

Welcome × postgres/postgres@localhost\*

postgres/postgres@localhost

No limit

Query History

Query

```
1 CREATE OR REPLACE FUNCTION get_flight_info(p_flight_id INT)
2 RETURNS TABLE (
3     flight_id INT,
4     departing_airport INT,
5     arriving_airport INT,
6     scheduled_departure TIMESTAMP,
7     scheduled_arrival TIMESTAMP
8 )
9 AS $$$
10 BEGIN
11     RETURN QUERY
12     SELECT f.flight_id,
13            f.departing_airport_id,
14            f.arriving_airport_id,
15            f.scheduled_departure,
16            f.scheduled_arrival
17     FROM flights f
18     WHERE f.flight_id = p_flight_id;
19 END;
20 $$ LANGUAGE plpgsql;
```

Total rows: Query complete 00:00:00.623

✓ Query returned successfully in 623 msec. LF Ln 20, Col 21

Welcome × postgres/postgres@localhost\*

postgres/postgres@localhost No limit

Query History

```
1 CREATE OR REPLACE FUNCTION get_passengers_by_country(p_country TEXT)
2 RETURNS TABLE (
3     passenger_id INT,
4     first_name TEXT,
5     last_name TEXT,
6     passport_number TEXT
7 )
8 AS $$ 
9 BEGIN
10    RETURN QUERY
11    SELECT passenger_id, first_name, last_name, passport_number
12    FROM passengers
13    WHERE country_of_citizenship = p_country;
14 END;
15 $$ LANGUAGE plpgsql;
```

Total rows: Query complete 00:00:00.040 ✓ Query returned successfully in 40 msec. LF Ln 15, Col 21

Welcome × postgres/postgres@localhost\*

postgres/postgres@localhost No limit

Query History

```
1 CREATE OR REPLACE PROCEDURE add_passenger(
2     p_first_name TEXT,
3     p_last_name TEXT,
4     p_date_of_birth DATE,
5     p_country TEXT,
6     p_passport TEXT
7 )
8 AS $$ 
9 BEGIN
10    INSERT INTO passengers(first_name, last_name, date_of_birth,
11                           country_of_citizenship, passport_number)
12      VALUES (p_first_name, p_last_name, p_date_of_birth, p_country, p_passport);
13 END;
14 $$ LANGUAGE plpgsql;
```

Total rows: Query complete 00:00:00.039

✓ Query returned successfully in 39 msec. LF Ln 14, Col 21

Welcome × postgres/postgres@localhost\*

postgres/postgres@localhost No limit

Query History

```
1 CREATE OR REPLACE PROCEDURE add_baggage(
2     p_booking_id INT,
3     p_weight NUMERIC
4 )
5 AS $$*
6 BEGIN
7     INSERT INTO baggage(weight_in_kg, created_at, updated_at, booking_id)
8     VALUES (p_weight, NOW(), NOW(), p_booking_id);
9 END;
$ LANGUAGE plpgsql;
```

Total rows: Query complete 00:00:00.039 LF Ln 10, Col 21

Welcome × postgres/postgres@localhost\*

postgres/postgres@localhost No limit

Query History

```
1 CREATE OR REPLACE PROCEDURE update_ticket_price(
2     p_booking_id INT,
3     p_new_price NUMERIC
4 )
5 AS $$ 
6 BEGIN
7     UPDATE booking
8         SET ticket_price = p_new_price
9         WHERE booking_id = p_booking_id;
10 END;
$ LANGUAGE plpgsql;
```

Total rows: Query complete 00:00:00.039 LF Ln 11, Col 21

Welcome × postgres/postgres@localhost\*

postgres/postgres@localhost No limit

Query History

```
1 CREATE OR REPLACE PROCEDURE delay_flight(
2     p_flight_id INT,
3     p_delay_minutes INT
4 )
5 AS $$
6 BEGIN
7     UPDATE flights
8     SET scheduled_departure = scheduled_departure + (p_delay_minutes || ' minutes')::INTERVAL,
9         scheduled_arrival = scheduled_arrival + (p_delay_minutes || ' minutes')::INTERVAL
10    WHERE flight_id = p_flight_id;
11 END;
12 $$ LANGUAGE plpgsql;
```

Total rows: Query complete 00:00:00.031 ✓ Query returned successfully in 31 msec. LF Ln 12, Col 21

Welcome × postgres/postgres@localhost\*

postgres/postgres@localhost No limit

Query History

```
1 CREATE OR REPLACE FUNCTION get_fullname(p_passenger_id INT)
2 RETURNS TEXT
3 AS $$
4 DECLARE
5     fullname TEXT;
6 BEGIN
7     SELECT first_name || ' ' || last_name
8     INTO fullname
9     FROM passengers
10    WHERE passenger_id = p_passenger_id;
11
12    RETURN fullname;
13 END;
$ LANGUAGE plpgsql;
```

Total rows: Query complete 00:00:00.021

✓ Query returned successfully in 21 msec. LF Ln 14, Col 21

Welcome × postgres/postgres@localhost\*

postgres/postgres@localhost

No limit

Query History

Query

```
1 CREATE OR REPLACE FUNCTION count_baggage(p_booking_id INT)
2 RETURNS INT
3 AS $$ 
4 DECLARE
5     bag_count INT;
6 BEGIN
7     SELECT COUNT(*)
8     INTO bag_count
9     FROM baggage
10    WHERE booking_id = p_booking_id;
11
12    RETURN bag_count;
13 END;
$ LANGUAGE plpgsql;
```

Total rows: Query complete 00:00:00.032

✓ Query returned successfully in 32 msec. LF Ln 14, Col 21

Welcome × postgres/postgres@localhost\*

postgres/postgres@localhost No limit

Query History

```
1 CREATE OR REPLACE PROCEDURE delay_all_flights(p_minutes INT)
2 AS $$ 
3 BEGIN
4     UPDATE flights
5         SET scheduled_departure = scheduled_departure + (p_minutes || ' minutes')::INTERVAL,
6             scheduled_arrival = scheduled_arrival + (p_minutes || ' minutes')::INTERVAL;
7 END;
8 $$ LANGUAGE plpgsql;
```

Total rows: Query complete 00:00:00.032

LF Ln 8, Col 21

Welcome × postgres/postgres@localhost\*

postgres/postgres@localhost No limit

Query History

```
1 CREATE OR REPLACE FUNCTION find_passenger_by_passport(p_passport TEXT)
2 RETURNS TABLE(
3     passenger_id INT,
4     first_name TEXT,
5     last_name TEXT,
6     country_of_citizenship TEXT
7 )
8 AS $$ 
9 BEGIN
10    RETURN QUERY
11    SELECT passenger_id, first_name, last_name, country_of_citizenship
12    FROM passengers
13    WHERE passport_number = p_passport;
14 END;
15 $$ LANGUAGE plpgsql;
```

Total rows: Query complete 00:00:00.032 LF Ln 15, Col 21