Retail Sales analysis SQL

By Ashraf Ahmad Khan

Basic Queries

Select all products from the `info` table.

```
SELECT * FROM info;
```

Retrieve the product_name and description of all products.

```
SELECT product_name, description FROM info;
```

Get the product_id, listing_price, and sale_price from the finance table.

```
SELECT product_id, listing_price, sale_price FROM finance
```

Find all products with a discount greater than 10%.

```
SELECT * FROM finance WHERE discount > 10.00;
```

List all brands in the brands table.

```
SELECT * DISTINCT brand From brands;
```

Calculate the total revenue generated by all products.

```
SELECT SUM(revenue) AS total_revenue FROM finance;
```

Find the average sale price of all products.

```
SELECT AVG(sale_price) AS avg_sale_price FROM finance;
```

Count the total number of reviews for all products.

```
SELECT SUM(reviews) AS total_reviews FROM reviews;
```

Find the maximum discount given on any product

```
SELECT MAX(discount) AS max_discount FROM finance;
```

Retrieve the minimum rating received by any product.

```
SELECT MIN(rating) AS min_rating FROM reviews;
```

```
SELECT MIN(rating) AS min t FROM finance;
```

Filtering Queries

List all products with a listing_price greater than $500.

```
SELECT * FROM finance WHERE listing_price > 500.00\
```

Find all products that have not been visited recently (before '2024-08-20').

```
SELECT * FROM traffic WHERE last_visited < '2024-08-20';
```

Retrieve all products with a rating of 4.5 or higher.

```
SELECT * FROM reviews WHERE rating >= 4.5;
```

Find all products whose description is not NULL.

```
SELECT * FROM WHERE description is not null ;
```

List all products from a specific brand, e.g., 'Brand A'.

```
SELECT * FROM brands WHERE brand = 'Brand A';
```

Joining Queries

Get the product_name, listing_price, and sale_price by joining the info and finance tables.

```
SELECT i.product_name ,f.listing_price ,f.sale_price from info i
Join finance f ON i.product_id = f.product_id;
```

Find the product_name, brand, and rating by joining the info, brands, and reviews tables.

```
SELECT i.product_name, b.brand, r.rating

FROM info i

JOIN brands b ON i.product_id = b.product_id

JOIN reviews r ON i.product_id = r.product_id;
```

List all products along with their revenue and last_visited timestamp.

```
SELECT i.product_name, f.revenue, t.last_visited

FROM info i

JOIN finance f ON i.product_id = f.product_id

JOIN traffic t ON i.product_id = t.product_id;
```

Get the product_name and brand for all products that have a listing_price above $1000.

```
SELECT i.product_name, b.brand

FROM info i

JOIN finance f ON i.product_id = f.product_id

JOIN brands b ON i.product_id = b.product_id

WHERE f.listing_price > 1000.00;
```

Find the `product_name` and `description` of all products that have not received any reviews.

```
SELECT i.product_name, i.description

FROM info i

LEFT JOIN reviews r ON i.product_id = r.product_id

WHERE r.reviews IS NULL;
```

Subqueries

List all products that have a revenue greater than the average revenue.

```
SELECT * FROM finance WHERE revenue > (SELECT AVG(revenue) FROM
finance);
```

Find the product with the highest sale price.

```
SELECT * FROM finance WHERE sale_price = (SELECT MAX(sale_price) FROM
finance);
```

Retrieve the product_name for products that were last visited on the most recent date.

```
SELECT product_name

FROM info

WHERE product_id IN (SELECT product_id FROM traffic


WHERE last_visited = (SELECT MAX(last_visited) FROM traffic));
```

Conditional Queries

```
SELECT i.product_name, f.discount, r.rating

FROM info i

JOIN finance f ON i.product_id = f.product_id

JOIN reviews r ON i.product_id = r.product_id

WHERE f.discount > 20.00 OR r.rating < 4.0;
```

Find all products that are either from 'Brand A' or have a rating of 5.0.

```
SELECT i.product_name, b.brand, r.rating
FROM info i

JOIN brands b ON i.product_id = b.product_id

JOIN reviews r ON i.product_id = r.product_id

WHERE b.brand = 'Brand A' OR r.rating = 5.0;
```

List all products that have a NULL value in any column of the info table.

```
Select * from info where product_name is null or product_id is null

Or description is null ;
```

Get all products with a sale price between $500 and $1000.

```
SELECT * FROM finance WHERE sale_price BETWEEN 500.00 AND 1000.00;
```

Find all products where the listing price is equal to the sale price.

```
SELECT * FROM finance WHERE listing_price = sale_price;
```

Retrieve the top 5 products based on revenue.

```
SELECT * FROM finance ORDER BY revenue DESC LIMIT 5;
```

Find the brand with the most products listed.

```
SELECT brand, COUNT(*) AS product_count

FROM brands

GROUP BY brand

ORDER BY product_count DESC

LIMIT 1;
```

Extract the first 10 characters of each product's description.

```
SELECT LEFT(description, 10) AS short_description FROM info;
```


Find all products whose description contains the word 'quality'.

```
SELECT * FROM info WHERE description LIKE '%quality%';
```

Count the number of products with 'Product' in their name.

```
SELECT COUNT(*) AS product_count FROM info WHERE product_name LIKE
'Product%';
```

Convert the product_name of all products to uppercase.

```
SELECT UPPER(product_name) FROM info;
```

Find the most recent visit date across all products.

```
SELECT MAX(last_visited) AS most_recent_visit FROM traffic;
```

Identify the top 3 most visited products.

```
SELECT product_id, COUNT(*) AS visit_count
FROM traffic
GROUP BY product_id
ORDER BY visit_count DESC
LIMIT 3;
```

Retail Sales analysis

Apr
May
200

Retail Sales analysis