

## Q.1

PRACTICAL3.SHOPPING ▾ Settings ▾

```
1 -- 1. Find all records where Size is missing and the purchase_amount is greater than 50.
2 --Expected Columns: Customer ID, Size, purchase_amount, Item_Purchased
3
4 | select CUSTOMER_ID,Size,purchase_amount, Item_purchased
5 | from shoping_trends
6 | where size is null and purchase_amount>50;
7
```

Results

# CUSTOMER_ID	SIZE	# PURCHASE_AMOUNT	ITEM_PURCHASED
1	null	74.0	Handbag
2	null	54.0	Jeans
3	null	88.0	Shirt
4	null	54.0	Blouse
5	null	57.0	Blouse
6	null	65.0	Sandals
7	null	54.0	Shoes
8	null	56.0	Shoes
9	null	55.0	Sneakers
10	null	84.0	Coat
11	null	96.0	Sandals

## Q.2 Using IFNULL

Databases Worksheets

PRACTICAL3.SHOPPING ▾ Settings ▾

```
8 --List the total number of purchases grouped by Season, treating NULL values as 'Unknown Season'.
9 --Expected Columns: Season, Total Purchases
10 |
11 | select count(*) as total_purchase,
12 | IFNULL(season, 'Unknown season') as season
13 | from shoping_trends
14 | group by season;
15
```

Results

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

## Q.2 Using Coalesce

Databases Worksheets

PRACTICAL3.SHOPPING Settings

```

7
8 --List the total number of purchases grouped by Season, treating NULL values as 'Unknown Season'
9 --Expected Columns: Season, Total Purchases
10
11 select count (*) as total_purchase,
12 coalesce (season, 'Unknown season') as season
13 from shoping_trends
14 group by season;
15

```

Results

# TOTAL_PURCHASE	SEASON
1	65
2	80
3	55
4	27
5	73

### Q.3 Using IFNULL

Databases Worksheets

PRACTICAL3.SHOPPING Settings

```

16 --3. Count how many customers used each Payment Method, treating NULLs as 'Not Provided'.
17 --Expected Columns: Payment Method, Customer Count
18
19 select IFNULL (payment_method,'Not provided') AS payment_method,
20 count (customer_id) as customer_count
21 from shoping_trends
22 group by payment_method;
23

```

Results

PAYMENT_METHOD	CUSTOMER_COUNT
Credit Card	44
PayPal	51
Debit Card	42
Not provided	30
Cash	42
Bank Transfer	38
Venmo	53

### Q.3 Using Coalesce

```

16 --3. Count how many customers used each Payment Method, treating NULLs as 'Not Provided'.
17 --Expected Columns: Payment Method, Customer Count
18
19 select coalesce (payment_method,'Not provided') AS payment_method,
20 count (customer_id) as customer_count
21 from shoping_trends
22 group by payment_method;
23
24 --4. Show customers where Promo Code Used is NULL and Review Rating is below 3.0.

```

Results

PAYMENT_METHOD	CUSTOMER_COUNT
Credit Card	44
PayPal	51
Debit Card	42
Not provided	30
Cash	42
Bank Transfer	38
Venmo	53

### Q.4

Databases Worksheets

PRACTICAL3.SHOPPING Settings

```

24 --4. Show customers where Promo Code Used is NULL and Review Rating is below 3.0.
25 --Expected Columns: Customer ID, Promo Code Used, Review Rating, Item Purchased
26
27 select customer_id, promo_code_used, review_rating, item_purchased
28 from shopping_trends
29 WHERE promo_code_used IS NULL AND review_rating < 3;
30
31

```

Results

CUSTOMER_ID	PROMO_CODE_USED	REVIEW_RATING	ITEM_PURCHASED
1	null	2.5	Jeans
2	null	2.6	Jeans
3	null	2.5	Jeans
4	null	2.6	Sneakers
5	null	2.8	Sneakers
6	null	2.5	Shoes
7	null	2.5	Shorts
8	null	2.9	Blouse

## Q.5 Using IfNull

Databases Worksheets

PRACTICAL3.SHOPPING Settings

```

31 --5. Group customers by Shipping Type, and return the average purchase_amount, treating missing values as 0.
32 --Expected Columns: Shipping Type, Average purchase_amount
33
34
35 select shipping_type,
36 avg (IFNULL(purchase_amount, 0)) as Average_purchase_amount
37 from shopping_trends
38 group by shipping_type;
39
40

```

Results

SHIPPING_TYPE	AVERAGE_PURCHASE_AMOUNT
Standard	47.6666667
Express	53.4545455
Store Pickup	55.3333333
null	52.7037037
Free Shipping	50.2142857
Next Day Air	54.8666667
2-Day Shipping	51.5576923

## Q.5 Using Coalesce

PRACTICAL3.SHOPPING Settings

```

32 --5. Group customers by Shipping Type, and return the average purchase_amount, treating missing values as 0.
33 --Expected Columns: Shipping Type, Average purchase_amount
34
35 select shipping_type,
36 avg (COALESCE(purchase_amount, 0)) as Average_purchase_amount
37 from shopping_trends
38 group by shipping_type;
39
40 --6. Display the number of purchases per Location only for those with more than 5 purchases and no NULL Payment Method.
41
42

```

Results

SHIPPING_TYPE	AVERAGE_PURCHASE_AMOUNT
Standard	47.6666667
Express	53.4545455
Store Pickup	55.3333333
null	52.7037037
Free Shipping	50.2142857
Next Day Air	54.8666667
2-Day Shipping	51.5576923

## Q.6

PRACTICAL3.SHOPPING

```

40 --> 40. Display the number of purchases per Location only for those with more than 5 purchases and no NULL Payment Method.
41 --Expected Columns: Location, Total Purchases
42
43 Select location,
44 count(*) as total_purchases
45 from shoping_trends
46 WHERE PAYMENT_METHOD IS not null
47 group by location
48 Having total_purchases>5;

```

**Results**

LOCATION	TOTAL_PURCHASES
Maine	41
Kentucky	30
null	24
New York	31
Oregon	30
Rhode Island	29
Florida	32
Massachusetts	31
Texas	22

## Q.7 Using IFNULL

PRACTICAL3.SHOPPING

```

50 --7. Create a column Spender Category that classifies customers using CASE:
51 --"High" if amount > 80, "Medium" if BETWEEN 50 AND 80,
52 --"Low" otherwise. Replace NULLs in purchase_amount with 0.
53 --Expected Columns: Customer ID, purchase_amount, Spender Category
54
55 SELECT customer_id,
56 IFNULL (purchase_amount,0) AS PURCHASE_AMOUNT,
57 Case
58 when purchase_amount>80 THEN 'High'
59 when purchase_amount between 50 And 80 Then 'Medium'
60 else 'Low'
61 end as spender_category
from shoping_trends;

```

**Results**

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

## Q.7 Using Coalesce

PRACTICAL3.SHOPPING ▾     Settings ▾

Open in Workspaces     Code Versions    

```

50  --7. Create a column Spender Category that classifies customers using CASE:
51  --'High' if amount > 80, 'Medium' if BETWEEN 50 AND 80,
52  --'Low' otherwise. Replace NULLs in purchase_amount with 0.
53  --Expected Columns: Customer ID, purchase_amount, Spender Category
54  SELECT customer_id,
55    coalesce (purchase_amount,0) AS PURCHASE_AMOUNT,
56    Case
57    when purchase_amount>80 THEN 'High'
58    when purchase_amount between 50 And 80 Then 'Medium'
59    else 'Low'
60  end as spender_category
61  from shoping_trends;
62

```

↳ Results    ↵ Chart   

# 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	20.0 Low

## Q.8

Databases   Worksheets

Search objects

PRACTICAL1   PRACTICAL2   PRACTICAL3   RETAIL   SNOWFLAKE   SNOWFLAKE\_LEARNING\_DB   SNOWFLAKE\_SAMPLE\_DATA   STORE

PRACTICAL3.SHOPPING ▾     Settings ▾

ACCOUNTADMIN + COMPUTE\_WH (X-Small)     Share    

Open in Workspaces     Code Versions    

```

62  --08.Find customers who have no Previous Purchases value but whose Color is not NULL.
63  --Expected Columns: Customer ID, Color, Previous Purchases
64
65  select customer_id,
66    color,
67    previous_purchases,
68    from shoping_trends
69
70  where color is not null AND previous_purchases is 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

## Q.9

Databases Worksheets

Search objects

```

    PRACTICAL3.SHOPPING Settings ▾
    71 --9. Group records by Frequency of
    72 --Purchases and show the total amount spent per group, treating NULL frequencies as 'Unknown'.
    73 --Expected Columns: Frequency of Purchases, Total purchase_amount
    74
    75 select ifnull (frequency_of_purchases,'Unknown'),
    76 sum(purchase_amount) as total_purchase_amount
    77 from shopping_trends
    78 group by frequency_of_purchases;
    79
  
```

Results Chart

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

## Q.10

Databases Worksheets

Search objects

```

    PRACTICAL3.SHOPPING Settings ▾
    80 --10.Display a list of all Category values with the number of times each was purchased,
    81 --excluding rows where Category is NULL.
    82 --Expected Columns: Category, Total Purchases
    83
    84 select category,
    85 count(*) total_purchases
    86 from shopping_trends
    87 where category is not null
    88 group by category;
    89
  
```

Results Chart

1	Outerwear	60
2	Footwear	70
3	Clothing	59
4	Accessories	78

## Q.11

Databases Worksheets

Search objects

```

    PRACTICAL3.SHOPPING Settings ▾
    89
    90 --11.Return the top5 Locations with the highest total purchase_amount, replacing NULLs in amount with 0.
    91 --Expected Columns: Location, Total purchase_amount
    92
    93 select top 5 location,
    94 sum(ifnull (purchase_amount, 0)) as Total_purchase_amount
    95 from shopping_trends
    96 group by location
    97 order by Total_purchase_amount desc;;
  
```

Results Chart

1	Maine	2294.0
2	Florida	1980.0
3	Massachusetts	1899.0
4	Rhode Island	1876.0
5	Kentucky	1798.0

## Q.12

Databases Worksheets

PRACTICAL3.SHOPPING Settings

```
98 --12.Group customers by Gender and Size, and count how many entries have a NULL Color.
99 --Expected Columns: Gender, Size, Null Color Count
100
101
102 select gender, size,
103 count(*) as Null_color_count
104 from shoping_trends
105 Where color is null
106 group by gender, size;
```

Results

GENDER	SIZE	NULL_COLOR_COUNT
Male	null	6
Male	M	7
Male	L	6
Male	S	5
Male	XL	5

## Q13.

Databases Worksheets

PRACTICAL3.SHOPPING Settings

```
107
108 --13.Identify all Item Purchased where more than 3 purchases had NULL Shipping Type.
109 --Expected Columns: Item Purchased, NULL Shipping Type Count
110
111
112 select item_purchased,
113 count(*) as Null_shipping_Type_Count
114 from shoping_trends
115 WHERE shipping_type is null
116 group by item_purchased
117 Having Null_shipping_Type_Count >3;
118
```

Results

ITEM_PURCHASED	NULL_SHIPPING_TYPE_COUNT
null	4
Shirt	5
Shoes	4

## Q.14

Databases Worksheets

PRACTICAL3.SHOPPING Settings

```
115
116 group by item_purchased
117 Having Null_shipping_Type_Count >3;
118
119 --14.Show a count of how many customers per Payment Method have NULL Review Rating.
120 --Expected Columns: Payment Method, Missing Review Rating Count
121
122 select payment_method,
123 count (*) as missing_review_rating_count
124 from shoping_trends
125 Where review_rating is null
126 group by payment_method;
127
```

Results

PAYMENT_METHOD	MISSING REVIEW RATING COUNT
Credit Card	8
Cash	4
null	2
Debit Card	7
Venmo	9
PayPal	3
Bank Transfer	4

## Q.15 Using ifnull

The screenshot shows the Snowflake SQL interface. The top navigation bar includes tabs for Onlineandstoresales, 2025-10-15 10:10pm, Practical1, Practical2, Practical3, and a plus sign for creating new workspaces. The current workspace is Practical3. The sidebar on the left lists databases like PRACTICAL1, PRACTICAL2, PRACTICAL3, RETAIL, SNOWFLAKE, SNOWFLAKE\_LEARNING\_DB, SNOWFLAKE\_SAMPLE\_DATA, and STORE. The main area displays a query in the PRACTICAL3.SHOPPING schema:

```
126 --15.Group by Category and return the average Review Rating, replacing NULLs with 0.
127 --and filter only where average is greater than 3.5.
128 --Expected Columns: Category, Average Review Rating
129
130 select category,
131     avg(ifnull(review_rating, 0)) as average_review_rating
132     from shopping_trends
133     group by category
134 Having average_review_rating>3.5;
135
```

The results pane shows a single row with the message "Query produced no results".

## Q.15 Using Coalesce

The screenshot shows the Snowflake SQL interface. The top navigation bar includes tabs for Onlineandstoresales, 2025-10-15 10:10pm, Practical1, Practical2, Practical3, and a plus sign for creating new workspaces. The current workspace is Practical3. The sidebar on the left lists databases like PRACTICAL1, PRACTICAL2, PRACTICAL3, RETAIL, SNOWFLAKE, SNOWFLAKE\_LEARNING\_DB, SNOWFLAKE\_SAMPLE\_DATA, and STORE. The main area displays a query in the PRACTICAL3.SHOPPING schema:

```
125
126 --15.Group by Category and return the average Review Rating, replacing NULLs with 0,
127 --and filter only where average is greater than 3.5.
128 --Expected Columns: Category, Average Review Rating
129
130 select category,
131     avg(coalesce(review_rating, 0)) as average_review_rating
132     from shopping_trends
133     group by category
134 Having average_review_rating>3.5;
135
```

The results pane shows a single row with the message "Query produced no results".

## Q.16

The screenshot shows the Snowflake SQL interface. The top navigation bar includes tabs for Onlineandstoresales, 2025-10-15 10:10pm, Practical1, Practical2, Practical3, and a plus sign for creating new workspaces. The current workspace is Practical3. The sidebar on the left lists databases like PRACTICAL1, PRACTICAL2, PRACTICAL3, RETAIL, SNOWFLAKE, SNOWFLAKE\_LEARNING\_DB, SNOWFLAKE\_SAMPLE\_DATA, and STORE. The main area displays a query in the PRACTICAL3.SHOPPING schema:

```
135
136 --16. List all Colors that are missing (NULL) in at least 2 rows and the average Age of customers for those rows.
137 --Expected Columns: Color, Average Age
138
139 select color,
140     avg(age) as average_age
141     from shopping_trends
142     where color is null
143     group by color
144 Having average_age>=2;
145
```

The results pane shows a single row with the message "Query produced no results".

## Q.17

Databases Worksheets

PRACTICAL3.SHOPPING Settings

```
145 --17.Use CASE to create a column Delivery Speed: 'Fast' if Shipping Type is 'Express' or
146 --'Next Day Air', 'Slow' if 'Standard',
147 --'Other' for all else including NULL. Then count how many customers fall into each category,
148 -- Expected Columns: Delivery Speed, Customer Count
149 select
150 case
151 when shipping_type='Express' or shipping_type='Next Day Air' then 'fast'
152 when shipping_type = 'Standard' then 'slow'
153 else 'other'
154 end as delivery_speed,
155 count (customer_id) as customer_count
156 from shopping_trends
157 Group by delivery_speed;
```

Results

DELIVERY_SPEED	CUSTOMER_COUNT
other	166
slow	45
fast	89

Query Details

- Query duration: 612ms
- Rows: 3
- Query ID: 01bfe285-000c-b21f-00...

DELIVERY\_SPEED

## Q.18

Databases Worksheets

PRACTICAL3.SHOPPING Settings

```
158 --18.Find customers whose purchase_amount is NULL and whose Promo Code Used is 'Yes'.
159 --Expected Columns: Customer ID, purchase_amount, Promo Code Used
160
161 select customer_id,
162 purchase_amount,
163 promo_code_used
164 from shopping_trends
165 where purchase_amount is null and promo_code_used = 'Yes';
166
```

Results

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

## Q.19

Databases Worksheets

PRACTICAL3.SHOPPING Settings

```

167 --19.Group by Location and show the maximum Previous
168 --Purchases, replacing NULLs with 0, only where the average rating is above 4.0.
169 --Expected Columns: Location, Max Previous Purchases, Average Review Rating
170 | Select location,
171 | Max(IFNULL(previous_purchases,0)) as Max_previous_purchases,
172 | AVG(Review_rating) as average_review_rating
173 | from shoping_trends
174 | group by location
175 | Having average_review_rating>4;

```

↳ Results ↵ Chart

LOCATION	MAX_PREVIOUS_PURCHASES	AVERAGE REVIEW RATING
Query produced no results		

## Q.20

Databases Worksheets

PRACTICAL3.SHOPPING Settings

```

176 --20.Show customers who have a NULL Shipping
177 --Type but made a purchase in the range of 30 to 70 USD.
178 --Expected Columns: Customer ID, Shipping
179 --Type, purchase_amount, Item Purchased
180 | Select Customer_id, shipping_Type,purchase_amount, item_purchased
181 | from shoping_trends
182 | Where shipping_type is null and purchase_amount between 30 and 70;
183 |
184 |

```

↳ Results ↵ Chart

# 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