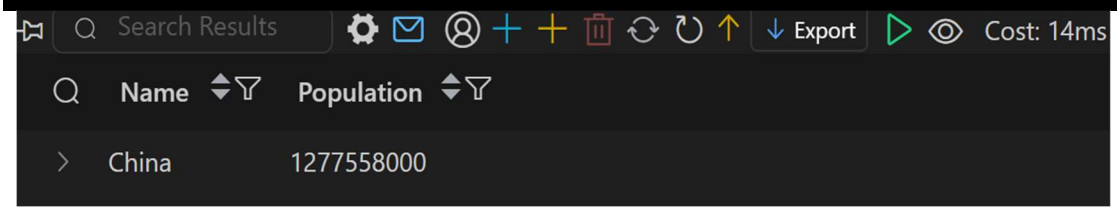
DPP Session 23 – SQL FUNCTIONS ASSIGNMENT

Question 1: Write an SQL query to find the country with the maximum population in the country table.
SQL Query:

```
SELECT Name, Population FROM country ORDER BY Population DESC LIMIT 1;
```

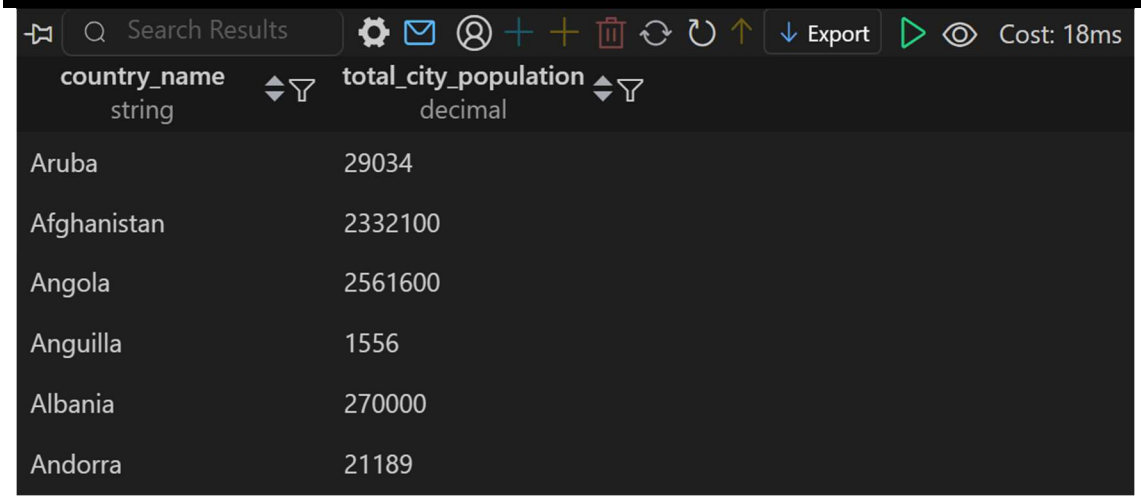


Explanation:

We sort countries by population in descending order and pick the top record.

Question 2: Write an SQL query to sum the populations of all cities per country.
SQL Query:

```
SELECT c.Name AS country_name, SUM(ci.Population) AS total_city_population
FROM country c
JOIN city ci ON c.Code = ci.CountryCode
GROUP BY c.Name;
```

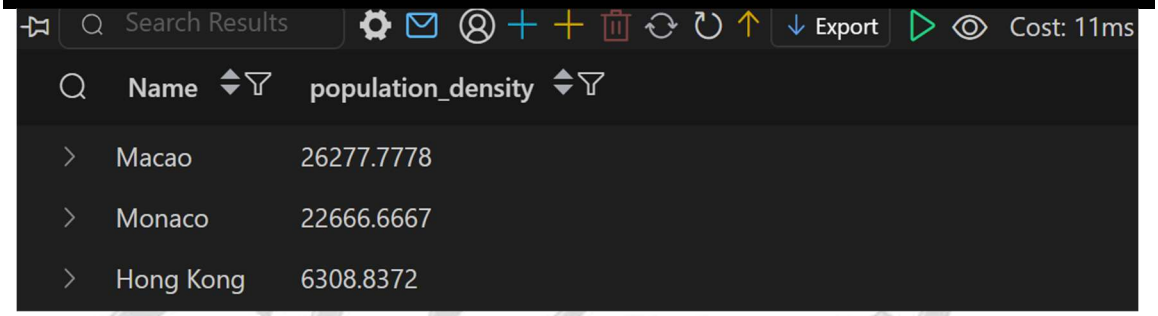


Explanation:

We join country and city tables and aggregate city populations per country.

Question 3: Find the top 3 countries with the highest population density.
SQL Query:

```
SELECT Name, (Population / SurfaceArea) AS population_density
FROM country
WHERE SurfaceArea > 0
ORDER BY population_density DESC
LIMIT 3;
```



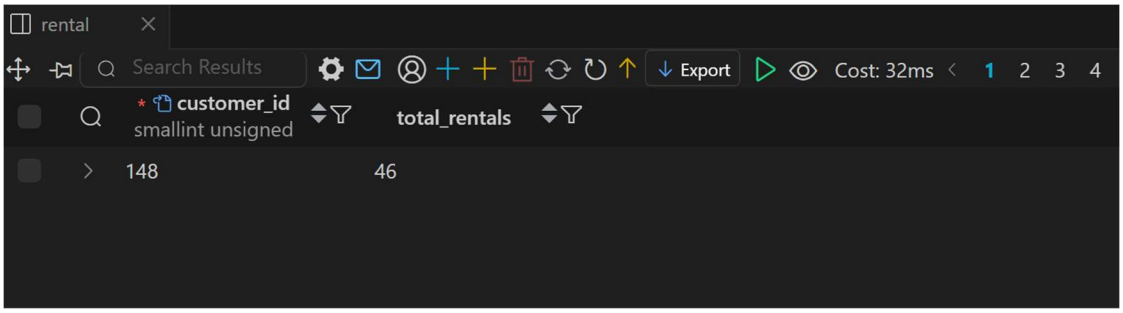
Explanation:

Population density is calculated as population divided by surface area.

Sakila Database Questions

Question 4: Write an SQL query to find the customer_id who has the highest number of rentals.
SQL Query:

```
SELECT customer_id, COUNT(*) AS total_rentals
FROM rental
GROUP BY customer_id
ORDER BY total_rentals DESC
LIMIT 1;
```

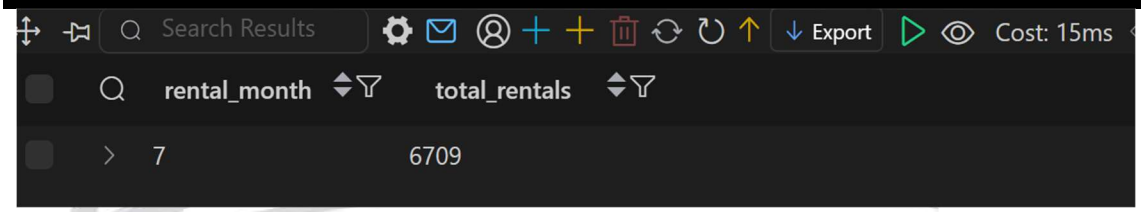


Explanation:

We count rentals per customer and select the maximum.

Question 5: Write an SQL query to identify the month with the most rentals.
SQL Query:

```
SELECT MONTH(rental_date) AS rental_month, COUNT(*) AS total_rentals
FROM rental
GROUP BY rental_month
ORDER BY total_rentals DESC
LIMIT 1;
```

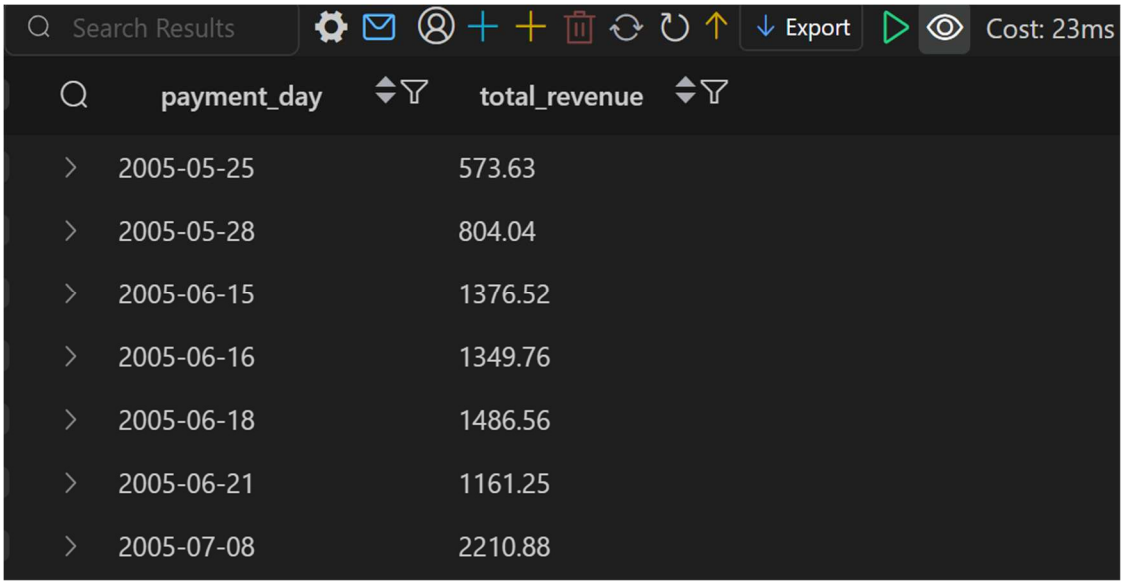


Explanation:

We extract the month from rental_date and count rentals per month.

Question 6: Find the total revenue generated per day.
SQL Query:

```
SELECT DATE(payment_date) AS payment_day, SUM(amount) AS total_revenue
FROM payment
GROUP BY payment_day;
```

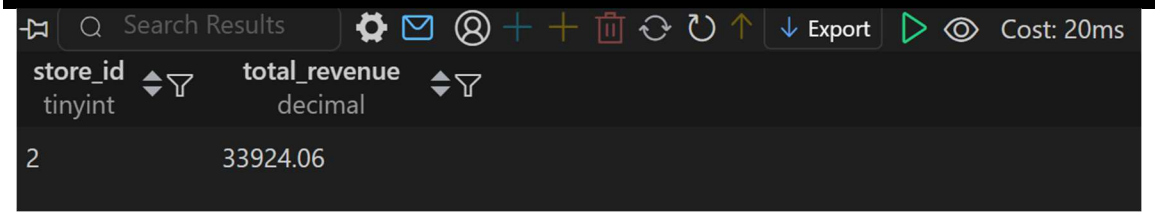


Explanation:

Payments are grouped by date to calculate daily revenue.

Question 7: Find the store that generated the highest total revenue.
SQL Query:

```
SELECT s.store_id, SUM(p.amount) AS total_revenue
FROM store s
JOIN staff st ON s.store_id = st.store_id
JOIN payment p ON st.staff_id = p.staff_id
GROUP BY s.store_id
ORDER BY total_revenue DESC
LIMIT 1;
```

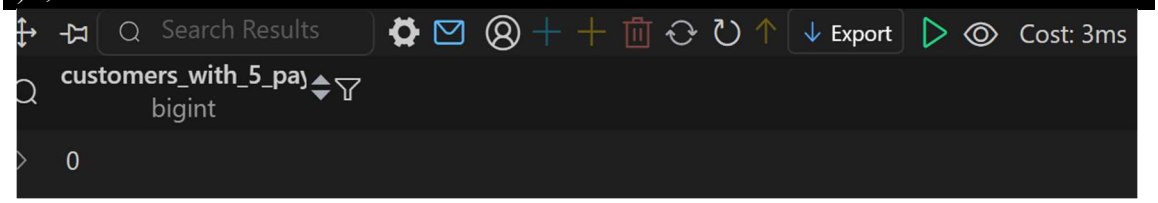


Explanation:

We join store, staff, and payment tables and sum revenue per store.

Question 8: Find the customers who have made exactly 5 payments.
SQL Query:

```
SELECT
    COUNT(*) AS customers_with_5_payments
FROM (
    SELECT customer_id
    FROM payment
    GROUP BY customer_id
    HAVING COUNT(*) = 5
) t;
```



Explanation:

We use HAVING to filter customers with exactly five payments but there are no such customers who have made exactly 5 payments.