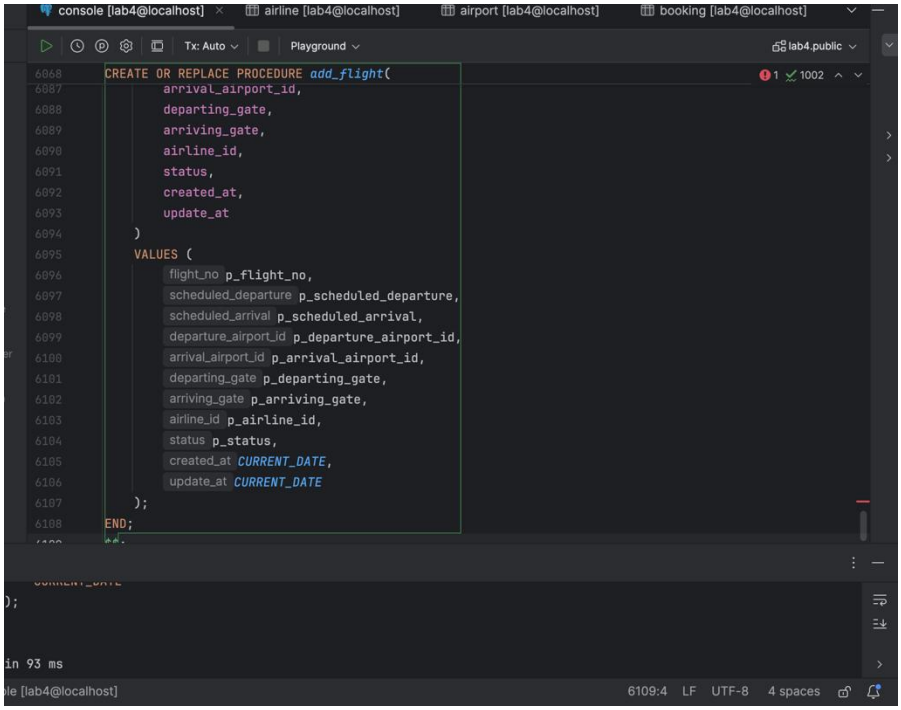
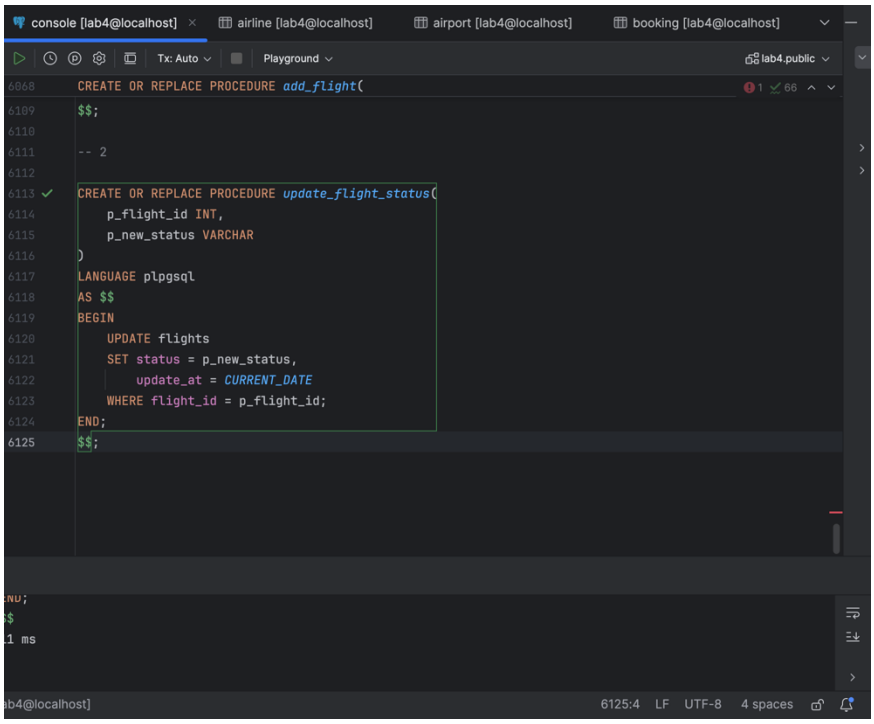
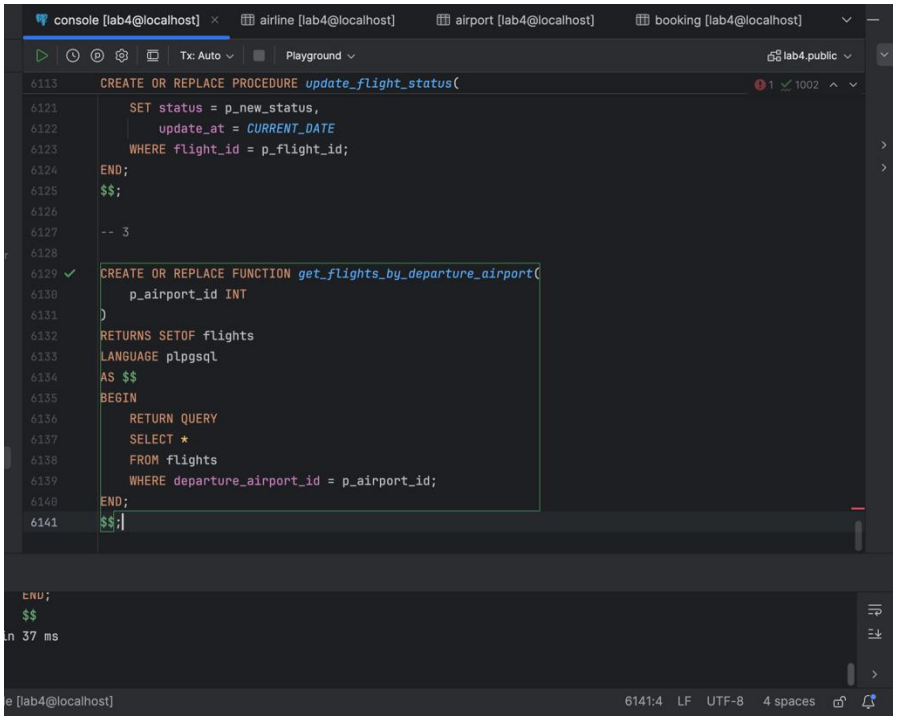
1.


console [lab4@localhost]
airline [lab4@localhost]
airport [lab4@localhost]
booking [lab4@localhost]
Tx: Auto
Playground
lab4.public
CREATE OR REPLACE PROCEDURE add_flight(
arrival_airport_id,
departing_gate,
arriving_gate,
airline_id,
status,
created_at,
update_at
)
VALUES (
flight_no p_flight_no,
scheduled_departure p_scheduled_departure,
scheduled_arrival p_scheduled_arrival,
departure_airport_id p_departure_airport_id,
arrival_airport_id p_arrival_airport_id,
departing_gate p_departing_gate,
arriving_gate p_arriving_gate,
airline_id p_airline_id,
status p_status,
created_at CURRENT_DATE,
update_at CURRENT_DATE
);
END;
in 93 ms
6109:4 LF UTF-8 4 spaces

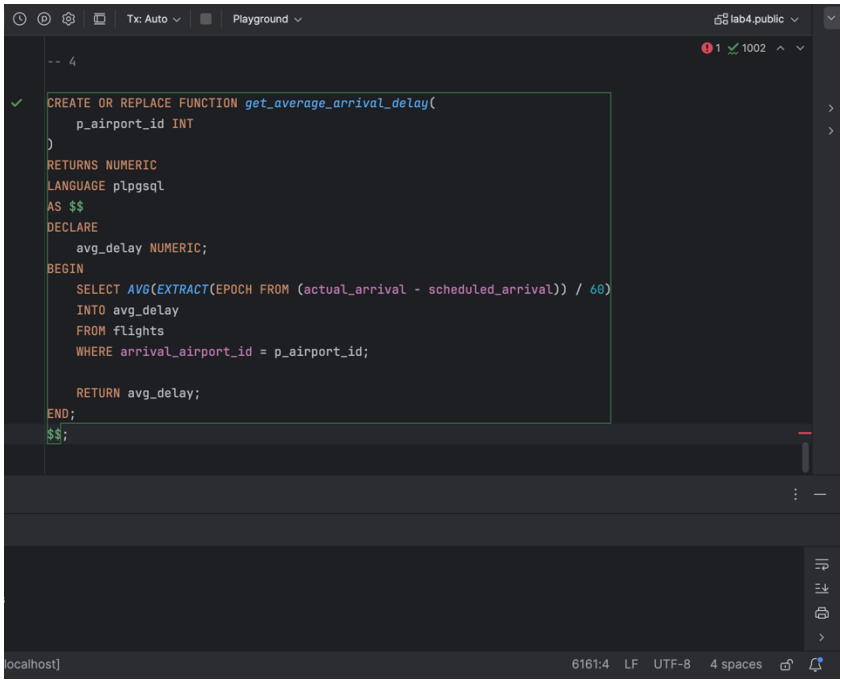
2.


console [lab4@localhost]
airline [lab4@localhost]
airport [lab4@localhost]
booking [lab4@localhost]
Tx: Auto
Playground
lab4.public
CREATE OR REPLACE PROCEDURE add_flight(
$$;
-- 2
CREATE OR REPLACE PROCEDURE update_flight_status(
p_flight_id INT,
p_new_status VARCHAR
)
LANGUAGE plpgsql
AS $$
BEGIN
UPDATE flights
SET status = p_new_status,
update_at = CURRENT_DATE
WHERE flight_id = p_flight_id;
END;
$$;
1 ms
6125:4 LF UTF-8 4 spaces

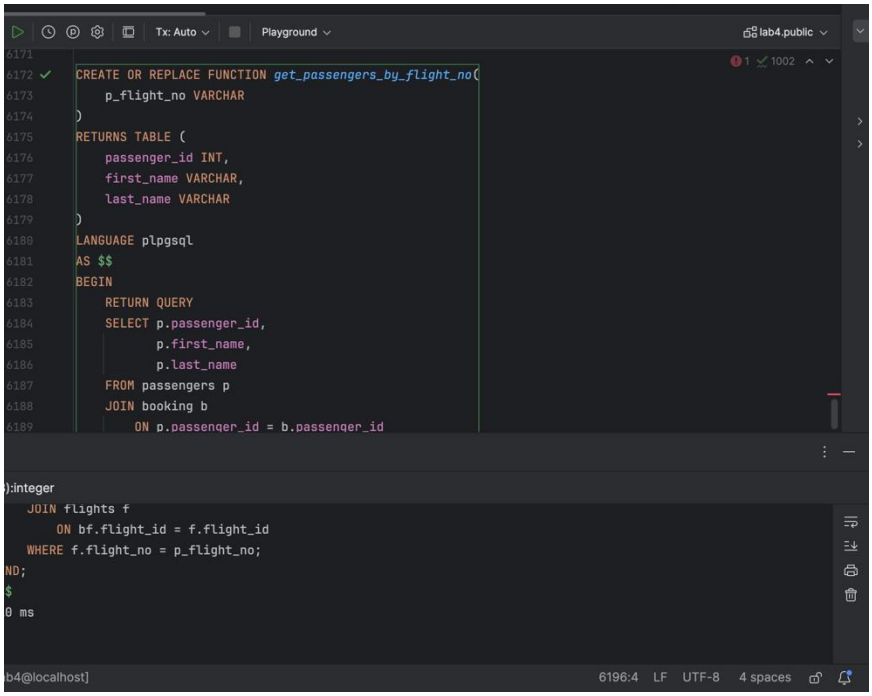
3.


console [lab4@localhost]
airline [lab4@localhost]
airport [lab4@localhost]
booking [lab4@localhost]
Tx: Auto
Playground
lab4.public
CREATE OR REPLACE PROCEDURE update_flight_status(
SET status = p_new_status,
update_at = CURRENT_DATE
WHERE flight_id = p_flight_id;
END;
$$;
-- 3
CREATE OR REPLACE FUNCTION get_flights_by_departure_airport(
p_airport_id INT
)
RETURNS SETOF flights
LANGUAGE plpgsql
AS $$
BEGIN
RETURN QUERY
SELECT *
FROM flights
WHERE departure_airport_id = p_airport_id;
END;
$$;
in 37 ms
6141:4 LF UTF-8 4 spaces

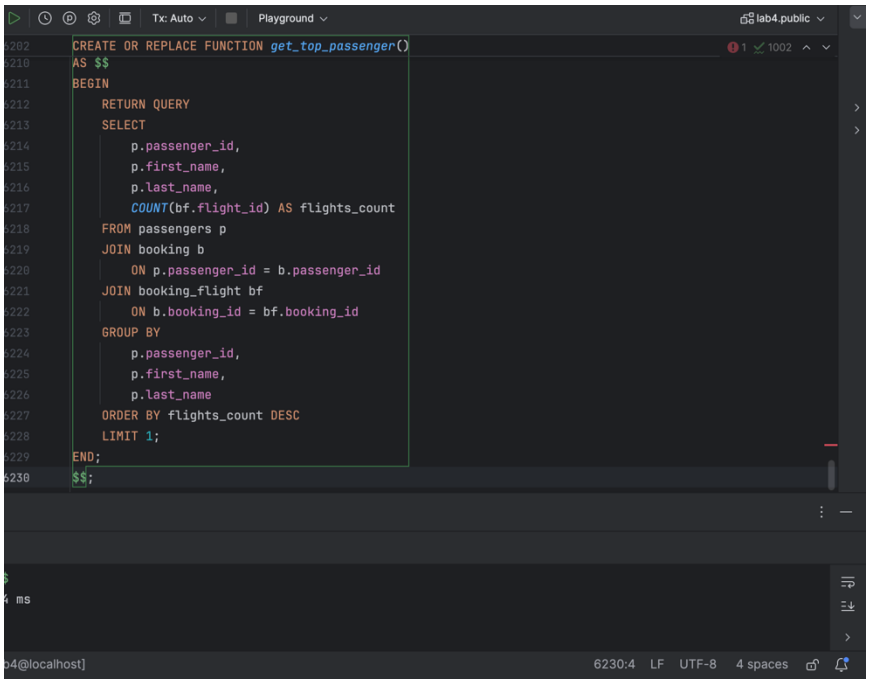
4.


Tx: Auto
Playground
lab4.public
-- 4
CREATE OR REPLACE FUNCTION get_average_arrival_delay(
p_airport_id INT
)
RETURNS NUMERIC
LANGUAGE plpgsql
AS $$
DECLARE
avg_delay NUMERIC;
BEGIN
SELECT AVG(EXTRACT(EPOCH FROM (actual_arrival - scheduled_arrival)) / 60)
INTO avg_delay
FROM flights
WHERE arrival_airport_id = p_airport_id;
RETURN avg_delay;
END;
$$;
6161:4 LF UTF-8 4 spaces

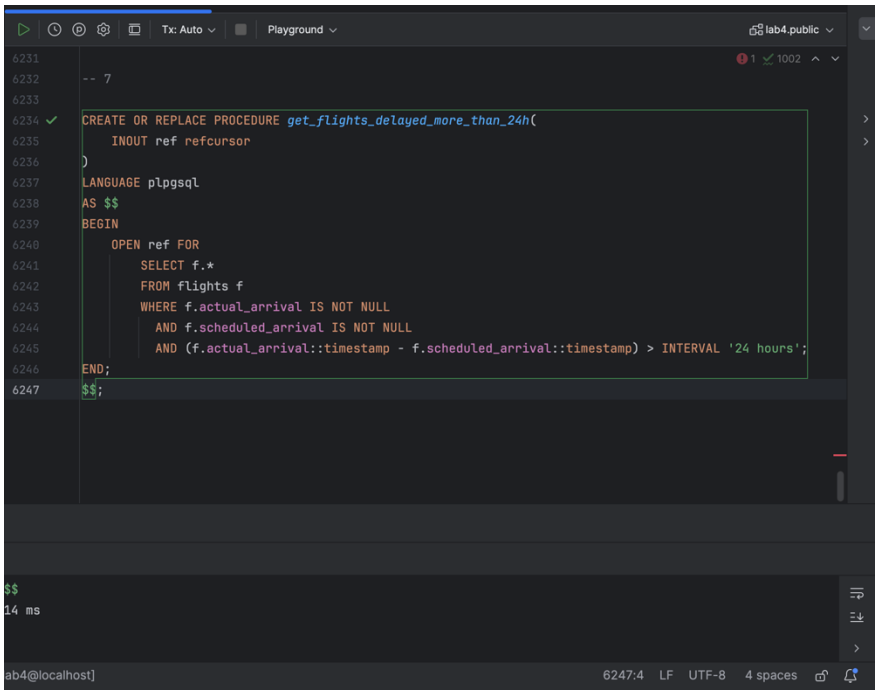
5.


Tx: Auto
Playground
lab4.public
CREATE OR REPLACE FUNCTION get_passengers_by_flight_no(
p_flight_no VARCHAR
)
RETURNS TABLE (
passenger_id INT,
first_name VARCHAR,
last_name VARCHAR
)
LANGUAGE plpgsql
AS $$
BEGIN
RETURN QUERY
SELECT p.passenger_id,
p.first_name,
p.last_name
FROM passengers p
JOIN booking b
ON p.passenger_id = b.passenger_id
):integer
JOIN flights f
ON bf.flight_id = f.flight_id
WHERE f.flight_no = p_flight_no;
ND;
$
0 ms
6196:4 LF UTF-8 4 spaces

6.


Tx: Auto
Playground
lab4.public
CREATE OR REPLACE FUNCTION get_top_passenger()
AS $$
BEGIN
RETURN QUERY
SELECT
p.passenger_id,
p.first_name,
p.last_name,
COUNT(bf.flight_id) AS flights_count
FROM passengers p
JOIN booking b
ON p.passenger_id = b.passenger_id
JOIN booking_flight bf
ON b.booking_id = bf.booking_id
GROUP BY
p.passenger_id,
p.first_name,
p.last_name
ORDER BY flights_count DESC
LIMIT 1;
END;
$$;
ms
6230:4 LF UTF-8 4 spaces

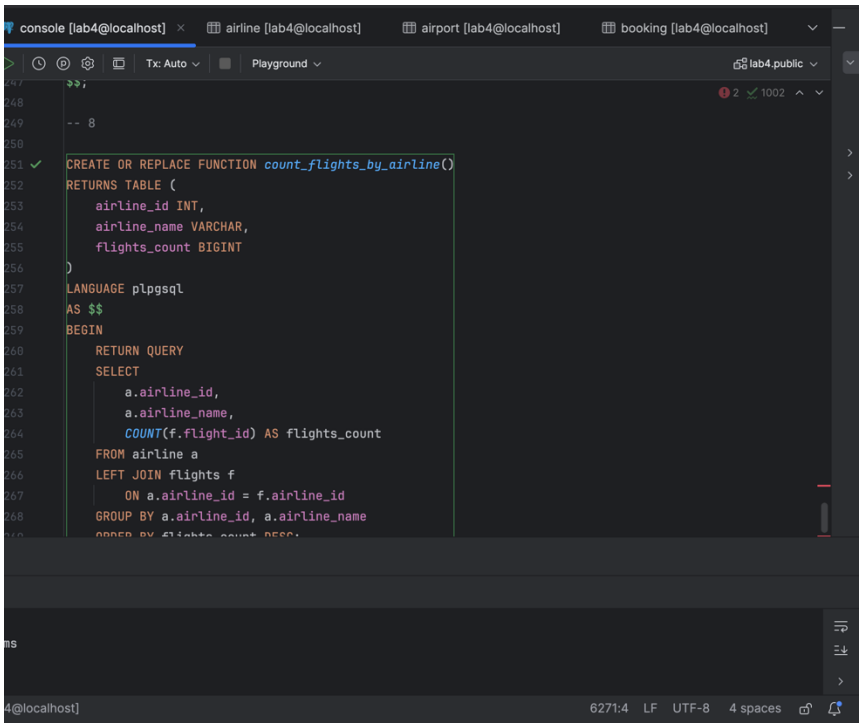
7.



```
-- 7

CREATE OR REPLACE PROCEDURE get_flights_delayed_more_than_24h(
    INOUT ref refcursor
)
LANGUAGE plpgsql
AS $$
BEGIN
    OPEN ref FOR
        SELECT f.*
        FROM flights f
        WHERE f.actual_arrival IS NOT NULL
          AND f.scheduled_arrival IS NOT NULL
          AND (f.actual_arrival::timestamp - f.scheduled_arrival::timestamp) > INTERVAL '24 hours';
END;
$$;
```

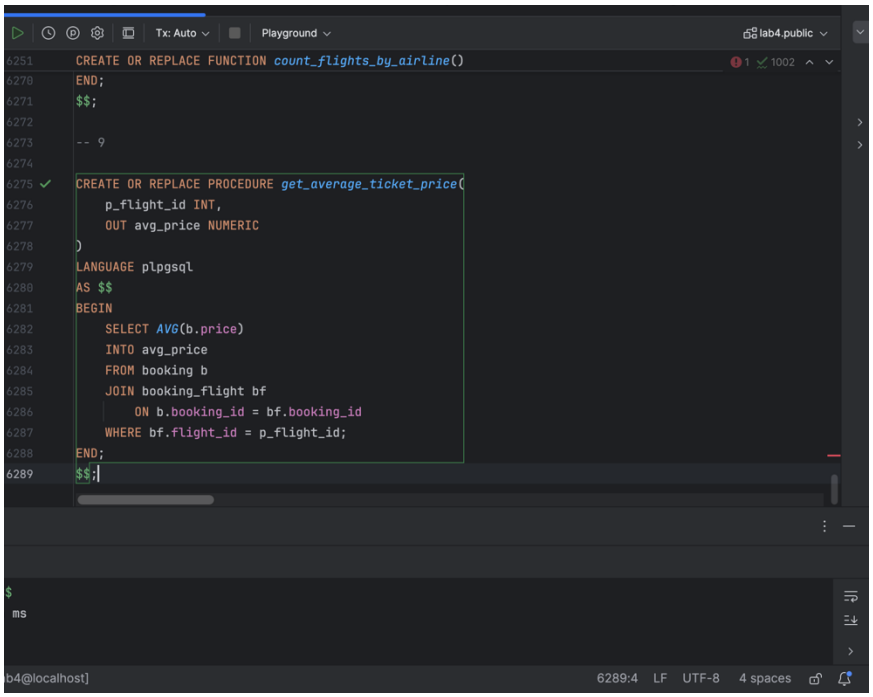
```
$$
14 ms
```

8.

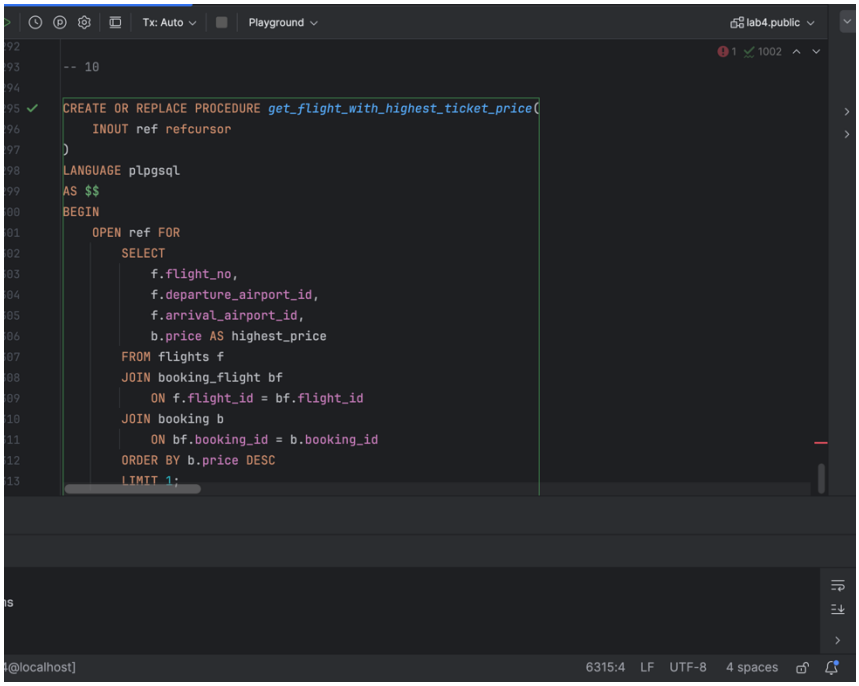


```
$$;

-- 8

CREATE OR REPLACE FUNCTION count_flights_by_airline()
RETURNS TABLE (
    airline_id INT,
    airline_name VARCHAR,
    flights_count BIGINT
)
LANGUAGE plpgsql
AS $$
BEGIN
    RETURN QUERY
    SELECT
        a.airline_id,
        a.airline_name,
        COUNT(f.flight_id) AS flights_count
    FROM airline a
    LEFT JOIN flights f
        ON a.airline_id = f.airline_id
    GROUP BY a.airline_id, a.airline_name
```

9.


Tx: Auto
Playground
lab4.public
CREATE OR REPLACE FUNCTION count_flights_by_airline()
END;
$$;
-- 9
CREATE OR REPLACE PROCEDURE get_average_ticket_price(
p_flight_id INT,
OUT avg_price NUMERIC
)
LANGUAGE plpgsql
AS $$
BEGIN
SELECT AVG(b.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;
END;
$$;
ms
6289:4 LF UTF-8 4 spaces

10.


Tx: Auto
Playground
lab4.public
-- 10
CREATE OR REPLACE PROCEDURE get_flight_with_highest_ticket_price(
INOUT ref refcursor
)
LANGUAGE plpgsql
AS $$
BEGIN
OPEN ref FOR
SELECT
f.flight_no,
f.departure_airport_id,
f.arrival_airport_id,
b.price AS highest_price
FROM flights f
JOIN booking_flight bf
ON f.flight_id = bf.flight_id
JOIN booking b
ON bf.booking_id = b.booking_id
ORDER BY b.price DESC
LIMIT 1;
6315:4 LF UTF-8 4 spaces