```sql
CREATE OR REPLACE PROCEDURE update_flight_status(
    p_flight_id  INT,
    p_new_status TEXT
)
LANGUAGE plpgsql
AS $$
BEGIN
    UPDATE flights
    SET status = p_new_status
    WHERE flight_id = p_flight_id;

    IF NOT FOUND THEN
        RAISE NOTICE 'Flight with id % not found', p_flight_id;
    END IF;
END;
$$;

```

```
CREATE OR REPLACE PROCEDURE list_flights_from_airport(
    p_departing_airport_id INT
)
LANGUAGE plpgsql
AS $$
BEGIN
    RAISE NOTICE '%', p_departing_airport_id;
END;
$$;

```

```sql
CREATE OR REPLACE FUNCTION avg_arrival_delay_for_airport(
    p_airport_id INT
)
RETURNS INTERVAL
LANGUAGE plpgsql
AS $$
DECLARE
    v_avg_delay INTERVAL;
BEGIN
    SELECT AVG(act_arrival_time - sch_arrival_time)
    INTO v_avg_delay
    FROM flights
    WHERE arriving_airport_id = p_airport_id
      AND act_arrival_time IS NOT NULL;

    RETURN v_avg_delay;
END;
$$;
```

```sql
CREATE OR REPLACE PROCEDURE add_flight(
    p_sch_departure_time   TIMESTAMP,
    p_sch_arrival_time     TIMESTAMP,
    p_departing_airport_id INT,
    p_arriving_airport_id  INT,
    p_airline_id           INT,
    p_departing_gate       TEXT,
    p_arriving_gate        TEXT
)
LANGUAGE plpgsql
AS $$
BEGIN
    INSERT INTO flights (
        sch_departure_time,
        sch_arrival_time,
        departing_airport_id,
        arriving_airport_id,
        airline_id,
        departing_gate,
        arriving_gate
    )
    VALUES (
        p_sch_departure_time,
        p_sch_arrival_time,
        p_departing_airport_id,
        p_arriving_airport_id,
        p_airline_id,
        p_departing_gate,
        p_arriving_gate
    );
END;
$$;
```

```
CREATE OR REPLACE PROCEDURE list_passengers_for_flight(
    p_flight_id INT
)
LANGUAGE plpgsql
AS $$
BEGIN
    RAISE NOTICE '%', p_flight_id;
END;
$$;
```

```
1
2  CREATE OR REPLACE PROCEDURE top_passenger_by_flights()
3  LANGUAGE plpgsql
4  AS $$
5  DECLARE
6      v_passenger_id INT;
7      v_first_name TEXT;
8      v_last_name TEXT;
9      v_flights_cnt INT;
10 BEGIN
11     SELECT b.passenger_id,
12            P.first_name,
13            P.last_name,
14            COUNT(*)
15     INTO v_passenger_id, v_first_name, v_last_name, v_flights_cnt
16     FROM booking b
17     JOIN passengers P ON P.passenger_id = b.passenger_id
18     GROUP BY b.passenger_id, P.first_name, P.last_name
19     ORDER BY COUNT(*) DESC
20     LIMIT 1;
21
22     RAISE NOTICE '% % % %', v_passenger_id, v_first_name, v_last_name, v_flights_cnt;
23 END;
24 $$;
```

```
1
2 CREATE OR REPLACE PROCEDURE list_flights_delayed_more_than_24h()
3 LANGUAGE plpgsql
4 AS $$
5 BEGIN
6     RAISE NOTICE 'OK';
7 END;
8 $$;
9
```

```sql

CREATE OR REPLACE PROCEDURE most_expensive_flight()
LANGUAGE plpgsql
AS $$
DECLARE
    v_flight_id INT;
    v_max_price NUMERIC(10,2);
    v_dep_airport TEXT;
    v_arr_airport TEXT;
BEGIN
    SELECT b.flight_id,
           MAX(b.ticket_price)
    INTO v_flight_id, v_max_price
    FROM booking b
    GROUP BY b.flight_id
    ORDER BY MAX(b.ticket_price) DESC
    LIMIT 1;

    SELECT ap_from.airport_name,
           ap_to.airport_name
    INTO v_dep_airport, v_arr_airport
    FROM flights f
    JOIN airport ap_from ON ap_from.airport_id = f.departing_airport_id
    JOIN airport ap_to   ON ap_to.airport_id   = f.arriving_airport_id
    WHERE f.flight_id = v_flight_id;

    RAISE NOTICE '% % % %', v_flight_id, v_dep_airport, v_arr_airport, v_max_price;
```

```sql
CREATE OR REPLACE PROCEDURE avg_ticket_price_for_flight(
    p_flight_id INT
)
LANGUAGE plpgsql
AS $$
DECLARE
    v_avg NUMERIC(10,2);
BEGIN
    SELECT AVG(ticket_price)
    INTO v_avg
    FROM booking
    WHERE flight_id = p_flight_id;

    RAISE NOTICE '%', v_avg;
END;
$$;
```

```sql
CREATE OR REPLACE FUNCTION flights_count_by_airline()
RETURNS TABLE (
    airline_id INT,
    airline_name TEXT,
    flights_count BIGINT
)
LANGUAGE plpgsql
AS $$
BEGIN
    RETURN QUERY
    SELECT A.airline_id,
           A.airline_name,
           COUNT(f.flight_id)
    FROM airline A
    LEFT JOIN flights f ON f.airline_id = A.airline_id
    GROUP BY A.airline_id, A.airline_name
    ORDER BY COUNT(f.flight_id) DESC;
END;
$$;
```