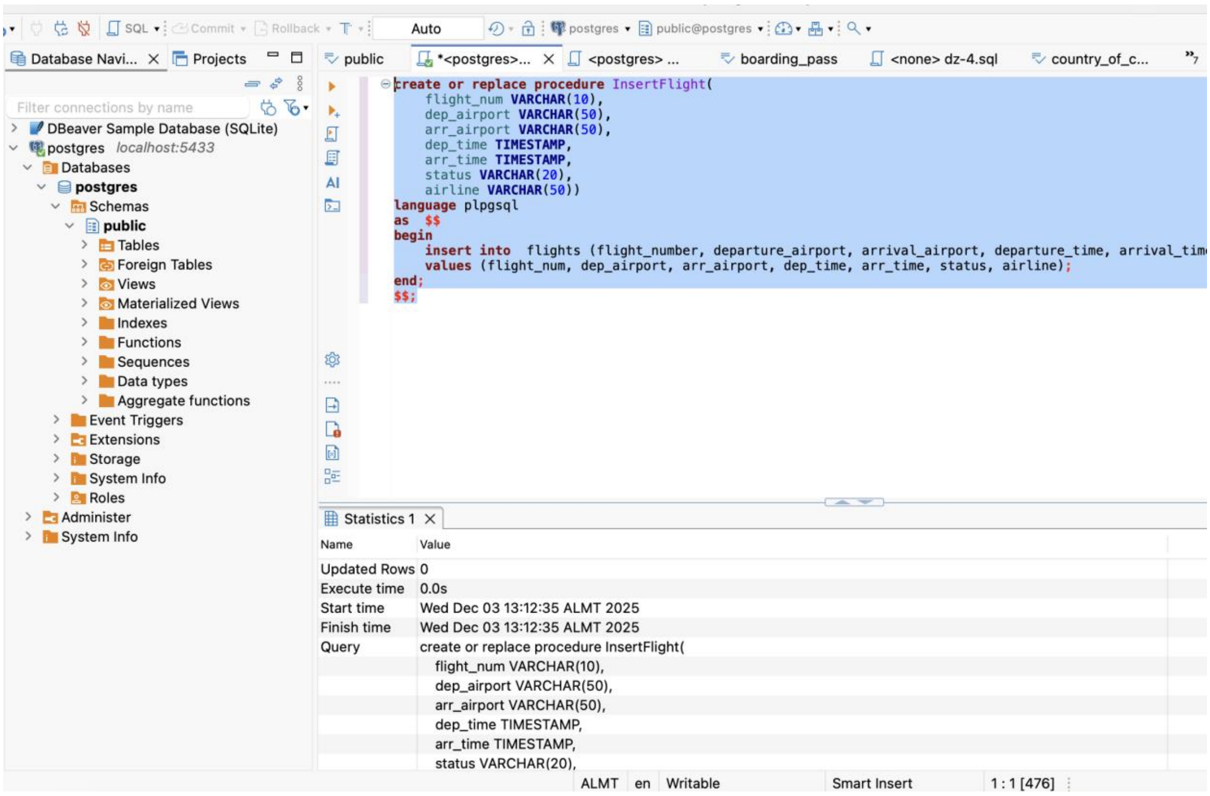
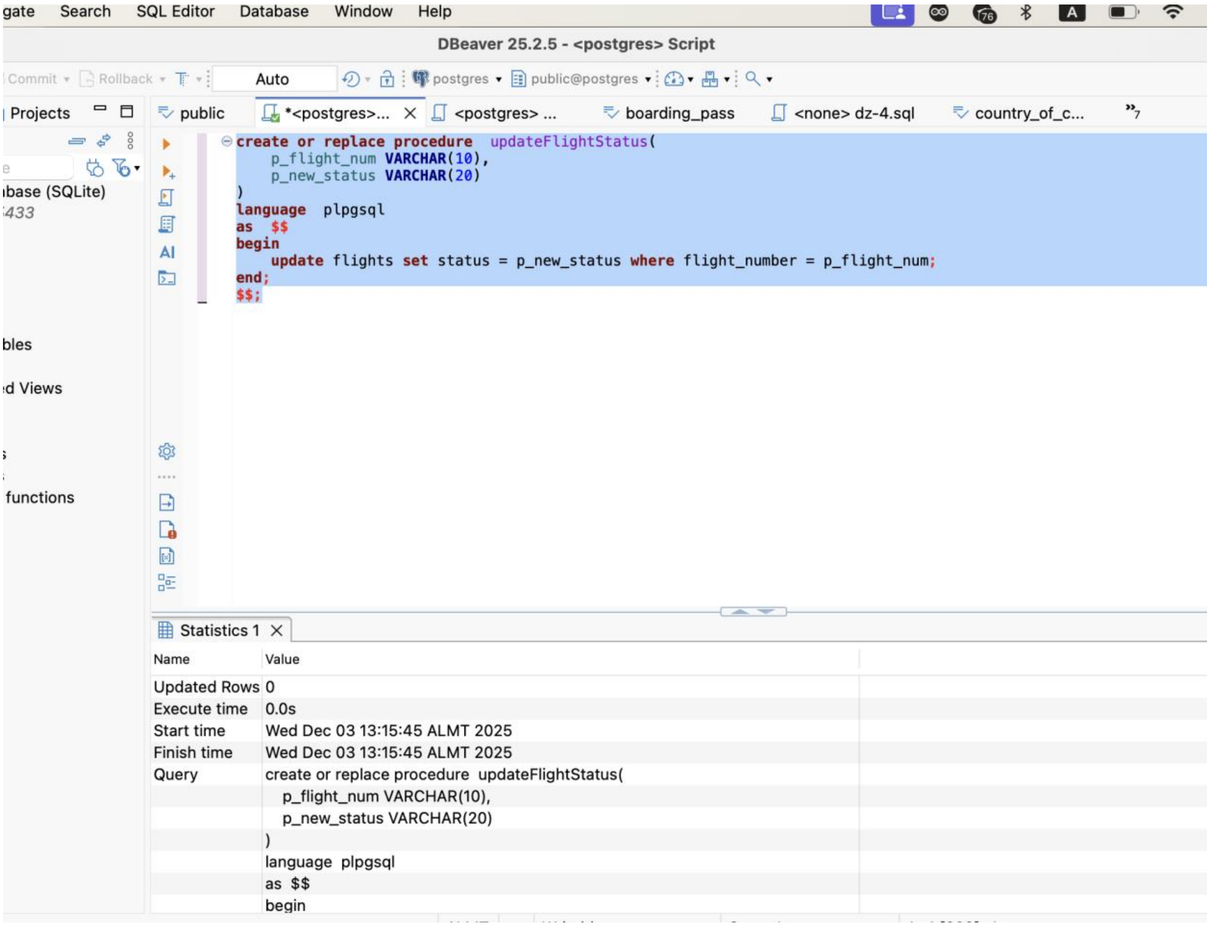


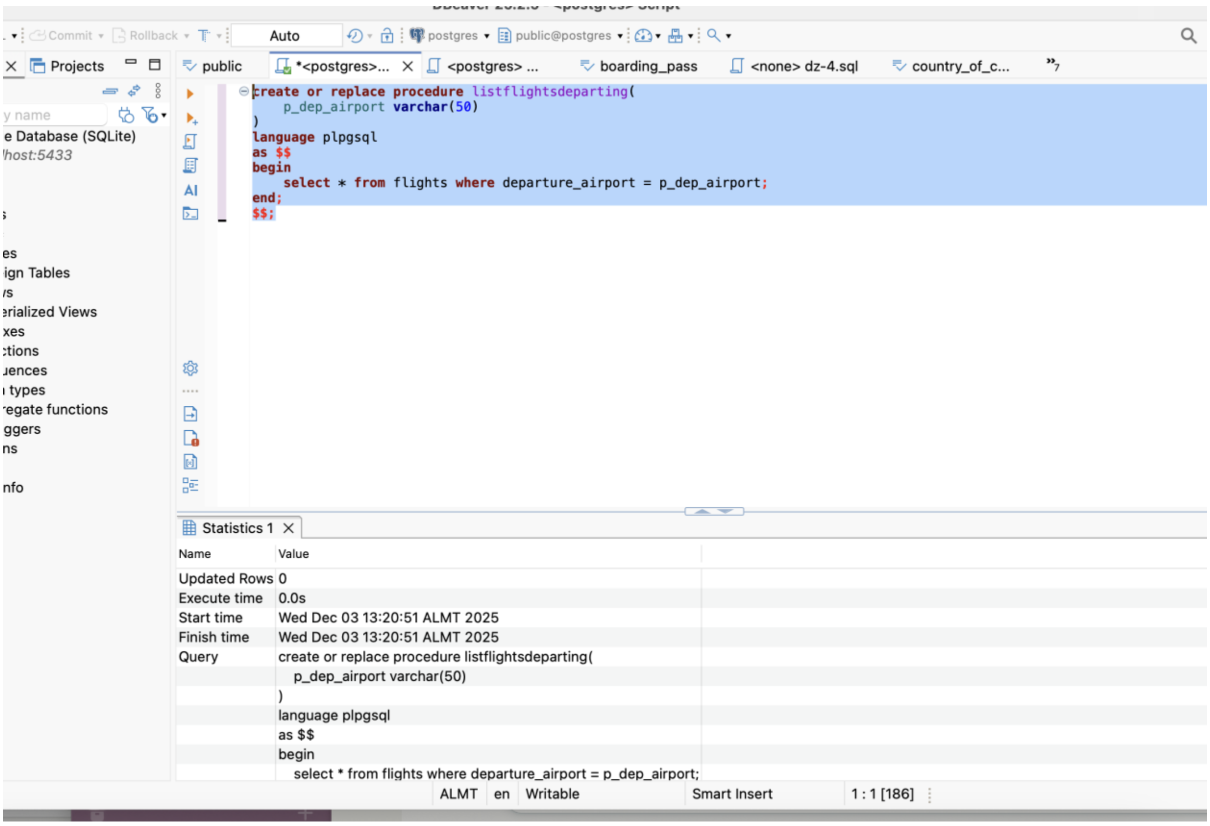
```
create or replace procedure InsertFlight(
    flight_num VARCHAR(10),
    dep_airport VARCHAR(50),
    arr_airport VARCHAR(50),
    dep_time TIMESTAMP,
    arr_time TIMESTAMP,
    status VARCHAR(20),
    airline VARCHAR(50))
language plpgsql
as  $$
begin
    insert into  flights (flight_number, departure_airport, arrival_airport, departure_time, arrival_tim
    values (flight_num, dep_airport, arr_airport, dep_time, arr_time, status, airline);
end;
$$;
```

Name	Value
Updated Rows	0
Execute time	0.0s
Start time	Wed Dec 03 13:12:35 ALMT 2025
Finish time	Wed Dec 03 13:12:35 ALMT 2025
Query	create or replace procedure InsertFlight(flight_num VARCHAR(10), dep_airport VARCHAR(50), arr_airport VARCHAR(50), dep_time TIMESTAMP, arr_time TIMESTAMP, status VARCHAR(20),



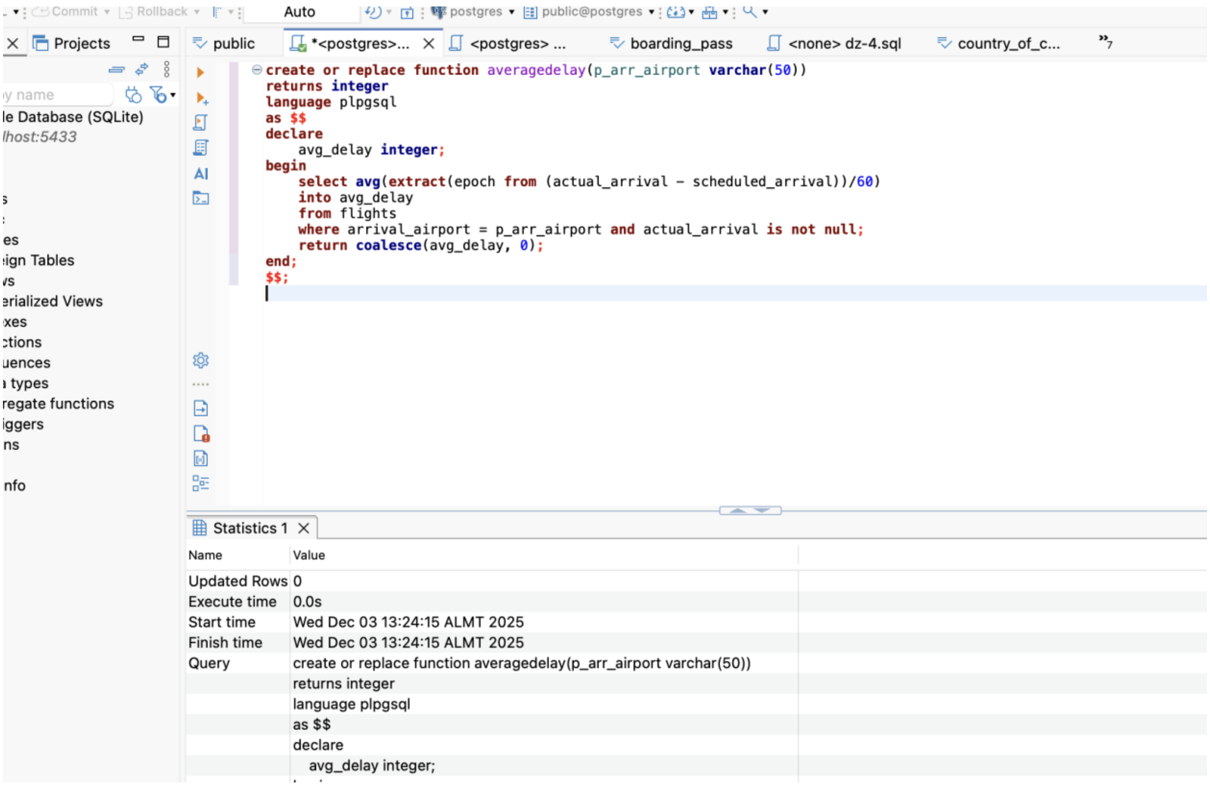
```
create or replace procedure  updateFlightStatus(
    p_flight_num VARCHAR(10),
    p_new_status VARCHAR(20)
)
language  plpgsql
as  $$
begin
    update flights set status = p_new_status where flight_number = p_flight_num;
end;
$$;
```

Name	Value
Updated Rows	0
Execute time	0.0s
Start time	Wed Dec 03 13:15:45 ALMT 2025
Finish time	Wed Dec 03 13:15:45 ALMT 2025
Query	create or replace procedure updateFlightStatus(p_flight_num VARCHAR(10), p_new_status VARCHAR(20)) language plpgsql as $$ begin



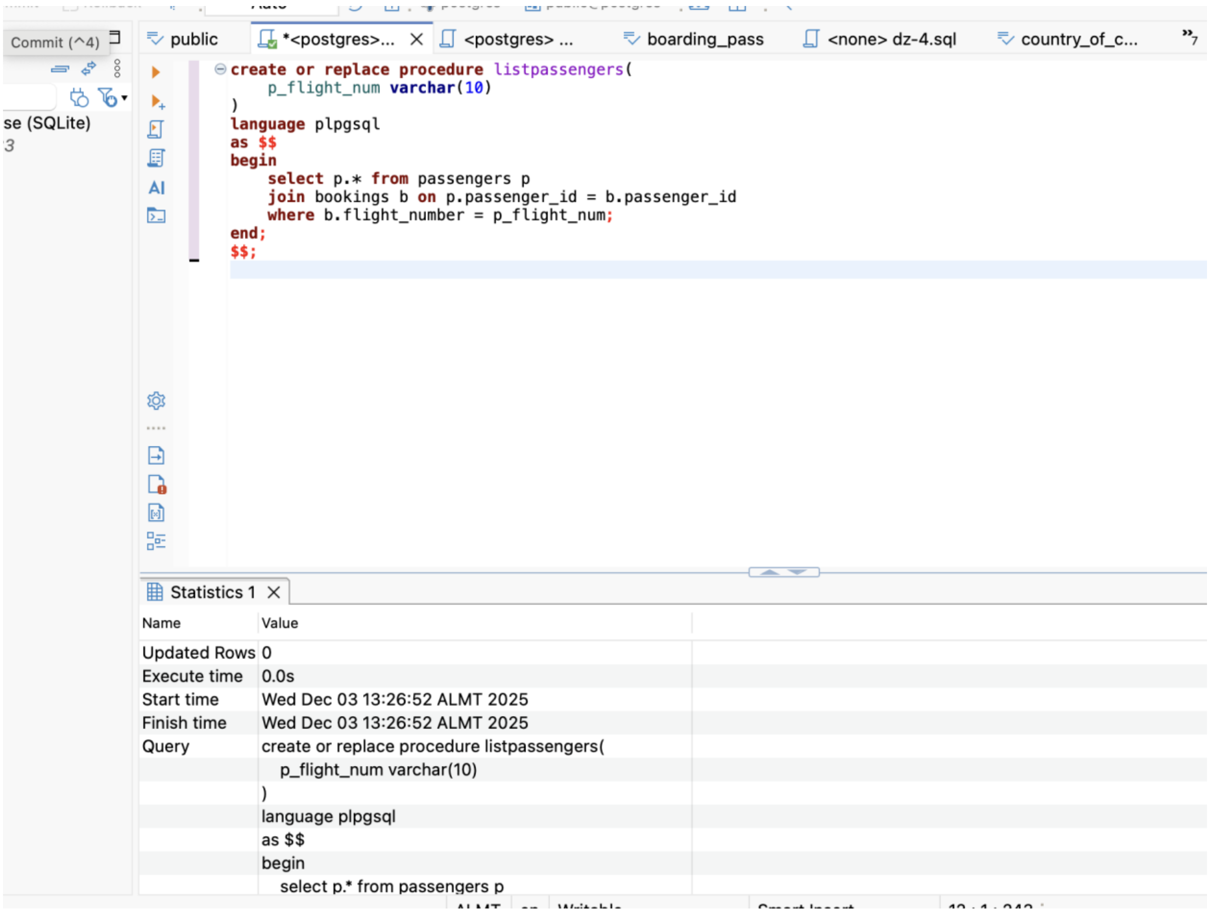
```sql
create or replace procedure listflightsdeparting(
    p_dep_airport varchar(50)
)
language plpgsql
as $$
begin
    select * from flights where departure_airport = p_dep_airport;
end;
$$;
```

```
Name          Value
Updated Rows  0
Execute time  0.0s
Start time    Wed Dec 03 13:20:51 ALMT 2025
Finish time   Wed Dec 03 13:20:51 ALMT 2025
Query         create or replace procedure listflightsdeparting(
                  p_dep_airport varchar(50)
              )
              language plpgsql
              as $$
              begin
                  select * from flights where departure_airport = p_dep_airport;
```



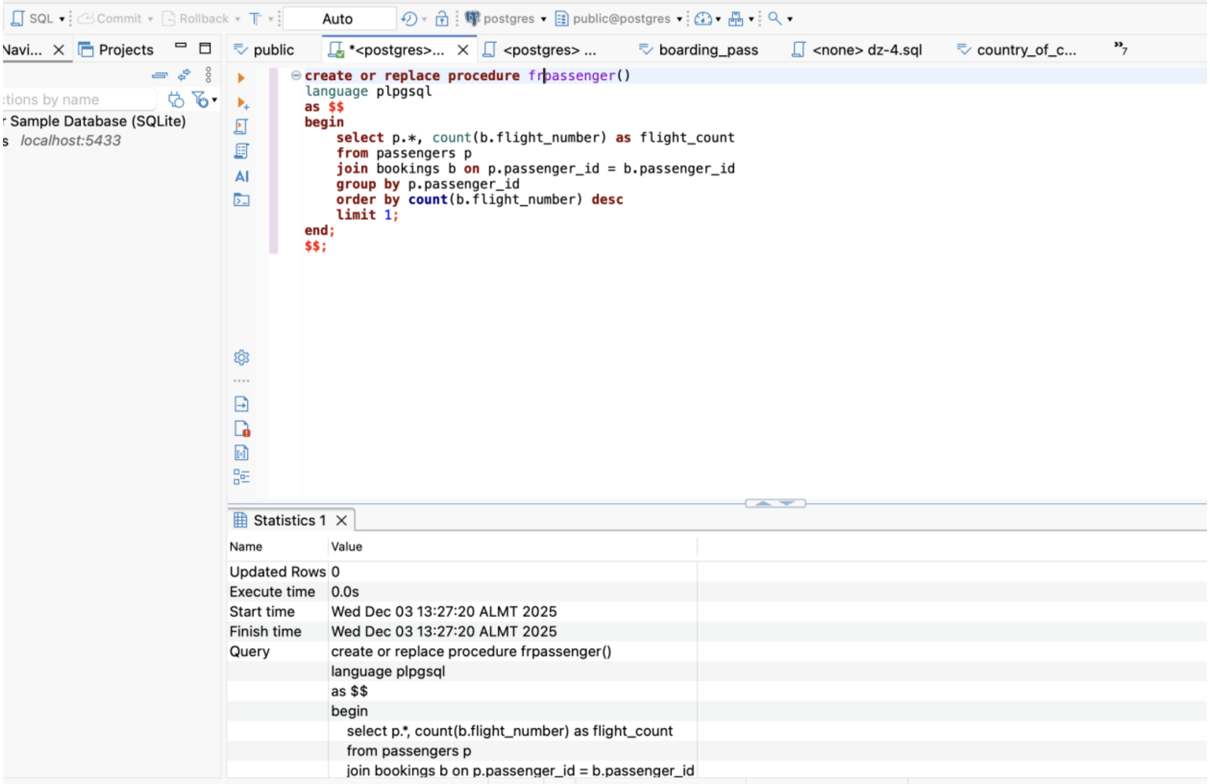
```sql
create or replace function averagedelay(p_arr_airport varchar(50))
returns integer
language plpgsql
as $$
declare
    avg_delay integer;
begin
    select avg(extract(epoch from (actual_arrival - scheduled_arrival))/60)
    into avg_delay
    from flights
    where arrival_airport = p_arr_airport and actual_arrival is not null;
    return coalesce(avg_delay, 0);
end;
$$;
```

```
Name          Value
Updated Rows  0
Execute time  0.0s
Start time    Wed Dec 03 13:24:15 ALMT 2025
Finish time   Wed Dec 03 13:24:15 ALMT 2025
Query         create or replace function averagedelay(p_arr_airport varchar(50))
              returns integer
              language plpgsql
              as $$
              declare
                  avg_delay integer;
```



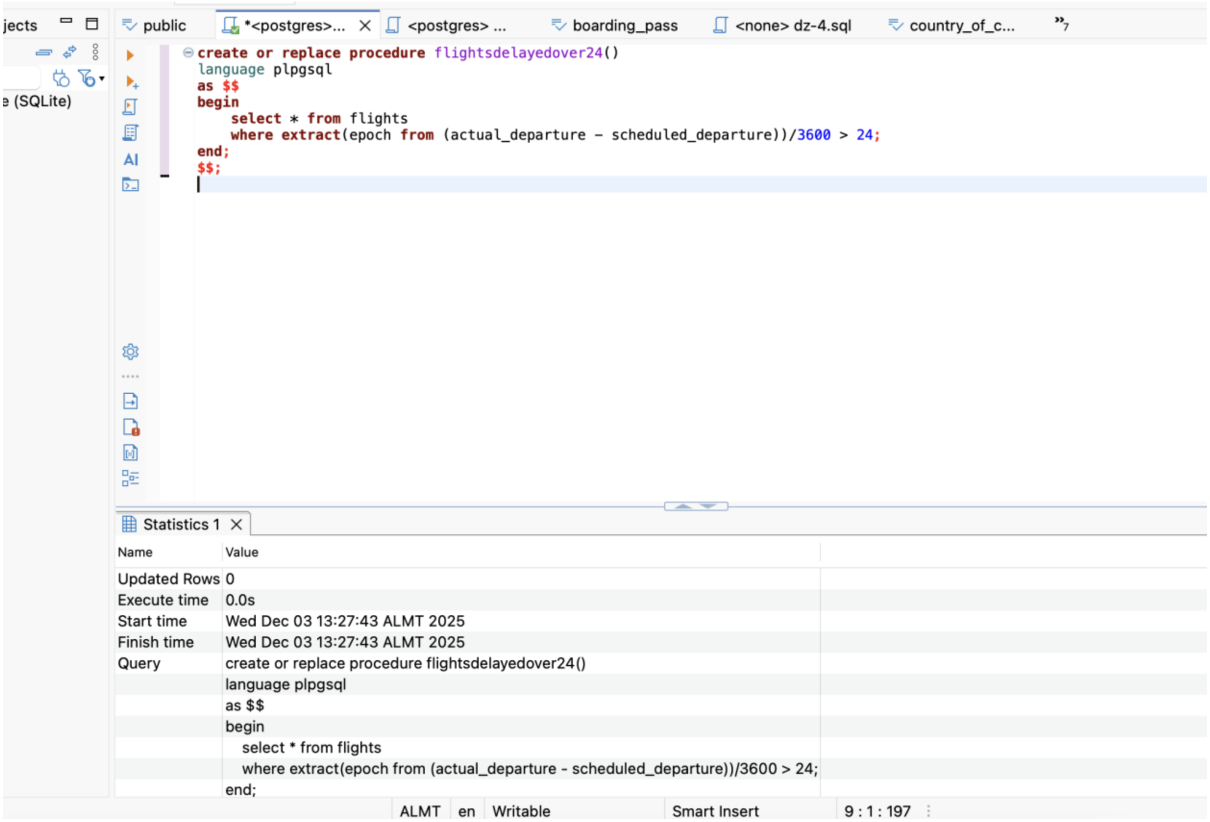
```sql
create or replace procedure listpassengers(
    p_flight_num varchar(10)
)
language plpgsql
as $$
begin
    select p.* from passengers p
    join bookings b on p.passenger_id = b.passenger_id
    where b.flight_number = p_flight_num;
end;
$$;
```

Name	Value
Updated Rows	0
Execute time	0.0s
Start time	Wed Dec 03 13:26:52 ALMT 2025
Finish time	Wed Dec 03 13:26:52 ALMT 2025
Query	create or replace procedure listpassengers(
	p_flight_num varchar(10)
	)
	language plpgsql
	as $$
	begin
	select p.* from passengers p



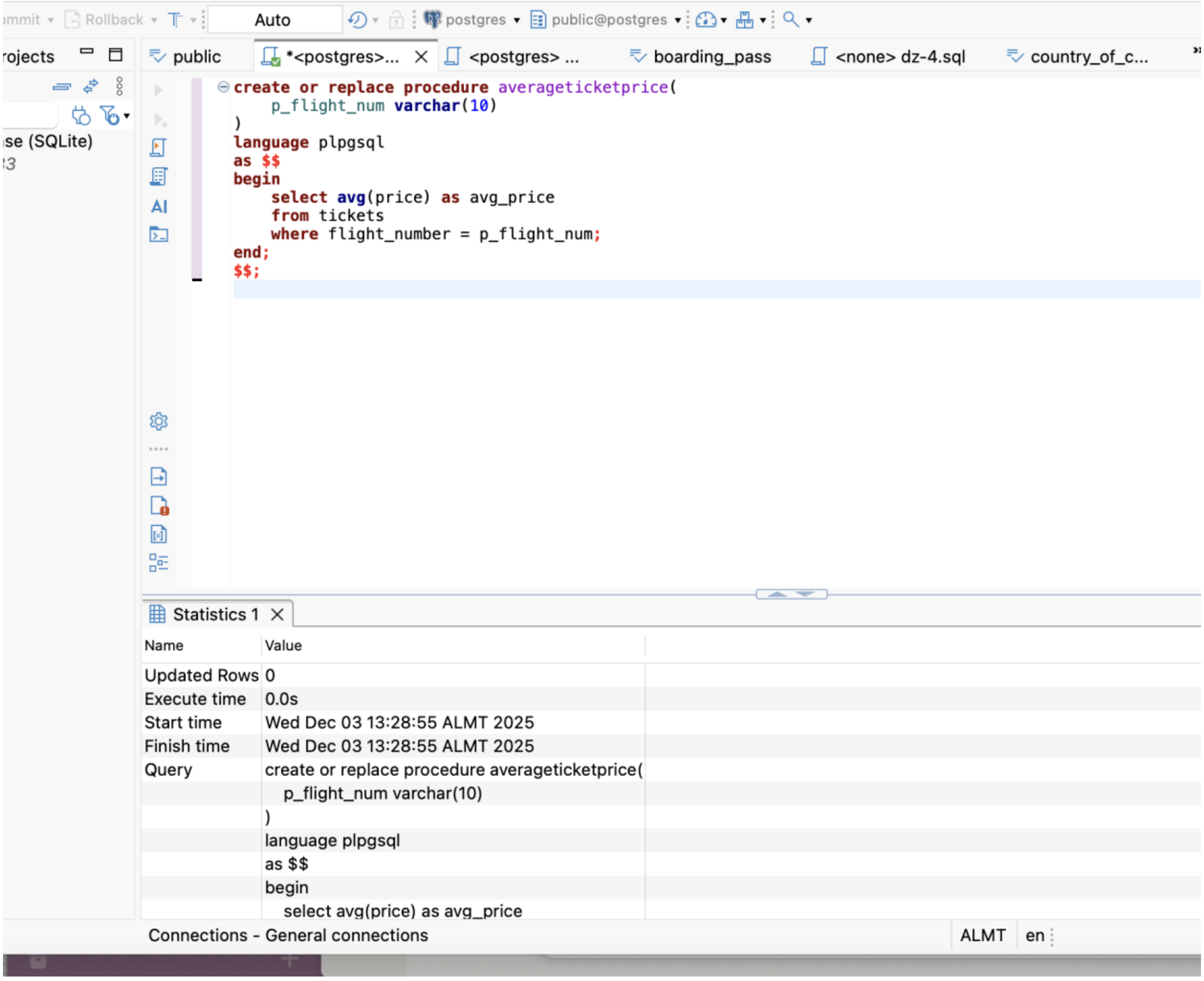

```sql
create or replace procedure frpassenger()
language plpgsql
as $$
begin
    select p.*, count(b.flight_number) as flight_count
    from passengers p
    join bookings b on p.passenger_id = b.passenger_id
    group by p.passenger_id
    order by count(b.flight_number) desc
    limit 1;
end;
$$;
```

Name	Value
Updated Rows	0
Execute time	0.0s
Start time	Wed Dec 03 13:27:20 ALMT 2025
Finish time	Wed Dec 03 13:27:20 ALMT 2025
Query	create or replace procedure frpassenger()
	language plpgsql
	as $$
	begin
	select p.*, count(b.flight_number) as flight_count
	from passengers p
	join bookings b on p.passenger_id = b.passenger_id



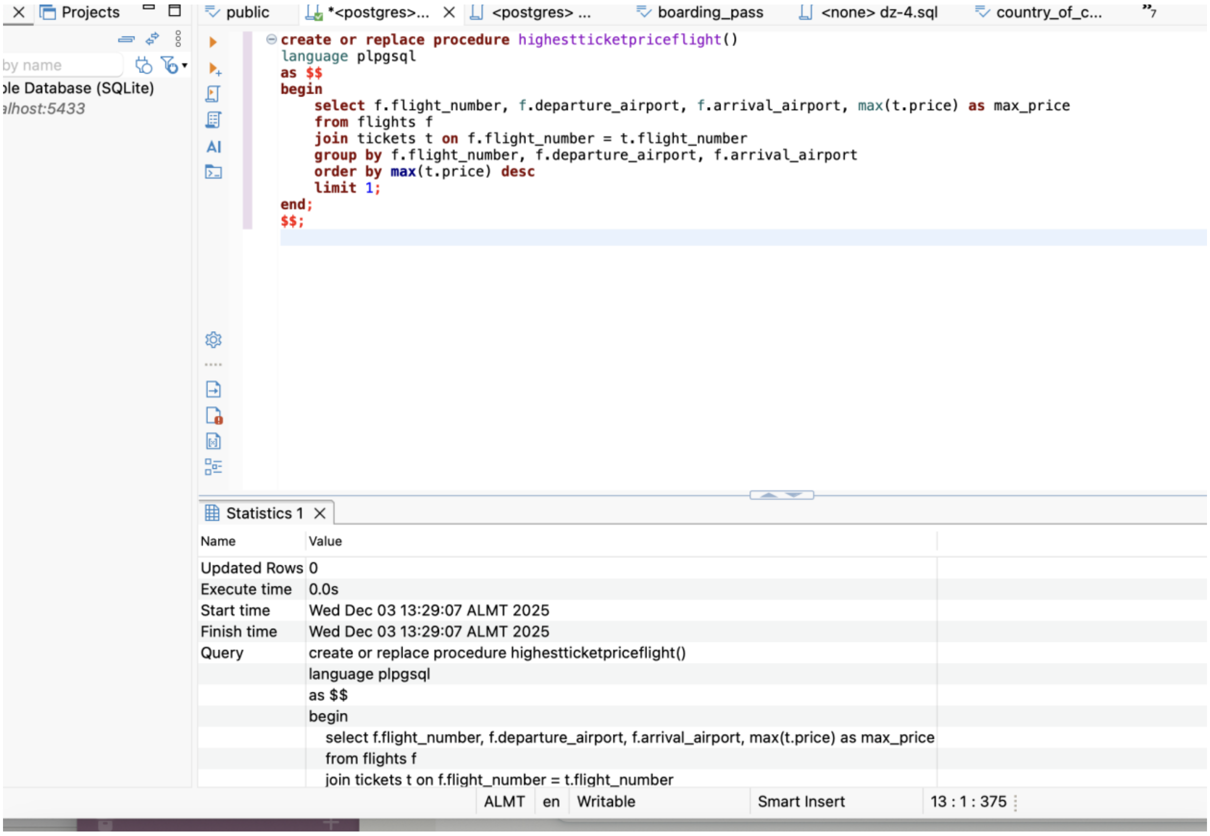
```
create or replace procedure flightsdelayedover24()
language plpgsql
as $$
begin
    select * from flights
    where extract(epoch from (actual_departure - scheduled_departure))/3600 > 24;
end;
$$;
```

Name	Value
Updated Rows	0
Execute time	0.0s
Start time	Wed Dec 03 13:27:43 ALMT 2025
Finish time	Wed Dec 03 13:27:43 ALMT 2025
Query	create or replace procedure flightsdelayedover24() language plpgsql as $$ begin select * from flights where extract(epoch from (actual_departure - scheduled_departure))/3600 > 24; end;




```
create or replace procedure averageticketprice(
    p_flight_num varchar(10)
)
language plpgsql
as $$
begin
    select avg(price) as avg_price
    from tickets
    where flight_number = p_flight_num;
end;
$$;
```

Name	Value
Updated Rows	0
Execute time	0.0s
Start time	Wed Dec 03 13:28:55 ALMT 2025
Finish time	Wed Dec 03 13:28:55 ALMT 2025
Query	create or replace procedure averageticketprice(p_flight_num varchar(10)) language plpgsql as $$ begin select avg(price) as avg_price



```sql
create or replace procedure highestticketpriceflight()
language plpgsql
as $$
begin
    select f.flight_number, f.departure_airport, f.arrival_airport, max(t.price) as max_price
    from flights f
    join tickets t on f.flight_number = t.flight_number
    group by f.flight_number, f.departure_airport, f.arrival_airport
    order by max(t.price) desc
    limit 1;
end;
$$;
```

Name	Value
Updated Rows	0
Execute time	0.0s
Start time	Wed Dec 03 13:29:07 ALMT 2025
Finish time	Wed Dec 03 13:29:07 ALMT 2025
Query	create or replace procedure highestticketpriceflight()
	language plpgsql
	as $$
	begin
	select f.flight_number, f.departure_airport, f.arrival_airport, max(t.price) as max_price
	from flights f
	join tickets t on f.flight_number = t.flight_number