```sql
-- Retrieve the count of listings for each borough and room type
SELECT borough, room_type, COUNT(*) AS listing_count
FROM room_types
JOIN prices ON room_types.listing_id = prices.listing_id
GROUP BY borough, room_type
ORDER BY borough, room_type;

-- Retrieve the minimum, maximum, and average price for each borough
SELECT borough, MIN(price), MAX(price), AVG(price)
FROM prices
GROUP BY borough;

-- Retrieve the total revenue for each borough by multiplying booked days and price,
-- considering the join with reviews
SELECT borough, SUM(booked_days_365 * price)
FROM prices AS p
JOIN reviews AS r
ON r.listing_id = p.listing_id
GROUP BY borough;

-- Retrieve the average price per month for each neighborhood and room type, filtering by
--specific room type and neighborhoods
SELECT neighbourhood, room_type, AVG(price_per_month)
FROM prices
JOIN room_types AS r ON r.listing_id = prices.listing_id
WHERE room_type = 'entire home/apt'
AND neighbourhood IN ('Sea Gate', 'Tribeca', 'Bayside')
GROUP BY 1,2;

-- Retrieve the listing IDs where calculated_host_listings_count is 0
SELECT listing_id
FROM reviews
WHERE calculated_host_listings_count = 0;

-- Calculate the correlation between booked days and price using the reviews and prices tables
SELECT CORR(r.booked_days_365, p.price) AS correlation
FROM reviews r
JOIN prices p ON r.listing_id = p.listing_id;

-- Retrieve the count of listings for each borough and room type, including subtotals and grand
-- total
SELECT borough, room_type, count(*)
FROM room_types
```

```sql
JOIN prices ON room_types.listing_id = prices.listing_id
GROUP BY ROLLUP(borough, room_type);

-- Retrieve the earliest and latest last review dates, as well as the difference between them
SELECT min(last_review) AS start, max(last_review) AS end_, max(last_review) -
min(last_review)
FROM reviews;

-- Retrieve the count of listings for each borough and room type, including borough and room
type names
SELECT COUNT(*), borough, room_type
FROM prices JOIN room_types USING(listing_id)
GROUP BY 2,3;

-- Retrieve the count of listings exceeding a price of 500 for each room type
SELECT room_type, SUM(CASE WHEN price>500 THEN 1 ELSE 0 END)
FROM room_types
JOIN prices ON prices.listing_id = room_types.listing_id
GROUP BY room_type;

-- Retrieve the row count for each table: prices, reviews, and room_types
SELECT 'prices' AS TableName, COUNT(*) AS RowCount FROM prices
UNION ALL
SELECT 'reviews' AS TableName, COUNT(*) AS RowCount FROM reviews
UNION ALL
SELECT 'room_types' AS TableName, COUNT(*) AS RowCount FROM room_types;

-- Retrieve all columns from reviews, room_types, and prices tables using left joins
SELECT p.*, r.*, rt.*
FROM reviews r
LEFT JOIN room_types rt ON r.listing_id = rt.listing_id
LEFT JOIN prices p ON r.listing_id = p.listing_id;
```