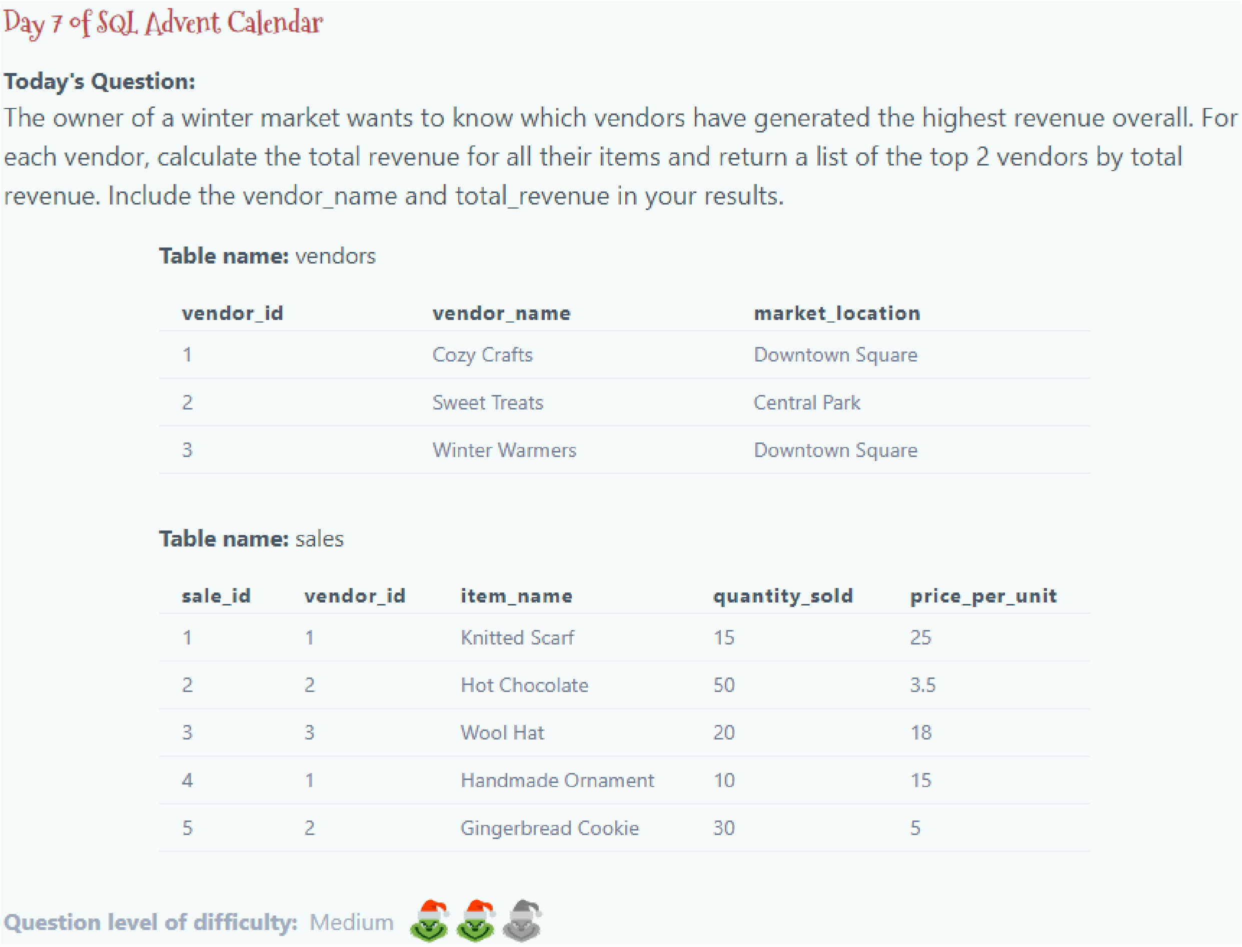
SQL Advent Calendar 2024 - IInd Quarter



Day 7 of SQL Advent Calendar

Today's Question:
The owner of a winter market wants to know which vendors have generated the highest revenue overall. For each vendor, calculate the total revenue for all their items and return a list of the top 2 vendors by total revenue. Include the vendor_name and total_revenue in your results.

Table name: vendors

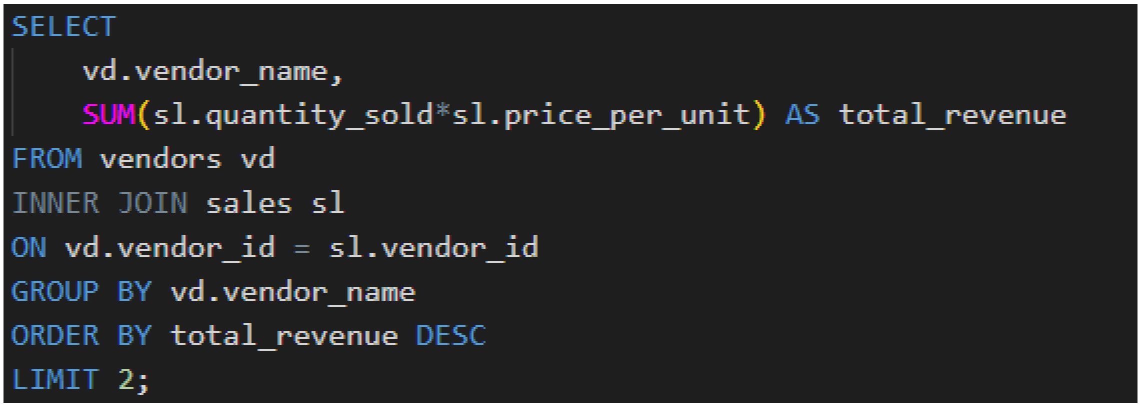
vendor_id	vendor_name	market_location
1	Cozy Crafts	Downtown Square
2	Sweet Treats	Central Park
3	Winter Warmers	Downtown Square

Table name: sales

sale_id	vendor_id	item_name	quantity_sold	price_per_unit
1	1	Knitted Scarf	15	25
2	2	Hot Chocolate	50	3.5
3	3	Wool Hat	20	18
4	1	Handmade Ornament	10	15
5	2	Gingerbread Cookie	30	5

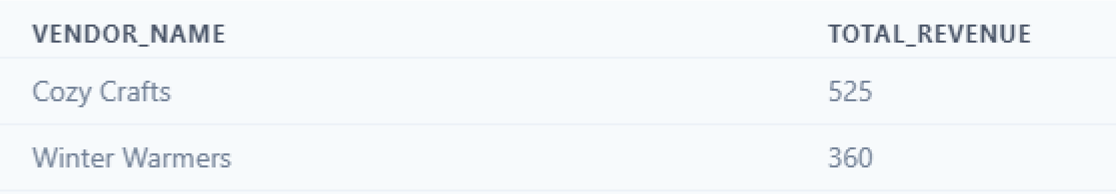
Question level of difficulty: Medium

Code:

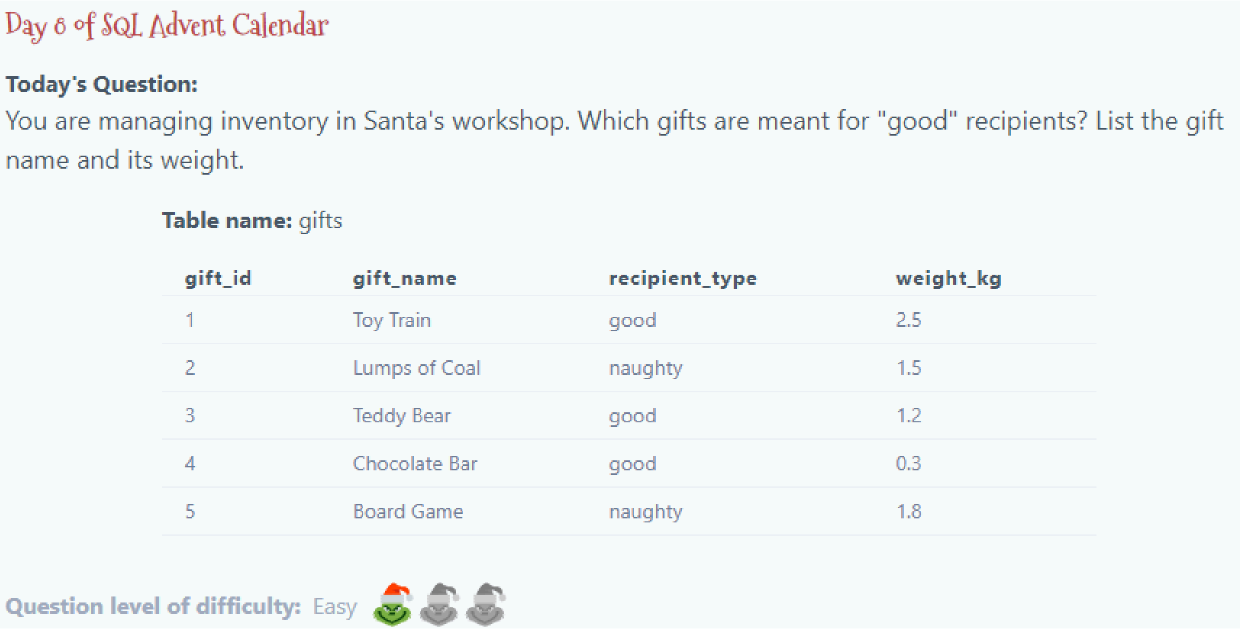


```
SELECT
    vd.vendor_name,
    SUM(sl.quantity_sold*sl.price_per_unit) AS total_revenue
FROM vendors vd
INNER JOIN sales sl
ON vd.vendor_id = sl.vendor_id
GROUP BY vd.vendor_name
ORDER BY total_revenue DESC
LIMIT 2;
```

Output:



VENDOR_NAME	TOTAL_REVENUE
Cozy Crafts	525
Winter Warmers	360



Day 6 of SQL Advent Calendar

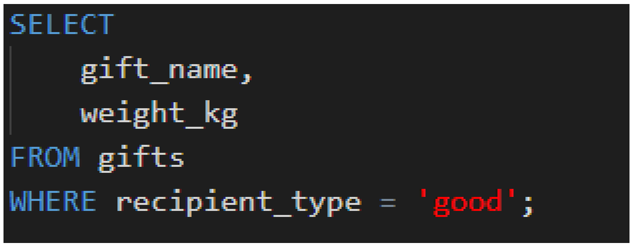
Today's Question:
You are managing inventory in Santa's workshop. Which gifts are meant for "good" recipients? List the gift name and its weight.

Table name: gifts

gift_id	gift_name	recipient_type	weight_kg
1	Toy Train	good	2.5
2	Lumps of Coal	naughty	1.5
3	Teddy Bear	good	1.2
4	Chocolate Bar	good	0.3
5	Board Game	naughty	1.8

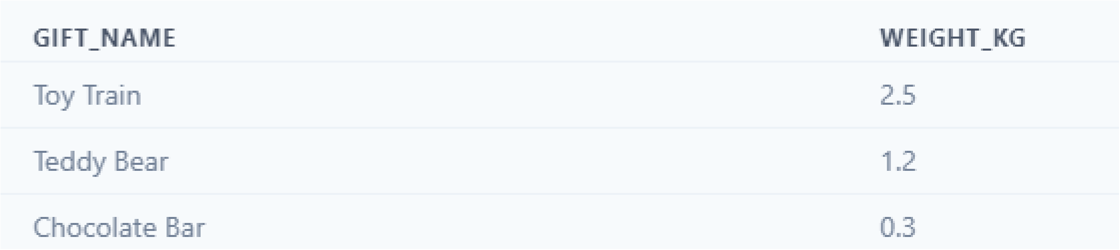
Question level of difficulty: Easy

Code:



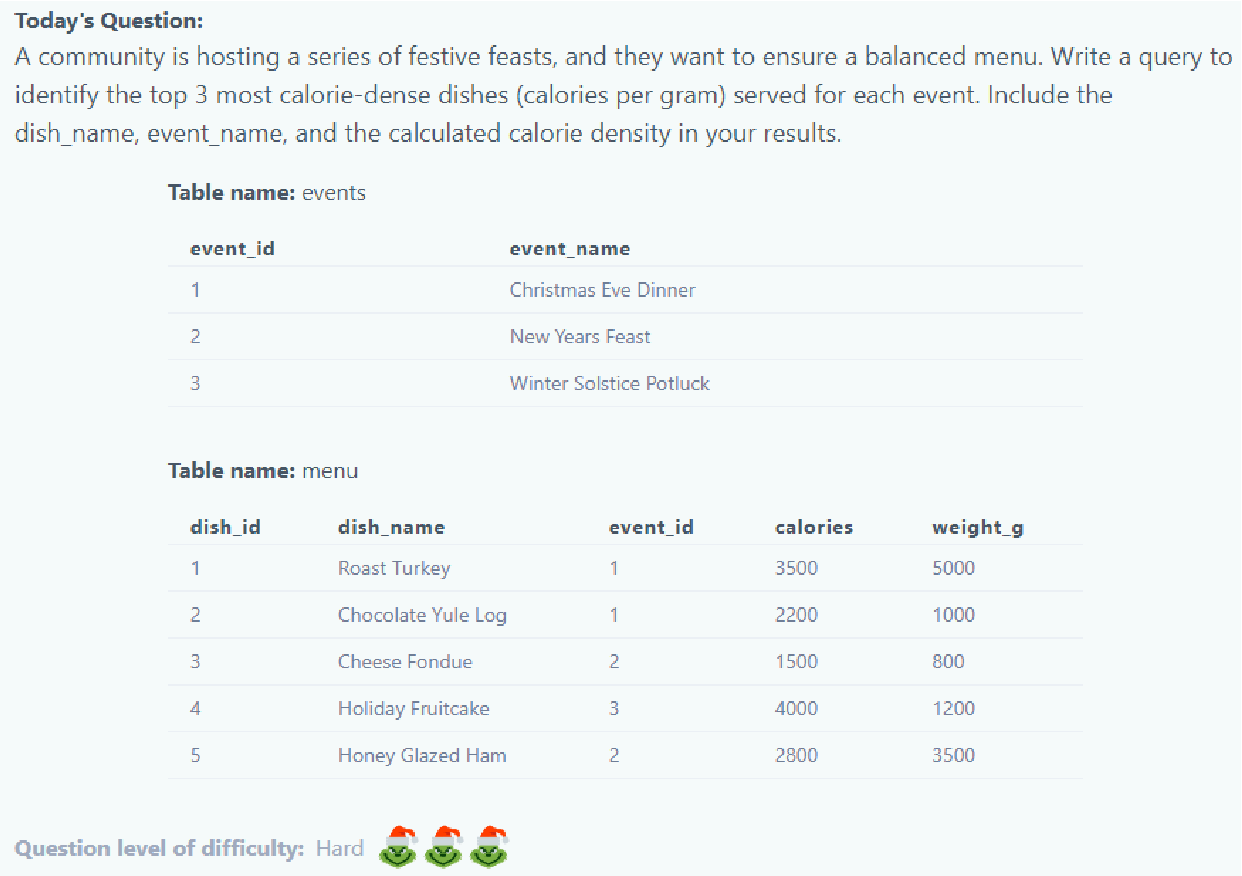
```sql
SELECT
    gift_name,
    weight_kg
FROM gifts
WHERE recipient_type = 'good';
```

Output:



GIFT_NAME	WEIGHT_KG
Toy Train	2.5
Teddy Bear	1.2
Chocolate Bar	0.3

DAY 9:



Today's Question:
A community is hosting a series of festive feasts, and they want to ensure a balanced menu. Write a query to identify the top 3 most calorie-dense dishes (calories per gram) served for each event. Include the dish_name, event_name, and the calculated calorie density in your results.

Table name: events

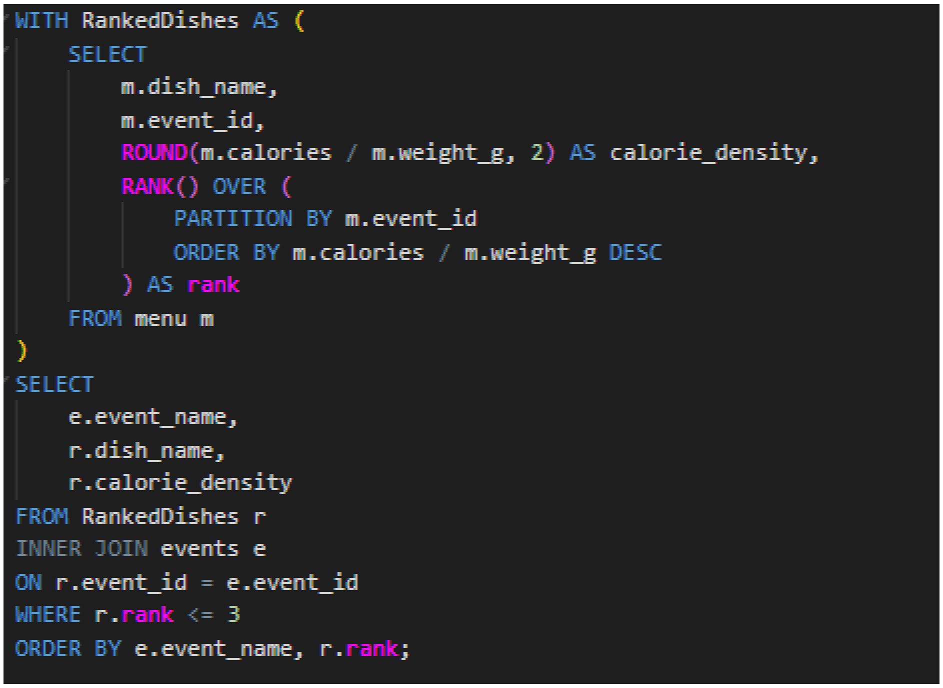
event_id	event_name
1	Christmas Eve Dinner
2	New Years Feast
3	Winter Solstice Potluck

Table name: menu

dish_id	dish_name	event_id	calories	weight_g
1	Roast Turkey	1	3500	5000
2	Chocolate Yule Log	1	2200	1000
3	Cheese Fondue	2	1500	800
4	Holiday Fruitcake	3	4000	1200
5	Honey Glazed Ham	2	2800	3500

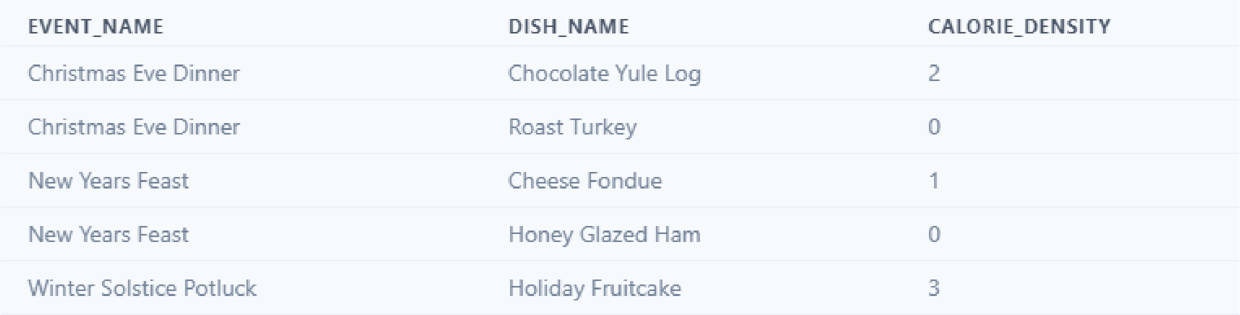
Question level of difficulty: Hard

Code:

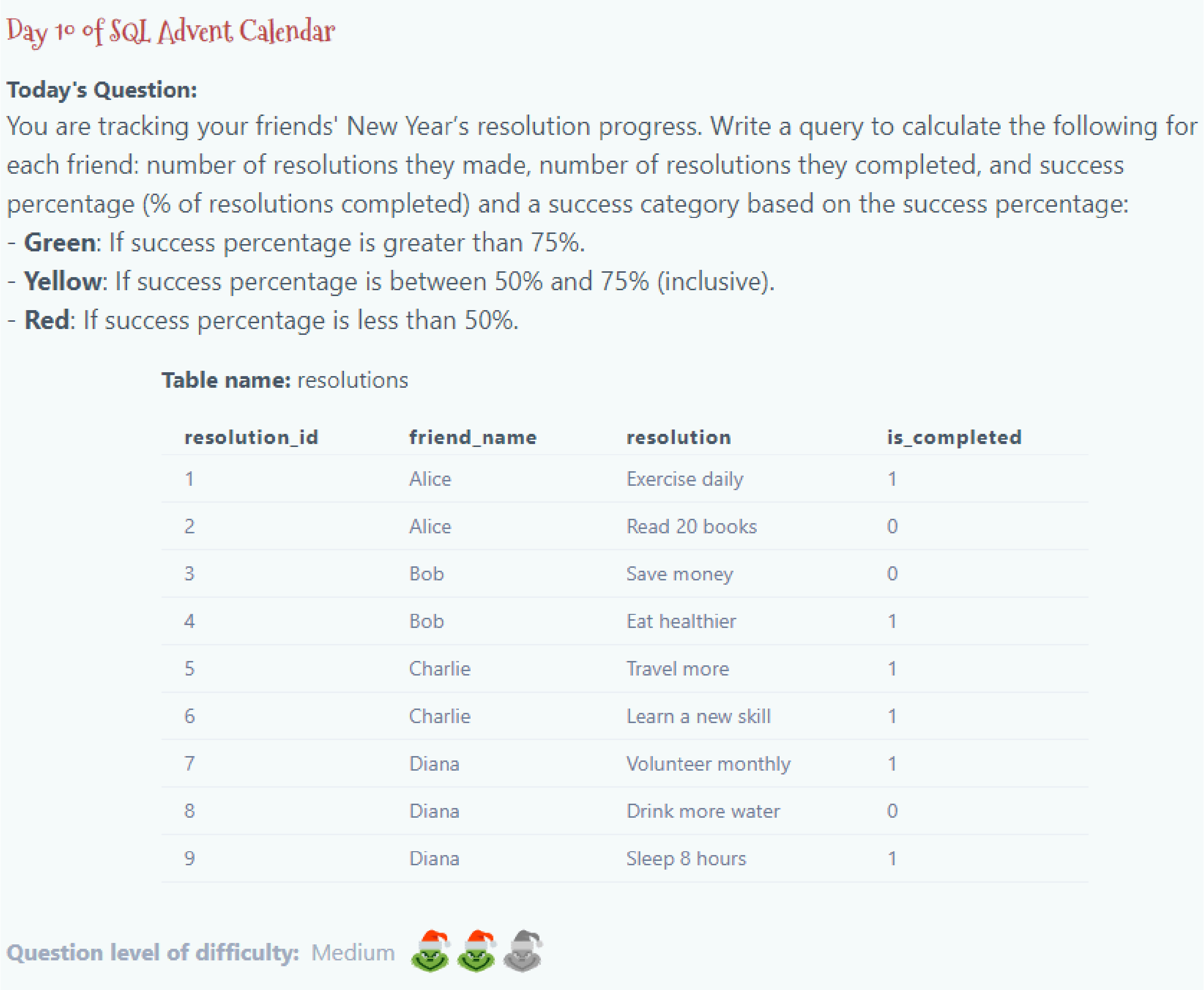


```sql
WITH RankedDishes AS (
    SELECT
        m.dish_name,
        m.event_id,
        ROUND(m.calories / m.weight_g, 2) AS calorie_density,
        RANK() OVER (
            PARTITION BY m.event_id
            ORDER BY m.calories / m.weight_g DESC
        ) AS rank
    FROM menu m
)
SELECT
    e.event_name,
    r.dish_name,
    r.calorie_density
FROM RankedDishes r
INNER JOIN events e
ON r.event_id = e.event_id
WHERE r.rank <= 3
ORDER BY e.event_name, r.rank;
```

Output:



EVENT_NAME	DISH_NAME	CALORIE_DENSITY
Christmas Eve Dinner	Chocolate Yule Log	2
Christmas Eve Dinner	Roast Turkey	0
New Years Feast	Cheese Fondue	1
New Years Feast	Honey Glazed Ham	0
Winter Solstice Potluck	Holiday Fruitcake	3



Day 10 of SQL Advent Calendar

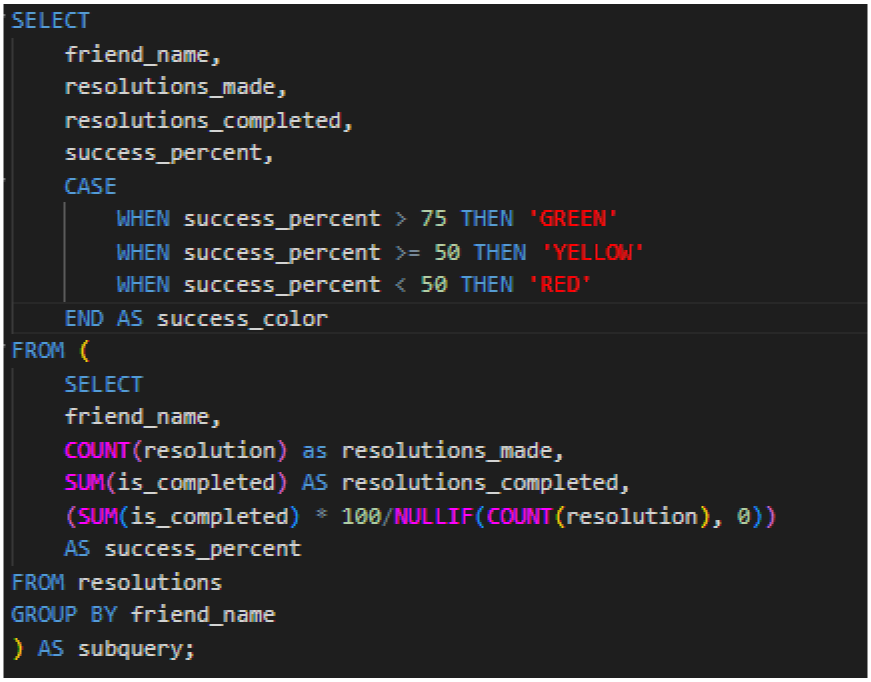
Today's Question:
You are tracking your friends' New Year’s resolution progress. Write a query to calculate the following for each friend: number of resolutions they made, number of resolutions they completed, and success percentage (% of resolutions completed) and a success category based on the success percentage:
- **Green**: If success percentage is greater than 75%.
- **Yellow**: If success percentage is between 50% and 75% (inclusive).
- **Red**: If success percentage is less than 50%.

Table name: resolutions

resolution_id	friend_name	resolution	is_completed
1	Alice	Exercise daily	1
2	Alice	Read 20 books	0
3	Bob	Save money	0
4	Bob	Eat healthier	1
5	Charlie	Travel more	1
6	Charlie	Learn a new skill	1
7	Diana	Volunteer monthly	1
8	Diana	Drink more water	0
9	Diana	Sleep 8 hours	1

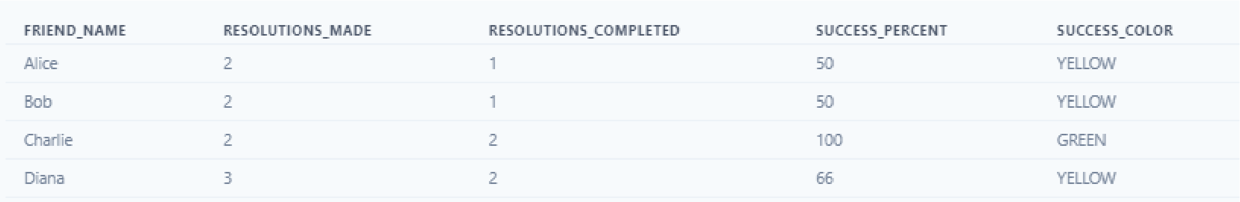
Question level of difficulty: Medium

Code:

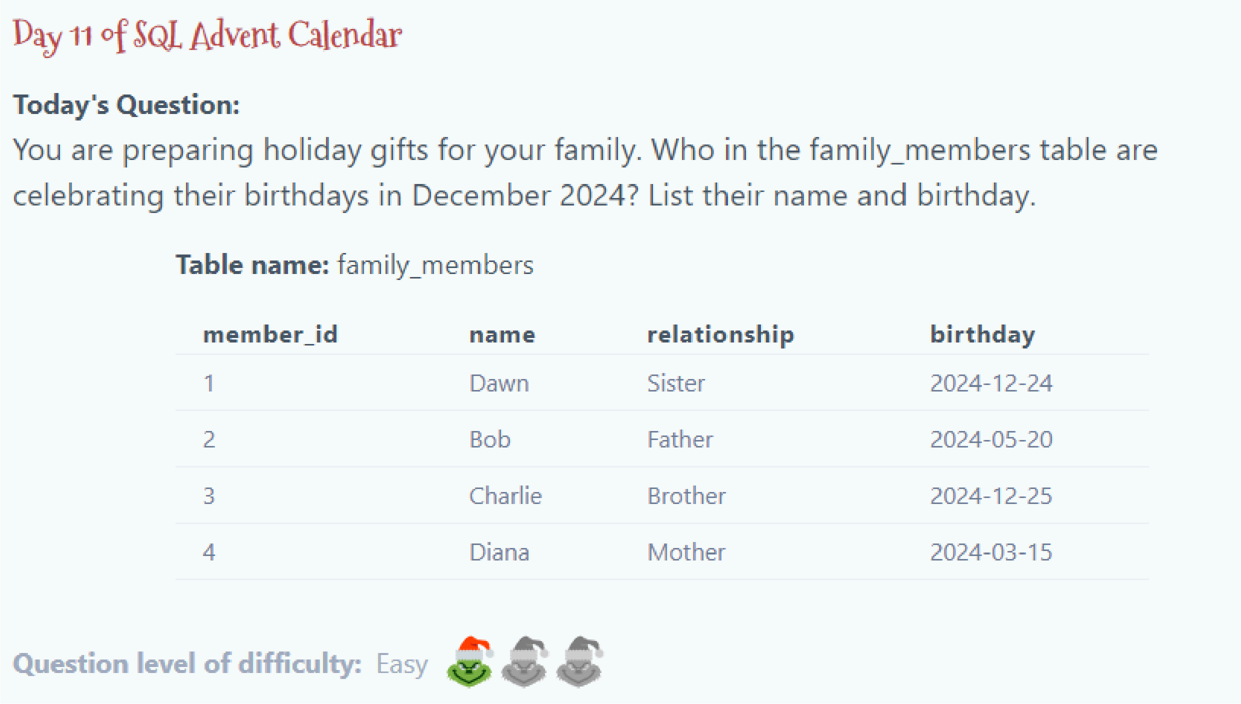


```
SELECT
    friend_name,
    resolutions_made,
    resolutions_completed,
    success_percent,
    CASE
        WHEN success_percent > 75 THEN 'GREEN'
        WHEN success_percent >= 50 THEN 'YELLOW'
        WHEN success_percent < 50 THEN 'RED'
    END AS success_color
FROM (
    SELECT
    friend_name,
    COUNT(resolution) as resolutions_made,
    SUM(is_completed) AS resolutions_completed,
    (SUM(is_completed) * 100/NULLIF(COUNT(resolution), 0))
    AS success_percent
FROM resolutions
GROUP BY friend_name
) AS subquery;
```

Output:



FRIEND_NAME	RESOLUTIONS_MADE	RESOLUTIONS_COMPLETED	SUCCESS_PERCENT	SUCCESS_COLOR
Alice	2	1	50	YELLOW
Bob	2	1	50	YELLOW
Charlie	2	2	100	GREEN
Diana	3	2	66	YELLOW



Day 11 of SQL Advent Calendar

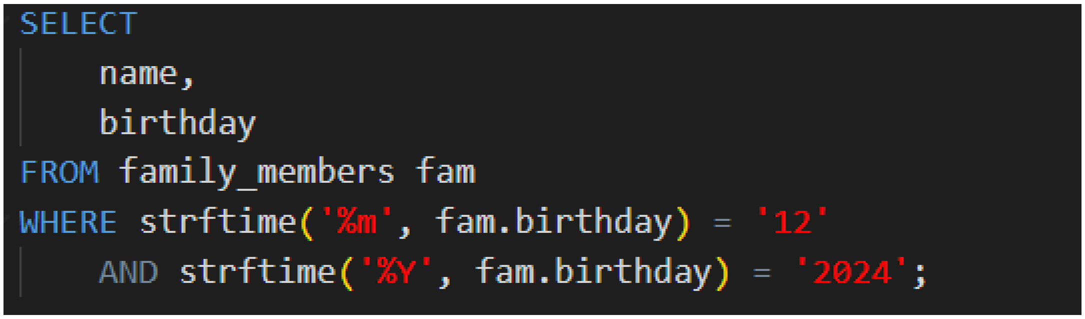
Today's Question:
You are preparing holiday gifts for your family. Who in the family_members table are celebrating their birthdays in December 2024? List their name and birthday.

Table name: family_members

member_id	name	relationship	birthday
1	Dawn	Sister	2024-12-24
2	Bob	Father	2024-05-20
3	Charlie	Brother	2024-12-25
4	Diana	Mother	2024-03-15

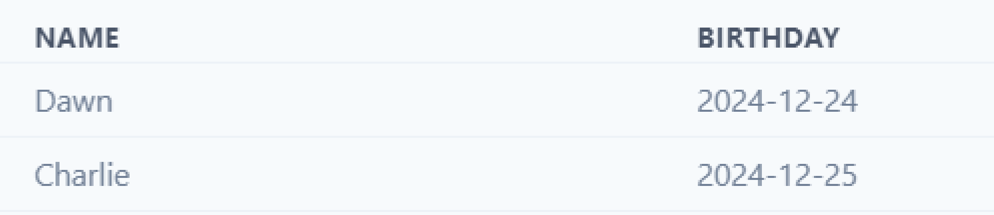
Question level of difficulty: Easy

Code:

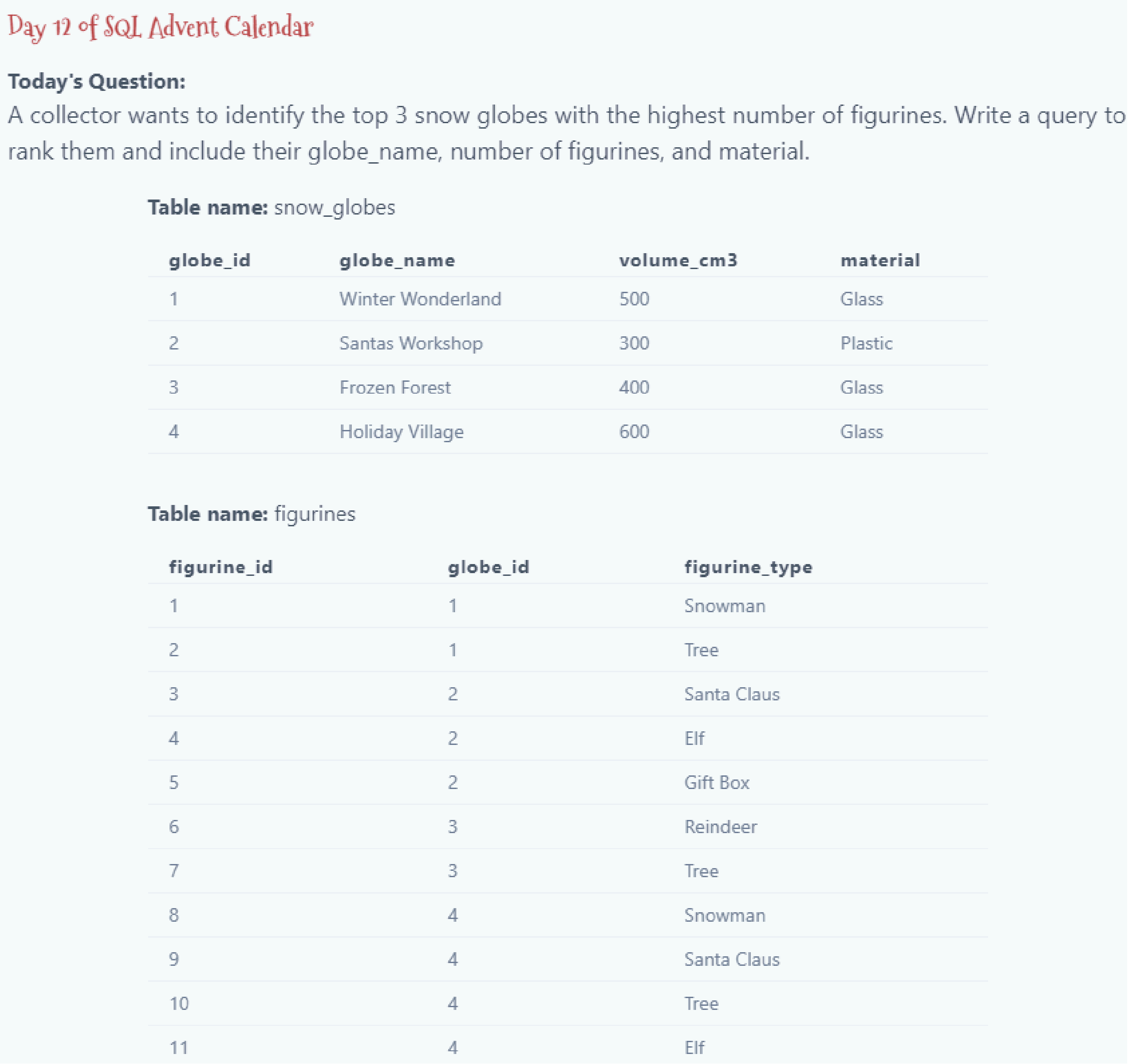


```sql
SELECT
    name,
    birthday
FROM family_members fam
WHERE strftime('%m', fam.birthday) = '12'
    AND strftime('%Y', fam.birthday) = '2024';
```

Output:



NAME	BIRTHDAY
Dawn	2024-12-24
Charlie	2024-12-25



Day 12 of SQL Advent Calendar

Today's Question:
A collector wants to identify the top 3 snow globes with the highest number of figurines. Write a query to rank them and include their globe_name, number of figurines, and material.

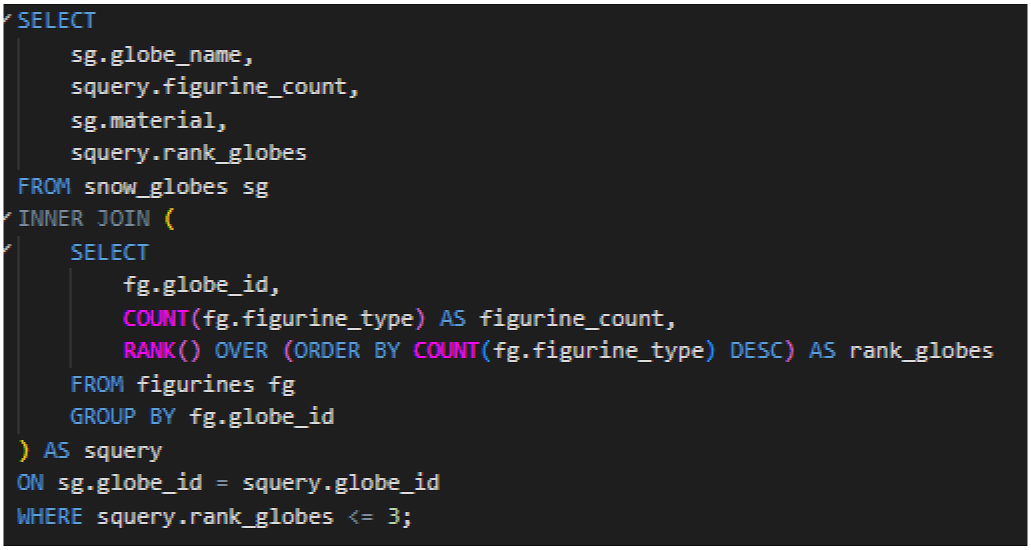
Table name: snow_globes

globe_id	globe_name	volume_cm3	material
1	Winter Wonderland	500	Glass
2	Santas Workshop	300	Plastic
3	Frozen Forest	400	Glass
4	Holiday Village	600	Glass

Table name: figurines

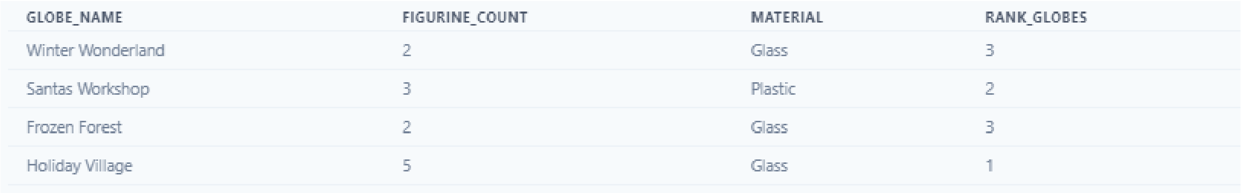
figurine_id	globe_id	figurine_type
1	1	Snowman
2	1	Tree
3	2	Santa Claus
4	2	Elf
5	2	Gift Box
6	3	Reindeer
7	3	Tree
8	4	Snowman
9	4	Santa Claus
10	4	Tree
11	4	Elf

Code:



```
SELECT
    sg.globe_name,
    squery.figurine_count,
    sg.material,
    squery.rank_globes
FROM snow_globes sg
INNER JOIN (
    SELECT
        fg.globe_id,
        COUNT(fg.figurine_type) AS figurine_count,
        RANK() OVER (ORDER BY COUNT(fg.figurine_type) DESC) AS rank_globes
    FROM figurines fg
    GROUP BY fg.globe_id
) AS squery
ON sg.globe_id = squery.globe_id
WHERE squery.rank_globes <= 3;
```

Output:



GLOBE_NAME	FIGURINE_COUNT	MATERIAL	RANK_GLOBES
Winter Wonderland	2	Glass	3
Santas Workshop	3	Plastic	2
Frozen Forest	2	Glass	3
Holiday Village	5	Glass	1