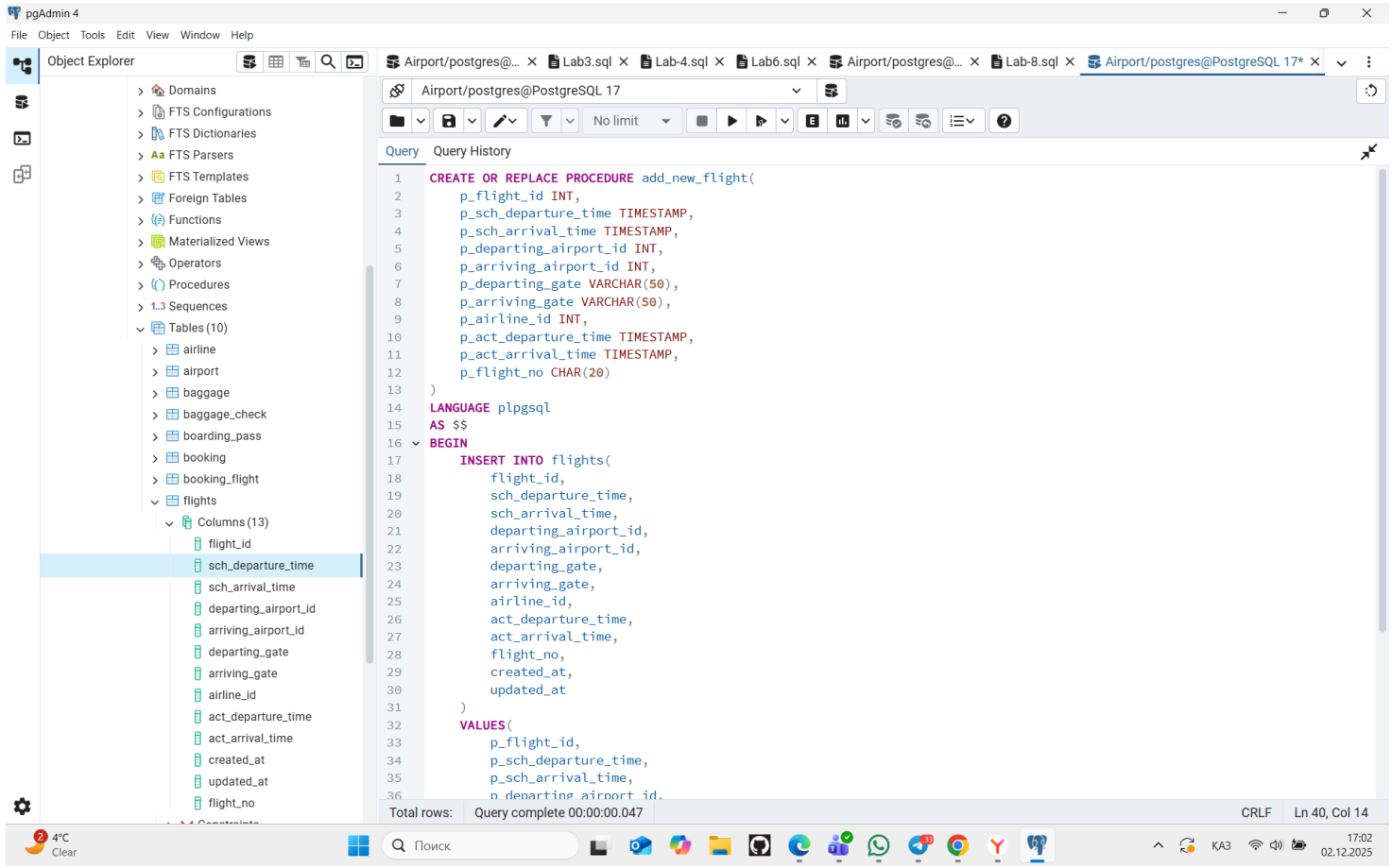
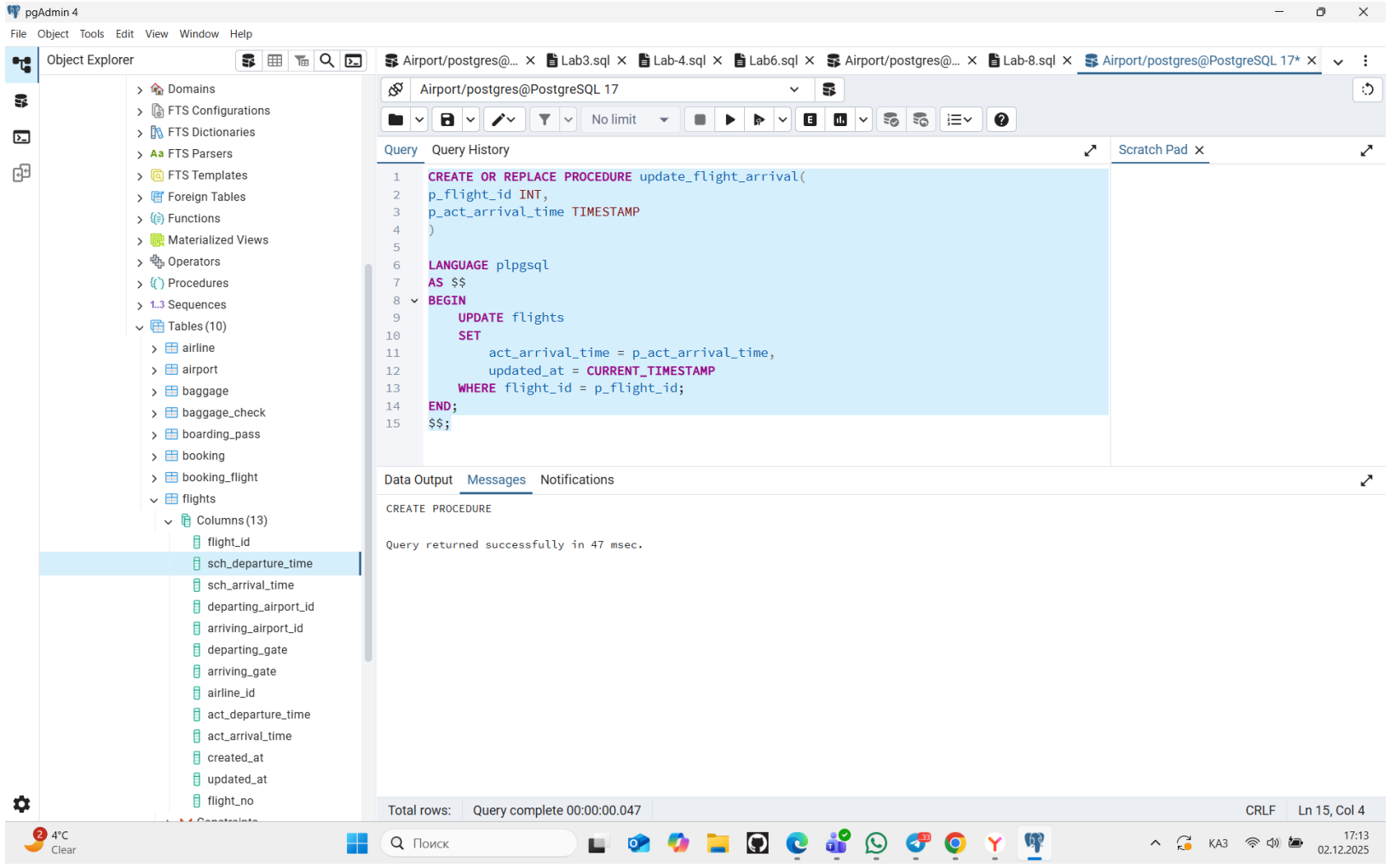
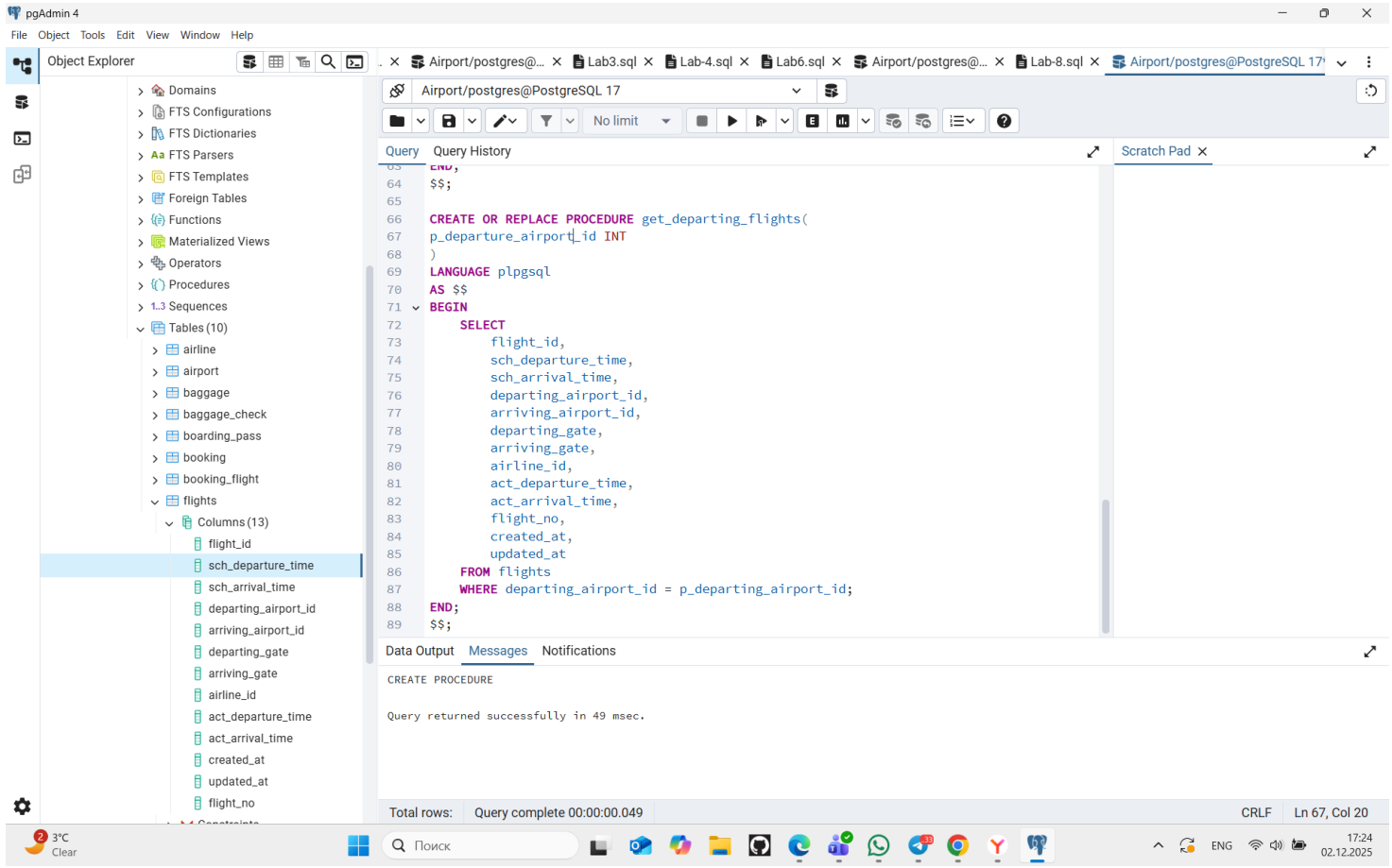
task1


pgAdmin 4
File Object Tools Edit View Window Help
Object Explorer
Airport/postgres@PostgreSQL 17
Query Query History
CREATE OR REPLACE PROCEDURE add_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(50),
p_arriving_gate VARCHAR(50),
p_airline_id INT,
p_act_departure_time TIMESTAMP,
p_act_arrival_time TIMESTAMP,
p_flight_no CHAR(20)
)
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,
flight_no,
created_at,
updated_at
)
VALUES(
p_flight_id,
p_sch_departure_time,
p_sch_arrival_time,
Total rows:
Query complete 00:00:00.047
CRLF
Ln 40, Col 14

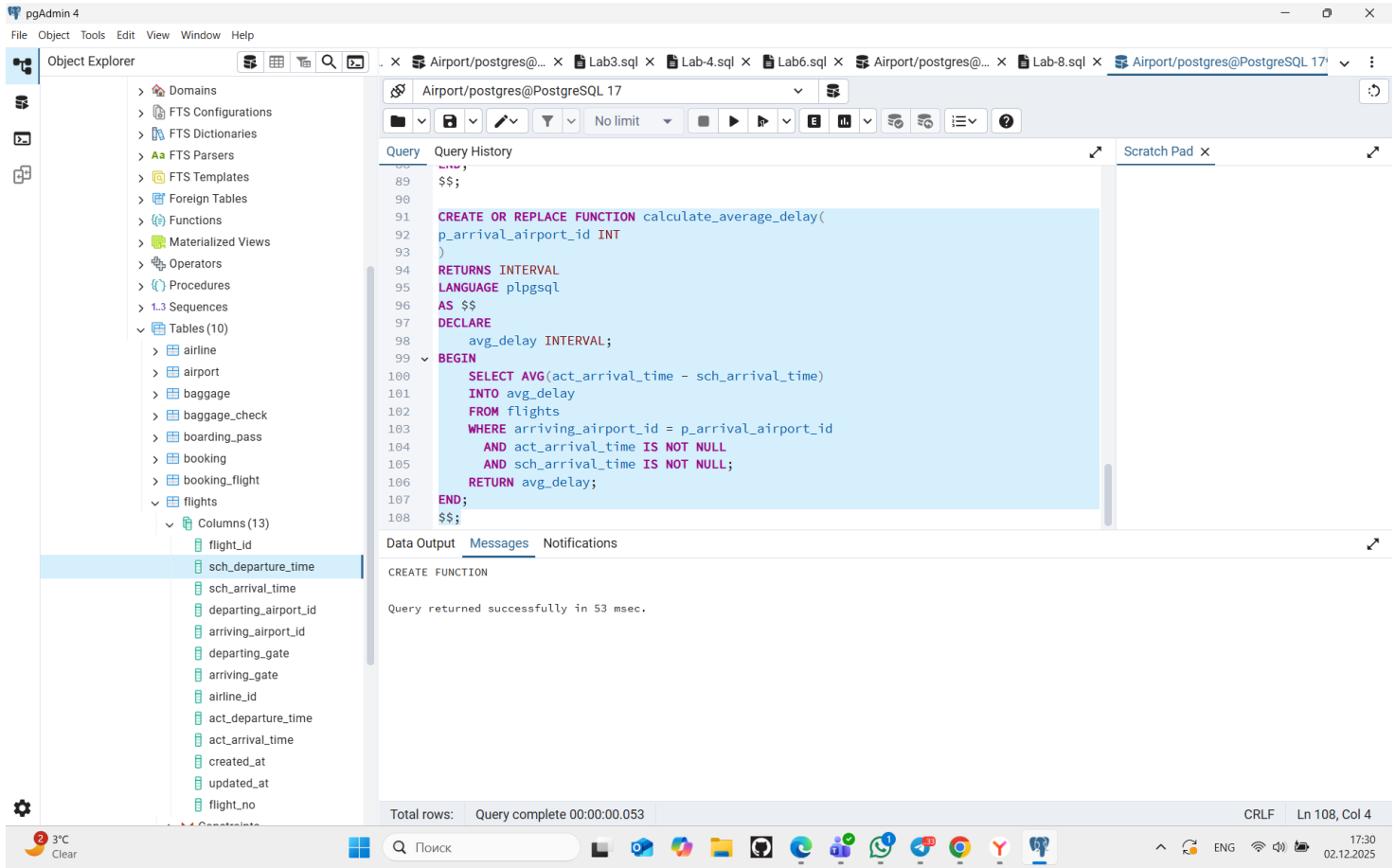
TASK 2


pgAdmin 4
File Object Tools Edit View Window Help
Object Explorer
Domains
FTS Configurations
FTS Dictionaries
FTS Parsers
FTS Templates
Foreign Tables
Functions
Materialized Views
Operators
Procedures
Sequences
Tables (10)
airline
airport
baggage
baggage_check
boarding_pass
booking
booking_flight
flights
Columns (13)
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
flight_no
Airport/postgres@... Lab3.sql Lab-4.sql Lab6.sql Airport/postgres@... Lab-8.sql Airport/postgres@PostgreSQL 17*
Airport/postgres@PostgreSQL 17
No limit
Query Query History
Scratch Pad
1 CREATE OR REPLACE PROCEDURE update_flight_arrival(
2 p_flight_id INT,
3 p_act_arrival_time TIMESTAMP
4)
5
6 LANGUAGE plpgsql
7 AS $$
8 BEGIN
9 UPDATE flights
10 SET
11 act_arrival_time = p_act_arrival_time,
12 updated_at = CURRENT_TIMESTAMP
13 WHERE flight_id = p_flight_id;
14 END;
15 $$;
Data Output Messages Notifications
CREATE PROCEDURE
Query returned successfully in 47 msec.
Total rows:
Query complete 00:00:00.047
CRLF
Ln 15, Col 4
4°C Clear
Поиск
КАЗ
17:13
02.12.2025

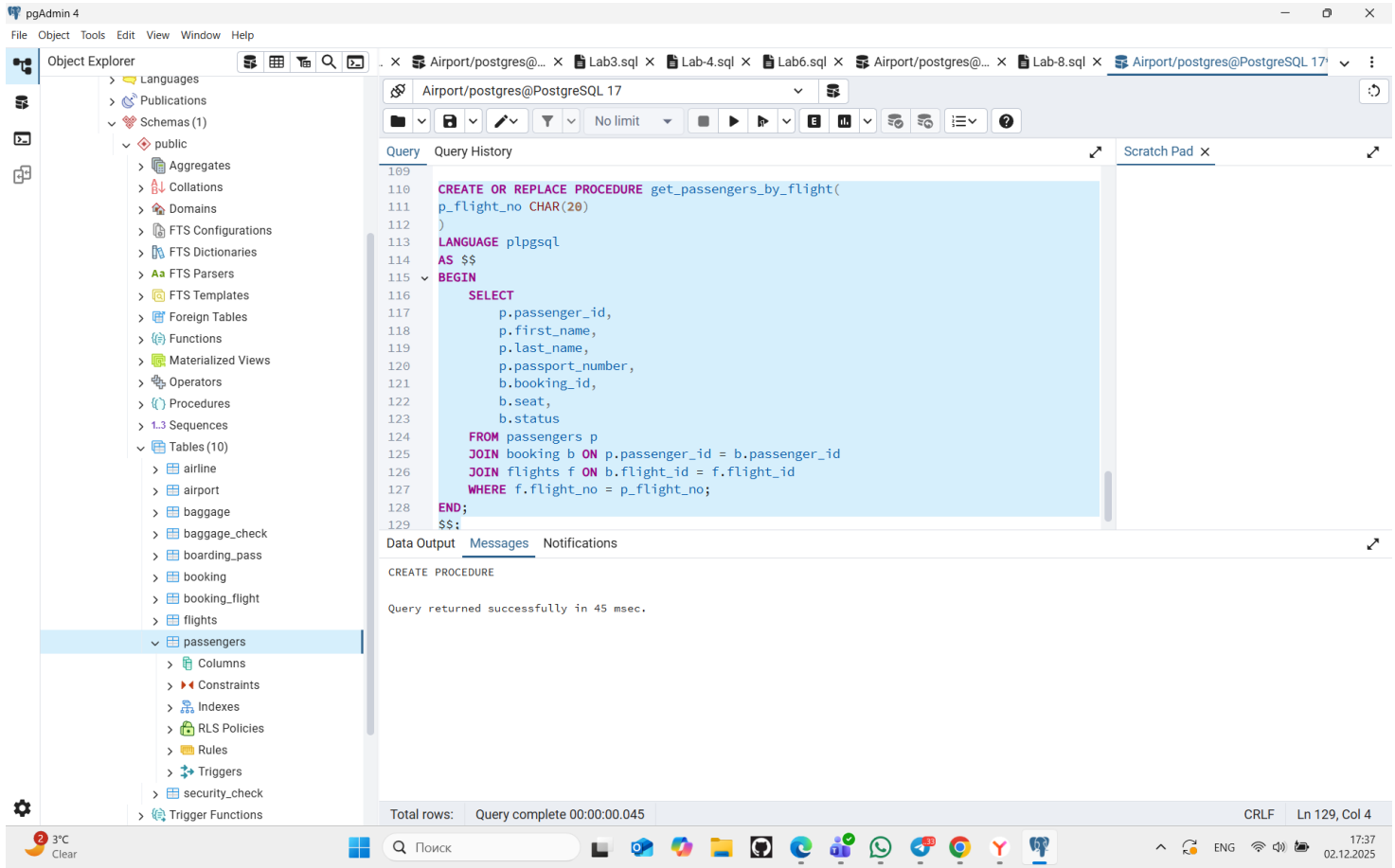
TASK #3


pgAdmin 4
File Object Tools Edit View Window Help
Object Explorer
Airport/postgres@PostgreSQL 17
No limit
Query Query History
Scratch Pad
64 $$;
65
66 CREATE OR REPLACE PROCEDURE get_departing_flights(
67 p_departure_airport_id INT
68)
69 LANGUAGE plpgsql
70 AS $$
71 BEGIN
72 SELECT
73 flight_id,
74 sch_departure_time,
75 sch_arrival_time,
76 departing_airport_id,
77 arriving_airport_id,
78 departing_gate,
79 arriving_gate,
80 airline_id,
81 act_departure_time,
82 act_arrival_time,
83 flight_no,
84 created_at,
85 updated_at
86 FROM flights
87 WHERE departing_airport_id = p_departing_airport_id;
88 END;
89 $$;
Domains
FTS Configurations
FTS Dictionaries
FTS Parsers
FTS Templates
Foreign Tables
Functions
Materialized Views
Operators
Procedures
Sequences
Tables (10)
airline
airport
baggage
baggage_check
boarding_pass
booking
booking_flight
flights
Columns (13)
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
flight_no
Data Output Messages Notifications
CREATE PROCEDURE
Query returned successfully in 49 msec.
Total rows: Query complete 00:00:00.049
CRLF Ln 67, Col 20
3°C Clear
Поиск
ENG
17:24
02.12.2025

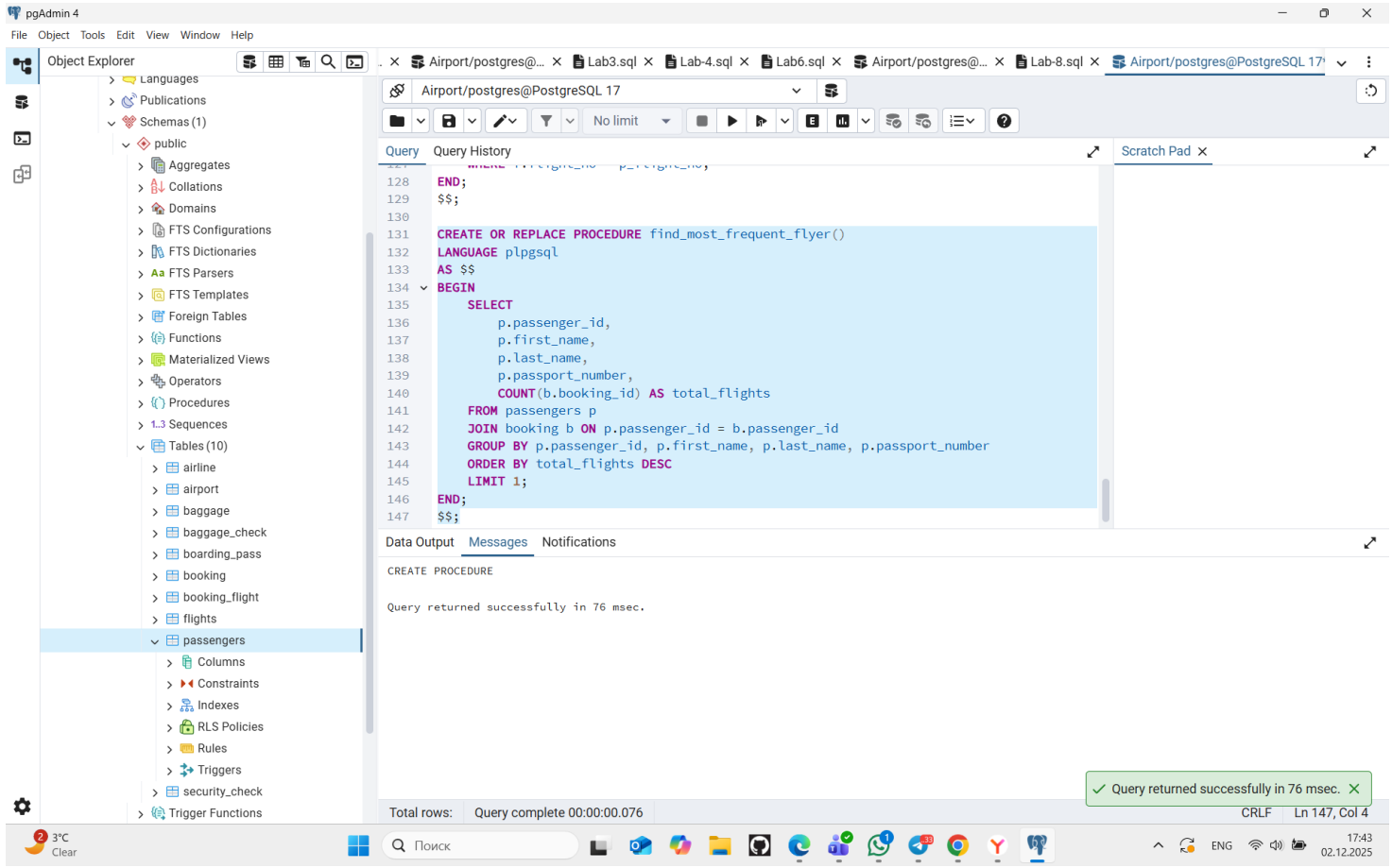
TASK #4


pgAdmin 4
File Object Tools Edit View Window Help
Object Explorer
Airport/postgres@PostgreSQL 17
No limit
Query Query History
Scratch Pad
$$;
CREATE OR REPLACE FUNCTION calculate_average_delay(
p_arrival_airport_id INT
)
RETURNS INTERVAL
LANGUAGE plpgsql
AS $$
DECLARE
avg_delay INTERVAL;
BEGIN
SELECT AVG(act_arrival_time - sch_arrival_time)
INTO avg_delay
FROM flights
WHERE arriving_airport_id = p_arrival_airport_id
AND act_arrival_time IS NOT NULL
AND sch_arrival_time IS NOT NULL;
RETURN avg_delay;
END;
$$;
Data Output Messages Notifications
CREATE FUNCTION
Query returned successfully in 53 msec.
Total rows:
Query complete 00:00:00.053
CRLF
Ln 108, Col 4
Domains
FTS Configurations
FTS Dictionaries
FTS Parsers
FTS Templates
Foreign Tables
Functions
Materialized Views
Operators
Procedures
Sequences
Tables (10)
airline
airport
baggage
baggage_check
boarding_pass
booking
booking_flight
flights
Columns (13)
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
flight_no

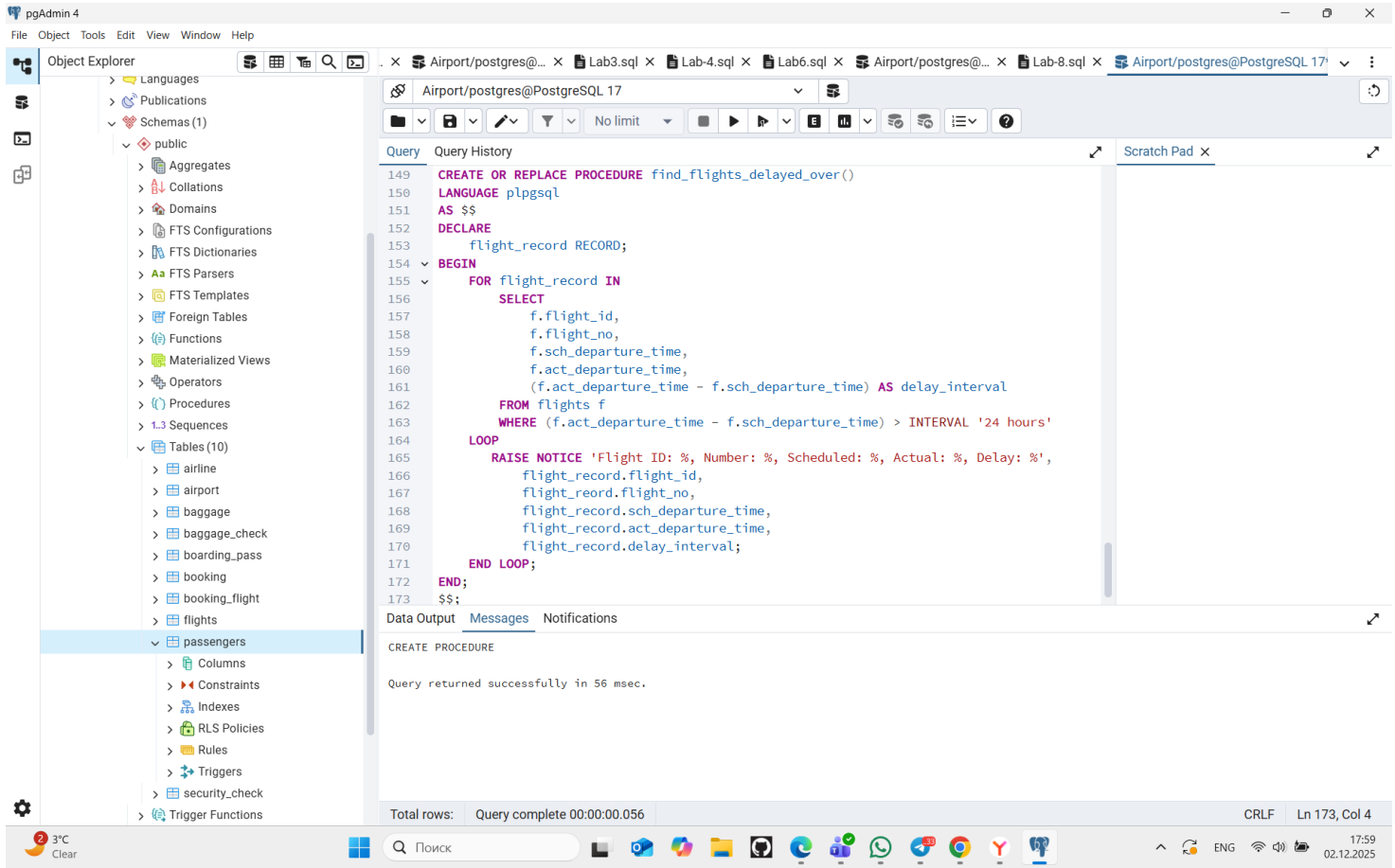
TASK #5


pgAdmin 4
File Object Tools Edit View Window Help
Object Explorer
Airport/postgres@PostgreSQL 17
No limit
Query Query History
Scratch Pad
CREATE OR REPLACE PROCEDURE get_passengers_by_flight(
p_flight_no CHAR(20)
)
LANGUAGE plpgsql
AS $$
BEGIN
SELECT
p.passenger_id,
p.first_name,
p.last_name,
p.passport_number,
b.booking_id,
b.seat,
b.status
FROM passengers p
JOIN booking b ON p.passenger_id = b.passenger_id
JOIN flights f ON b.flight_id = f.flight_id
WHERE f.flight_no = p_flight_no;
END;
$$;
Data Output Messages Notifications
CREATE PROCEDURE
Query returned successfully in 45 msec.
Total rows: Query complete 00:00:00.045
CRLF Ln 129, Col 4

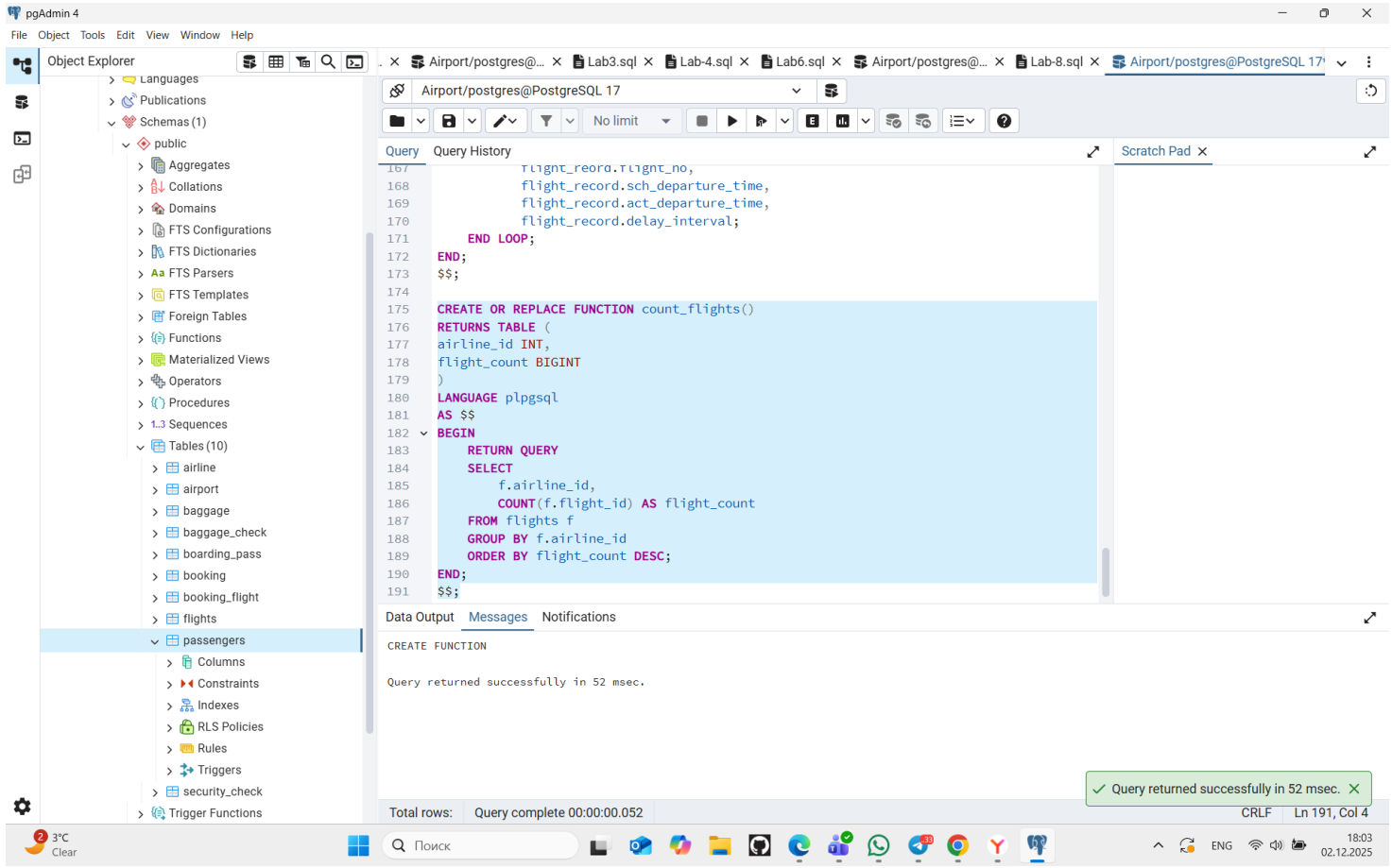
TASK #6


pgAdmin 4
File Object Tools Edit View Window Help
Object Explorer
Airport/postgres@PostgreSQL 17
No limit
Query Query History
Scratch Pad
CREATE OR REPLACE PROCEDURE find_most_frequent_flyer()
LANGUAGE plpgsql
AS $$
BEGIN
SELECT
p.passenger_id,
p.first_name,
p.last_name,
p.passport_number,
COUNT(b.booking_id) AS total_flights
FROM passengers p
JOIN booking b ON p.passenger_id = b.passenger_id
GROUP BY p.passenger_id, p.first_name, p.last_name, p.passport_number
ORDER BY total_flights DESC
LIMIT 1;
END;
$$;
Data Output Messages Notifications
CREATE PROCEDURE
Query returned successfully in 76 msec.
Total rows: Query complete 00:00:00.076
CRLF Ln 147, Col 4
Query returned successfully in 76 msec.

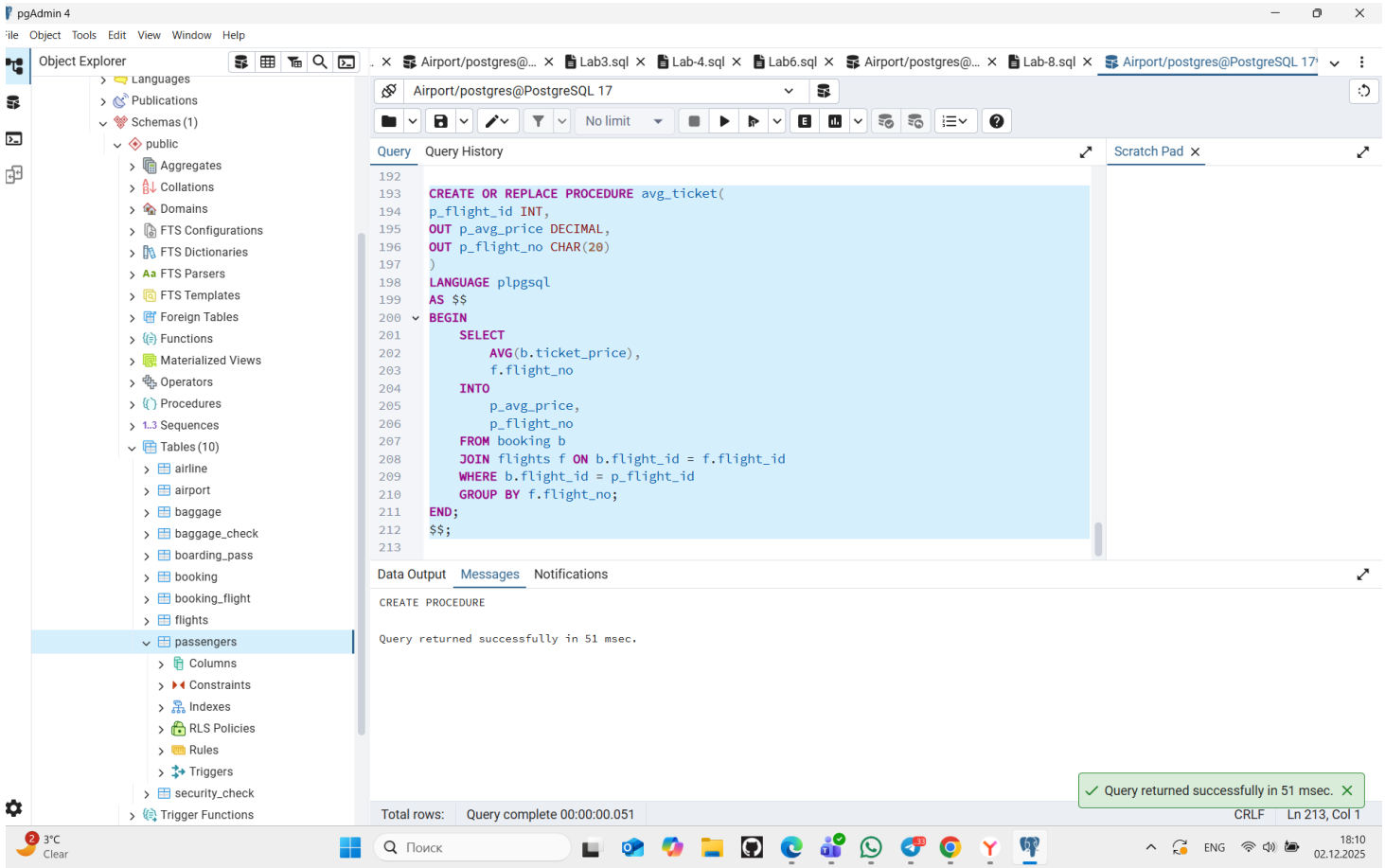
TASK #7


pgAdmin 4
File Object Tools Edit View Window Help
Object Explorer
Airport/postgres@PostgreSQL 17
No limit
Query Query History
Scratch Pad
CREATE OR REPLACE PROCEDURE find_flights_delayed_over()
LANGUAGE plpgsql
AS $$
DECLARE
flight_record RECORD;
BEGIN
FOR flight_record IN
SELECT
f.flight_id,
f.flight_no,
f.sch_departure_time,
f.act_departure_time,
(f.act_departure_time - f.sch_departure_time) AS delay_interval
FROM flights f
WHERE (f.act_departure_time - f.sch_departure_time) > INTERVAL '24 hours'
LOOP
RAISE NOTICE 'Flight ID: %, Number: %, Scheduled: %, Actual: %, Delay: %',
flight_record.flight_id,
flight_reord.flight_no,
flight_record.sch_departure_time,
flight_record.act_departure_time,
flight_record.delay_interval;
END LOOP;
END;
$$;
Data Output Messages Notifications
CREATE PROCEDURE
Query returned successfully in 56 msec.
Total rows:
Query complete 00:00:00.056
CRLF
Ln 173, Col 4
Поиск
ENG
17:59
02.12.2025

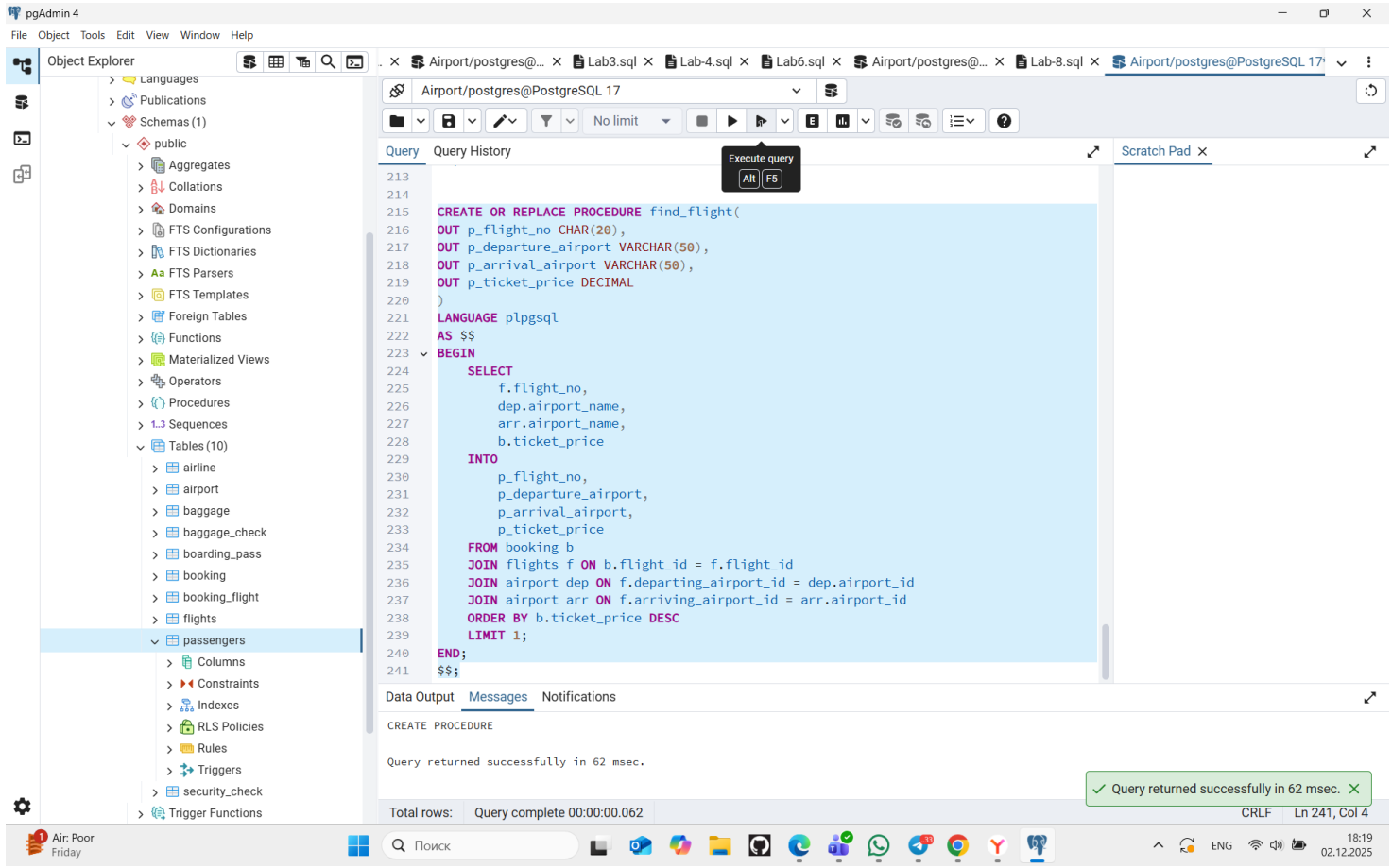
TASK #8


pgAdmin 4
File Object Tools Edit View Window Help
Object Explorer
Airport/postgres@PostgreSQL 17
No limit
Query Query History
Scratch Pad
CREATE OR REPLACE FUNCTION count_flights()
RETURNS TABLE (
airline_id INT,
flight_count BIGINT
)
LANGUAGE plpgsql
AS $$
BEGIN
RETURN QUERY
SELECT
f.airline_id,
COUNT(f.flight_id) AS flight_count
FROM flights f
GROUP BY f.airline_id
ORDER BY flight_count DESC;
END;
$$;
Data Output Messages Notifications
CREATE FUNCTION
Query returned successfully in 52 msec.
Query returned successfully in 52 msec.
Total rows:
Query complete 00:00:00.052
CRLF
Ln 191, Col 4

TASK #9


pgAdmin 4
File Object Tools Edit View Window Help
Object Explorer
Airport/postgres@PostgreSQL 17
Query Query History
Scratch Pad
CREATE OR REPLACE PROCEDURE avg_ticket(
p_flight_id INT,
OUT p_avg_price DECIMAL,
OUT p_flight_no CHAR(20)
)
LANGUAGE plpgsql
AS $$
BEGIN
SELECT
AVG(b.ticket_price),
f.flight_no
INTO
p_avg_price,
p_flight_no
FROM booking b
JOIN flights f ON b.flight_id = f.flight_id
WHERE b.flight_id = p_flight_id
GROUP BY f.flight_no;
END;
$$;
Data Output Messages Notifications
CREATE PROCEDURE
Query returned successfully in 51 msec.
Query returned successfully in 51 msec.
Total rows:
Query complete 00:00:00.051
CRLF
Ln 213, Col 1

TASK #10


pgAdmin 4
File Object Tools Edit View Window Help
Object Explorer
Airport/postgres@PostgreSQL 17
No limit
Execute query
Alt F5
Query Query History
Scratch Pad
CREATE OR REPLACE PROCEDURE find_flight(
OUT p_flight_no CHAR(20),
OUT p_departure_airport VARCHAR(50),
OUT p_arrival_airport VARCHAR(50),
OUT p_ticket_price DECIMAL
)
LANGUAGE plpgsql
AS $$
BEGIN
SELECT
f.flight_no,
dep.airport_name,
arr.airport_name,
b.ticket_price
INTO
p_flight_no,
p_departure_airport,
p_arrival_airport,
p_ticket_price
FROM booking b
JOIN flights f ON b.flight_id = f.flight_id
JOIN airport dep ON f.departing_airport_id = dep.airport_id
JOIN airport arr ON f.arriving_airport_id = arr.airport_id
ORDER BY b.ticket_price DESC
LIMIT 1;
END;
$$;
Data Output Messages Notifications
CREATE PROCEDURE
Query returned successfully in 62 msec.
Query returned successfully in 62 msec.
Total rows: Query complete 00:00:00.062
CRLF Ln 241, Col 4
Languages
Publications
Schemas (1)
public
Aggregates
Collations
Domains
FTS Configurations
FTS Dictionaries
FTS Parsers
FTS Templates
Foreign Tables
Functions
Materialized Views
Operators
Procedures
Sequences
Tables (10)
airline
airport
baggage
baggage_check
boarding_pass
booking
booking_flight
flights
passengers
Columns
Constraints
Indexes
RLS Policies
Rules
Triggers
security_check
Trigger Functions
Air: Poor
Friday
Поиск
ENG
18:19
02.12.2025