```sql
-- task 1
CREATE OR REPLACE PROCEDURE insert_flight(
    p_sch_dep timestamp,
    p_sch_arr timestamp,
    p_dep_airport int,
    p_arr_airport int,
    p_dep_gate varchar,
    p_arr_gate varchar,
    p_airline_id int
)
LANGUAGE plpgsql
AS $$
BEGIN
    INSERT INTO flights(
        sch_departure_time, sch_arrival_time,
        departing_airport_id, arriving_airport_id,
        departing_gate, arriving_gate, airline_id,
        created_at, updated_at
    ) VALUES (
        p_sch_dep, p_sch_arr,
        p_dep_airport, p_arr_airport,
        p_dep_gate, p_arr_gate, p_airline_id,
        NOW(), NOW()
    );
END;
$$;
```

Data Output Messages Notifications

CREATE PROCEDURE

Query returned successfully in 172 msec.

```
22        p_dep_gate, p_arr_gate, p_airline_id,
23            NOW(), NOW()
24      );
25  END;
26  $$;
27
28  -- task 2
29  CREATE OR REPLACE PROCEDURE update_flight_status(
30      p_flight_id int,
31      p_status varchar
32  )
33  LANGUAGE plpgsql
34  AS $$
35  BEGIN
36      UPDATE booking
37      SET status = p_status,
38          updated_at = NOW()
39      WHERE flight_id = p_flight_id;
40  END;
41  $$;
```

Data Output   **Messages**   Notifications

```
CREATE PROCEDURE

Query returned successfully in 137 msec.
```

```sql
BEGIN
    UPDATE booking
    SET status = p_status,
        updated_at = NOW()
    WHERE flight_id = p_flight_id;
END;
$$;

-- task 3
CREATE OR REPLACE PROCEDURE flights_from_airport(
    p_airport_id int
)
LANGUAGE plpgsql
AS $$
BEGIN
    SELECT *
    FROM flights
    WHERE departing_airport_id = p_airport_id;
END;
$$;
```

Data Output · Messages · Notifications

```
CREATE PROCEDURE

Query returned successfully in 107 msec.
```

✓ Query returned successfully in 107 msec. ✕

```
54    $$;
55
56    -- task 4
57    CREATE OR REPLACE FUNCTION avg_arrival_delay(p_airport_id int)
58    RETURNS interval
59    LANGUAGE plpgsql
60    AS $$
61    DECLARE
62        result interval;
63    BEGIN
64        SELECT AVG(act_arrival_time - sch_arrival_time)
65        INTO result
66        FROM flights
67        WHERE arriving_airport_id = p_airport_id;
68
69        RETURN result;
70    END;
71    $$;
72
73    -- task 5
```

Data Output   Messages   Notifications

```
CREATE FUNCTION

Query returned successfully in 146 msec.
```

```
65            INTO result
66        FROM flights
67        WHERE arriving_airport_id = p_airport_id;
68
69        RETURN result;
70    END;
71    $$;
72
73    -- task 5
74    CREATE OR REPLACE PROCEDURE passengers_of_flight(p_flight_id int)
75    LANGUAGE plpgsql
76    AS $$
77    BEGIN
78        SELECT p.*
79        FROM passengers p
80        JOIN booking b ON b.passenger_id = p.passenger_id
81        WHERE b.flight_id = p_flight_id;
82    END;
83    $$;
84
```

Data Output   Messages   Notifications

```
CREATE PROCEDURE

Query returned successfully in 109 msec.
```

Total rows: Query complete 00:00:00.109

```sql
        FROM passengers p
        JOIN booking b ON b.passenger_id = p.passenger_id
        WHERE b.flight_id = p_flight_id;
END;
$$;

-- task 6
CREATE OR REPLACE PROCEDURE passenger_with_most_flights()
LANGUAGE plpgsql
AS $$
BEGIN
    SELECT p.passenger_id, p.first_name, p.last_name, COUNT(b.flight_id) AS total_flights
    FROM passengers p
    JOIN booking b ON b.passenger_id = p.passenger_id
    GROUP BY p.passenger_id
    ORDER BY total_flights DESC
    LIMIT 1;
END;
$$;
```

Data Output   Messages   Notifications

```
CREATE PROCEDURE

Query returned successfully in 169 msec.
```

Object Explorer

t_db/postgre... × airport_db/postgre... × airport_db/postgre... × public.booking/air... × airport_db/postgres@PostgreSQL 17* × public.booking/air... ×

airport_db/postgres@PostgreSQL 17

Query   Query History   Scratch Pad ×

```
 89 ∨  BEGIN
 90        SELECT p.passenger_id, p.first_name, p.last_name, COUNT(b.flight_id) AS total_flights
 91        FROM passengers p
 92        JOIN booking b ON b.passenger_id = p.passenger_id
 93        GROUP BY p.passenger_id
 94        ORDER BY total_flights DESC
 95        LIMIT 1;
 96    END;
 97    $$;
 98
 99    -- task 7
100    CREATE OR REPLACE PROCEDURE flights_delayed_24h()
101    LANGUAGE plpgsql
102    AS $$
103 ∨  BEGIN
104        SELECT *
105        FROM flights
106        WHERE act_departure_time - sch_departure_time > INTERVAL '24 hours';
107    END;
108    $$;
```

Data Output   Messages   Notifications

```
CREATE PROCEDURE

Query returned successfully in 130 msec.
```

Total rows: Query complete 00:00:00.130

✓ Query returned successfully in 130 msec. ✕

CRLF   Ln 108, Col 4

Object Explorer columns:
- Columns (8)
  - booking_id
  - flight_id
  - passenger_id
  - booking_platforr
  - created_at
  - updated_at
  - status
  - ticket_price
- Constraints
- Indexes
- RLS Policies
- Rules
- Triggers
- booking_flight
- flights
- passengers
  - Columns (10)
    - passenger_id
    - first_name
    - last_name
    - date_of_birth
    - gender
    - country_of_citize
    - country_of_resid
    - passport_numbe
    - created_at
    - updated_at
  - Constraints
  - Indexes

```
106      WHERE act_departure_time - sch_departure_time > INTERVAL '24 hours';
107  END;
108  $$;
109
110  -- task 8
111  CREATE OR REPLACE FUNCTION count_flights_per_airline()
112  RETURNS TABLE(
113      airline_id int,
114      total_flights bigint
115  )
116  LANGUAGE plpgsql
117  AS $$
118  BEGIN
119      RETURN QUERY
120      SELECT airline_id, COUNT(*)
121      FROM flights
122      GROUP BY airline_id;
123  END;
124  $$;
125
```

Data Output   Messages   Notifications

CREATE FUNCTION

Query returned successfully in 139 msec.

```sql
BEGIN
    RETURN QUERY
    SELECT airline_id, COUNT(*)
    FROM flights
    GROUP BY airline_id;
END;
$$;

-- task 9
CREATE OR REPLACE PROCEDURE avg_ticket_price(p_flight_id int)
LANGUAGE plpgsql
AS $$
BEGIN
    SELECT AVG(price)
    FROM booking
    WHERE flight_id = p_flight_id;
END;
$$;

-- task 10
```

Data Output   Messages   Notifications

CREATE PROCEDURE

Query returned successfully in 591 msec.

✓ Query returned successfully in 591 msec. ✕

```
132          FROM booking
133          WHERE flight_id = p_flight_id;
134      END;
135      $$;
136
137      -- task 10
138      CREATE OR REPLACE PROCEDURE highest_ticket_flight()
139      LANGUAGE plpgsql
140      AS $$
141    ∨ BEGIN
142          SELECT f.flight_id,
143                 f.departing_airport_id,
144                 f.arriving_airport_id,
145                 b.price
146          FROM flights f
147          JOIN booking b ON b.flight_id = f.flight_id
148          ORDER BY b.price DESC
149          LIMIT 1;
150      END;
151      $$;
```

Data Output   Messages   Notifications

```
CREATE PROCEDURE

Query returned successfully in 1 secs 236 msec.
```