



# BrightLight Data Analytics Coding Practical

## Practical 2.1: Advanced SQL

1. Find all records where Size is missing and the purchase\_amount is greater than 50.

```
1  --1
2  select customer_id,
3  size,
4  purchase_amount,
5  item_purchased
6  from shopping_trends
7  Where size is NULL AND purchase_amount>50;
8
```

Results Chart

	# CUSTOMER_ID	A SIZE	# PURCHASE_AMOUNT	A 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
5	62	null	57.0	Blouse
6	73	null	65.0	Sandals
7	91	null	54.0	Shoes
8	97	null	56.0	Shoes
9	100	null	55.0	Sneakers
10	160	null	84.0	Coat

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

```
9      --2
10     SELECT
11         COALESCE(Season, 'Unknown Season') AS Season,
12         COUNT(*) AS total_purchases
13     FROM
14         shopping_trends
15     GROUP BY
16         COALESCE(Season, 'Unknown Season');
```

Results Chart

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

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

```
20 SELECT
21     COALESCE(payment_method, 'Not Provided') AS "Payment Method",
22     COUNT(DISTINCT customer_id) AS "Customer Count"
23 FROM
24     shopping_trends
25 GROUP BY
26     COALESCE(payment_method, 'Not Provided');
```

Results Chart

	Payment Method	Customer Count
1	PayPal	51
2	Cash	42
3	Credit Card	44
4	Bank Transfer	38
5	Not Provided	30
6	Venmo	53
7	Debit Card	42

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

```
29 SELECT
30     customer_id,
31     promo_code_used,
32     review_rating,
33     item_purchased
34 FROM
35     shopping_trends
36 WHERE
37     promo_code_used IS NULL
38     AND review_rating < 3.0;
```

Results Chart

	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.

```
41 SELECT
42     shipping_type,
43     AVG(COALESCE(purchase_amount, 0)) AS "Average purchase_amount"
44 FROM
45     shopping_trends
46 GROUP BY
47     shipping_type;
```

-- 6  
SELECT

Results Chart

	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

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

```
50 SELECT
51     location,
52     COUNT(*) AS "Total Purchases"
53 FROM
54     shopping_trends
55 WHERE
56     payment_method IS NOT NULL
57 GROUP BY
58     location
59 HAVING
60     COUNT(*) > 5;
```

Results Chart

	LOCATION	Total Purchases
1	null	24
2	Maine	41
3	Oregon	30
4	Kentucky	30
5	Florida	32
6	Massachusetts	31

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.

```
64 SELECT
65     customer_id,
66     COALESCE(purchase_amount, 0) AS "purchase_amount",
67     CASE
68         WHEN COALESCE(purchase_amount, 0) > 80 THEN 'High'
69         WHEN COALESCE(purchase_amount, 0) BETWEEN 50 AND 80 THEN 'Medium'
70         ELSE 'Low'
71     END AS "Spender Category"
72 FROM
73     shopping_trends;
74
```

	# 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

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

```
76 SELECT
77     customer_id,
78     color,
79     previous_purchases
80 FROM
81     shopping_trends
82 WHERE
83     previous_purchases IS NULL
84     AND color IS NOT NULL;
85
```

	# 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	30	White	null

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

```
86 --9
87 SELECT
88     COALESCE(frequency_of_purchases, 'Unknown') AS "Frequency of Purchases",
89     SUM(COALESCE(purchase_amount, 0)) AS "Total purchase_amount"
90 FROM
91     shopping_trends
92 GROUP BY
93     COALESCE(frequency_of_purchases, 'Unknown');
94
```

	Frequency of Purchases	Total purchase_amount
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	Estimated	2022.0

10. Display a list of all Category values with the number of times each was purchased, excluding rows where Category is NULL.

```
96 SELECT
97     category,
98     COUNT(*) AS "Total Purchases"
99 FROM
100     shopping_trends
101 WHERE
102     category IS NOT NULL
103 GROUP BY
104     category;
105
```

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

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

```
107 --11
108 SELECT
109     location,
110     SUM(COALESCE(purchase_amount, 0)) AS "Total purchase_amount"
111 FROM
112     shopping_trends
113 GROUP BY
114     location
115 ORDER BY
116     "Total purchase_amount" DESC
117 LIMIT 5;
```

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

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

```
120 SELECT
121     gender,
122     size,
123     COUNT(*) AS "Null Color Count"
124 FROM
125     shopping_trends
126 WHERE
127     color IS NULL
128 GROUP BY
129     gender, size;
```

	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



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

```
133 SELECT
134     item_purchased,
135     COUNT(*) AS "NULL Shipping Type Count"
136 FROM
137     shopping_trends
138 WHERE
139     shipping_type IS NULL
140 GROUP BY
141     item_purchased
142 HAVING
143     COUNT(*) > 3;
144
```

	ITEM_PURCHASED	# NULL Shipping Type Count
1	Shirt	5
2	null	4
3	Shoes	4

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

```
145 --14
146 SELECT
147     payment_method,
148     COUNT(DISTINCT customer_id) AS "Missing Review Rating Count"
149 FROM
150     shopping_trends
151 WHERE
152     review_rating IS NULL
153 GROUP BY
154     payment_method;
155
```

	PAYMENT_METHOD	# Missing Review Rating Count
1	Bank Transfer	4
2	PayPal	3
3	null	2
4	Credit Card	8
5	Venmo	9

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

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

```
168 SELECT
169     color,
170     AVG(age) AS "Average Age"
171 FROM
172     shopping_trends
173 WHERE
174     color IS NULL
175 GROUP BY
176     color
177 HAVING
178     COUNT(*) >= 2;
179
```

Results		Chart	🔍	🗑️	⬇️	📄	🕒
	📄 COLOR	# Average Age					
1	null	47.8461538					

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.

```
181 SELECT
182     CASE
183         WHEN shipping_type IN ('Express', 'Next Day Air') THEN 'Fast'
184         WHEN shipping_type = 'Standard' THEN 'Slow'
185         ELSE 'Other'
186     END AS "Delivery Speed",
187     COUNT(DISTINCT customer_id) AS "Customer Count"
188 FROM
189     shopping_trends
190 GROUP BY
191     CASE
192         WHEN shipping_type IN ('Express', 'Next Day Air') THEN 'Fast'
193         WHEN shipping_type = 'Standard' THEN 'Slow'
194         ELSE 'Other'
195     END;
196
```

Results		Chart	🔍	🗑️	⬇️	📄	🕒
	📄 Delivery Speed	# Customer Count					
1	Slow	45					
2	Fast	89					
3	Other	166					

18. Find customers whose purchase\_amount is NULL and whose Promo Code Used is 'Yes'.

```
198 SELECT
199     customer_id,
200     purchase_amount,
201     promo_code_used AS "Promo Code Used"
202 FROM
203     shopping_trends
204 WHERE
205     purchase_amount IS NULL
206     AND promo_code_used = 'Yes';
207
```

Results Chart

	# CUSTOMER_ID	# PURCHASE_AMOUNT	0 1 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

19. Group by Location and show the maximum Previous Purchases, replacing NULLs with 0, only where the average rating is above 4.0.

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

221

222

223

224

225

226

227

228

229

230

231

SELECT

customer\_id AS "Customer ID",

shipping\_type AS "Shipping Type",

purchase\_amount,

item\_purchased AS "Item Purchased"

FROM

shopping\_trends

WHERE

shipping\_type IS NULL

AND purchase\_amount BETWEEN 30 AND 70;

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