERE category IS NOT NULL
GROUP BY category;
```

Results (just now)

Table | Chart — 4 rows — 32ms

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

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(purchase_amount, 0 )) AS Total_purchase_amount
FROM practical3.shopping.shoppingtrends
GROUP BY location
ORDER BY Total_purchase_amount DESC
LIMIT 5;
```

Results (1 minute ago)

Table | Chart — 5 rows — 26ms

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

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(*) AS Null_Color_Count
FROM practical3.shopping.shoppingtrends
WHERE color IS NULL
GROUP BY gender, size;
```

Results (just now)

Table | Chart — 5 rows — 26ms

	GENDER	SIZE	NULL_COLOR_COUNT
1	Male	null	6
2	Male	M	7
3	Male	L	6
4	Male	S	5
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(*) AS Null_Shipping_Type_Count
FROM practical3.shopping.shoppingtrends
WHERE shipping_type IS NULL
GROUP BY item_purchased
HAVING COUNT(*) > 3;
```

Results (just now)

Table | Chart — 3 rows — 35ms

	ITEM_PURCHASED	NULL_SHIPPING_TYPE_COUNT
1	null	4
2	Shirt	5
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(*) AS Missing_review_rating_count
FROM practical3.shopping.shoppingtrends
WHERE review_rating IS NULL
GROUP BY payment_method;
```

Results (just now)

Table | Chart — 7 rows — 38ms

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

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.shoppingtrends
GROUP BY category
HAVING AVG(IFNULL(review_rating, 0 )) > 3.5;
```

Results (just now)

Table | Chart — 0 rows — 46ms

CATEGORY	AVERAGE_REVIEW_RATING

Query produced no results

After checking the data, there is no value above 3.5

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.shoppingtrends
WHERE color IS NULL
GROUP BY color
HAVING COUNT(*) >= 2;
```

Results (just now)

0 rows 20ms

SQL compilation error: syntax error line 135 at position 6 unexpected ' C'.

I don't know whether there is an issue wit my snowflake or or maybe there something else I am missing but to the best of my knowledge this code is right. I couldn't as Lerato or Ketro because it was way past their work times.

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
    CASE
        WHEN shipping_type IN ('Express', 'Next Day Air') THEN 'Fast'
        WHEN shipping_type IN ('Standard') THEN 'Standard'
        ELSE 'Other'
    END AS Delivery_Speed,
    COUNT(*) AS Customer_Count
FROM practical3.shopping.shoppingtrends
GROUP BY 1;
```

Results (just now)

Table Chart 3 rows 27ms

	DELIVERY_SPEED	CUSTOMER_COUNT
1	Other	166
2	Standard	45
3	Fast	89

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.shoppingtrends
WHERE purchase_amount IS NULL AND promo_code_used = 'Yes';
```

Results (just now)

Table | Chart — 20 rows — 37ms

	CUSTOMER_ID (13 – 298)	PURCHASE_AMOUNT (All values are null)	PROMO_CODE_USED (true 100.0%)
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

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 Maximum_Previous_Purchases,
       AVG(review_rating) AS Average_Review_Rating
FROM practical3.shopping.shoppingtrends
GROUP BY location
HAVING AVG(review_rating) > 4.0;
```

Results (just now)

Table | Chart — 0 rows — 28ms

LOCATION	MAXIMUM_PREVIOUS_PURCHASES	AVERAGE_REVIEW_RATING

Query produced no results

After checking the data, there is no value above 4.0

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.shoppingtrends
WHERE shipping_type IS NULL AND purchase_amount BETWEEN 30 AND 70;
```

Results (1 minute ago)

Table | Chart — 7 rows — 63ms

	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