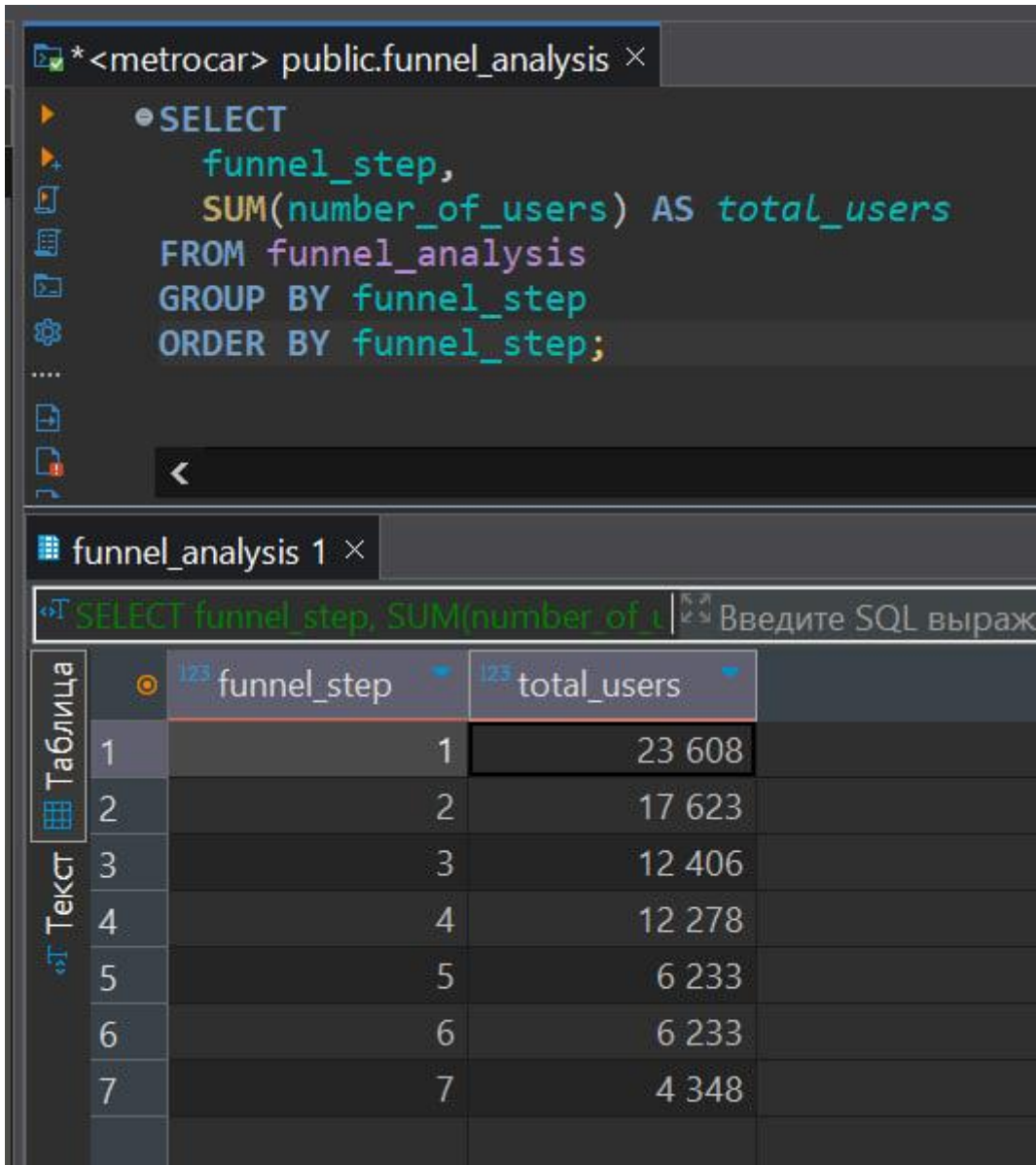


1. Базовий запит до funnel_analysis для виведення кількості користувачів на кожному з етапів.

```
SELECT
  funnel_step,
  SUM(number_of_users) AS total_users
FROM funnel_analysis
GROUP BY funnel_step
ORDER BY funnel_step;
```

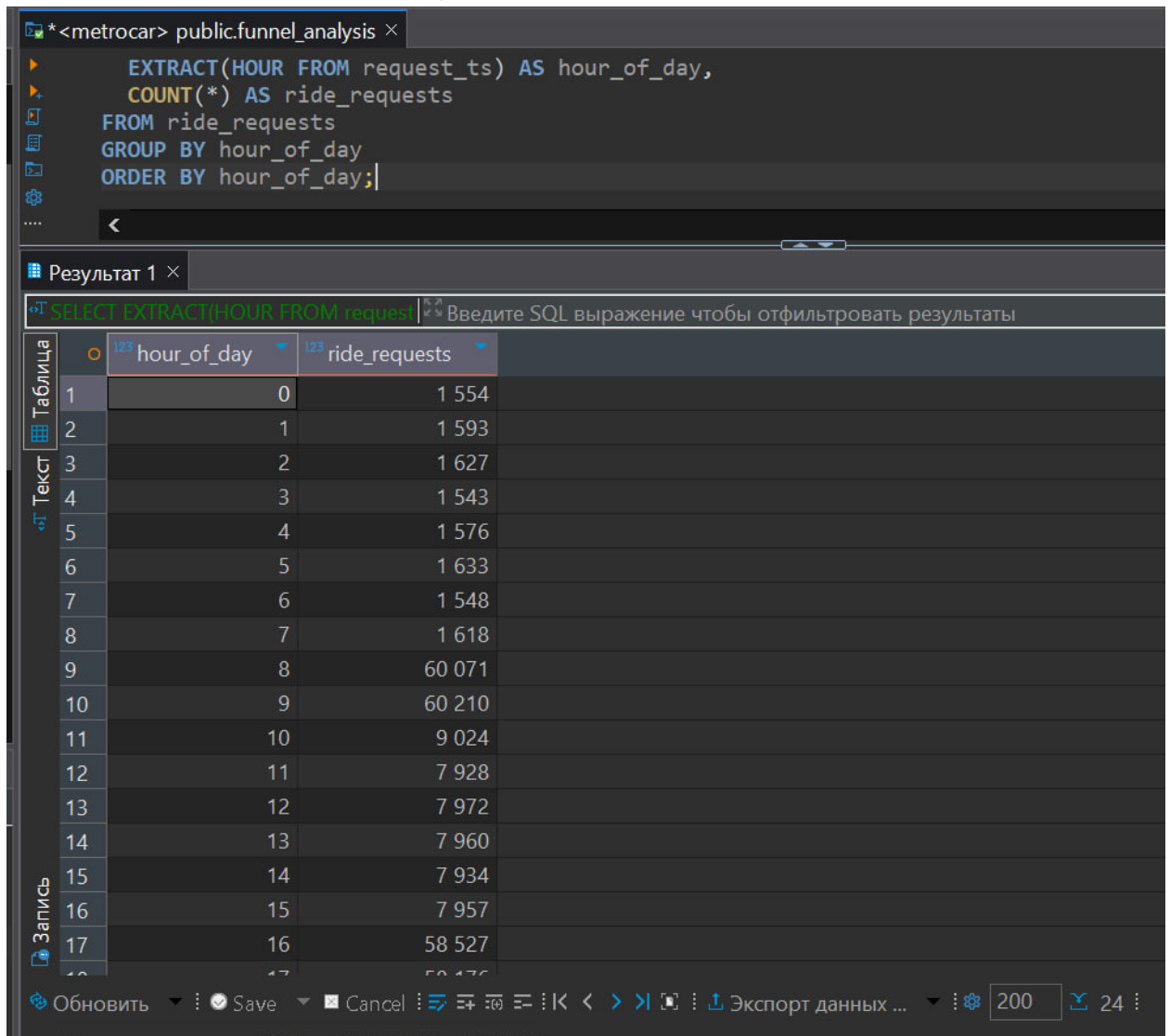


The screenshot shows a SQL IDE interface. The top pane displays a query for the 'public.funnel_analysis' table. The bottom pane shows the results of the query in a table format. The table has two columns: 'funnel_step' and 'total_users'. The results are ordered by 'funnel_step' from 1 to 7.

funnel_step	total_users
1	23 608
2	17 623
3	12 406
4	12 278
5	6 233
6	6 233
7	4 348

2. Запит для виведення кількості запитів в кожну годину доби.

```
SELECT
  EXTRACT(HOUR FROM request_ts) AS hour_of_day,
  COUNT(*) AS ride_requests
FROM ride_requests
GROUP BY hour_of_day
ORDER BY hour_of_day;
```



The screenshot shows a SQL query editor with a query and its results. The query is:

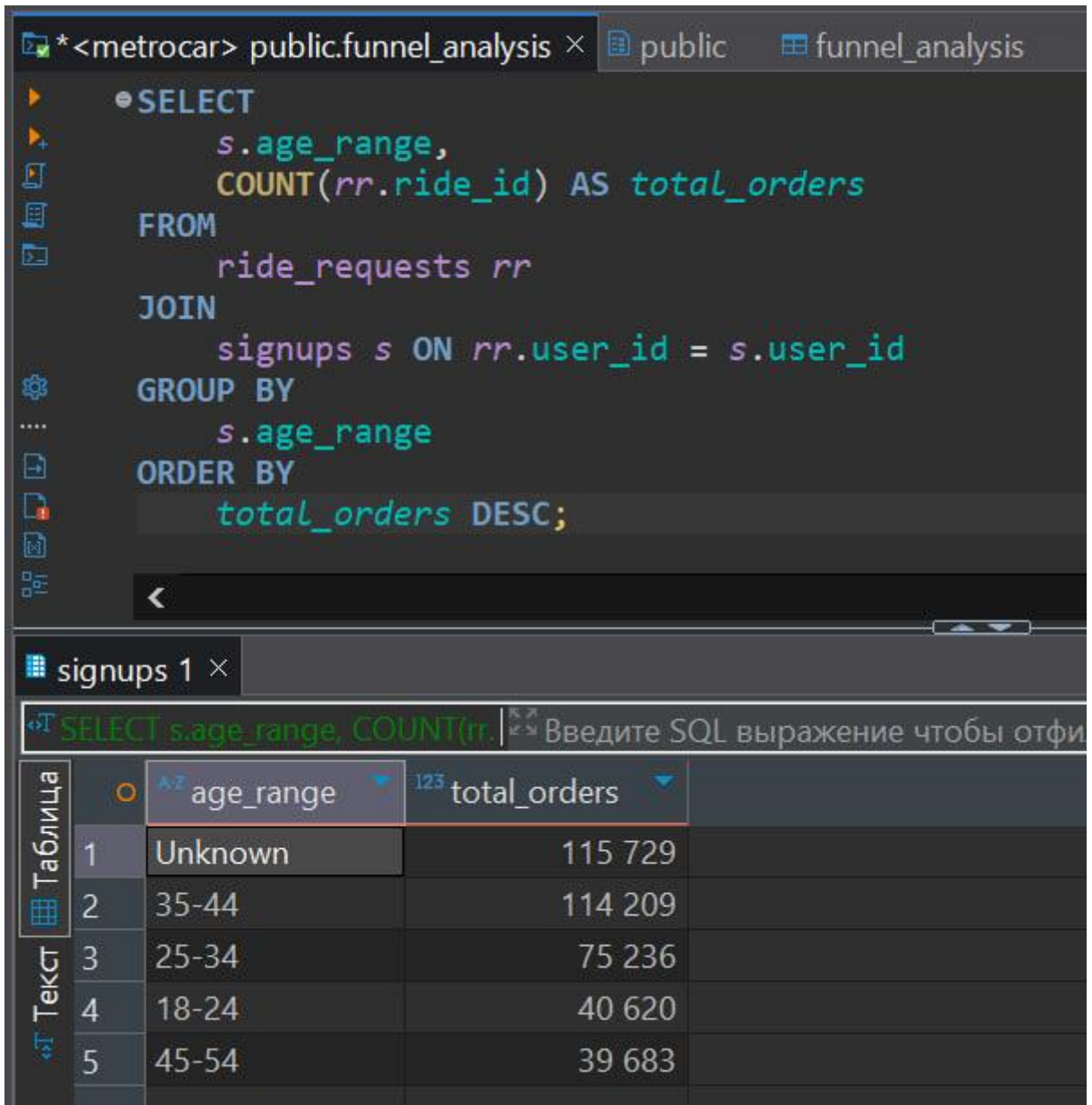
```
EXTRACT(HOUR FROM request_ts) AS hour_of_day,
COUNT(*) AS ride_requests
FROM ride_requests
GROUP BY hour_of_day
ORDER BY hour_of_day;
```

The results are displayed in a table with two columns: **hour_of_day** and **ride_requests**. The table shows the number of ride requests for each hour of the day, ordered by hour.

hour_of_day	ride_requests
0	1 554
1	1 593
2	1 627
3	1 543
4	1 576
5	1 633
6	1 548
7	1 618
8	60 071
9	60 210
10	9 024
11	7 928
12	7 972
13	7 960
14	7 934
15	7 957
16	58 527
17	58 476

3. Запит для отримання кількості замовлень для кожної вікової категорії.

```
SELECT s.age_range, COUNT(rr.ride_id) AS total_orders
FROM
    ride_requests rr
JOIN
    signups s ON rr.user_id = s.user_id
GROUP BY
    s.age_range
ORDER BY
    total_orders DESC;
```



The screenshot shows a SQL IDE interface. The top pane displays a SQL query. The bottom pane shows the results of the query in a table format. The table has two columns: 'age_range' and 'total_orders'. The results are ordered by 'total_orders' in descending order.

	age_range	total_orders
1	Unknown	115 729
2	35-44	114 209
3	25-34	75 236
4	18-24	40 620
5	45-54	39 683

4. Запит для отримання кількості успішних поїздок, загальної кількості запитів та % завершення.

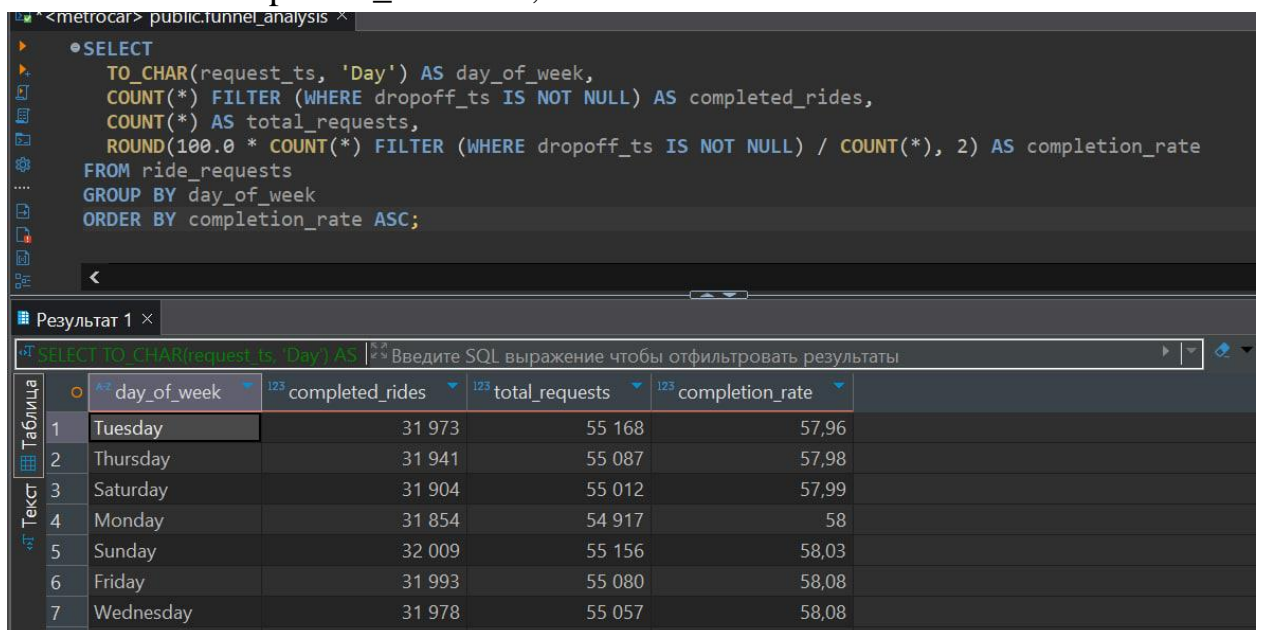
SELECT

```
TO_CHAR(request_ts, 'Day') AS day_of_week,  
COUNT(*) FILTER (WHERE dropoff_ts IS NOT NULL) AS completed_rides,  
COUNT(*) AS total_requests,  
ROUND(100.0 * COUNT(*) FILTER (WHERE dropoff_ts IS NOT NULL) /  
COUNT(*), 2) AS completion_rate
```

FROM ride_requests

GROUP BY day_of_week

ORDER BY completion_rate ASC;



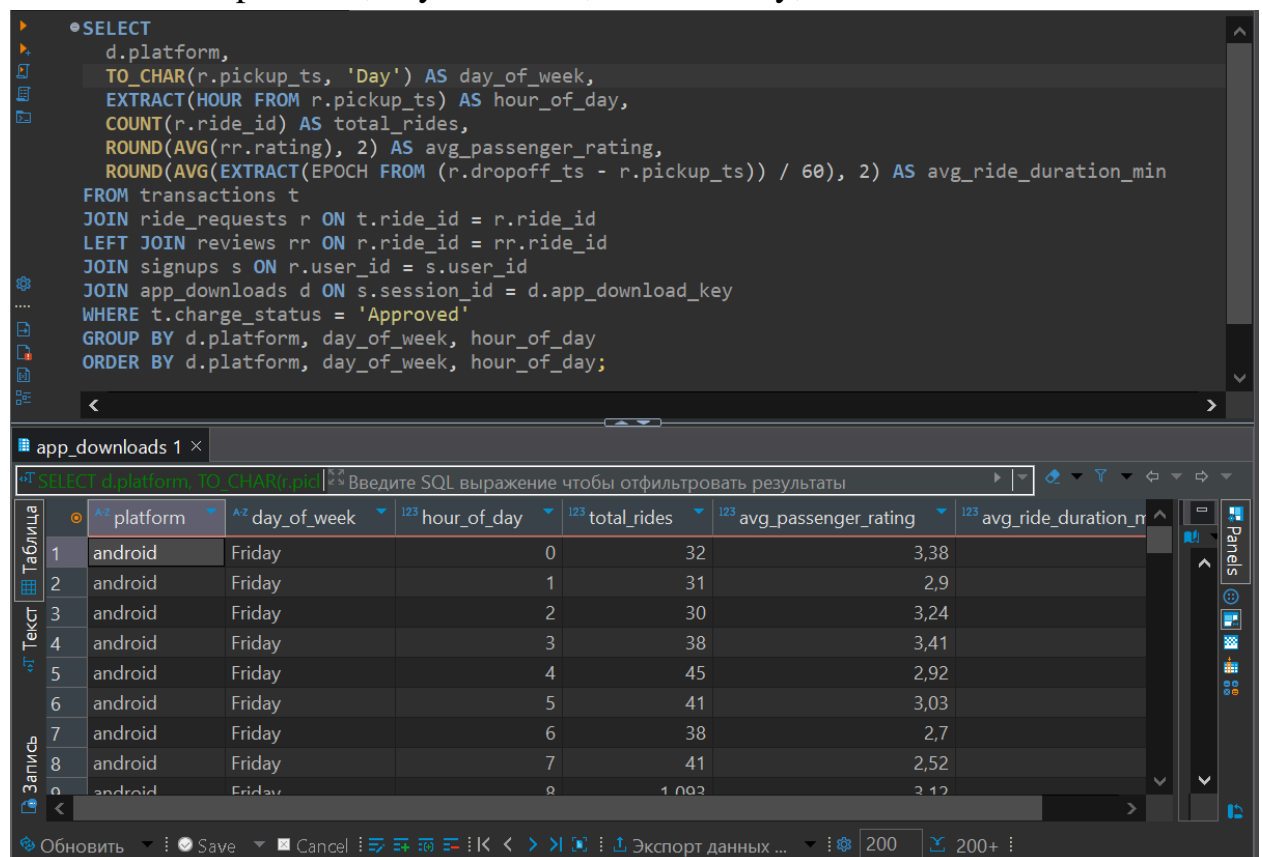
Результат 1 ×

Введите SQL выражение чтобы отфильтровать результаты

	day_of_week	completed_rides	total_requests	completion_rate
1	Tuesday	31 973	55 168	57,96
2	Thursday	31 941	55 087	57,98
3	Saturday	31 904	55 012	57,99
4	Monday	31 854	54 917	58
5	Sunday	32 009	55 156	58,03
6	Friday	31 993	55 080	58,08
7	Wednesday	31 978	55 057	58,08

5. Запит для відображення загальної кількості поїздов, середнього рейтингу, тривалості поїздки в залежності від платформи, дня тижня та часу доби.

```
SELECT
d.platform,
TO_CHAR(r.pickup_ts, 'Day') AS day_of_week, -- Названия дней недели
EXTRACT(HOUR FROM r.pickup_ts) AS hour_of_day,
COUNT(r.ride_id) AS total_rides,
ROUND(AVG(rr.rating), 2) AS avg_passenger_rating,
ROUND(AVG(EXTRACT(EPOCH FROM (r.dropoff_ts - r.pickup_ts)) / 60), 2)
AS avg_ride_duration_min
FROM transactions t
JOIN ride_requests r ON t.ride_id = r.ride_id
LEFT JOIN reviews rr ON r.ride_id = rr.ride_id
JOIN signups s ON r.user_id = s.user_id
JOIN app_downloads d ON s.session_id = d.app_download_key
WHERE t.charge_status = 'Approved'
GROUP BY d.platform, day_of_week, hour_of_day
ORDER BY d.platform, day_of_week, hour_of_day;
```

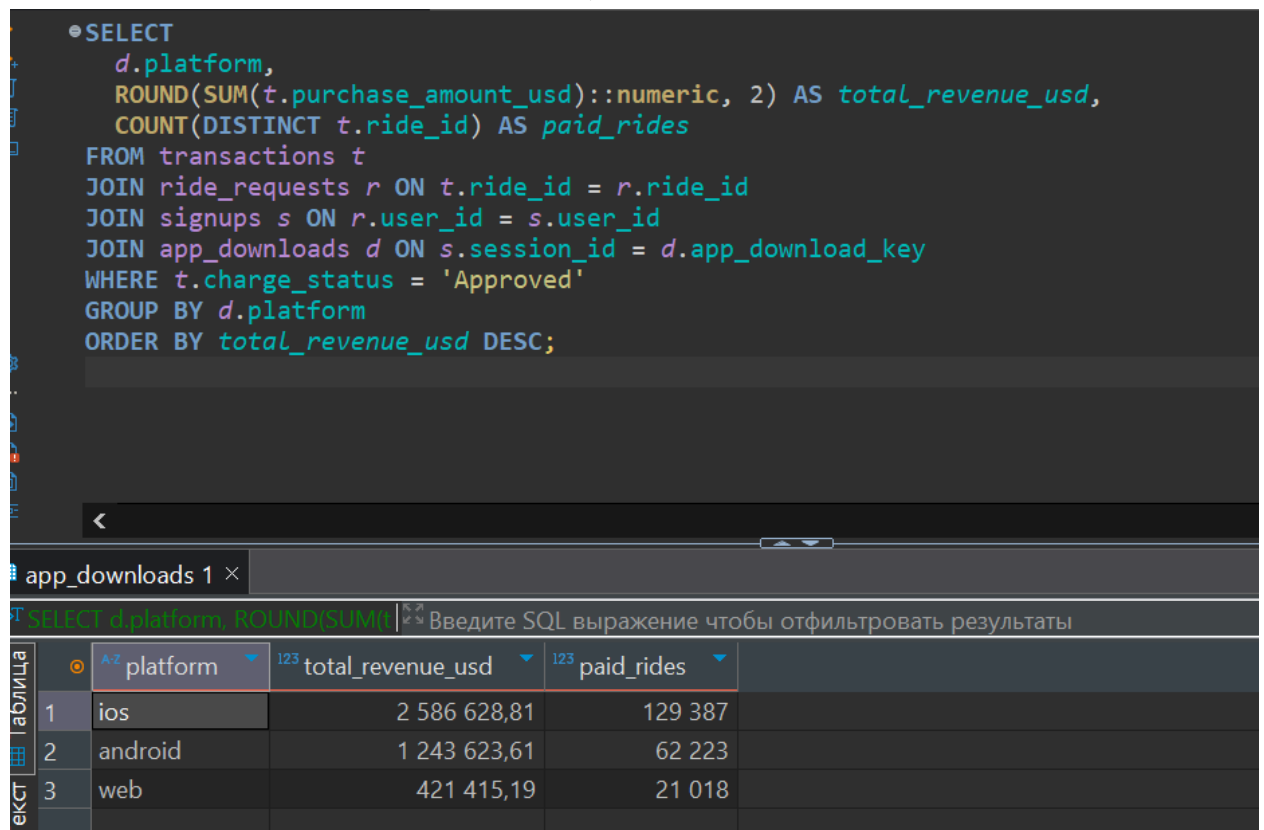


The screenshot shows a SQL IDE interface. The top pane contains the SQL query. The bottom pane shows the results of the query in a table format. The table has 8 columns: platform, day_of_week, hour_of_day, total_rides, avg_passenger_rating, and avg_ride_duration_min. The results are grouped by platform (android) and day_of_week (Friday), with rows representing different hours of the day (0 to 8).

	platform	day_of_week	hour_of_day	total_rides	avg_passenger_rating	avg_ride_duration_min
1	android	Friday	0	32	3,38	
2	android	Friday	1	31	2,9	
3	android	Friday	2	30	3,24	
4	android	Friday	3	38	3,41	
5	android	Friday	4	45	2,92	
6	android	Friday	5	41	3,03	
7	android	Friday	6	38	2,7	
8	android	Friday	7	41	2,52	
9	android	Friday	8	1 093	3,12	

6. Запит для перегляду, скільки доходу ми отримуємо з кожної платформи та скільки поїздок було оплачено.

```
SELECT
  d.platform,
  ROUND(SUM(t.purchase_amount_usd)::numeric, 2) AS total_revenue_usd,
  COUNT(DISTINCT t.ride_id) AS paid_rides
FROM transactions t
JOIN ride_requests r ON t.ride_id = r.ride_id
JOIN signups s ON r.user_id = s.user_id
JOIN app_downloads d ON s.session_id = d.app_download_key
WHERE t.charge_status = 'Approved'
GROUP BY d.platform
ORDER BY total_revenue_usd DESC;
```



The screenshot shows a SQL IDE interface. The top pane contains the SQL query. The bottom pane shows the results of the query in a table format. The table has three columns: platform, total_revenue_usd, and paid_rides. The results are ordered by total_revenue_usd in descending order.

```
SELECT
  d.platform,
  ROUND(SUM(t.purchase_amount_usd)::numeric, 2) AS total_revenue_usd,
  COUNT(DISTINCT t.ride_id) AS paid_rides
FROM transactions t
JOIN ride_requests r ON t.ride_id = r.ride_id
JOIN signups s ON r.user_id = s.user_id
JOIN app_downloads d ON s.session_id = d.app_download_key
WHERE t.charge_status = 'Approved'
GROUP BY d.platform
ORDER BY total_revenue_usd DESC;
```

	platform	total_revenue_usd	paid_rides
1	ios	2 586 628,81	129 387
2	android	1 243 623,61	62 223
3	web	421 415,19	21 018