

# task1

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer pane displays a tree structure of database objects, including Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (10). The 'flights' table under 'Tables(10)' is currently selected, and its columns are listed in the 'Columns (13)' section. One column, 'sch\_departure\_time', is highlighted with a blue selection bar. The main window contains a SQL query editor with the following code:

```
1 CREATE OR REPLACE PROCEDURE add_new_flight(
2     p_flight_id INT,
3     p_sch_departure_time TIMESTAMP,
4     p_sch_arrival_time TIMESTAMP,
5     p_departing_airport_id INT,
6     p_arriving_airport_id INT,
7     p_departing_gate VARCHAR(50),
8     p_arriving_gate VARCHAR(50),
9     p_airline_id INT,
10    p_act_departure_time TIMESTAMP,
11    p_act_arrival_time TIMESTAMP,
12    p_flight_no CHAR(20)
13 )
14 LANGUAGE plpgsql
15 AS $$
16 BEGIN
17     INSERT INTO flights(
18         flight_id,
19         sch_departure_time,
20         sch_arrival_time,
21         departing_airport_id,
22         arriving_airport_id,
23         departing_gate,
24         arriving_gate,
25         airline_id,
26         act_departure_time,
27         act_arrival_time,
28         flight_no,
29         created_at,
30         updated_at
31     )
32     VALUES(
33         p_flight_id,
34         p_sch_departure_time,
35         p_sch_arrival_time,
36         p_departing_airport_id,
37     );
38 END;
39 $$;
```

The status bar at the bottom indicates "Total rows: 0" and "Query complete 00:00:00.047". The bottom right corner shows the date and time: "02.12.2025 17:02".

# TASK 2

The screenshot shows the pgAdmin 4 interface with the following details:

- File Object Tools Edit View Window Help**: The top menu bar.
- Object Explorer**: A tree view on the left showing database objects. It includes sections like Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, and Tables (10). Under Tables, there are entries for airline, airport, baggage, baggage\_check, boarding\_pass, booking, booking\_flight, flights, and columns for each.
- Query Editor**: The main workspace where the following SQL code is written:

```
CREATE OR REPLACE PROCEDURE update_flight_arrival(
    p_flight_id INT,
    p_act_arrival_time TIMESTAMP
)
LANGUAGE plpgsql
AS $$

BEGIN
    UPDATE flights
    SET
        act_arrival_time = p_act_arrival_time,
        updated_at = CURRENT_TIMESTAMP
    WHERE flight_id = p_flight_id;
END;
$$;
```

Below the code, the **Data Output** tab shows the result of the query:

CREATE PROCEDURE

Query returned successfully in 47 msec.

At the bottom, the status bar displays:

- Total rows: 0
- Query complete 00:00:00.047
- CRLF | Ln 15, Col 4
- 17:13 02.12.2025

The system tray at the bottom of the screen shows the date and time (17:13 02.12.2025), battery level (4°C), and other system icons.

# TASK #3

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows a tree view of database objects. Under "Tables(10)", the "flights" table is selected, and its columns are listed below it.
- Query Editor:** Displays the SQL code for creating a procedure.

```
CREATE OR REPLACE PROCEDURE get_departing_flights(
    p_departure_airport_id INT
)
LANGUAGE plpgsql
AS $$

BEGIN
    SELECT
        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
    FROM flights
    WHERE departing_airport_id = p_departing_airport_id;
END;
$$;
```
- Data Output:** Shows the message: "Query returned successfully in 49 msec."
- Bottom Status Bar:** Shows "Total rows: 0" and "Query complete 00:00:00.049".

# TASK #4

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** Shows the database structure with tables like `airline`, `airport`, `baggage`, etc., and the `flights` table which has 13 columns.
- Query Editor:** Displays the SQL code for creating a function:

```
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:** Shows the message "CREATE FUNCTION" and "Query returned successfully in 53 msec."
- System Tray:** Shows icons for battery level (3°C), search, file explorer, task manager, and other system utilities.

# TASK #5

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer tree view displays various database objects under the 'public' schema, including Languages, Publications, Schemas, Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, Tables, and the 'passenger' table. The 'passenger' table is currently selected.

The main window contains a Query Editor tab with the following SQL code:

```
109
110 CREATE OR REPLACE PROCEDURE get_passengers_by_flight(
111     p_flight_no CHAR(20)
112 )
113 LANGUAGE plpgsql
114 AS $$
115 BEGIN
116     SELECT
117         p.passenger_id,
118         p.first_name,
119         p.last_name,
120         p.passport_number,
121         b.booking_id,
122         b.seat,
123         b.status
124     FROM passengers p
125     JOIN booking b ON p.passenger_id = b.passenger_id
126     JOIN flights f ON b.flight_id = f.flight_id
127     WHERE f.flight_no = p_flight_no;
128 END;
129 $$;
```

Below the code, the Data Output tab shows the message: "Query returned successfully in 45 msec." The status bar at the bottom right indicates the date and time: 02.12.2025, 17:37, and the system language: ENG.

# TASK #6

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** On the left, under the "Tables (10)" section, the "passenger" table is selected.
- Query Editor:** The main window contains the following PostgreSQL code:

```
END;
$$;

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;
$$;
```
- Status Bar:** At the bottom, it says "Query returned successfully in 76 msec." and "CRLF Ln 147, Col 4".
- Taskbar:** The taskbar at the bottom includes icons for Start, Search, File Explorer, Task View, Edge, Mail, Calendar, Photos, OneDrive, Microsoft Store, and Google Chrome.

# TASK #7

The screenshot shows the pgAdmin 4 interface. On the left, the Object Explorer tree view is expanded to show the 'public' schema, which contains various objects like Aggregates, Collations, Domains, FTS Configurations, FTS Dictionaries, FTS Parsers, FTS Templates, Foreign Tables, Functions, Materialized Views, Operators, Procedures, Sequences, Tables (10), and Passengers. The 'passengers' table is currently selected. The main right pane displays a SQL query editor with the following code:

```
149 CREATE OR REPLACE PROCEDURE find_flights_delayed_over()
150 LANGUAGE plpgsql
151 AS $$
152 DECLARE
153     flight_record RECORD;
154 BEGIN
155     FOR flight_record IN
156         SELECT
157             f.flight_id,
158             f.flight_no,
159             f.sch_departure_time,
160             f.act_departure_time,
161             (f.act_departure_time - f.sch_departure_time) AS delay_interval
162         FROM flights f
163         WHERE (f.act_departure_time - f.sch_departure_time) > INTERVAL '24 hours'
164     LOOP
165         RAISE NOTICE 'Flight ID: %, Number: %, Scheduled: %, Actual: %, Delay: %',
166             flight_record.flight_id,
167             flight_record.flight_no,
168             flight_record.sch_departure_time,
169             flight_record.act_departure_time,
170             flight_record.delay_interval;
171     END LOOP;
172 END;
173 $$;
```

The 'Data Output' tab at the bottom of the query editor shows the message "Query returned successfully in 56 msec." The status bar at the bottom right indicates "CRLF" and "Ln 173, Col 4".

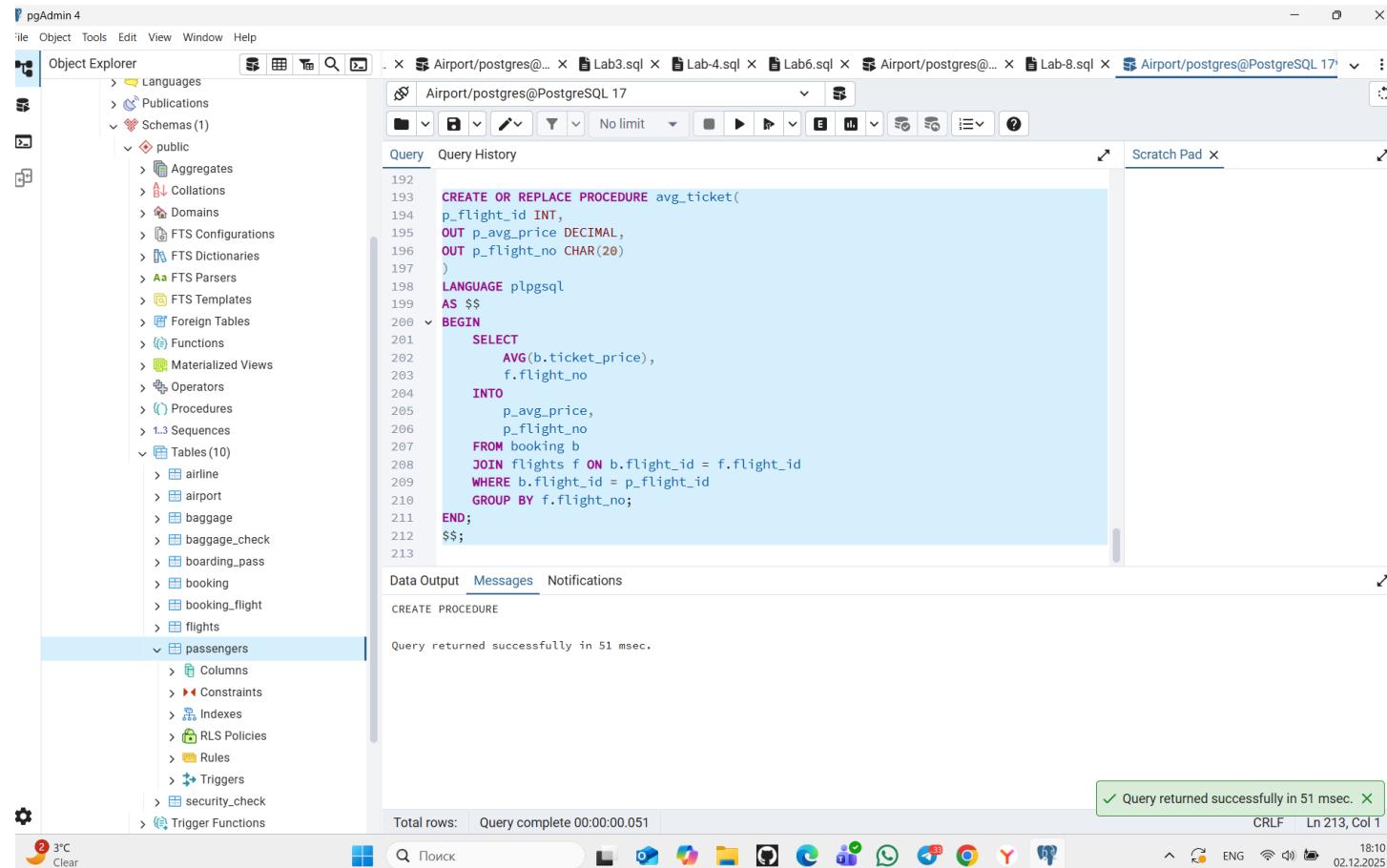
# TASK #8

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** On the left, under the "Tables (10)" section, the "passengers" table is selected.
- Query Editor:** The main window contains the following PostgreSQL code:

```
167     flight_reorder.flight_no,
168     flight_record.sch_departure_time,
169     flight_record.act_departure_time,
170     flight_record.delay_interval;
171   END LOOP;
172 END;
173 $$;
174
175 CREATE OR REPLACE FUNCTION count_flights()
176 RETURNS TABLE (
177   airline_id INT,
178   flight_count BIGINT
179 )
180 LANGUAGE plpgsql
181 AS $$
182 BEGIN
183   RETURN QUERY
184   SELECT
185     f.airline_id,
186     COUNT(f.flight_id) AS flight_count
187   FROM flights f
188   GROUP BY f.airline_id
189   ORDER BY flight_count DESC;
190 END;
191 $$;
```
- Data Output:** Below the code, it says "CREATE FUNCTION".
- Messages:** A green message box at the bottom right says "Query returned successfully in 52 msec."
- System Bar:** At the bottom, it shows "Total rows: Query complete 00:00:00.052" and "CRLF Ln 191, Col 4".
- Taskbar:** The taskbar at the very bottom includes icons for Start, Search, File Explorer, Task View, Edge, WhatsApp, Microsoft Teams, and Google Chrome.

# TASK #9



The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** On the left, under the "public" schema, the "Tables" node is expanded, showing 10 tables: airline, airport, baggage, baggage\_check, boarding\_pass, booking, booking\_flight, flights, and passengers.
- Query Editor:** The main window displays a SQL script for creating a procedure named "avg\_ticket".

```
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;
$$;
```
- Status Bar:** At the bottom right, a green message box indicates: "✓ Query returned successfully in 51 msec." Below it, the status bar shows: Total rows: 0, Query complete 00:00:00.051, CRLF, Ln 213, Col 1, 18:10, ENG, 02.12.2025.

# TASK #10

The screenshot shows the pgAdmin 4 interface with the following details:

- Object Explorer:** On the left, under the "Tables (10)" section, the "passengers" table is selected.
- Query Tab:** The main area contains a SQL procedure definition:

```
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:** Below the query, it says "Query returned successfully in 62 msec."
- Status Bar:** At the bottom right, it shows "18:19 ENG ⏺ 02.12.2025".