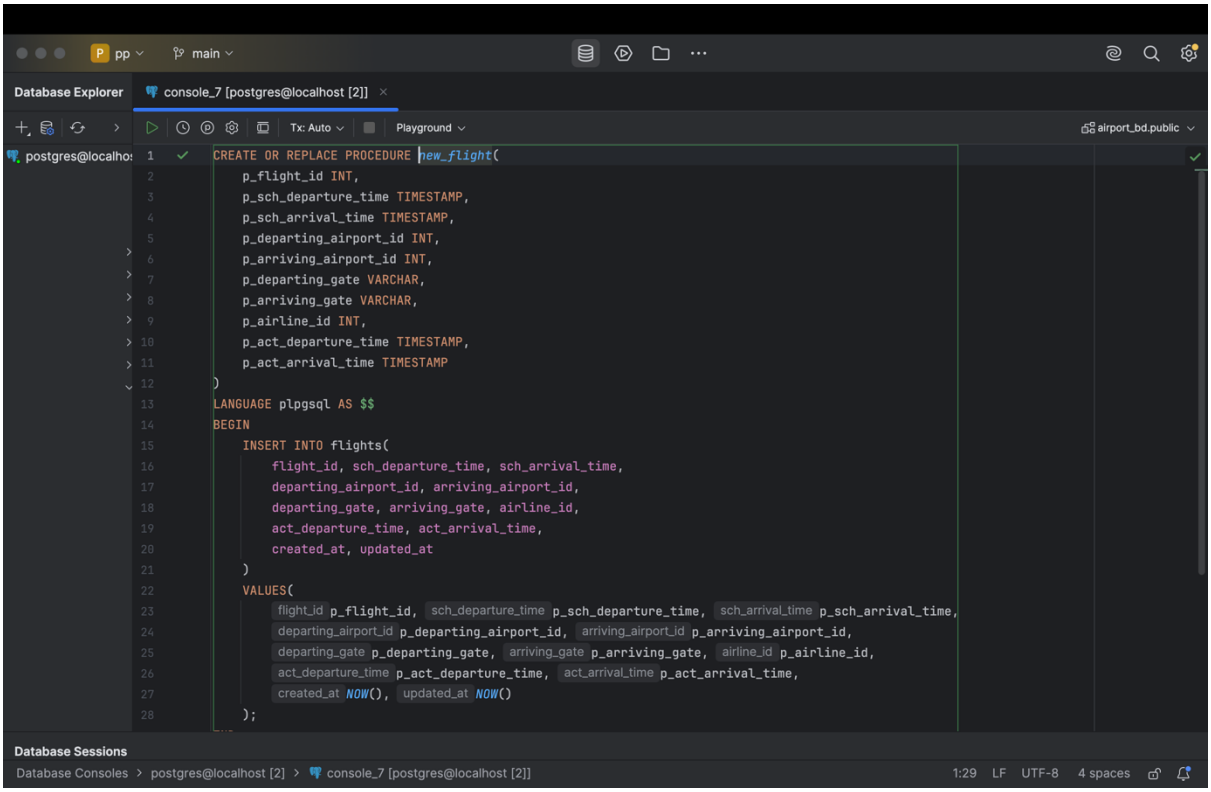
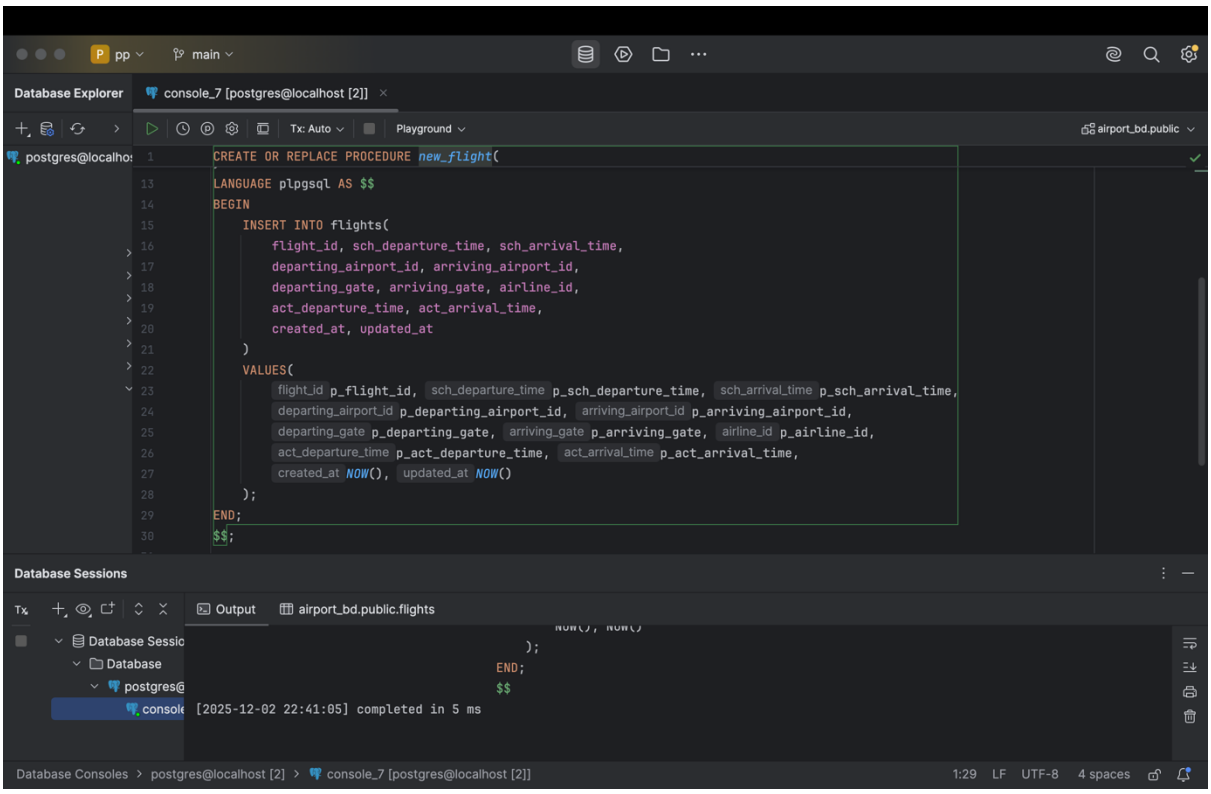
Laboratory work 10

1.



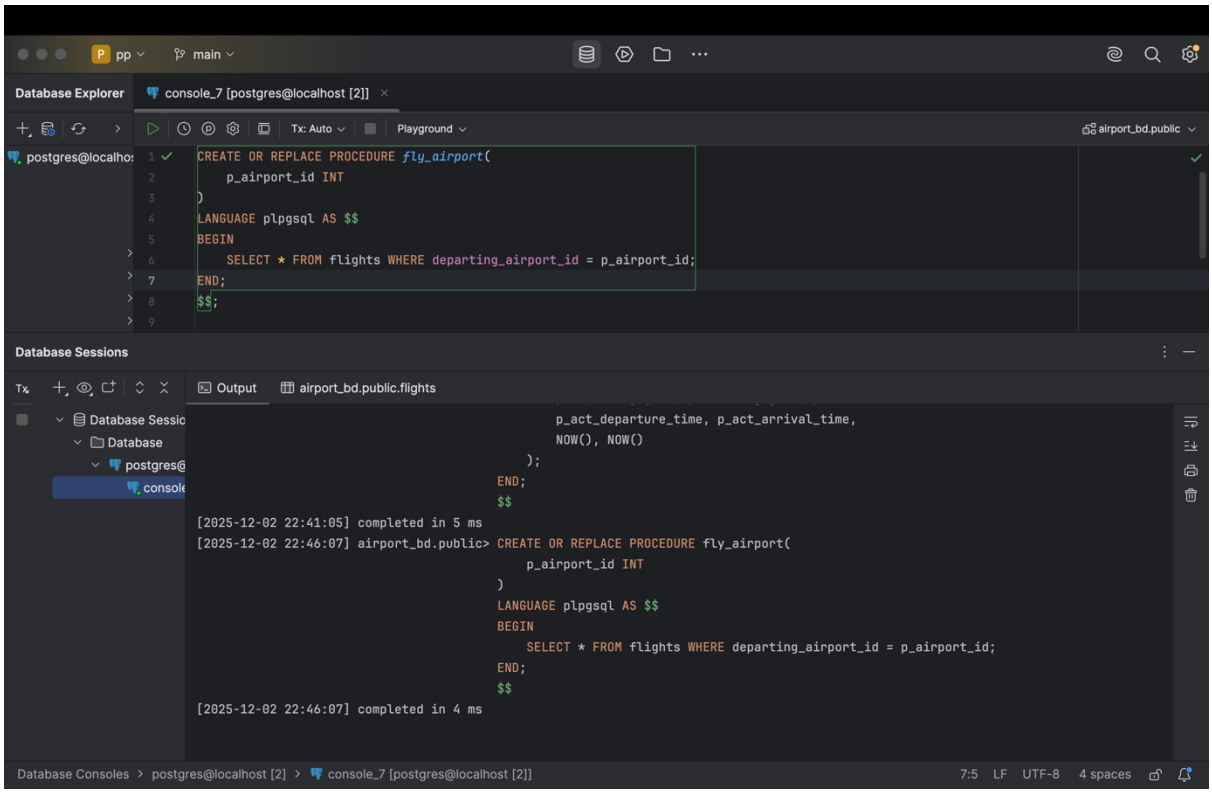
```
CREATE OR REPLACE PROCEDURE new_flight(
    p_flight_id INT,
    p_sch_departure_time TIMESTAMP,
    p_sch_arrival_time TIMESTAMP,
    p_departing_airport_id INT,
    p_arriving_airport_id INT,
    p_departing_gate VARCHAR,
    p_arriving_gate VARCHAR,
    p_airline_id INT,
    p_act_departure_time TIMESTAMP,
    p_act_arrival_time TIMESTAMP
)
LANGUAGE plpgsql AS $$
BEGIN
    INSERT INTO flights(
        flight_id, sch_departure_time, sch_arrival_time,
        departing_airport_id, arriving_airport_id,
        departing_gate, arriving_gate, airline_id,
        act_departure_time, act_arrival_time,
        created_at, updated_at
    )
    VALUES(
        flight_id p_flight_id, sch_departure_time p_sch_departure_time, sch_arrival_time p_sch_arrival_time,
        departing_airport_id p_departing_airport_id, arriving_airport_id p_arriving_airport_id,
        departing_gate p_departing_gate, arriving_gate p_arriving_gate, airline_id p_airline_id,
        act_departure_time p_act_departure_time, act_arrival_time p_act_arrival_time,
        created_at NOW(), updated_at NOW()
    );
```



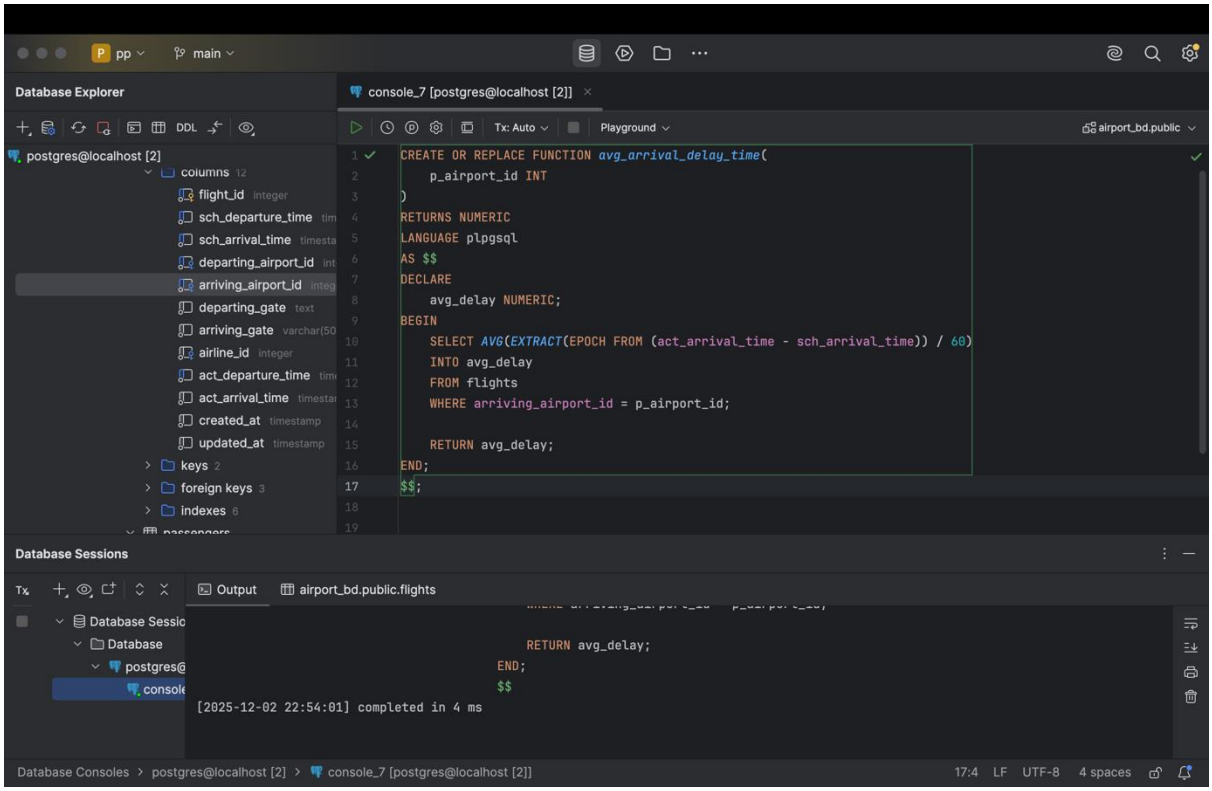
```
CREATE OR REPLACE PROCEDURE new_flight(
LANGUAGE plpgsql AS $$
BEGIN
    INSERT INTO flights(
        flight_id, sch_departure_time, sch_arrival_time,
        departing_airport_id, arriving_airport_id,
        departing_gate, arriving_gate, airline_id,
        act_departure_time, act_arrival_time,
        created_at, updated_at
    )
    VALUES(
        flight_id p_flight_id, sch_departure_time p_sch_departure_time, sch_arrival_time p_sch_arrival_time,
        departing_airport_id p_departing_airport_id, arriving_airport_id p_arriving_airport_id,
        departing_gate p_departing_gate, arriving_gate p_arriving_gate, airline_id p_airline_id,
        act_departure_time p_act_departure_time, act_arrival_time p_act_arrival_time,
        created_at NOW(), updated_at NOW()
    );
END;
$$;
```

```
);
END;
$$
[2025-12-02 22:41:05] completed in 5 ms
```

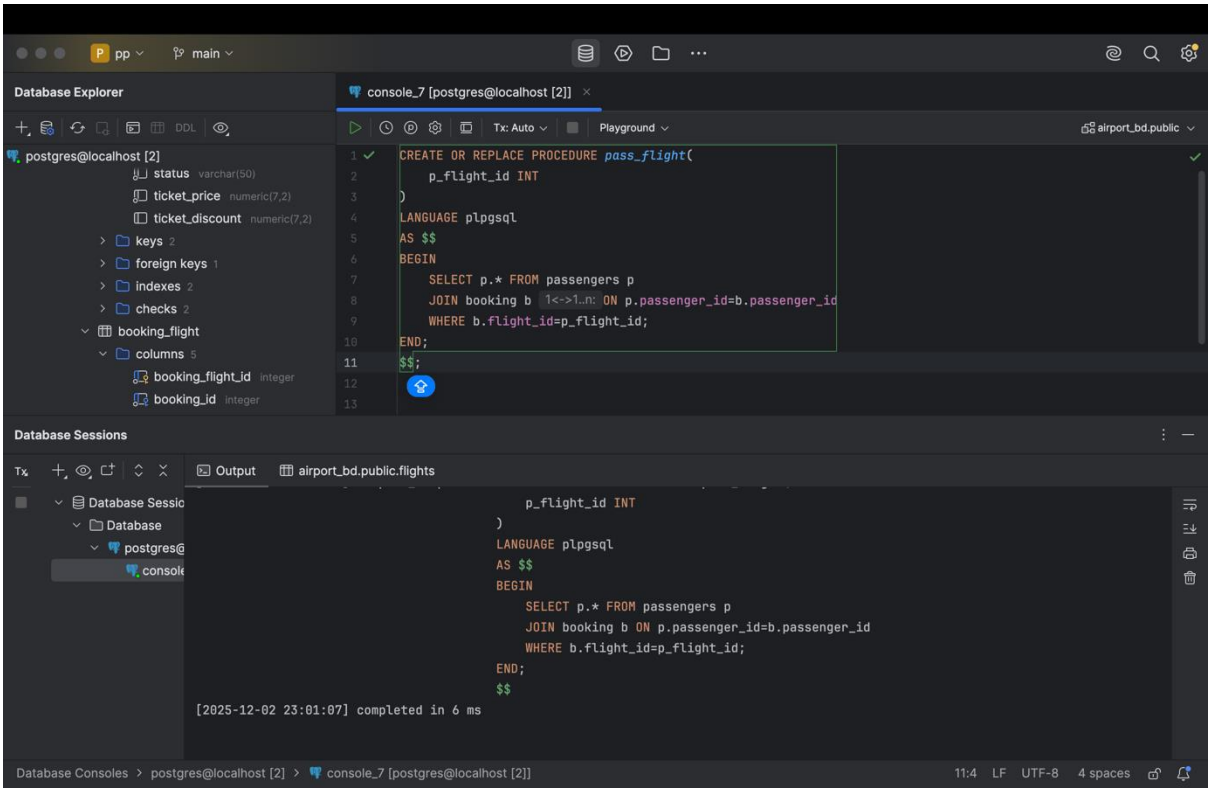
2.


pp
main
Database Explorer
console_7 [postgres@localhost [2]]
Tx: Auto
Playground
airport_bd.public
CREATE OR REPLACE PROCEDURE fly_airport(
p_airport_id INT
)
LANGUAGE plpgsql AS $$
BEGIN
SELECT * FROM flights WHERE departing_airport_id = p_airport_id;
END;
$$;
Database Sessions
Output
airport_bd.public.flights
p_act_departure_time, p_act_arrival_time,
NOW(), NOW()
);
END;
$$
[2025-12-02 22:41:05] completed in 5 ms
[2025-12-02 22:46:07] airport_bd.public> CREATE OR REPLACE PROCEDURE fly_airport(
p_airport_id INT
)
LANGUAGE plpgsql AS $$
BEGIN
SELECT * FROM flights WHERE departing_airport_id = p_airport_id;
END;
$$
[2025-12-02 22:46:07] completed in 4 ms
Database Consoles
postgres@localhost [2]
console_7 [postgres@localhost [2]]
7:5
LF
UTF-8
4 spaces

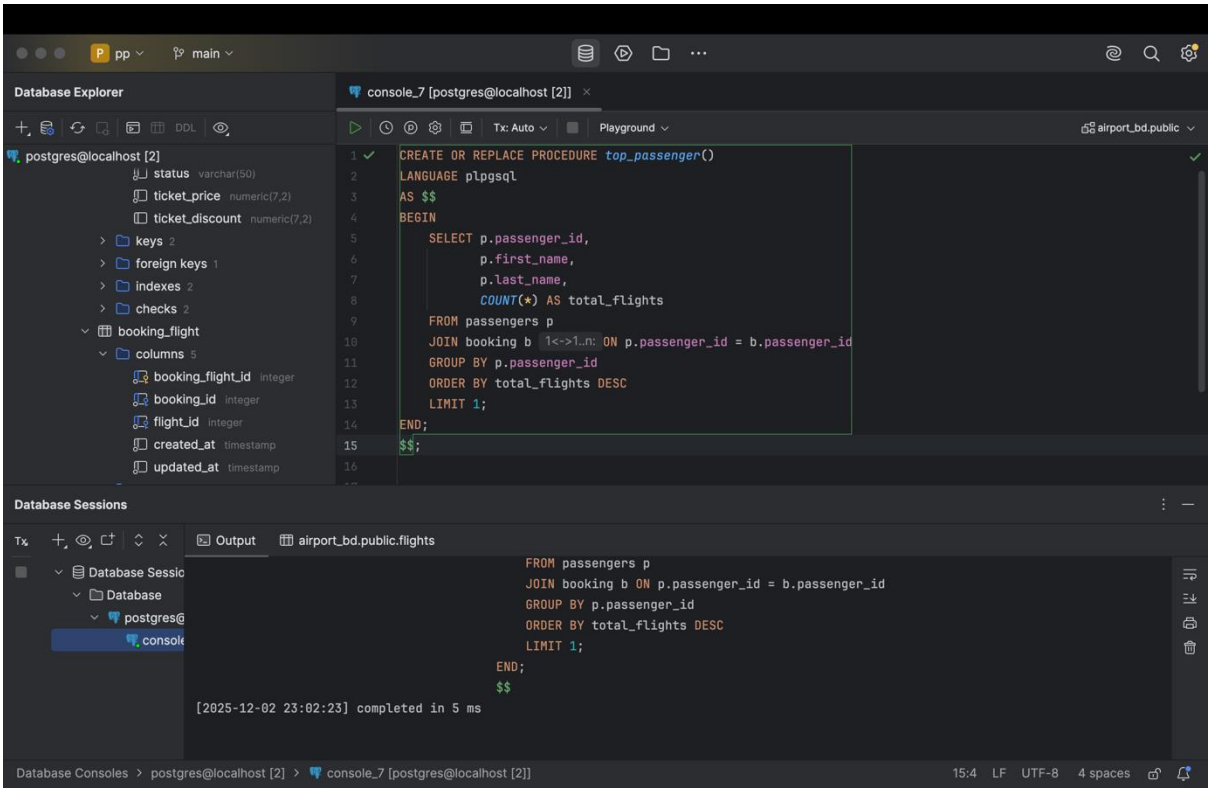
3.


pp
main
Database Explorer
console_7 [postgres@localhost [2]]
DDL
Tx: Auto
Playground
airport_bd.public
postgres@localhost [2]
columns 12
flight_id integer
sch_departure_time
sch_arrival_time
departing_airport_id
arriving_airport_id
departing_gate text
arriving_gate varchar(50
airline_id integer
act_departure_time
act_arrival_time
created_at timestamp
updated_at timestamp
keys 2
foreign keys 3
indexes 6
CREATE OR REPLACE FUNCTION avg_arrival_delay_time(
p_airport_id INT
)
RETURNS NUMERIC
LANGUAGE plpgsql
AS $$
DECLARE
avg_delay NUMERIC;
BEGIN
SELECT AVG(EXTRACT(EPOCH FROM (act_arrival_time - sch_arrival_time)) / 60)
INTO avg_delay
FROM flights
WHERE arriving_airport_id = p_airport_id;
RETURN avg_delay;
END;
$$;
Database Sessions
Output
airport_bd.public.flights
RETURN avg_delay;
END;
$$
[2025-12-02 22:54:01] completed in 4 ms
Database Consoles
postgres@localhost [2]
console_7 [postgres@localhost [2]]
17:4
LF
UTF-8
4 spaces

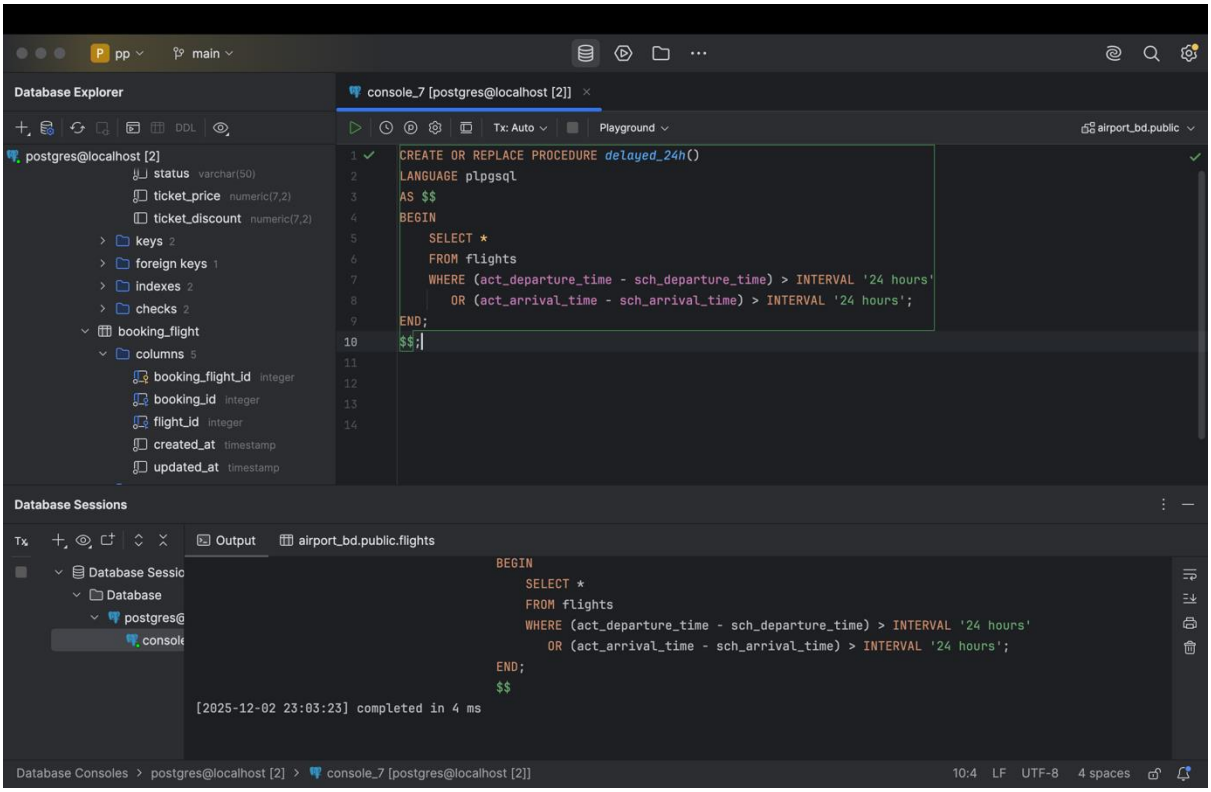
4.


pp
main
Database Explorer
console_7 [postgres@localhost [2]]
Tx: Auto
Playground
airport_bd.public
postgres@localhost [2]
status varchar(50)
ticket_price numeric(7,2)
ticket_discount numeric(7,2)
keys 2
foreign keys 1
indexes 2
checks 2
booking_flight
columns 5
booking_flight_id integer
booking_id integer
CREATE OR REPLACE PROCEDURE pass_flight(
p_flight_id INT
)
LANGUAGE plpgsql
AS $$
BEGIN
SELECT p.* FROM passengers p
JOIN booking b ON p.passenger_id=b.passenger_id
WHERE b.flight_id=p_flight_id;
END;
$$;
Database Sessions
Output
airport_bd.public.flights
[2025-12-02 23:01:07] completed in 6 ms
Database Consoles > postgres@localhost [2] > console_7 [postgres@localhost [2]]
11:4 LF UTF-8 4 spaces

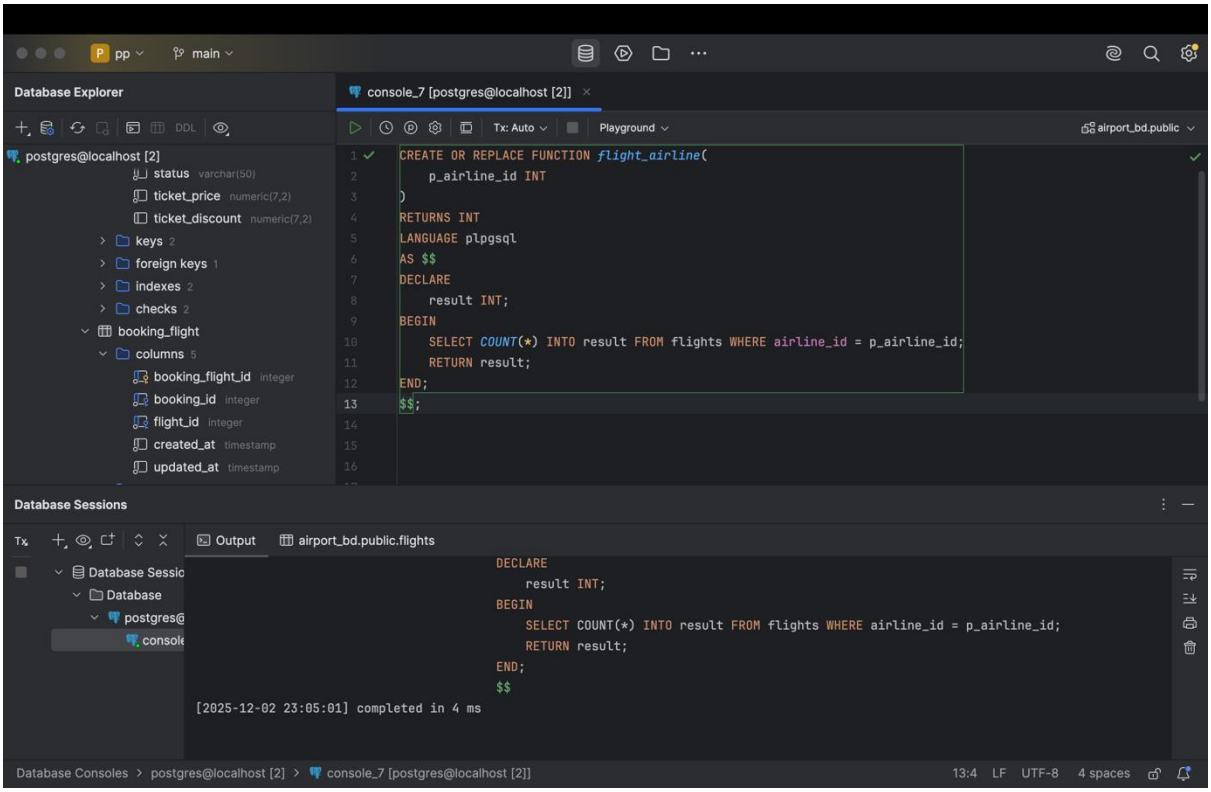
5.


pp
main
Database Explorer
console_7 [postgres@localhost [2]]
Tx: Auto
Playground
airport_bd.public
postgres@localhost [2]
status varchar(50)
ticket_price numeric(7,2)
ticket_discount numeric(7,2)
keys 2
foreign keys 1
indexes 2
checks 2
booking_flight
columns 5
booking_flight_id integer
booking_id integer
flight_id integer
created_at timestamp
updated_at timestamp
CREATE OR REPLACE PROCEDURE top_passenger()
LANGUAGE plpgsql
AS $$
BEGIN
SELECT p.passenger_id,
p.first_name,
p.last_name,
COUNT(*) AS total_flights
FROM passengers p
JOIN booking b ON p.passenger_id = b.passenger_id
GROUP BY p.passenger_id
ORDER BY total_flights DESC
LIMIT 1;
END;
$$;
Database Sessions
Output
airport_bd.public.flights
[2025-12-02 23:02:23] completed in 5 ms
Database Consoles > postgres@localhost [2] > console_7 [postgres@localhost [2]]
15:4 LF UTF-8 4 spaces

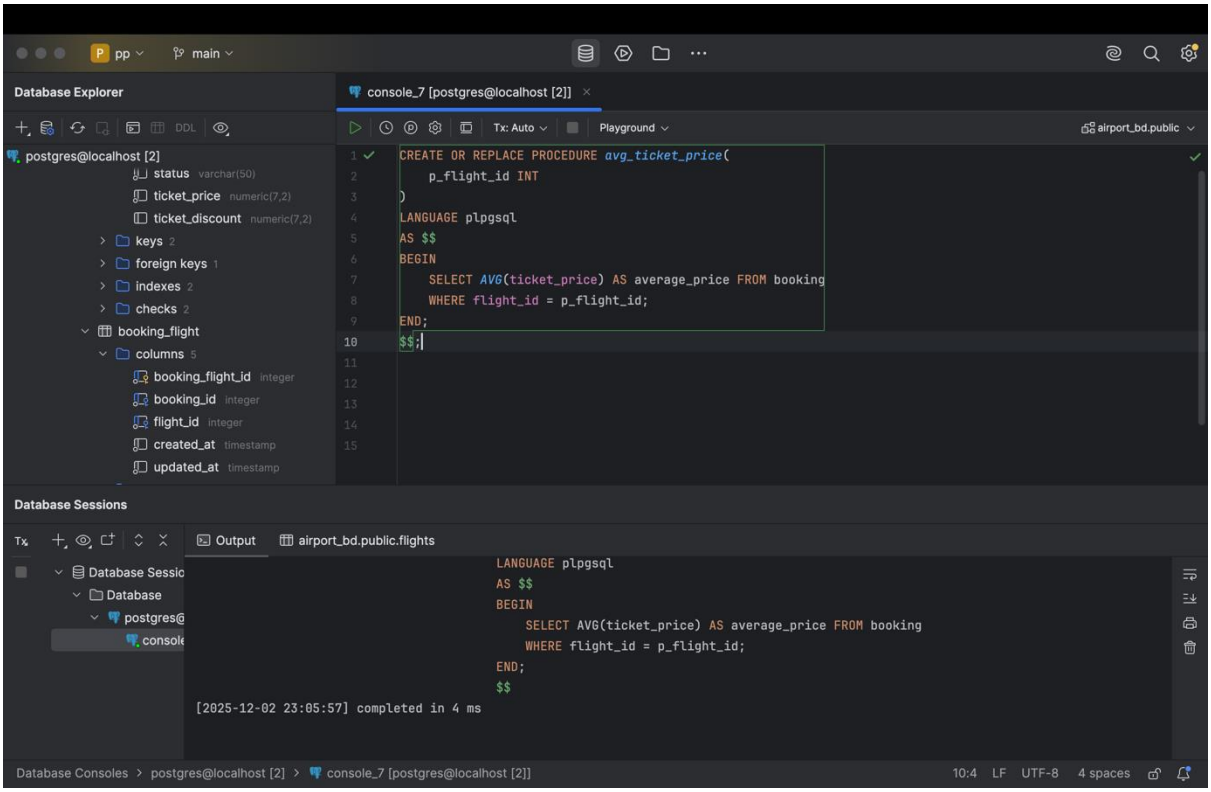
6.


pp
main
Database Explorer
console_7 [postgres@localhost [2]]
Tx: Auto
Playground
airport_bd.public
postgres@localhost [2]
status varchar(50)
ticket_price numeric(7,2)
ticket_discount numeric(7,2)
keys 2
foreign keys 1
indexes 2
checks 2
booking_flight
columns 5
booking_flight_id integer
booking_id integer
flight_id integer
created_at timestamp
updated_at timestamp
CREATE OR REPLACE PROCEDURE delayed_24h()
LANGUAGE plpgsql
AS $$
BEGIN
SELECT *
FROM flights
WHERE (act_departure_time - sch_departure_time) > INTERVAL '24 hours'
OR (act_arrival_time - sch_arrival_time) > INTERVAL '24 hours';
END;
$$;
Database Sessions
Output
airport_bd.public.flights
[2025-12-02 23:03:23] completed in 4 ms
Database Consoles > postgres@localhost [2] > console_7 [postgres@localhost [2]]
10:4 LF UTF-8 4 spaces

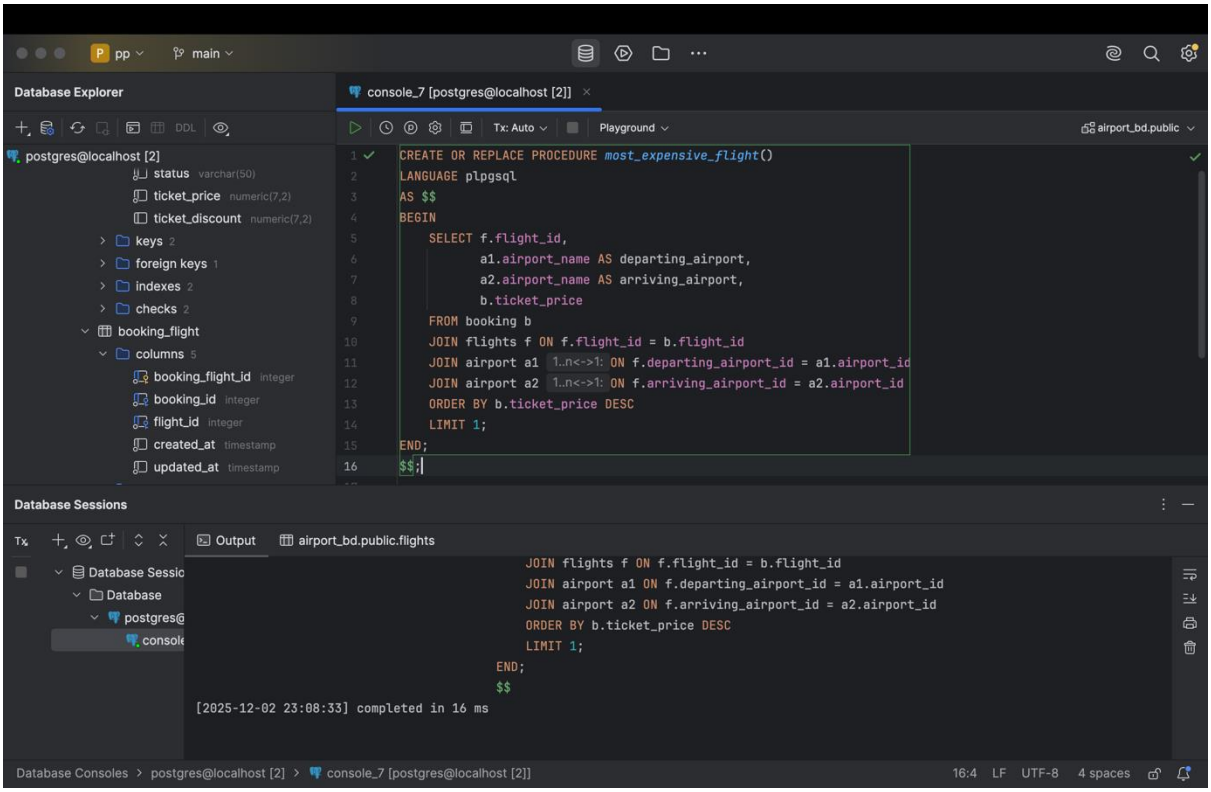
7.


pp
main
Database Explorer
console_7 [postgres@localhost [2]]
Tx: Auto
Playground
airport_bd.public
postgres@localhost [2]
status varchar(50)
ticket_price numeric(7,2)
ticket_discount numeric(7,2)
keys 2
foreign keys 1
indexes 2
checks 2
booking_flight
columns 5
booking_flight_id integer
booking_id integer
flight_id integer
created_at timestamp
updated_at timestamp
CREATE OR REPLACE FUNCTION flight_airline(
p_airline_id INT
)
RETURNS INT
LANGUAGE plpgsql
AS $$
DECLARE
result INT;
BEGIN
SELECT COUNT(*) INTO result FROM flights WHERE airline_id = p_airline_id;
RETURN result;
END;
$$;
Database Sessions
Output
airport_bd.public.flights
[2025-12-02 23:05:01] completed in 4 ms
Database Consoles > postgres@localhost [2] > console_7 [postgres@localhost [2]]
13:4 LF UTF-8 4 spaces

8.


pp
main
Database Explorer
console_7 [postgres@localhost [2]]
Tx: Auto
Playground
airport_bd.public
postgres@localhost [2]
status varchar(50)
ticket_price numeric(7,2)
ticket_discount numeric(7,2)
keys 2
foreign keys 1
indexes 2
checks 2
booking_flight
columns 5
booking_flight_id integer
booking_id integer
flight_id integer
created_at timestamp
updated_at timestamp
CREATE OR REPLACE PROCEDURE avg_ticket_price(
p_flight_id INT
)
LANGUAGE plpgsql
AS $$
BEGIN
SELECT AVG(ticket_price) AS average_price FROM booking
WHERE flight_id = p_flight_id;
END;
$$;
Database Sessions
Output
airport_bd.public.flights
LANGUAGE plpgsql
AS $$
BEGIN
SELECT AVG(ticket_price) AS average_price FROM booking
WHERE flight_id = p_flight_id;
END;
$$
[2025-12-02 23:05:57] completed in 4 ms
Database Consoles > postgres@localhost [2] > console_7 [postgres@localhost [2]]
10:4 LF UTF-8 4 spaces

9.


pp
main
Database Explorer
console_7 [postgres@localhost [2]]
Tx: Auto
Playground
airport_bd.public
postgres@localhost [2]
status varchar(50)
ticket_price numeric(7,2)
ticket_discount numeric(7,2)
keys 2
foreign keys 1
indexes 2
checks 2
booking_flight
columns 5
booking_flight_id integer
booking_id integer
flight_id integer
created_at timestamp
updated_at timestamp
CREATE OR REPLACE PROCEDURE most_expensive_flight()
LANGUAGE plpgsql
AS $$
BEGIN
SELECT f.flight_id,
a1.airport_name AS departing_airport,
a2.airport_name AS arriving_airport,
b.ticket_price
FROM booking b
JOIN flights f ON f.flight_id = b.flight_id
JOIN airport a1 1..n<->1: ON f.departing_airport_id = a1.airport_id
JOIN airport a2 1..n<->1: ON f.arriving_airport_id = a2.airport_id
ORDER BY b.ticket_price DESC
LIMIT 1;
END;
$$;
Database Sessions
Output
airport_bd.public.flights
JOIN flights f ON f.flight_id = b.flight_id
JOIN airport a1 ON f.departing_airport_id = a1.airport_id
JOIN airport a2 ON f.arriving_airport_id = a2.airport_id
ORDER BY b.ticket_price DESC
LIMIT 1;
END;
$$
[2025-12-02 23:08:33] completed in 16 ms
Database Consoles > postgres@localhost [2] > console_7 [postgres@localhost [2]]
16:4 LF UTF-8 4 spaces