

DAY 4 SQL Assignment

PostgreSQL

1. GROUP BY with WHERE - Orders by Year and Quarter..... 1
2. GROUP BY with HAVING - High Volume Ship Regions..... 2
3. Get all title designations across employees and customers (Try UNION & UNION ALL)..... 3
4. Find categories that have both discontinued and in-stock products 5
5. Find orders that have no discounted items (Display the order_id, EXCEPT) 6

1. GROUP BY with WHERE - Orders by Year and Quarter

Display, order year, quarter, order count, avg freight cost only for those orders where freight cost > 100

QUERY:

SELECT

EXTRACT(YEAR FROM ORDER_DATE) AS ORDER_YEAR,
EXTRACT(QUARTER FROM ORDER_DATE) AS QUARTER,
COUNT(*) AS ORDERCOUNT,
AVG(FREIGHT) AS FREIGHT

FROM ORDERS

WHERE FREIGHT > 100

GROUP BY EXTRACT(YEAR FROM ORDER_DATE),

EXTRACT(QUARTER FROM ORDER_DATE)

ORDER BY

ORDER_YEAR, QUARTER;

OUTPUT:

Data Output

Messages

Notifications

≡

📄

▼

📋

▼

🗑️

🗄️

⬇️

📈

SQL

	order_year numeric 🔒	quarter numeric 🔒	ordercount bigint 🔒	freight double precision 🔒
1	1996	3	12	163.842498143514
2	1996	4	22	203.84591015902433
3	1997	1	14	232.64714213779993
4	1997	2	21	269.5480975196475
5	1997	3	26	236.4411541865422
6	1997	4	33	206.7539388483221
7	1998	1	43	245.48162770825763
8	1998	2	16	292.79250144958496

Total rows: 8

Query complete 00:00:00.922

2. GROUP BY with HAVING - High Volume Ship Regions

Display, ship region, no of orders in each region, min and max freight cost
Filter regions where no of orders ≥ 5

QUERY:

SELECT

SHIP_REGION,

COUNT(ORDER_ID) AS ORDER_COUNT,

MIN(FREIGHT) AS MIN_FREIGHT,

MAX(FREIGHT) AS MAX_FREIGHT

FROM ORDERS

GROUP BY SHIP_REGION

HAVING

COUNT(ORDER_ID) ≥ 5 AND SHIP_REGION IS NOT NULL

ORDER BY ORDER_COUNT DESC

LIMIT 10;

OUTPUT:

Data Output Messages Notifications				
<div> <div>≡+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗄️</div> <div>⬇️</div> <div>📈</div> <div>SQL</div> </div>				
	ship_region character varying (15) 🔒	order_count bigint 🔒	min_freight real 🔒	max_freight real 🔒
1	SP	49	0.14	890.78
2	RJ	34	2.27	193.37
3	ID	31	8.19	830.75
4	OR	28	0.2	719.78
5	Co. Cork	19	16.74	603.54
6	WA	19	4.56	606.19
7	NM	18	8.53	708.95
8	Táchira	18	2.08	210.19
9	BC	17	0.94	243.73
Total rows: 10		Query complete 00:00:00.107		

- Get all title designations across employees and customers (Try UNION & UNION ALL)

QUERY:

SELECT TITLE AS DESIGNATION

FROM EMPLOYEES

UNION

SELECT CONTACT_TITLE AS DESIGNATION

FROM CUSTOMERS;

SELECT TITLE AS DESIGNATION








FROM EMPLOYEES

UNION ALL

SELECT CONTACT_TITLE AS DESIGNATION

FROM CUSTOMERS;

OUTPUT:

Data Output		Messages	Notifications
			
			SQL
designation			
character varying (30)			
1	Owner		
2	Sales Associate		
3	Sales Agent		
4	Inside Sales Coordinator		
5	Marketing Manager		
6	Assistant Sales Agent		
7	Assistant Sales Representative		
8	Accounting Manager		
9	Vice President, Sales		
Total rows: 14		Query complete 00:00:00.093	

Data Output		Messages	Notifications
<div> <div>≡+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🗄️</div> <div>⬇️</div> <div>📈</div> <div>SQL</div> </div>			
	designation character varying (30)		
1	Sales Representative		
2	Vice President, Sales		
3	Sales Representative		
4	Sales Representative		
5	Sales Manager		
6	Sales Representative		
7	Sales Representative		
8	Inside Sales Coordinator		
9	Sales Representative		
Total rows: 100		Query complete 00:00:00.178	

4. Find categories that have both discontinued and in-stock products

(Display category_id, instock means units_in_stock > 0, Intersect)

QUERY:

```

SELECT category_id
FROM products
WHERE units_in_stock > 0
INTERSECT
SELECT category_id
FROM products
WHERE discontinued = 1
ORDER BY category_id

```


OUTPUT:

Data Output


Messages

Notifications


≡+





▼




▼










SQL

	category_id  smallint
1	1
2	2
3	5
4	6
5	7

Total rows: 5

Query complete 00:00:00.115

5. Find orders that have no discounted items (Display the order_id, EXCEPT)

QUERY:

```
SELECT order_id
FROM order_details
EXCEPT
SELECT order_id
FROM order_details
WHERE discount != 0
ORDER BY order_id;
```

OUTPUT:

Data Output Messages Notifications



	order_id smallint
1	10782
2	10725
3	10423
4	10518
5	10356
6	10282
7	10283
8	10579
9	10896

Total rows: 450

Query complete 00:00:00.078