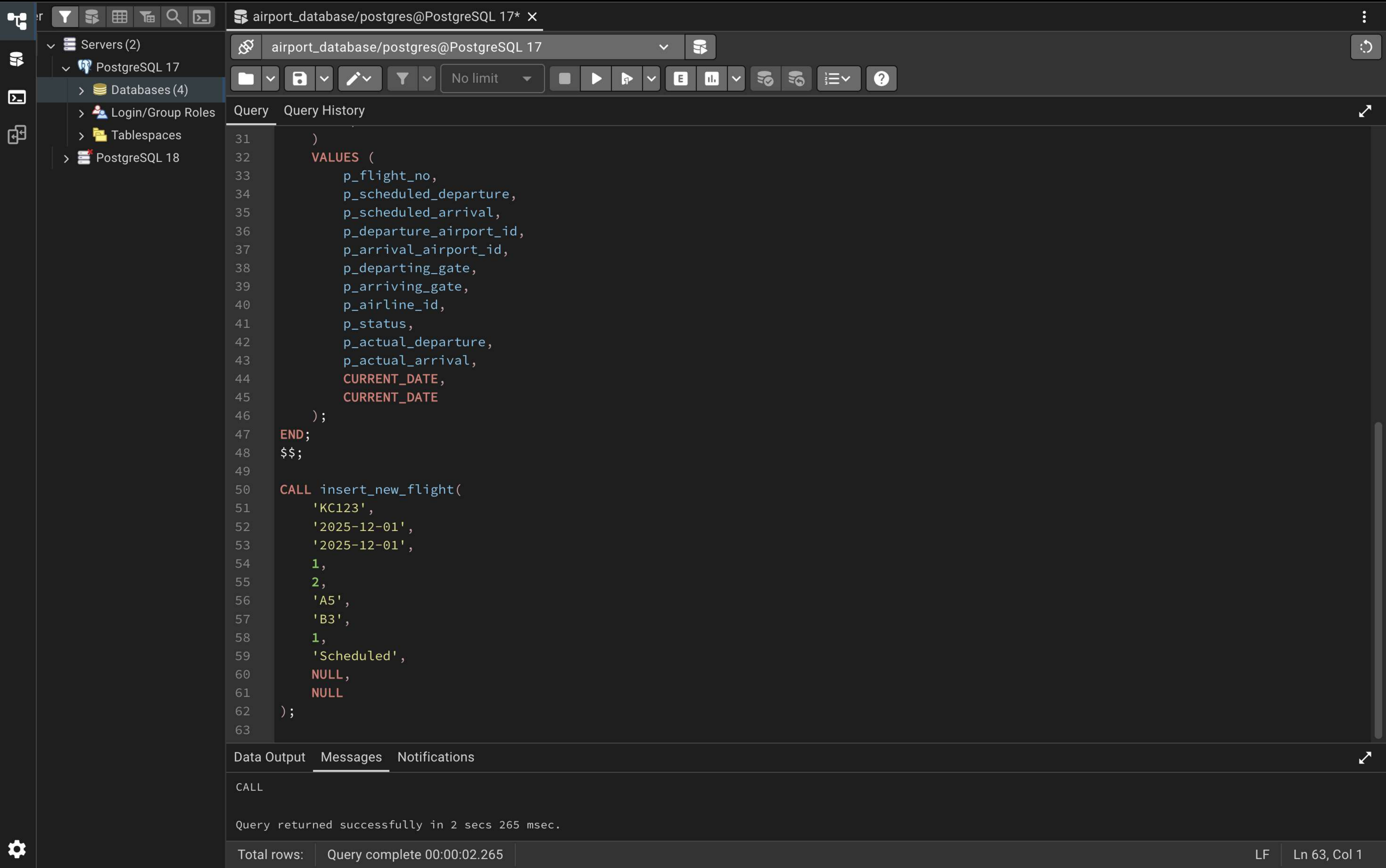
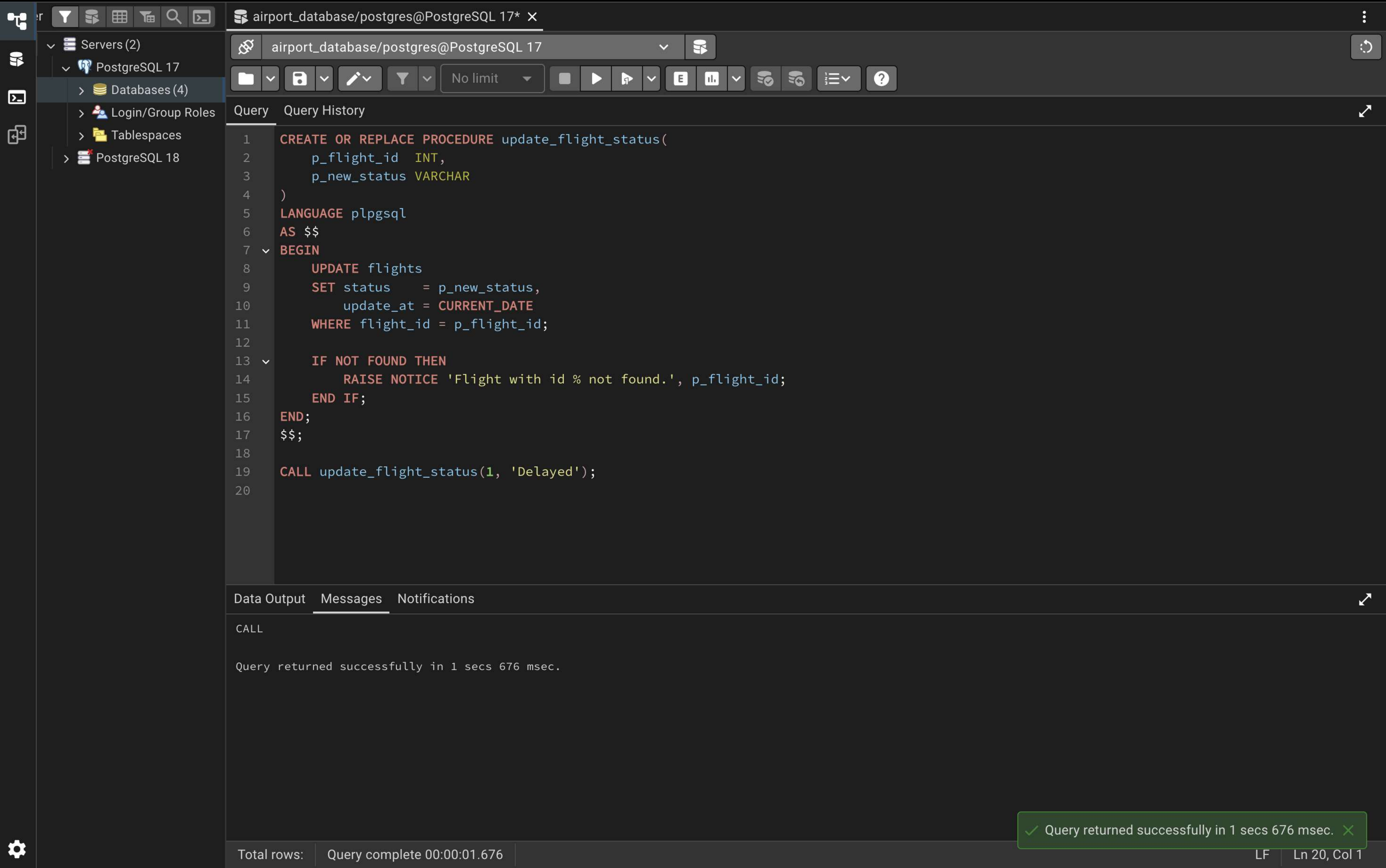


```
    )
    VALUES (
        p_flight_no,
        p_scheduled_departure,
        p_scheduled_arrival,
        p_departure_airport_id,
        p_arrival_airport_id,
        p_departing_gate,
        p_arriving_gate,
        p_airline_id,
        p_status,
        p_actual_departure,
        p_actual_arrival,
        CURRENT_DATE,
        CURRENT_DATE
    );
END;
$$;

CALL insert_new_flight(
    'KC123',
    '2025-12-01',
    '2025-12-01',
    1,
    2,
    'A5',
    'B3',
    1,
    'Scheduled',
    NULL,
    NULL
);
```

```
CALL

Query returned successfully in 2 secs 265 msec.
```

Total rows: Query complete 00:00:02.265 LF Ln 63, Col 1

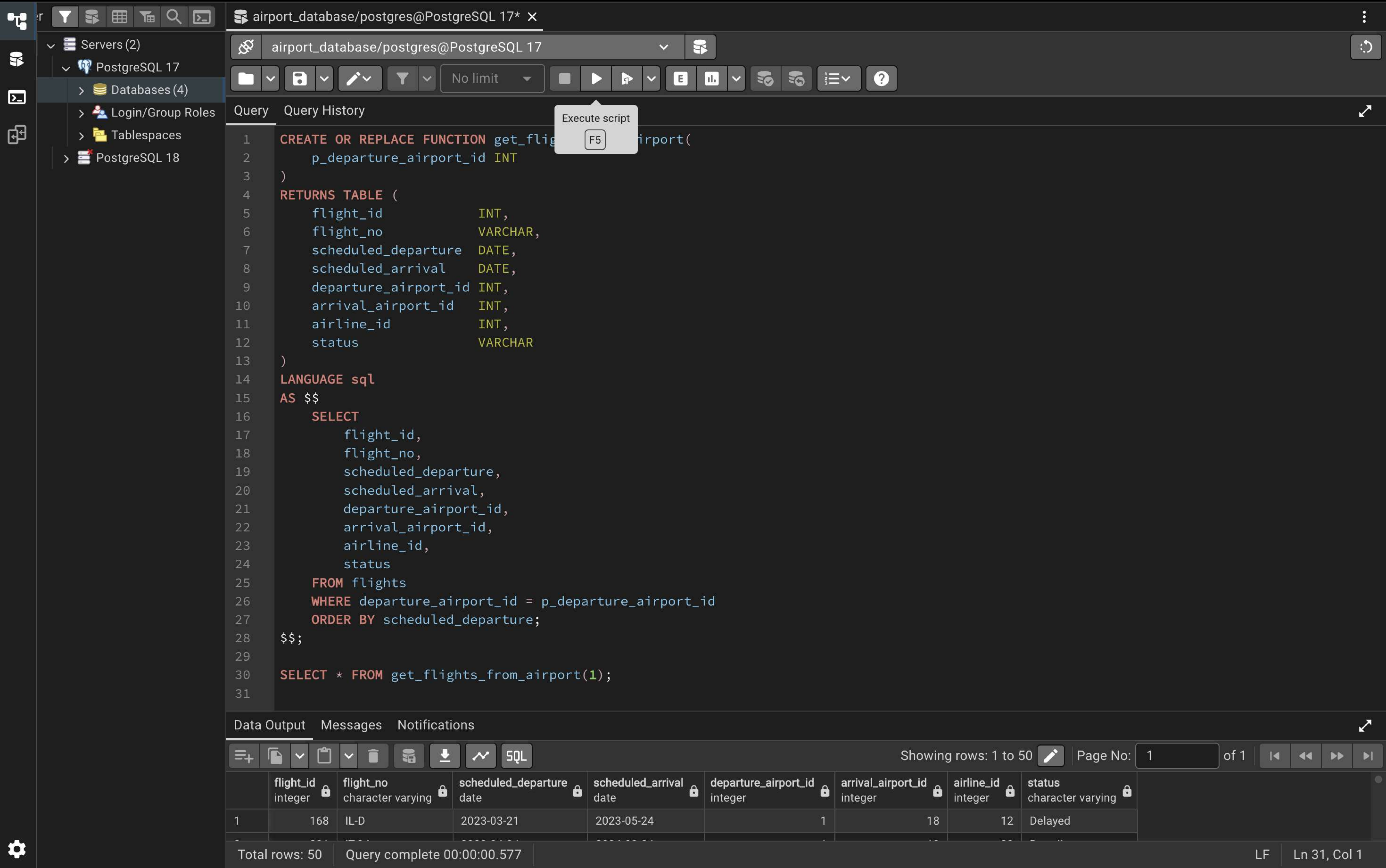


```sql
CREATE OR REPLACE PROCEDURE update_flight_status(
    p_flight_id  INT,
    p_new_status VARCHAR
)
LANGUAGE plpgsql
AS $$
BEGIN
    UPDATE flights
    SET status    = p_new_status,
        update_at = CURRENT_DATE
    WHERE flight_id = p_flight_id;

    IF NOT FOUND THEN
        RAISE NOTICE 'Flight with id % not found.', p_flight_id;
    END IF;
END;
$$;

CALL update_flight_status(1, 'Delayed');
```

```
CALL

Query returned successfully in 1 secs 676 msec.
```

Query returned successfully in 1 secs 676 msec.

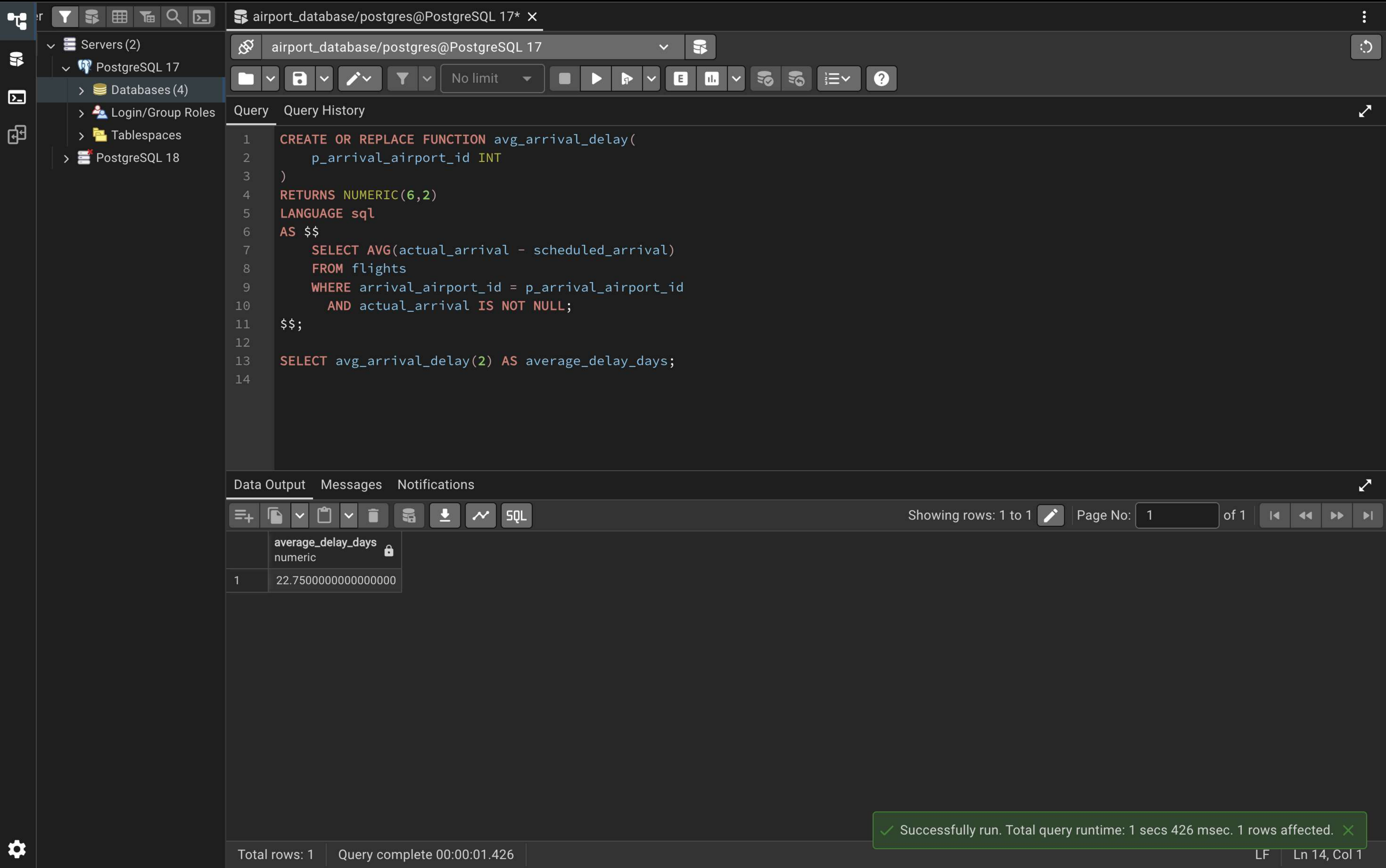
Total rows: Query complete 00:00:01.676   LF   Ln 20, Col 1



```
CREATE OR REPLACE FUNCTION get_flig                irport(
    p_departure_airport_id INT
)
RETURNS TABLE (
    flight_id            INT,
    flight_no            VARCHAR,
    scheduled_departure  DATE,
    scheduled_arrival    DATE,
    departure_airport_id INT,
    arrival_airport_id   INT,
    airline_id           INT,
    status               VARCHAR
)
LANGUAGE sql
AS $$
    SELECT
        flight_id,
        flight_no,
        scheduled_departure,
        scheduled_arrival,
        departure_airport_id,
        arrival_airport_id,
        airline_id,
        status
    FROM flights
    WHERE departure_airport_id = p_departure_airport_id
    ORDER BY scheduled_departure;
$$;

SELECT * FROM get_flights_from_airport(1);
```

	flight_id integer	flight_no character varying	scheduled_departure date	scheduled_arrival date	departure_airport_id integer	arrival_airport_id integer	airline_id integer	status character varying
1	168	IL-D	2023-03-21	2023-05-24	1	18	12	Delayed

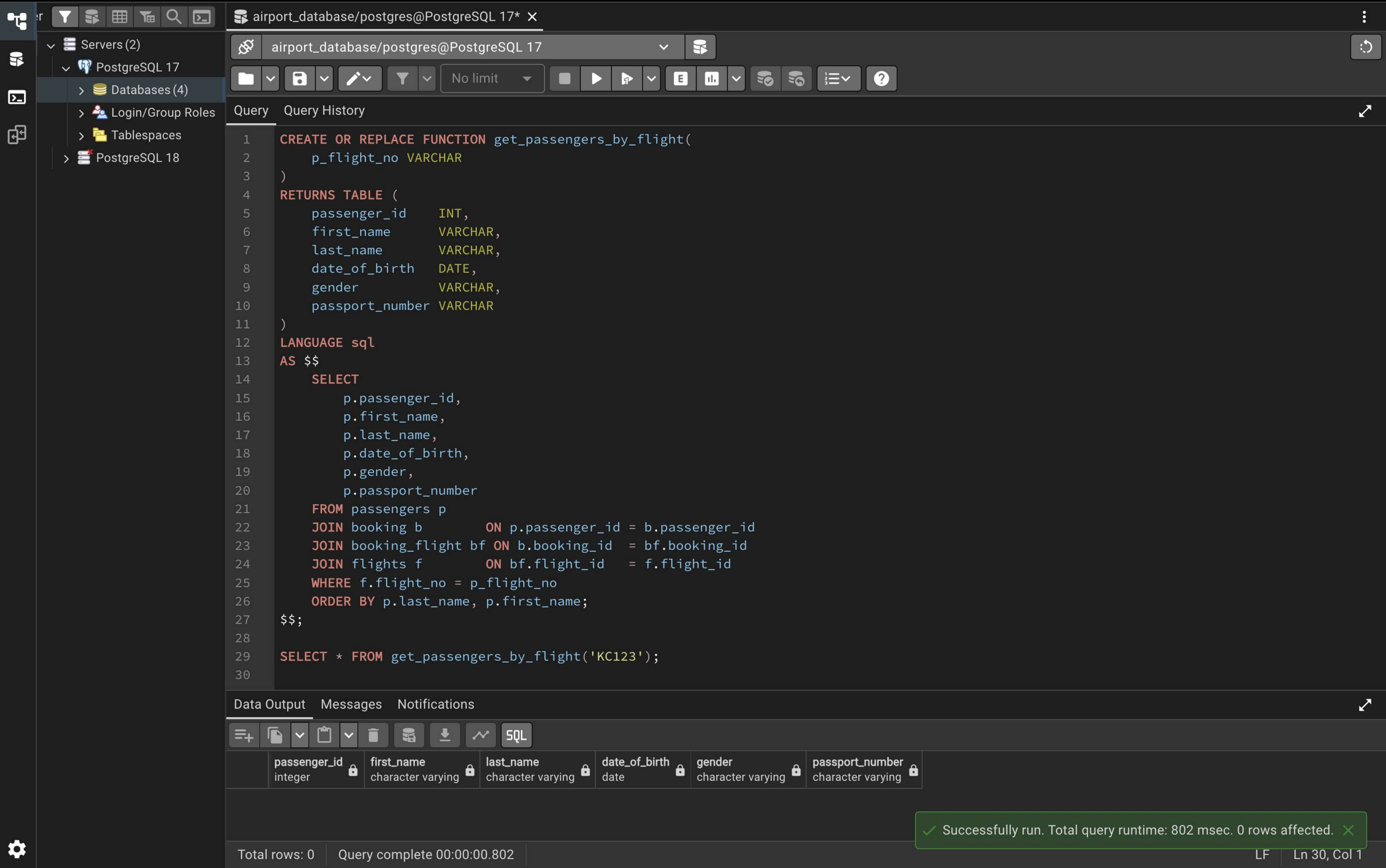
Total rows: 50   Query complete 00:00:00.577   LF   Ln 31, Col 1



```sql
CREATE OR REPLACE FUNCTION avg_arrival_delay(
    p_arrival_airport_id INT
)
RETURNS NUMERIC(6,2)
LANGUAGE sql
AS $$
    SELECT AVG(actual_arrival - scheduled_arrival)
    FROM flights
    WHERE arrival_airport_id = p_arrival_airport_id
      AND actual_arrival IS NOT NULL;
$$;

SELECT avg_arrival_delay(2) AS average_delay_days;
```

	average_delay_days numeric
1	22.7500000000000000

```
Successfully run. Total query runtime: 1 secs 426 msec. 1 rows affected.
Total rows: 1    Query complete 00:00:01.426    LF    Ln 14, Col 1
```

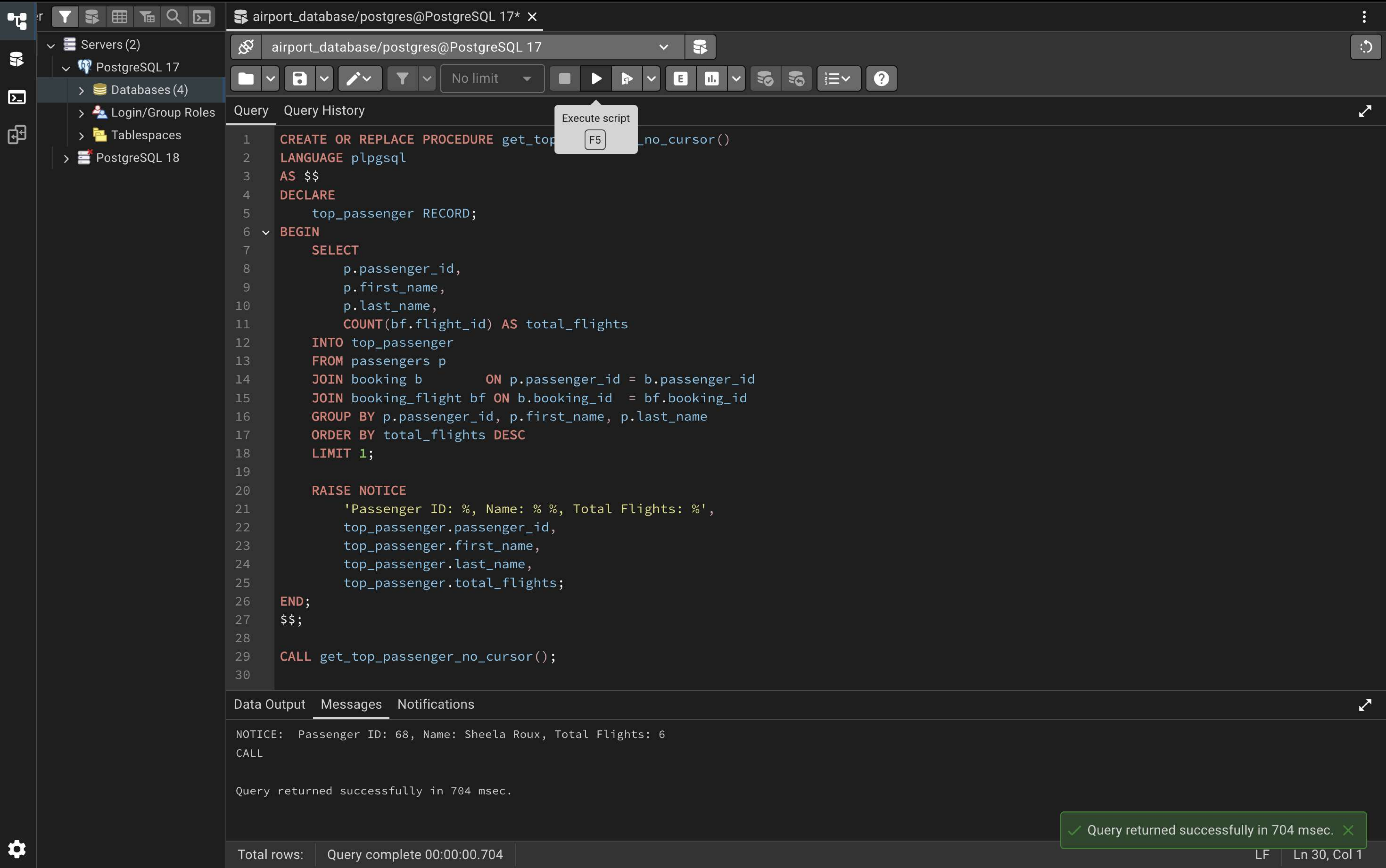


```
CREATE OR REPLACE FUNCTION get_passengers_by_flight(
    p_flight_no VARCHAR
)
RETURNS TABLE (
    passenger_id    INT,
    first_name      VARCHAR,
    last_name       VARCHAR,
    date_of_birth   DATE,
    gender          VARCHAR,
    passport_number VARCHAR
)
LANGUAGE sql
AS $$
    SELECT
        p.passenger_id,
        p.first_name,
        p.last_name,
        p.date_of_birth,
        p.gender,
        p.passport_number
    FROM passengers p
    JOIN booking b        ON p.passenger_id = b.passenger_id
    JOIN booking_flight bf ON b.booking_id  = bf.booking_id
    JOIN flights f        ON bf.flight_id   = f.flight_id
    WHERE f.flight_no = p_flight_no
    ORDER BY p.last_name, p.first_name;
$$;

SELECT * FROM get_passengers_by_flight('KC123');
```

passenger_id integer	first_name character varying	last_name character varying	date_of_birth date	gender character varying	passport_number character varying

Successfully run. Total query runtime: 802 msec. 0 rows affected.

Total rows: 0 Query complete 00:00:00.802 LF Ln 30, Col 1

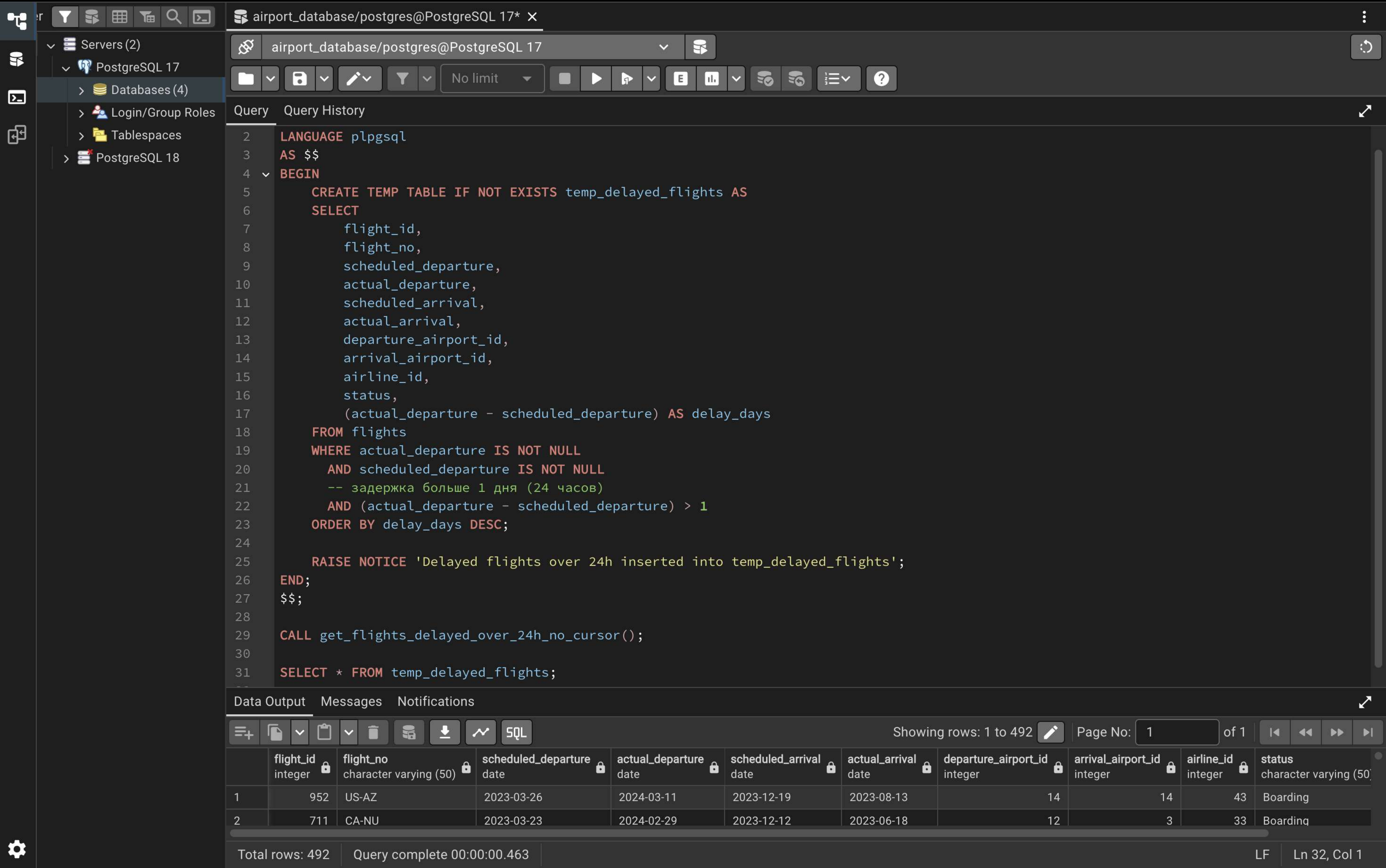


```
CREATE OR REPLACE PROCEDURE get_top          _no_cursor()
LANGUAGE plpgsql
AS $$
DECLARE
    top_passenger RECORD;
BEGIN
    SELECT
        p.passenger_id,
        p.first_name,
        p.last_name,
        COUNT(bf.flight_id) AS total_flights
    INTO top_passenger
    FROM passengers p
    JOIN booking b        ON p.passenger_id = b.passenger_id
    JOIN booking_flight bf ON b.booking_id  = bf.booking_id
    GROUP BY p.passenger_id, p.first_name, p.last_name
    ORDER BY total_flights DESC
    LIMIT 1;

    RAISE NOTICE
        'Passenger ID: %, Name: % %, Total Flights: %',
        top_passenger.passenger_id,
        top_passenger.first_name,
        top_passenger.last_name,
        top_passenger.total_flights;
END;
$$;

CALL get_top_passenger_no_cursor();
```

```
NOTICE:  Passenger ID: 68, Name: Sheela Roux, Total Flights: 6
CALL

Query returned successfully in 704 msec.
```

Query returned successfully in 704 msec.

Total rows: Query complete 00:00:00.704 LF Ln 30, Col 1

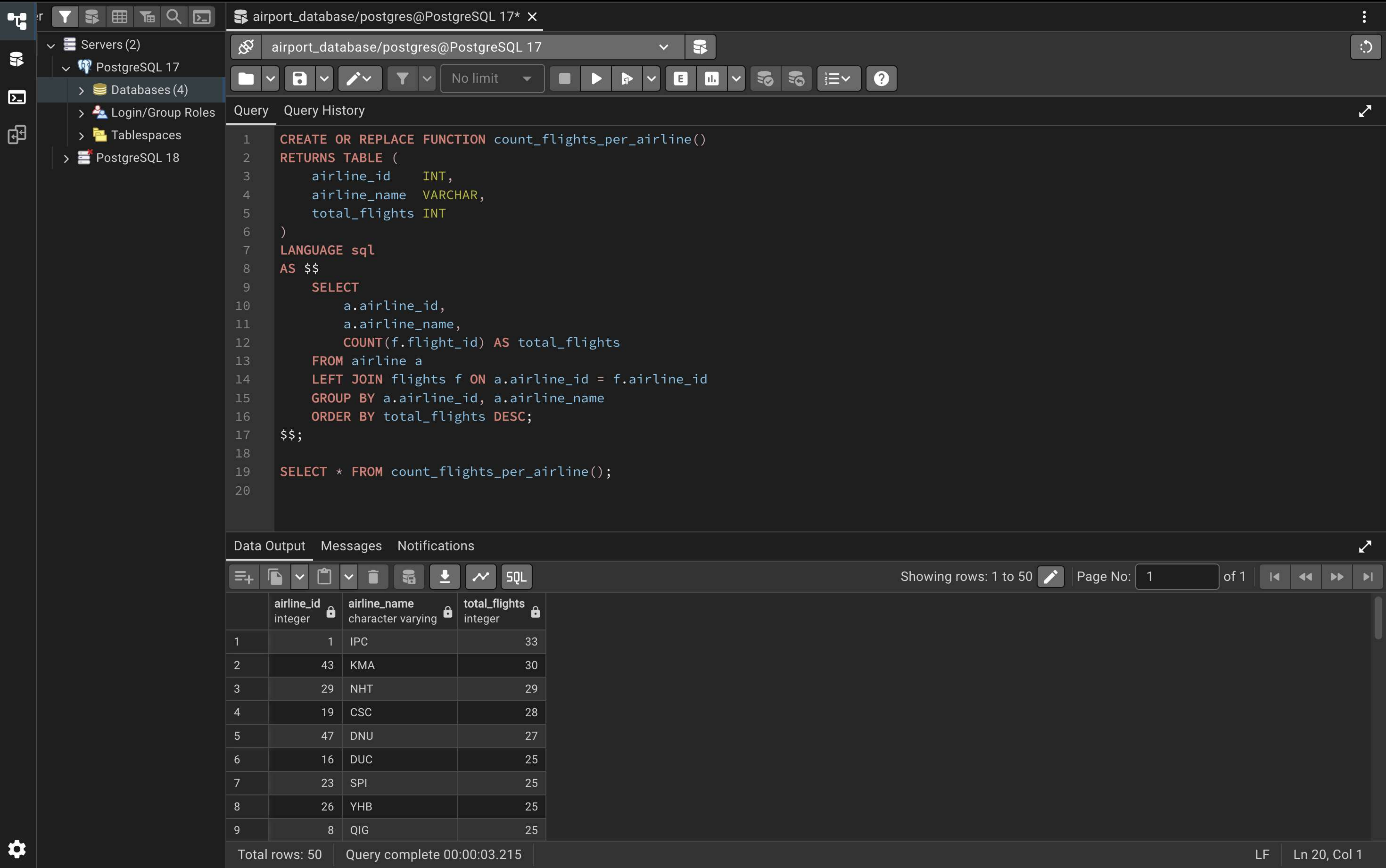


```sql
LANGUAGE plpgsql
AS $$
BEGIN
    CREATE TEMP TABLE IF NOT EXISTS temp_delayed_flights AS
    SELECT
        flight_id,
        flight_no,
        scheduled_departure,
        actual_departure,
        scheduled_arrival,
        actual_arrival,
        departure_airport_id,
        arrival_airport_id,
        airline_id,
        status,
        (actual_departure - scheduled_departure) AS delay_days
    FROM flights
    WHERE actual_departure IS NOT NULL
      AND scheduled_departure IS NOT NULL
      -- задержка больше 1 дня (24 часов)
      AND (actual_departure - scheduled_departure) > 1
    ORDER BY delay_days DESC;

    RAISE NOTICE 'Delayed flights over 24h inserted into temp_delayed_flights';
END;
$$;

CALL get_flights_delayed_over_24h_no_cursor();

SELECT * FROM temp_delayed_flights;
```

Showing rows: 1 to 492   Page No: 1 of 1

	flight_id integer	flight_no character varying (50)	scheduled_departure date	actual_departure date	scheduled_arrival date	actual_arrival date	departure_airport_id integer	arrival_airport_id integer	airline_id integer	status character varying (50)
1	952	US-AZ	2023-03-26	2024-03-11	2023-12-19	2023-08-13	14	14	43	Boarding
2	711	CA-NU	2023-03-23	2024-02-29	2023-12-12	2023-06-18	12	3	33	Boarding

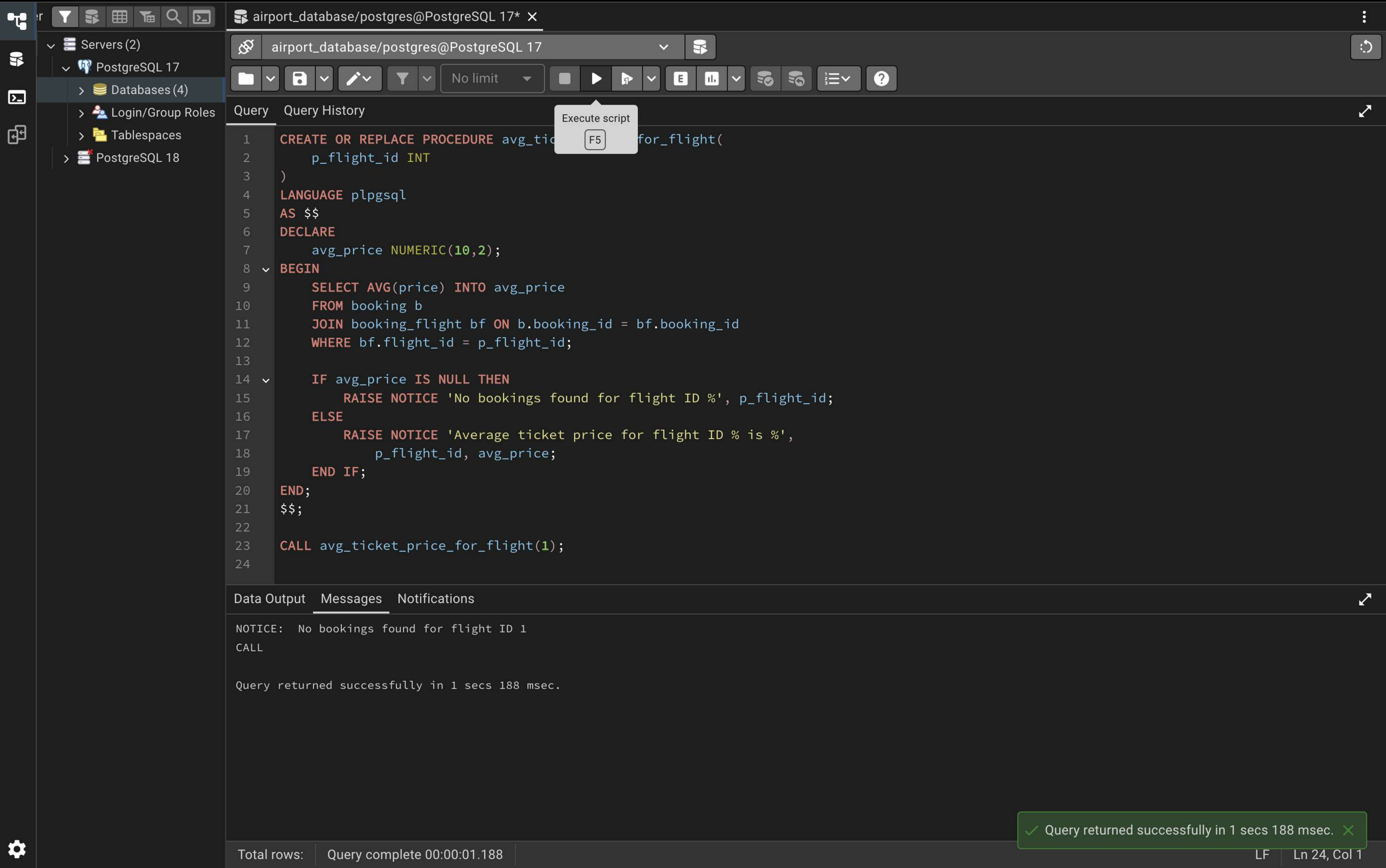
Total rows: 492   Query complete 00:00:00.463   LF   Ln 32, Col 1



```
CREATE OR REPLACE FUNCTION count_flights_per_airline()
RETURNS TABLE (
    airline_id    INT,
    airline_name  VARCHAR,
    total_flights INT
)
LANGUAGE sql
AS $$
    SELECT
        a.airline_id,
        a.airline_name,
        COUNT(f.flight_id) AS total_flights
    FROM airline a
    LEFT JOIN flights f ON a.airline_id = f.airline_id
    GROUP BY a.airline_id, a.airline_name
    ORDER BY total_flights DESC;
$$;

SELECT * FROM count_flights_per_airline();
```

	airline_id integer	airline_name character varying	total_flights integer
1	1	IPC	33
2	43	KMA	30
3	29	NHT	29
4	19	CSC	28
5	47	DNU	27
6	16	DUC	25
7	23	SPI	25
8	26	YHB	25
9	8	QIG	25

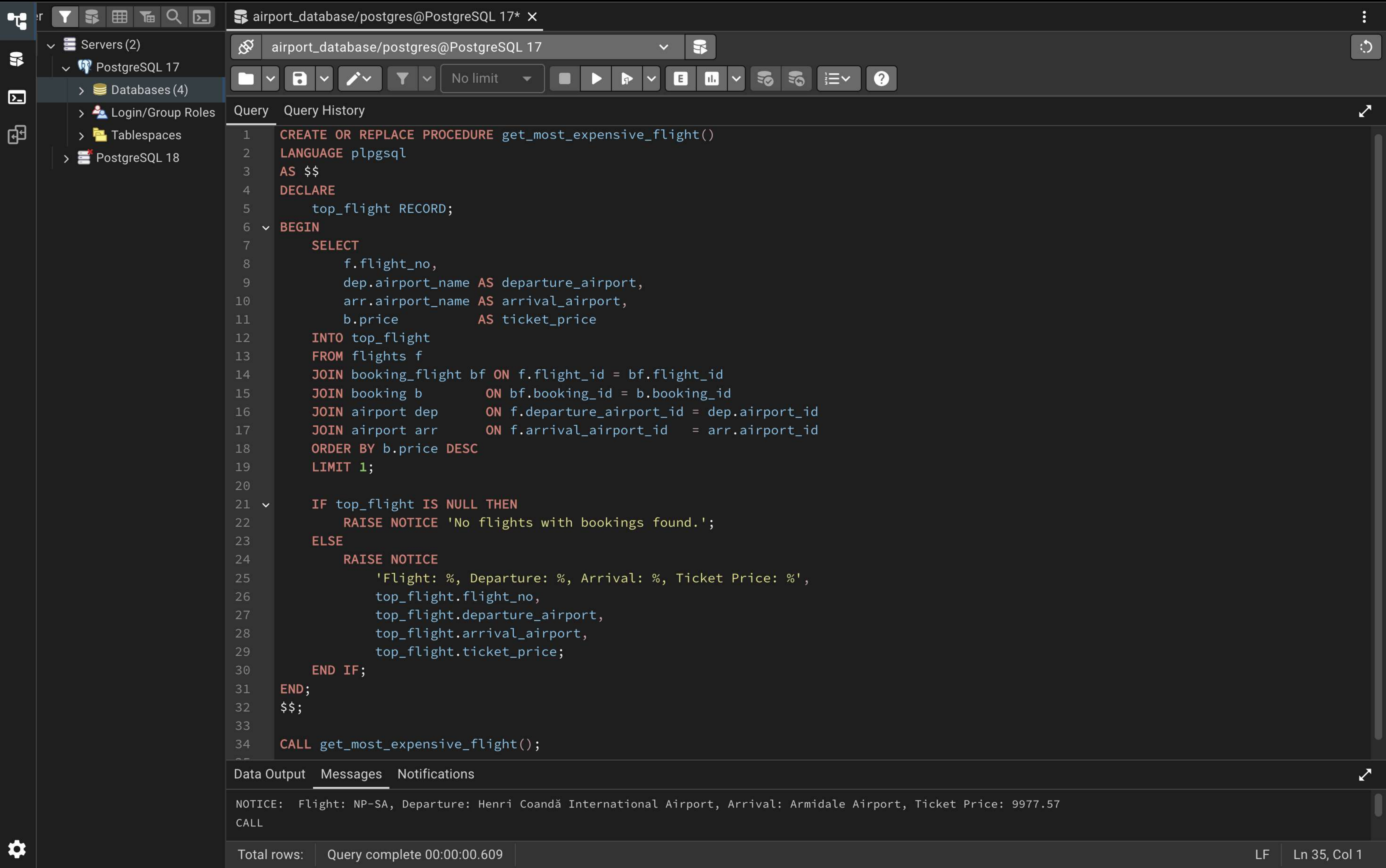
Total rows: 50   Query complete 00:00:03.215   LF   Ln 20, Col 1



```sql
CREATE OR REPLACE PROCEDURE avg_tic           for_flight(
    p_flight_id INT
)
LANGUAGE plpgsql
AS $$
DECLARE
    avg_price NUMERIC(10,2);
BEGIN
    SELECT AVG(price) INTO avg_price
    FROM booking b
    JOIN booking_flight bf ON b.booking_id = bf.booking_id
    WHERE bf.flight_id = p_flight_id;

    IF avg_price IS NULL THEN
        RAISE NOTICE 'No bookings found for flight ID %', p_flight_id;
    ELSE
        RAISE NOTICE 'Average ticket price for flight ID % is %',
            p_flight_id, avg_price;
    END IF;
END;
$$;

CALL avg_ticket_price_for_flight(1);
```

```
NOTICE:  No bookings found for flight ID 1
CALL

Query returned successfully in 1 secs 188 msec.
```

✓ Query returned successfully in 1 secs 188 msec.



```sql
CREATE OR REPLACE PROCEDURE get_most_expensive_flight()
LANGUAGE plpgsql
AS $$
DECLARE
    top_flight RECORD;
BEGIN
    SELECT
        f.flight_no,
        dep.airport_name AS departure_airport,
        arr.airport_name AS arrival_airport,
        b.price          AS ticket_price
    INTO top_flight
    FROM flights f
    JOIN booking_flight bf ON f.flight_id = bf.flight_id
    JOIN booking b        ON bf.booking_id = b.booking_id
    JOIN airport dep      ON f.departure_airport_id = dep.airport_id
    JOIN airport arr      ON f.arrival_airport_id   = arr.airport_id
    ORDER BY b.price DESC
    LIMIT 1;

    IF top_flight IS NULL THEN
        RAISE NOTICE 'No flights with bookings found.';
    ELSE
        RAISE NOTICE
            'Flight: %, Departure: %, Arrival: %, Ticket Price: %',
            top_flight.flight_no,
            top_flight.departure_airport,
            top_flight.arrival_airport,
            top_flight.ticket_price;
    END IF;
END;
$$;

CALL get_most_expensive_flight();
```

```
NOTICE:  Flight: NP-SA, Departure: Henri Coandă International Airport, Arrival: Armidale Airport, Ticket Price: 9977.57
CALL
```

Total rows: Query complete 00:00:00.609   LF   Ln 35, Col 1