

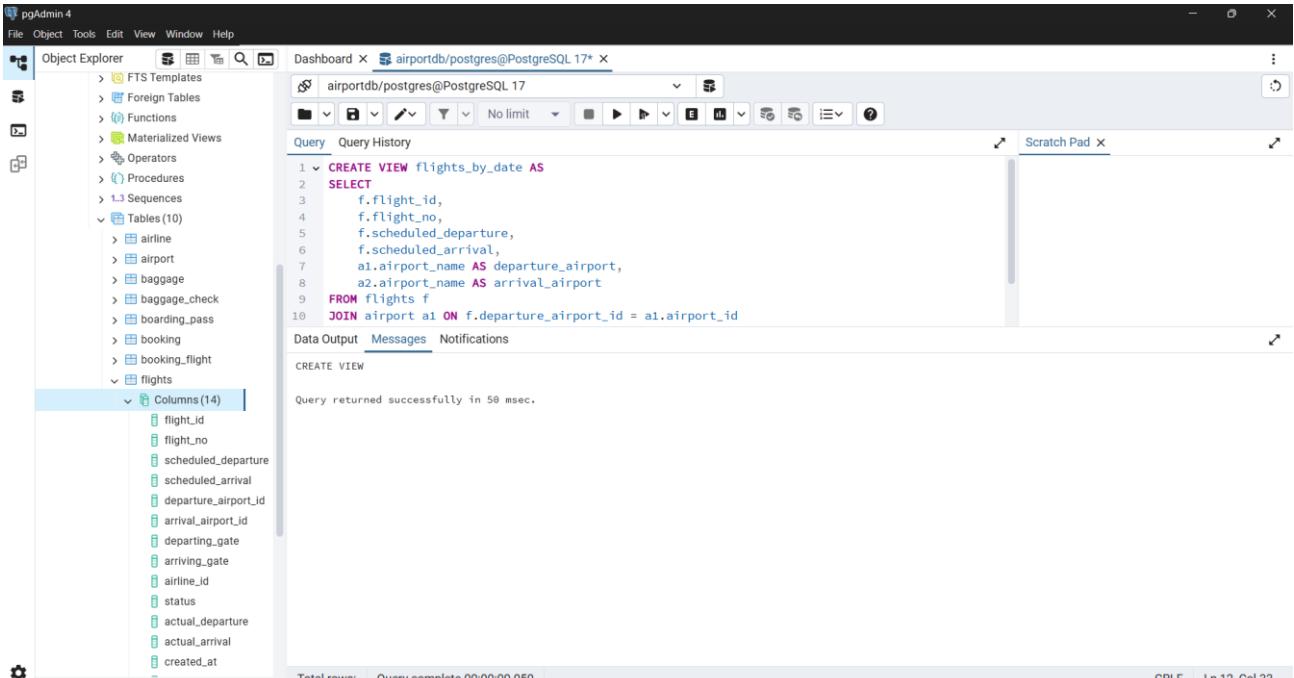
## Laboratory work 8

We continue to work with the database from the previous laboratory works.

Take a full-page screenshot that covers the code and results of each task.

VIEW.

1. Create a view to show details of all flights that are departing on a specific date.



The screenshot shows the pgAdmin 4 interface. The left sidebar is the Object Explorer, displaying various database objects like FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (10). The 'flights' table is selected, and its columns (14) are listed in the bottom-left pane. The main query editor window contains the following SQL code:

```
CREATE VIEW flights_by_date AS
SELECT
    f.flight_id,
    f.flight_no,
    f.scheduled_departure,
    f.scheduled_arrival,
    a1.airport_name AS departure_airport,
    a2.airport_name AS arrival_airport
FROM flights f
JOIN airport a1 ON f.departure_airport_id = a1.airport_id
JOIN airport a2 ON f.arrival_airport_id = a2.airport_id
```

The status bar at the bottom indicates "Query returned successfully in 50 msec." and "Total rows: 0".

2. Create a view that shows bookings for flights scheduled to depart within the next week.

```

CREATE VIEW bookings_next_week AS
SELECT
    b.booking_id,
    b.created_at,
    f.flight_no,
    f.scheduled_departure,
    p.first_name,
    p.last_name
FROM booking b
JOIN booking_flight bf ON b.booking_id = bf.booking_id

```

Total rows: Query complete 00:00:00.076 CRLF Ln 5, Col 16

3. Create a view to show the top 5 most popular flight routes based on the number of bookings.

```

CREATE VIEW top5_routes AS
SELECT
    a1.airport_name AS departure,
    a2.airport_name AS arrival,
    COUNT(bf.booking_id) AS total_bookings
FROM flights f
JOIN booking_flight bf ON f.flight_id = bf.flight_id
JOIN airport a1 ON f.departure_airport_id = a1.airport_id
JOIN airport a2 ON f.arrival_airport_id = a2.airport_id
GROUP BY a1.airport_name, a2.airport_name
ORDER BY total_bookings DESC
LIMIT 5;

```

Total rows: Query complete 00:00:00.095 CRLF Ln 1, Col 19

4. Create a view that lists all flights for a specific airline.

The screenshot shows the pgAdmin 4 interface with the 'Object Explorer' on the left and a 'Query' window on the right. The 'flights' table is selected in the Object Explorer. A query is being run in the Query window:

```

CREATE VIEW flights_by_airline AS
SELECT
    f.flight_id,
    f.flight_no,
    al.airline_name,
    f.scheduled_departure,
    f.scheduled_arrival
FROM flights f
JOIN airline al ON f.airline_id = al.airline_id
WHERE al.airline_name = 'Air Astana';

```

The 'Messages' tab in the results shows: 'CREATE VIEW'. Below it, it says 'Query returned successfully in 138 msec.'.

5. Modify the view created in task 4 to show only flights departing within the next 7 days for a specific airline.

The screenshot shows the pgAdmin 4 interface with the 'Object Explorer' on the left and a 'Query' window on the right. The 'flights' table is selected in the Object Explorer. A query is being run in the Query window:

```

CREATE OR REPLACE VIEW flights_by_airline AS
SELECT
    f.flight_id,
    f.flight_no,
    al.airline_name,
    f.scheduled_departure,
    f.scheduled_arrival
FROM flights f
JOIN airline al ON f.airline_id = al.airline_id
WHERE al.airline_name = 'Air Astana'
AND f.scheduled_departure BETWEEN NOW() AND NOW() + INTERVAL '7 days';

```

The 'Messages' tab in the results shows: 'CREATE VIEW'. Below it, it says 'Query returned successfully in 69 msec.'

6. Create a view to show flights that are delayed by more than 24 hours.

7. Create a view in which you can display the full name and country of origin of passengers who made bookings on Leffler-Thompson platform. Then show the list of that passengers.

The screenshot shows the pgAdmin 4 interface with the 'Object Explorer' on the left and a 'Query' window on the right. The 'Query' window contains the following SQL code:

```

1 v CREATE VIEW leffler_passengers AS
2 SELECT
3     p.first_name || ' ' || p.last_name AS full_name,
4     p.country_of_citizenship
5 FROM passengers p
6 JOIN booking b ON p.passenger_id = b.passenger_id
7 WHERE b.booking_platform = 'Leffler-Thompson';

```

The 'Messages' tab below the query window shows the message: "Query returned successfully in 122 msec." The status bar at the bottom indicates "Total rows: Query complete 00:00:00.122" and "CRLF Ln 7, Col 17".

## 8. Create a view that shows top 10 most visited countries.

The screenshot shows the pgAdmin 4 interface with the 'Object Explorer' on the left and a 'Query' window on the right. The 'Query' window contains the following SQL code:

```

1 v CREATE VIEW top10_countries AS
2 SELECT
3     a2.country AS destination_country,
4     COUNT(bf.booking_id) AS total_visits
5 FROM flights f
6 JOIN booking_flight bf ON f.flight_id = bf.flight_id
7 JOIN airport a2 ON bf.arrival_airport_id = a2.airport_id
8 GROUP BY a2.country
9 ORDER BY total_visits DESC
10 LIMIT 10;

```

The 'Messages' tab below the query window shows the message: "Query returned successfully in 67 msec." The status bar at the bottom indicates "Total rows: Query complete 00:00:00.067" and "CRLF Ln 11, Col 1".

## 9. Update any of the created views by adding new information in the view table. Show results.

The screenshot shows the pgAdmin 4 interface with the 'Object Explorer' on the left and a query editor on the right. The query editor contains the following SQL code:

```

CREATE OR REPLACE VIEW flights_by_date AS
SELECT
    f.flight_id,
    f.flight_no,
    f.scheduled_departure,
    f.scheduled_arrival,
    a1.airport_name AS departure_airport,
    a2.airport_name AS arrival_airport,
    f.status
FROM flights f
JOIN airport a1 ON f.departure_airport_id = a1.airport_id
JOIN airport a2 ON f.arrival_airport_id = a2.airport_id
WHERE DATE(f.scheduled_departure) = '2025-11-13';

```

The status bar at the bottom indicates "Query returned successfully in 148 msec." and "Total rows: Query complete 00:00:00.148".

## 10. Drop all existing views.

The screenshot shows the pgAdmin 4 interface with the 'Object Explorer' on the left and a query editor on the right. The query editor contains the following SQL code:

```

DROP VIEW IF EXISTS
    flights_by_date,
    bookings_next_week,
    top5_routes,
    flights_by_airline,
    delayed_flights,
    leffler_passengers,
    top10_countries
CASCADE;

```

The status bar at the bottom indicates "Query returned successfully in 132 msec." and "Total rows: Query complete 00:00:00.132". A note in the messages section says: "ЗАМЕЧАНИЕ: представление 'delayed\_flights' не существует, пропускается".