

17

Query returned successfully in 60 msec.

✗

CRLF	Ln 17, Col 1
------	--------------

```
1 ✓ create or replace procedure update_flight_status(  
2     p_flight_number VARCHAR,  
3     p_status VARCHAR  
4 )  
5 as $$  
6 begin  
7     update flights  
8     set status = p_status  
9     where flight_number = p_flight_number;  
10 end;  
11 $$ language plpgsql;
```

Data Output **Messages** Notifications

CREATE PROCEDURE

Query returned successfully in 42 msec.

Total rows: Query complete 00:00:00.042

CRLF Ln 11, Col 21

Welcome [airport/postgres@PostgreSQL 17*](#) x[airport/postgres@PostgreSQL 17](#) x

Query Query History

```
1 create or replace function avg_delay_arrival(p_airport VARCHAR)
2 returns numeric
3 as $$
4     select AVG(delay_minutes)
5     from flights
6     where arrival_airport = p_airport;
7 $$ language sql;
```

[Scratch Pad](#) xData Output [Messages](#) Notifications

```
ERROR:  столбец "delay_minutes" не существует
LINE 4:  select AVG(delay_minutes)
                   ^
```

Total rows: Query complete 00:00:00.041

CRLF Ln 7, Col 17

```
1 ✓ create or replace procedure get_passengers_by_flight(  
2     p_flight_number VARCHAR  
3 )  
4  
5 as $$  
6 begin  
7     select p.*  
8     from passengers p  
9     join booking b on p.passenger_id = b.passenger_id  
10    join booking_flight bf on b.booking_id = bf.booking_id  
11    join flights f on bf.flight_id = f.flight_id  
12    where f.flight_number = p_flight_number;  
13 end;  
14 $$ language plpgsql;
```



```

1  ✓ create or replace procedure get_top_frequent_passenger()
2  as $$
3  begin
4      select
5          p.passenger_id,
6          p.name,
7          count(*) as flights_count
8      from passengers p
9      join booking b on p.passenger_id = b.passenger_id
10     join booking_flight bf on b.booking_id = bf.booking_id
11     group by p.passenger_id, p.name
12     order by flights_count DESC
13     limit 1;
14 end;
15 $$ language plpgsql

```


CREATE PROCEDURE

Query returned successfully in 45 msec.

✓ Query returned successfully in 45 msec. ✗

Total rows: Query complete 00:00:00.045

CRLF Ln 15, Col 20



```
1  ✓ create or replace procedure get_long_delayed_flight()  
2    as $$  
3    begin  
4        select *  
5        from flights  
6        where dela_minutes > 1440;  
7    end;  
8    $$ language plpgsql;
```


CREATE PROCEDURE

Query returned successfully in 42 msec.

✓ Query returned successfully in 42 msec. ✗

Total rows: Query complete 00:00:00.042

CRLF Ln 8, Col 21


```
1  ✓ create or replace function flight_per_airline(p_airline_id INT)
2  returns INT
3  as $$
4      select count(*)
5      from flights
6      where airline_id = p_airline_id;
7  $$ language sql;
```


CREATE FUNCTION

Query returned successfully in 43 msec.

✓ Query returned successfully in 43 msec. ✗

Total rows: Query complete 00:00:00.043

CRLF Ln 7, Col 17


```
1 create or replace procedure avg_ticket_price_flight(  
2     p_flight_number VARCHAR  
3 )  
4 As $$  
5 begin  
6     select avg(ticket_price) as avg_price  
7     from flights  
8     where flight_number = p_flight_number;  
9 end;  
10 $$ language plpgsql;
```

Data Output Messages Notifications

CREATE PROCEDURE

Query returned successfully in 45 msec.

Total rows: Query complete 00:00:00.045

✓ Query returned successfully in 45 msec. ✗

CRLF Ln 10, Col 21

```
1  create or replace procedure get_most_expensive_flight()
2  as $$
3  begin
4      select flight_number, departure_airport, arrival_airport, ticket_price
5      from flights
6      order by ticket_price DESC
7      limit 1;
8  end;
9  $$ language plpgsql;
```

CREATE PROCEDURE

Query returned successfully in 46 msec.

✓ Query returned successfully in 46 msec. ✗

Total rows: Query complete 00:00:00.046

CRLF Ln 9, Col 21