HNG Ride Business Report

Dataset Overview

- Dataset Name: HNG Ride
- Data Source: provided by HNG
- Data Structure:

Table 1 – Drivers:

5 Columns: driver_id, driver_name, city, signup_date, rating

2000 rows

Table 2 – Riders:

5 Columns: rider_id, rider_name, city, signup_date, email

10000 rows

Table 3 – Payments:

5 Columns: payment_id, ride_id, amount, paid_date, method

50000 rows

Table 4 – rides:

11 Columns: rider_id, ride_id, driver_id, request_time, pickup_time, dropoff_time, pickup_city, dropoff_city, distance_km, Status, fare

50000 rows

Research Questions:

- Find the top 10 longest rides (by distance), including driver name, rider name, pickup/dropoff cities, and payment method.
- How many riders who signed up in 2021 still took rides in 2024?
- Compare quarterly revenue between 2021, 2022, 2023, and 2024. Which quarter had the biggest YoY growth?
- For each driver, calculate their average monthly rides since signup. Who are the top 5 drivers with the highest consistency (most rides per active month)?
- Calculate the cancellation rate per city and identify which city had the highest cancellation rate?
- Identify riders who have taken more than 10 rides but never paid with cash.
- Find the top 3 drivers in each city by total revenue earned between June 2021 and Dec 2024. If a driver has multiple cities, count revenue where they picked up passengers in that city.
- Management wants to know the top 10 drivers that are qualified to receive bonuses using the criteria below;

  at least 30 rides completed,

  an average rating ≥ 4.5, and

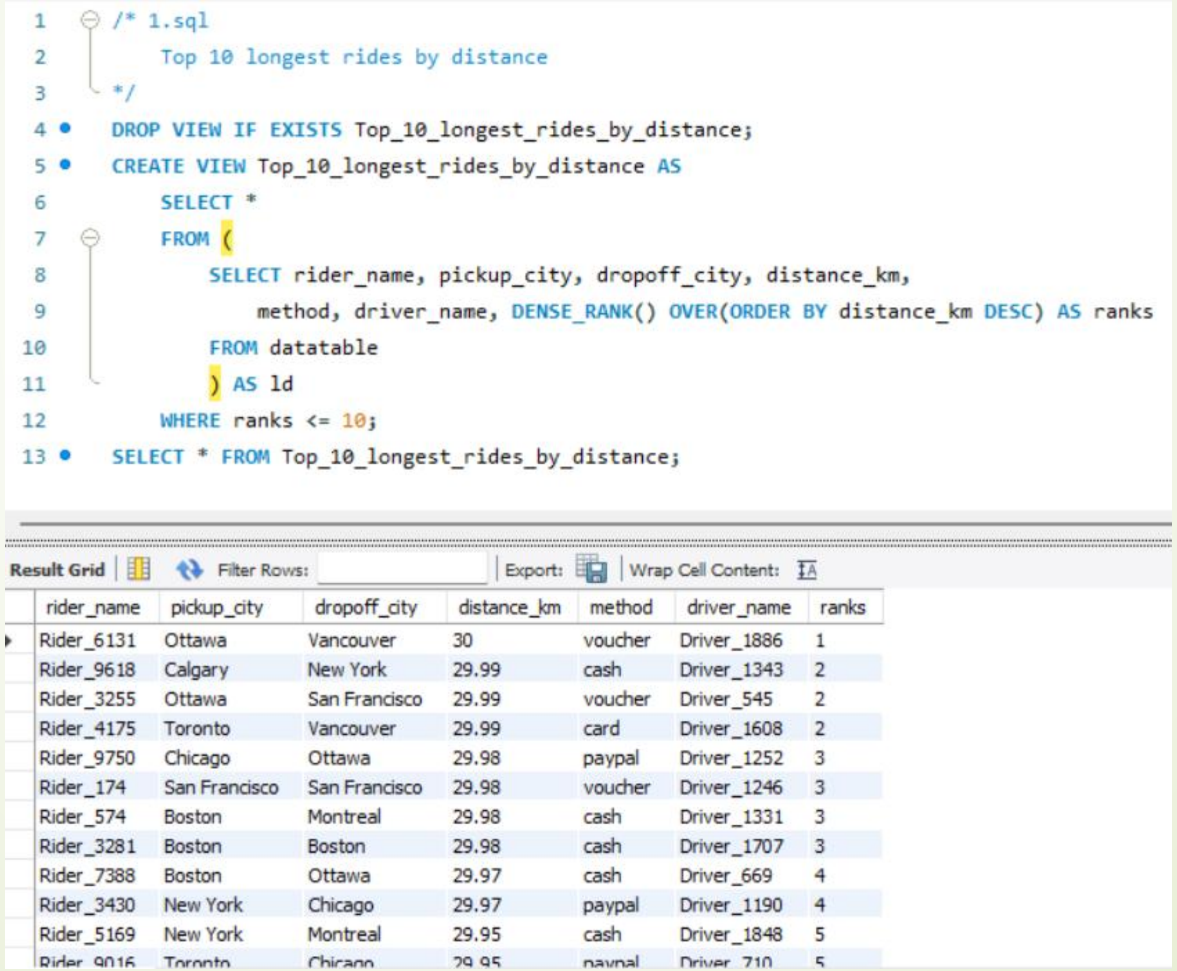
  a cancellation rate under 5%.

Data Cleaning Summary

- Joined tables: Drivers, Payments, Riders, Rides
- Formatted signup_date, request_time, pickup_time and paid_date columns as standard datetime columns
- Replaced inconsistent city name: L.A – Los Angeles, S.F. – San Francisco, N.Y – New York
- Filtered out data points outside the official reporting window June 2021 – December 2024
- Filtered out data points for incomplete transactions

Business Question 1:

Find the top 10 longest rides (by distance), including driver name, rider name, pickup/dropoff cities, and payment method.

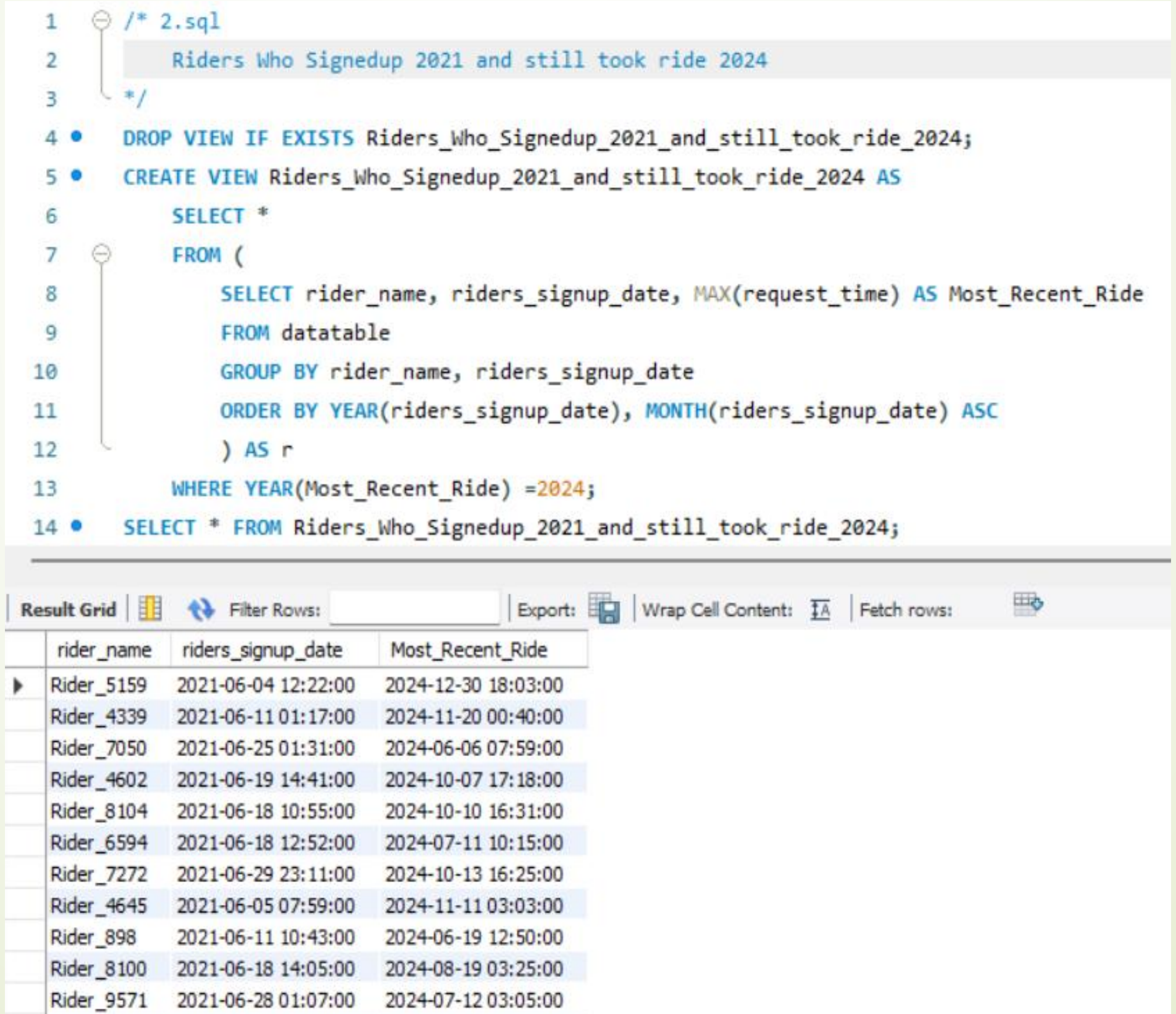


```sql
/* 1.sql
    Top 10 longest rides by distance
*/
DROP VIEW IF EXISTS Top_10_longest_rides_by_distance;
CREATE VIEW Top_10_longest_rides_by_distance AS
    SELECT *
    FROM (
        SELECT rider_name, pickup_city, dropoff_city, distance_km,
            method, driver_name, DENSE_RANK() OVER(ORDER BY distance_km DESC) AS ranks
        FROM datatable
        ) AS ld
    WHERE ranks <= 10;
SELECT * FROM Top_10_longest_rides_by_distance;
```

rider_name	pickup_city	dropoff_city	distance_km	method	driver_name	ranks
Rider_6131	Ottawa	Vancouver	30	voucher	Driver_1886	1
Rider_9618	Calgary	New York	29.99	cash	Driver_1343	2
Rider_3255	Ottawa	San Francisco	29.99	voucher	Driver_545	2
Rider_4175	Toronto	Vancouver	29.99	card	Driver_1608	2
Rider_9750	Chicago	Ottawa	29.98	paypal	Driver_1252	3
Rider_174	San Francisco	San Francisco	29.98	voucher	Driver_1246	3
Rider_574	Boston	Montreal	29.98	cash	Driver_1331	3
Rider_3281	Boston	Boston	29.98	cash	Driver_1707	3
Rider_7388	Boston	Ottawa	29.97	cash	Driver_669	4
Rider_3430	New York	Chicago	29.97	paypal	Driver_1190	4
Rider_5169	New York	Montreal	29.95	cash	Driver_1848	5
Rider_9016	Toronto	Chicago	29.95	paypal	Driver_710	5

Business Question 2:

How many riders who signed up in 2021 still took rides in 2024?

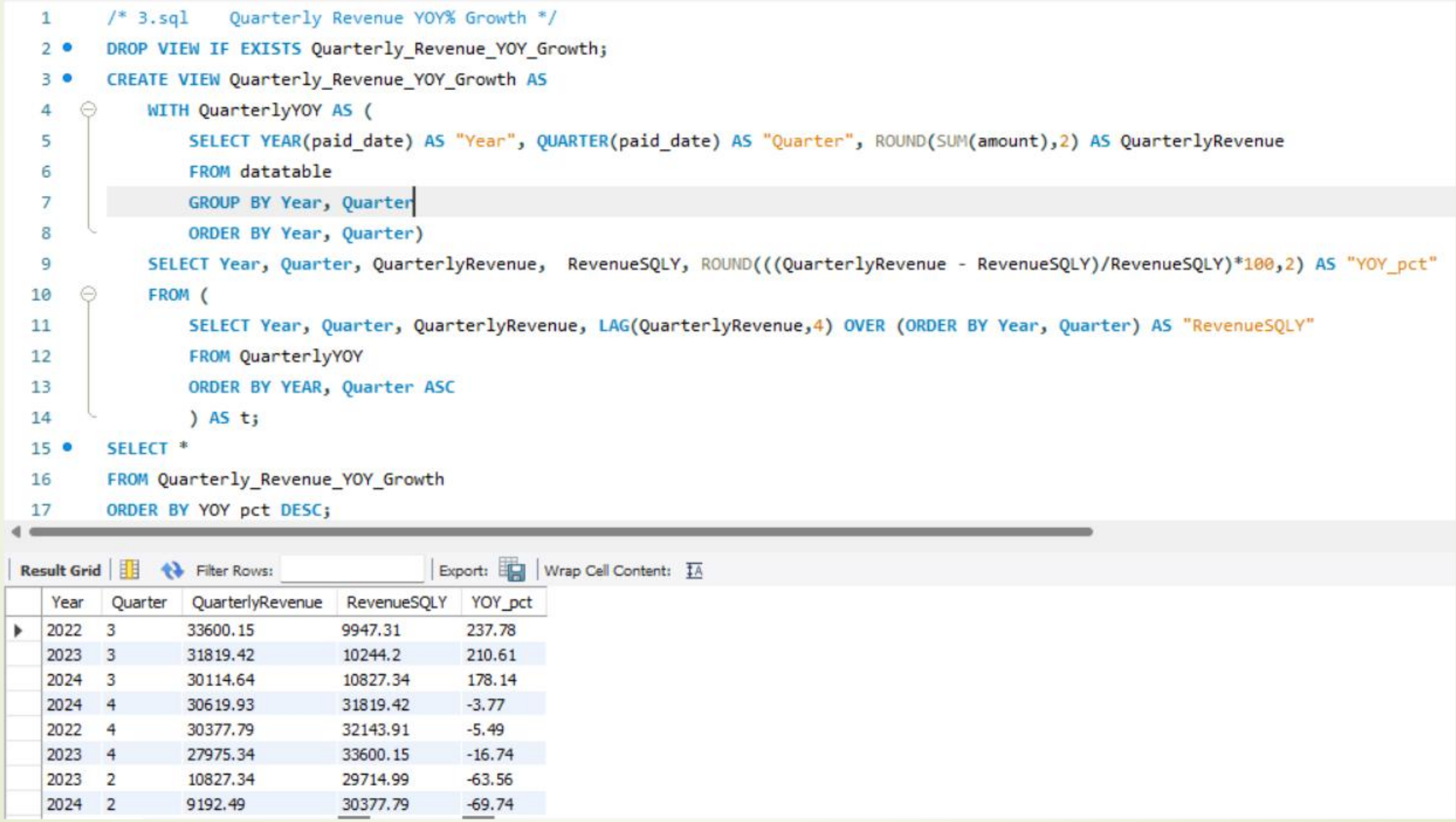


```sql
/* 2.sql
    Riders Who Signedup 2021 and still took ride 2024
*/
DROP VIEW IF EXISTS Riders_Who_Signedup_2021_and_still_took_ride_2024;
CREATE VIEW Riders_Who_Signedup_2021_and_still_took_ride_2024 AS
    SELECT *
    FROM (
        SELECT rider_name, riders_signup_date, MAX(request_time) AS Most_Recent_Ride
        FROM datatable
        GROUP BY rider_name, riders_signup_date
        ORDER BY YEAR(riders_signup_date), MONTH(riders_signup_date) ASC
        ) AS r
    WHERE YEAR(Most_Recent_Ride) =2024;
SELECT * FROM Riders_Who_Signedup_2021_and_still_took_ride_2024;
```

rider_name	riders_signup_date	Most_Recent_Ride
Rider_5159	2021-06-04 12:22:00	2024-12-30 18:03:00
Rider_4339	2021-06-11 01:17:00	2024-11-20 00:40:00
Rider_7050	2021-06-25 01:31:00	2024-06-06 07:59:00
Rider_4602	2021-06-19 14:41:00	2024-10-07 17:18:00
Rider_8104	2021-06-18 10:55:00	2024-10-10 16:31:00
Rider_6594	2021-06-18 12:52:00	2024-07-11 10:15:00
Rider_7272	2021-06-29 23:11:00	2024-10-13 16:25:00
Rider_4645	2021-06-05 07:59:00	2024-11-11 03:03:00
Rider_898	2021-06-11 10:43:00	2024-06-19 12:50:00
Rider_8100	2021-06-18 14:05:00	2024-08-19 03:25:00
Rider_9571	2021-06-28 01:07:00	2024-07-12 03:05:00

Business Question 3:

Compare quarterly revenue between 2021, 2022, 2023, and 2024. Which quarter had the biggest YoY growth?

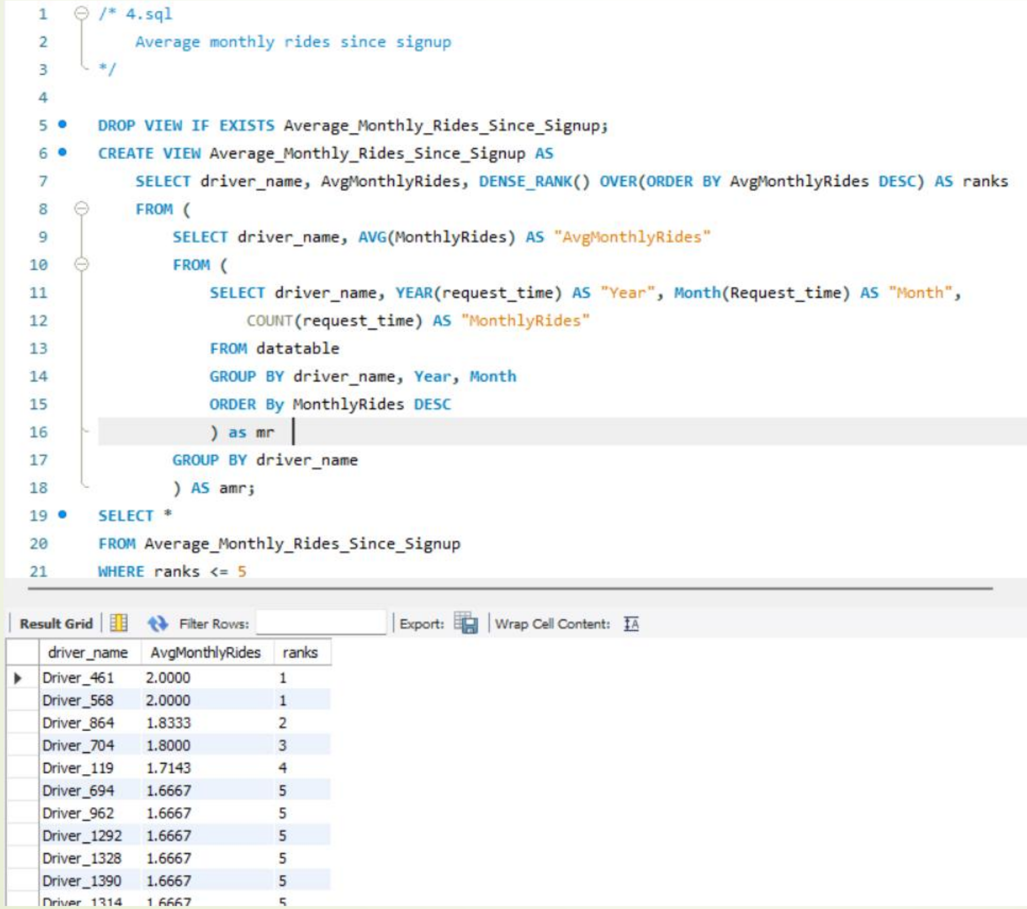


```sql
/* 3.sql    Quarterly Revenue YOY% Growth */
DROP VIEW IF EXISTS Quarterly_Revenue_YOY_Growth;
CREATE VIEW Quarterly_Revenue_YOY_Growth AS
    WITH QuarterlyYOY AS (
        SELECT YEAR(paid_date) AS "Year", QUARTER(paid_date) AS "Quarter", ROUND(SUM(amount),2) AS QuarterlyRevenue
        FROM datatable
        GROUP BY Year, Quarter
        ORDER BY Year, Quarter)
    SELECT Year, Quarter, QuarterlyRevenue,  RevenueSQLY, ROUND(((QuarterlyRevenue - RevenueSQLY)/RevenueSQLY)*100,2) AS "YOY_pct"
    FROM (
        SELECT Year, Quarter, QuarterlyRevenue, LAG(QuarterlyRevenue,4) OVER (ORDER BY Year, Quarter) AS "RevenueSQLY"
        FROM QuarterlyYOY
        ORDER BY YEAR, Quarter ASC
        ) AS t;
SELECT *
FROM Quarterly_Revenue_YOY_Growth
ORDER BY YOY pct DESC;
```

Year	Quarter	QuarterlyRevenue	RevenueSQLY	YOY_pct
2022	3	33600.15	9947.31	237.78
2023	3	31819.42	10244.2	210.61
2024	3	30114.64	10827.34	178.14
2024	4	30619.93	31819.42	-3.77
2022	4	30377.79	32143.91	-5.49
2023	4	27975.34	33600.15	-16.74
2023	2	10827.34	29714.99	-63.56
2024	2	9192.49	30377.79	-69.74

Business Question 4:

For each driver, calculate their average monthly rides since signup. Who are the top 5 drivers with the highest consistency (most rides per active month)?



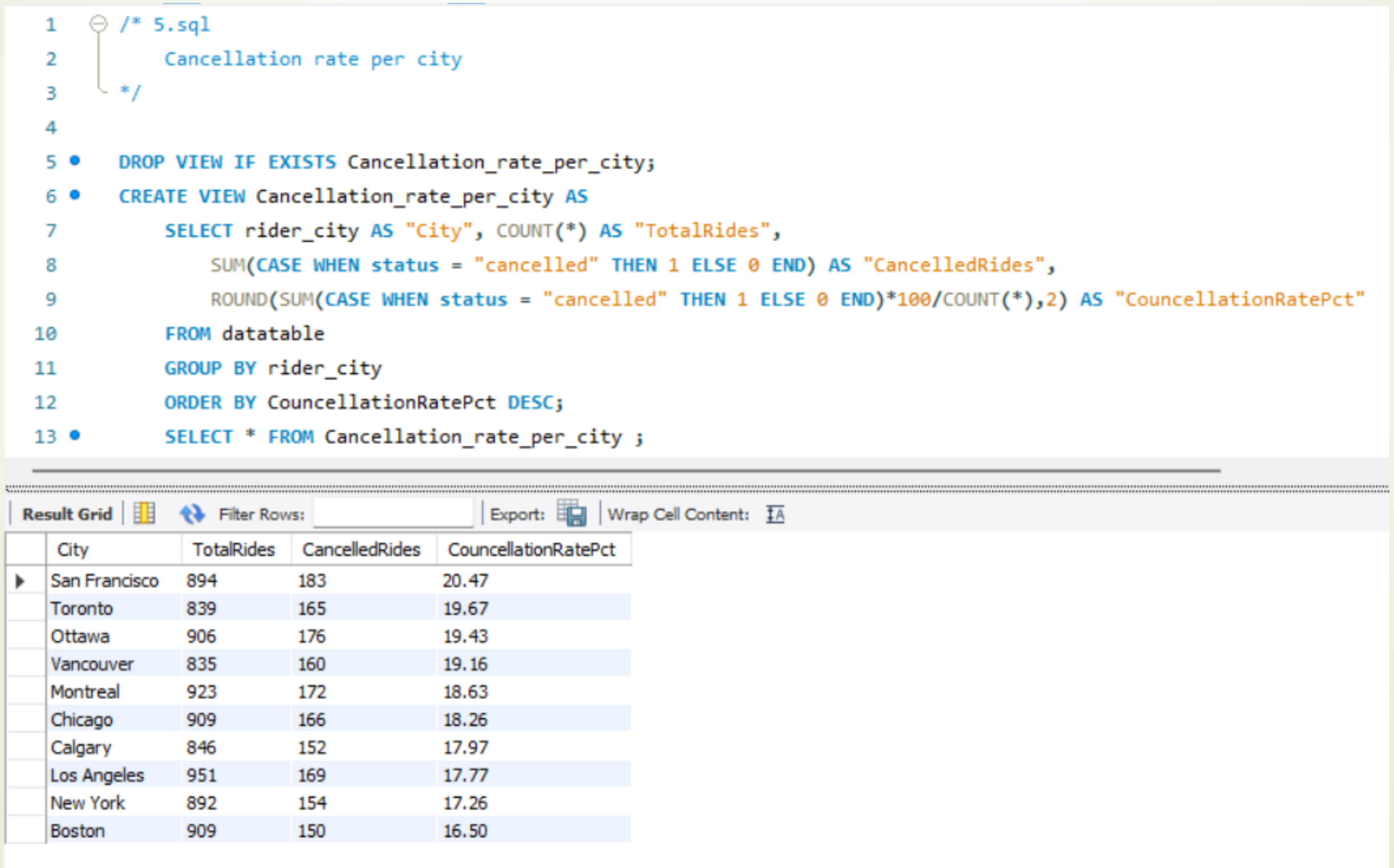
```
/* 4.sql
    Average monthly rides since signup
*/

DROP VIEW IF EXISTS Average_Monthly_Rides_Since_Signup;
CREATE VIEW Average_Monthly_Rides_Since_Signup AS
    SELECT driver_name, AvgMonthlyRides, DENSE_RANK() OVER(ORDER BY AvgMonthlyRides DESC) AS ranks
    FROM (
        SELECT driver_name, AVG(MonthlyRides) AS "AvgMonthlyRides"
        FROM (
            SELECT driver_name, YEAR(request_time) AS "Year", Month(Request_time) AS "Month",
                COUNT(request_time) AS "MonthlyRides"
            FROM datatable
            GROUP BY driver_name, Year, Month
            ORDER By MonthlyRides DESC
            ) as mr
        GROUP BY driver_name
        ) AS amr;
SELECT *
FROM Average_Monthly_Rides_Since_Signup
WHERE ranks <= 5
```

driver_name	AvgMonthlyRides	ranks
Driver_461	2.0000	1
Driver_568	2.0000	1
Driver_864	1.8333	2
Driver_704	1.8000	3
Driver_119	1.7143	4
Driver_694	1.6667	5
Driver_962	1.6667	5
Driver_1292	1.6667	5
Driver_1328	1.6667	5
Driver_1390	1.6667	5
Driver_1314	1.6667	5

Business Question 5:

Calculate the cancellation rate per city and identify which city had the highest cancellation rate?



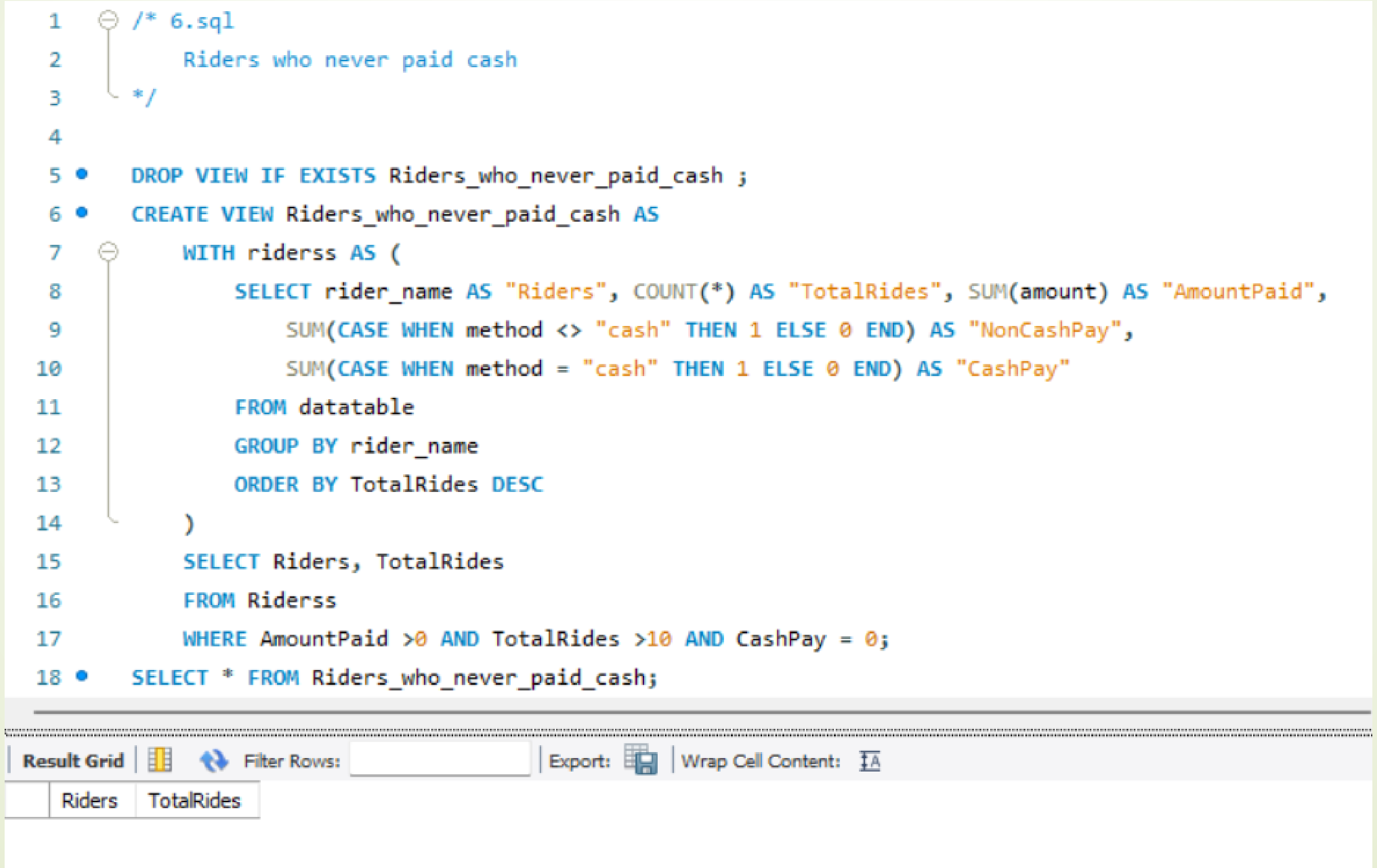
```
/* 5.sql
    Cancellation rate per city
*/

DROP VIEW IF EXISTS Cancellation_rate_per_city;
CREATE VIEW Cancellation_rate_per_city AS
    SELECT rider_city AS "City", COUNT(*) AS "TotalRides",
        SUM(CASE WHEN status = "cancelled" THEN 1 ELSE 0 END) AS "CancelledRides",
        ROUND(SUM(CASE WHEN status = "cancelled" THEN 1 ELSE 0 END)*100/COUNT(*),2) AS "CouncellationRatePct"
    FROM datatable
    GROUP BY rider_city
    ORDER BY CouncellationRatePct DESC;
    SELECT * FROM Cancellation_rate_per_city ;
```

City	TotalRides	CancelledRides	CouncellationRatePct
San Francisco	894	183	20.47
Toronto	839	165	19.67
Ottawa	906	176	19.43
Vancouver	835	160	19.16
Montreal	923	172	18.63
Chicago	909	166	18.26
Calgary	846	152	17.97
Los Angeles	951	169	17.77
New York	892	154	17.26
Boston	909	150	16.50

Business Question 6:

Identify riders who have taken more than 10 rides but never paid with cash.



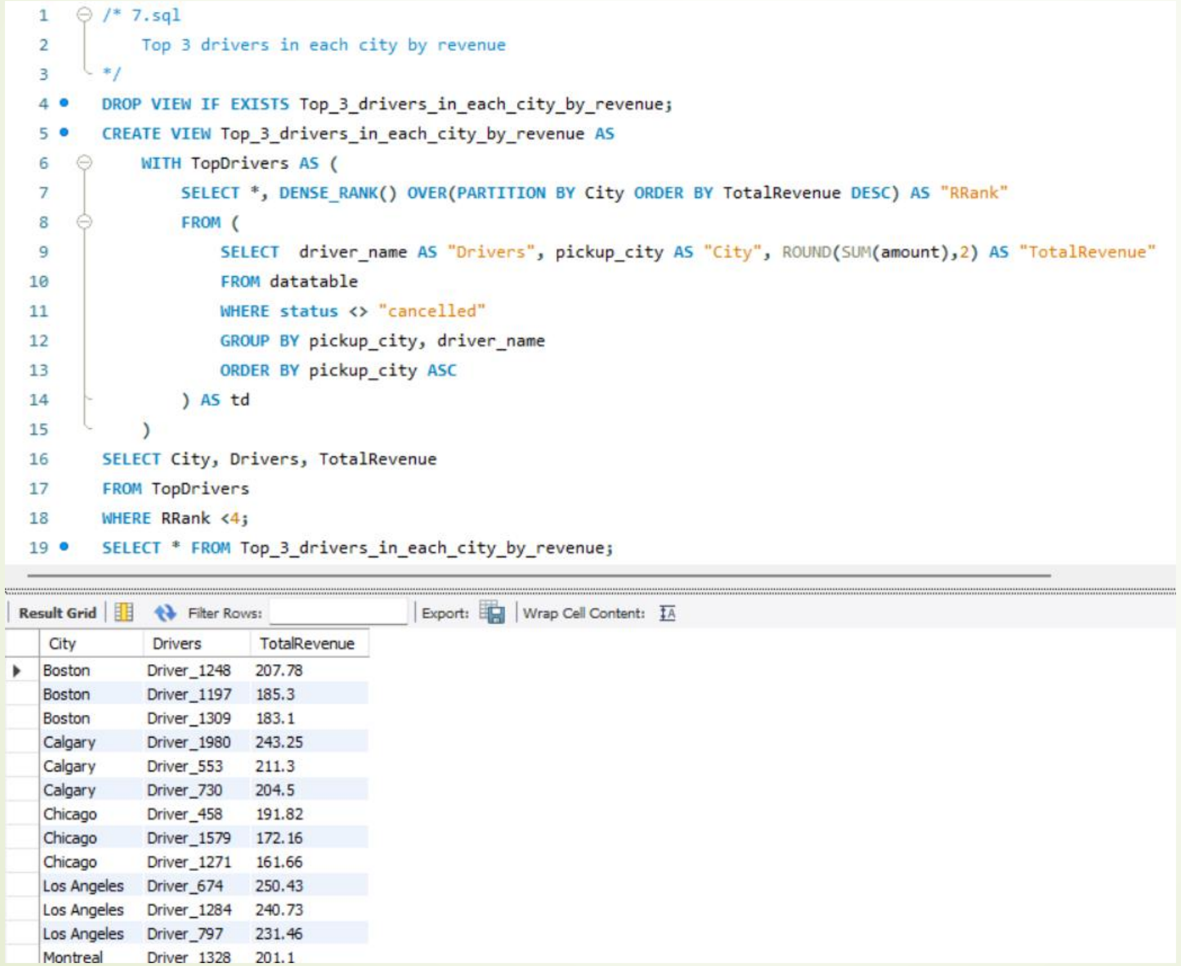
```sql
/* 6.sql
    Riders who never paid cash
*/

DROP VIEW IF EXISTS Riders_who_never_paid_cash ;
CREATE VIEW Riders_who_never_paid_cash AS
    WITH riderss AS (
        SELECT rider_name AS "Riders", COUNT(*) AS "TotalRides", SUM(amount) AS "AmountPaid",
            SUM(CASE WHEN method <> "cash" THEN 1 ELSE 0 END) AS "NonCashPay",
            SUM(CASE WHEN method = "cash" THEN 1 ELSE 0 END) AS "CashPay"
        FROM datatable
        GROUP BY rider_name
        ORDER BY TotalRides DESC
    )
    SELECT Riders, TotalRides
    FROM Riderss
    WHERE AmountPaid >0 AND TotalRides >10 AND CashPay = 0;
SELECT * FROM Riders_who_never_paid_cash;
```

Riders	TotalRides

Business Question 7:

Find the top 3 drivers in each city by total revenue earned between June 2021 and Dec 2024. If a driver has multiple cities, count revenue where they picked up passengers in that city.

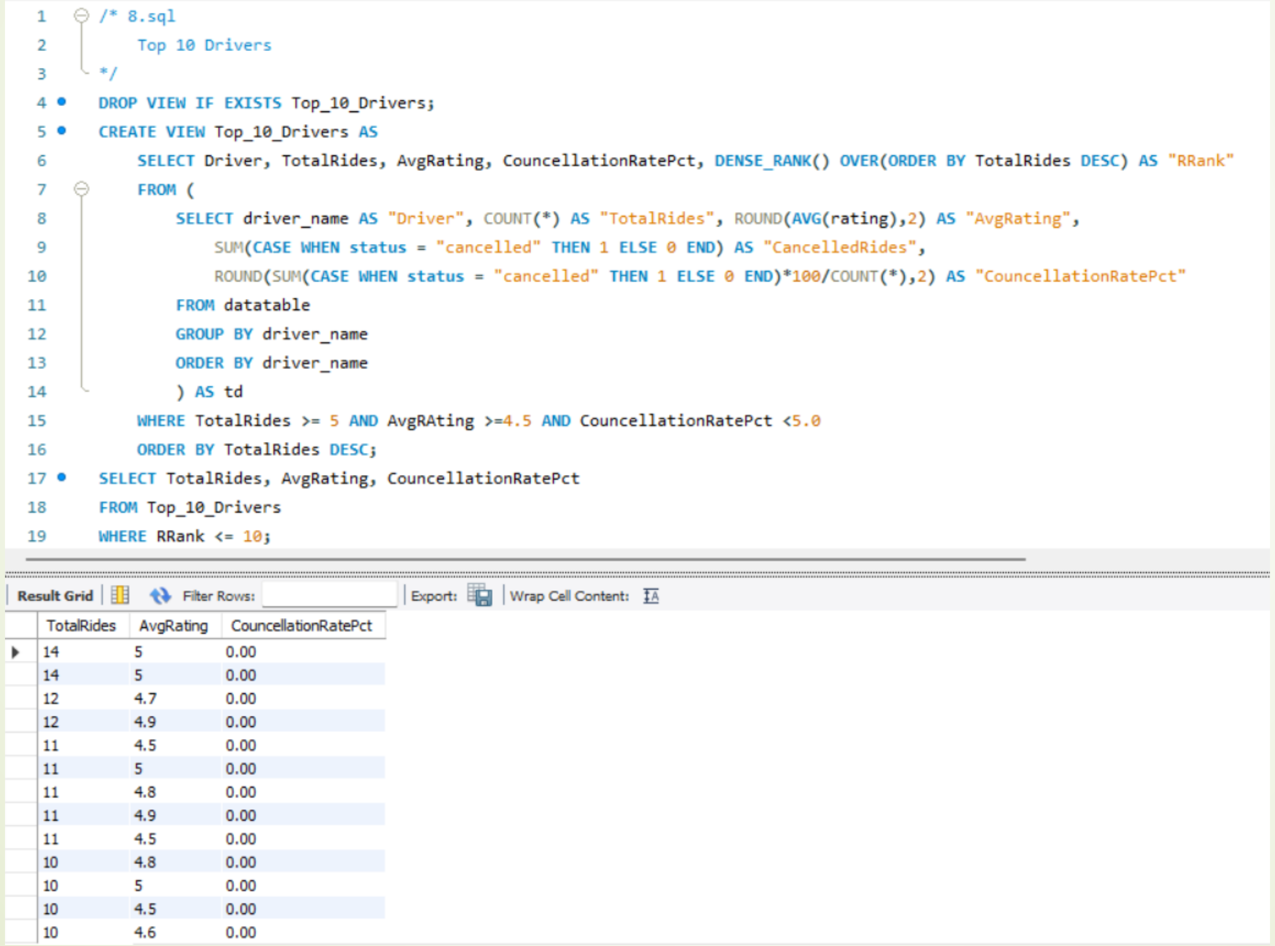


```sql
/* 7.sql
    Top 3 drivers in each city by revenue
*/
DROP VIEW IF EXISTS Top_3_drivers_in_each_city_by_revenue;
CREATE VIEW Top_3_drivers_in_each_city_by_revenue AS
    WITH TopDrivers AS (
        SELECT *, DENSE_RANK() OVER(PARTITION BY City ORDER BY TotalRevenue DESC) AS "RRank"
        FROM (
            SELECT  driver_name AS "Drivers", pickup_city AS "City", ROUND(SUM(amount),2) AS "TotalRevenue"
            FROM datatable
            WHERE status <> "cancelled"
            GROUP BY pickup_city, driver_name
            ORDER BY pickup_city ASC
        ) AS td
    )
SELECT City, Drivers, TotalRevenue
FROM TopDrivers
WHERE RRank <4;
SELECT * FROM Top_3_drivers_in_each_city_by_revenue;
```

City	Drivers	TotalRevenue
Boston	Driver_1248	207.78
Boston	Driver_1197	185.3
Boston	Driver_1309	183.1
Calgary	Driver_1980	243.25
Calgary	Driver_553	211.3
Calgary	Driver_730	204.5
Chicago	Driver_458	191.82
Chicago	Driver_1579	172.16
Chicago	Driver_1271	161.66
Los Angeles	Driver_674	250.43
Los Angeles	Driver_1284	240.73
Los Angeles	Driver_797	231.46
Montreal	Driver_1328	201.1

Business Question 8:

Management wants to know the top 10 drivers that are qualified to receive bonuses using the criteria below; at least 30 rides completed, an average rating ≥ 4.5, and a cancellation rate under 5%.



```sql
/* 8.sql
    Top 10 Drivers
*/
DROP VIEW IF EXISTS Top_10_Drivers;
CREATE VIEW Top_10_Drivers AS
    SELECT Driver, TotalRides, AvgRating, CouncellationRatePct, DENSE_RANK() OVER(ORDER BY TotalRides DESC) AS "RRank"
    FROM (
        SELECT driver_name AS "Driver", COUNT(*) AS "TotalRides", ROUND(AVG(rating),2) AS "AvgRating",
            SUM(CASE WHEN status = "cancelled" THEN 1 ELSE 0 END) AS "CancelledRides",
            ROUND(SUM(CASE WHEN status = "cancelled" THEN 1 ELSE 0 END)*100/COUNT(*),2) AS "CouncellationRatePct"
        FROM datatable
        GROUP BY driver_name
        ORDER BY driver_name
        ) AS td
    WHERE TotalRides >= 5 AND AvgRAting >=4.5 AND CouncellationRatePct <5.0
    ORDER BY TotalRides DESC;
SELECT TotalRides, AvgRating, CouncellationRatePct
FROM Top_10_Drivers
WHERE RRank <= 10;
```

TotalRides	AvgRating	CouncellationRatePct
14	5	0.00
14	5	0.00
12	4.7	0.00
12	4.9	0.00
11	4.5	0.00
11	5	0.00
11	4.8	0.00
11	4.9	0.00
11	4.5	0.00
10	4.8	0.00
10	5	0.00
10	4.5	0.00
10	4.6	0.00