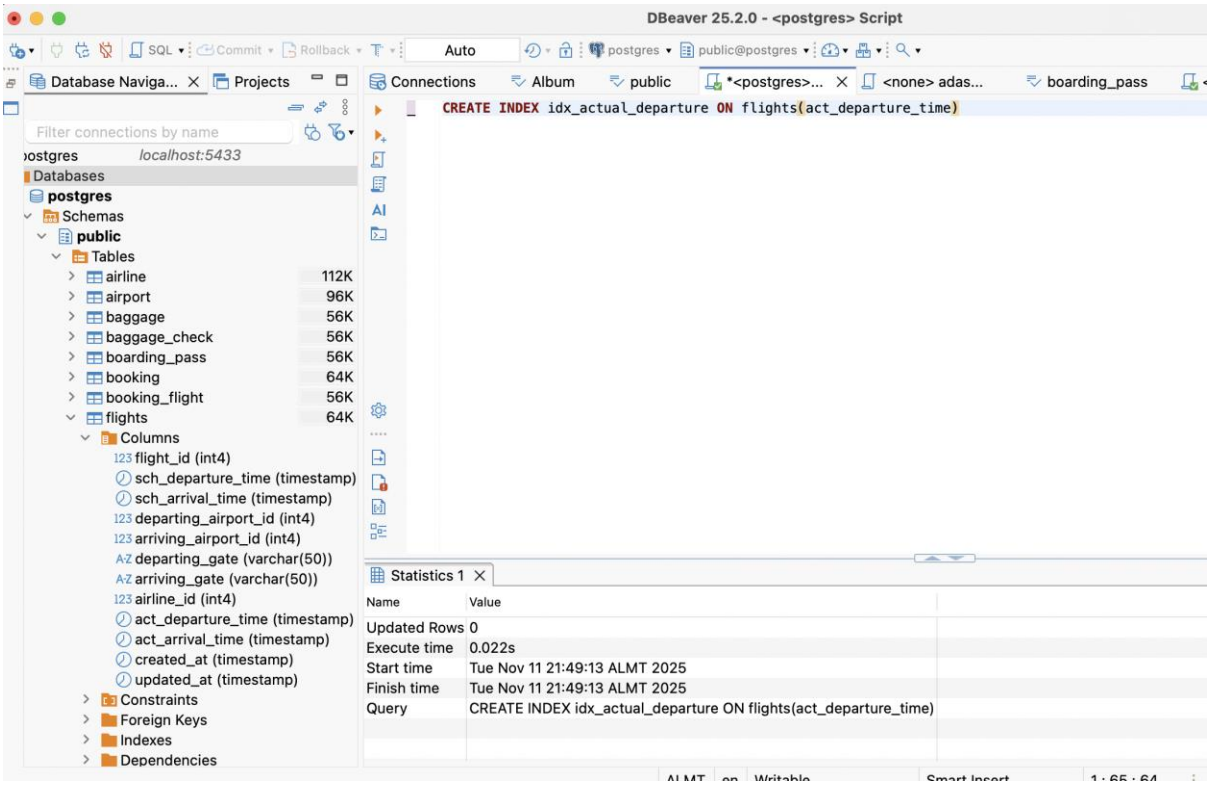
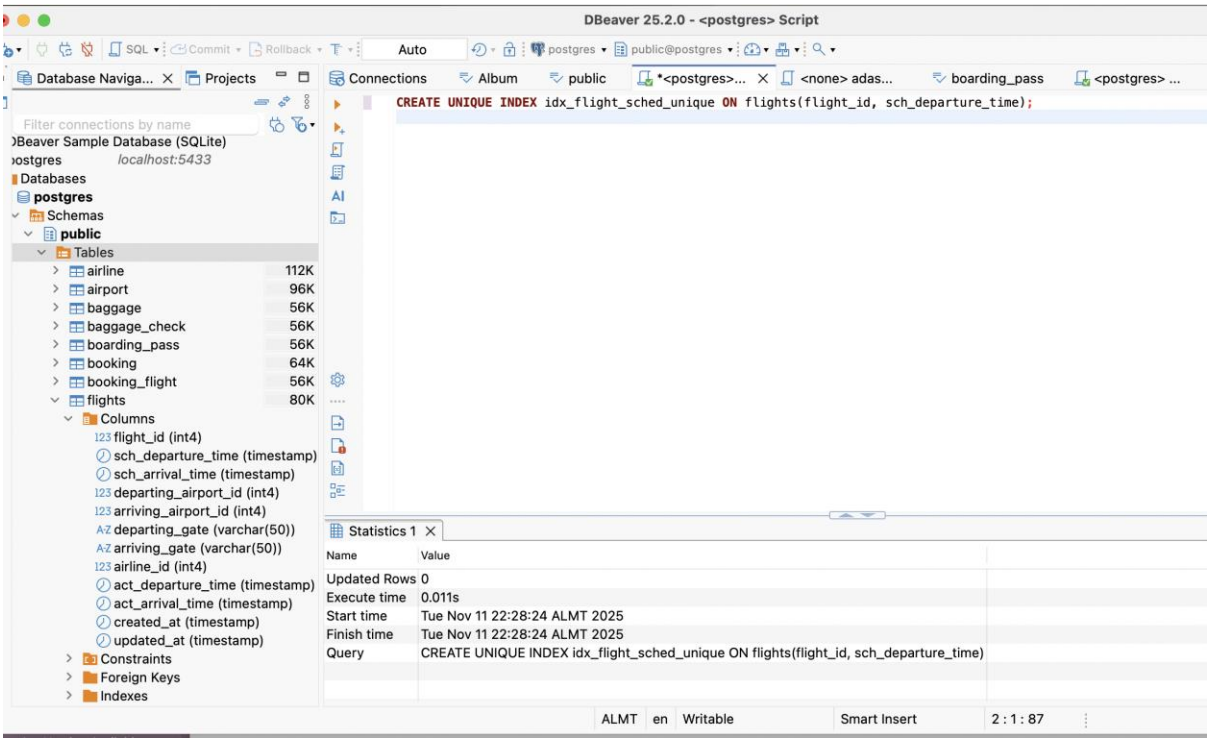
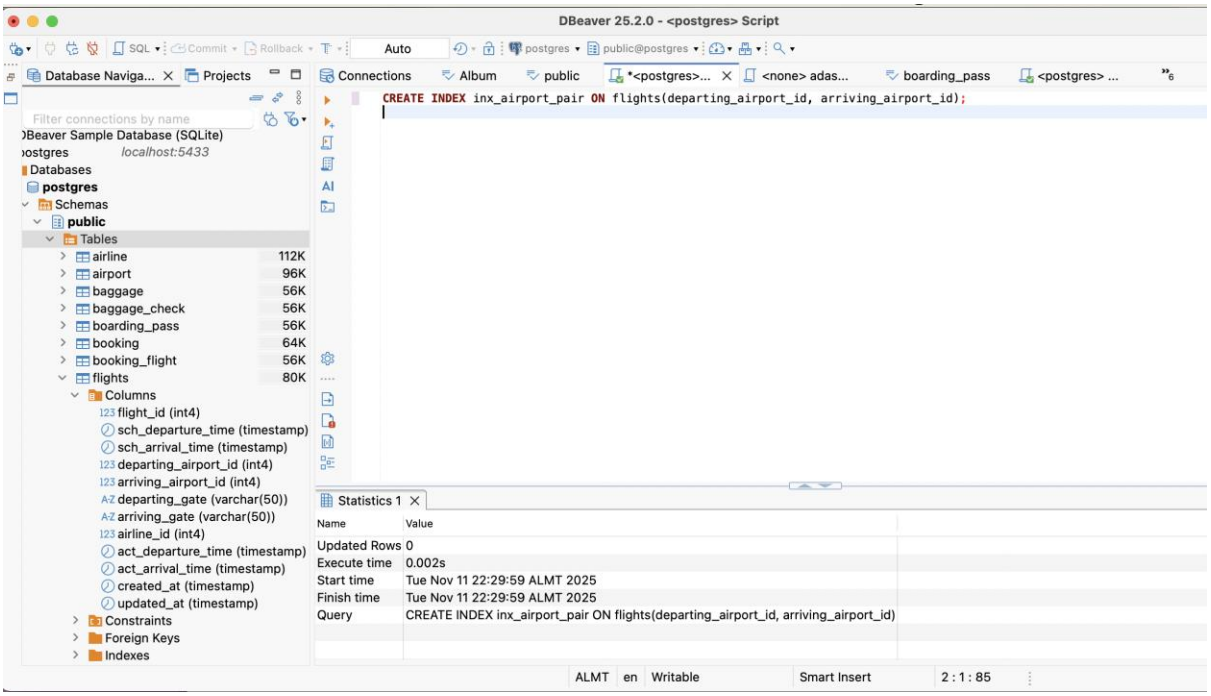
1. Create an index on the actual_departure column in the flights table.



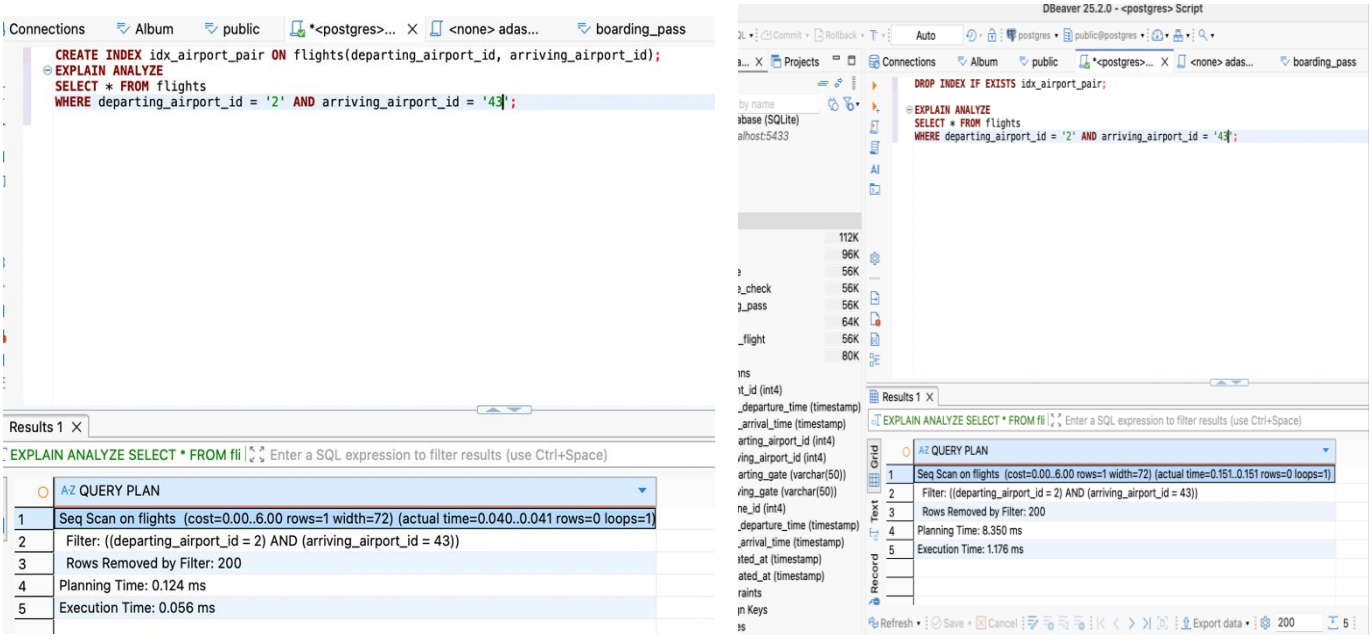
2. Create a unique index to ensure flight_no and scheduled_departure combinations are unique.



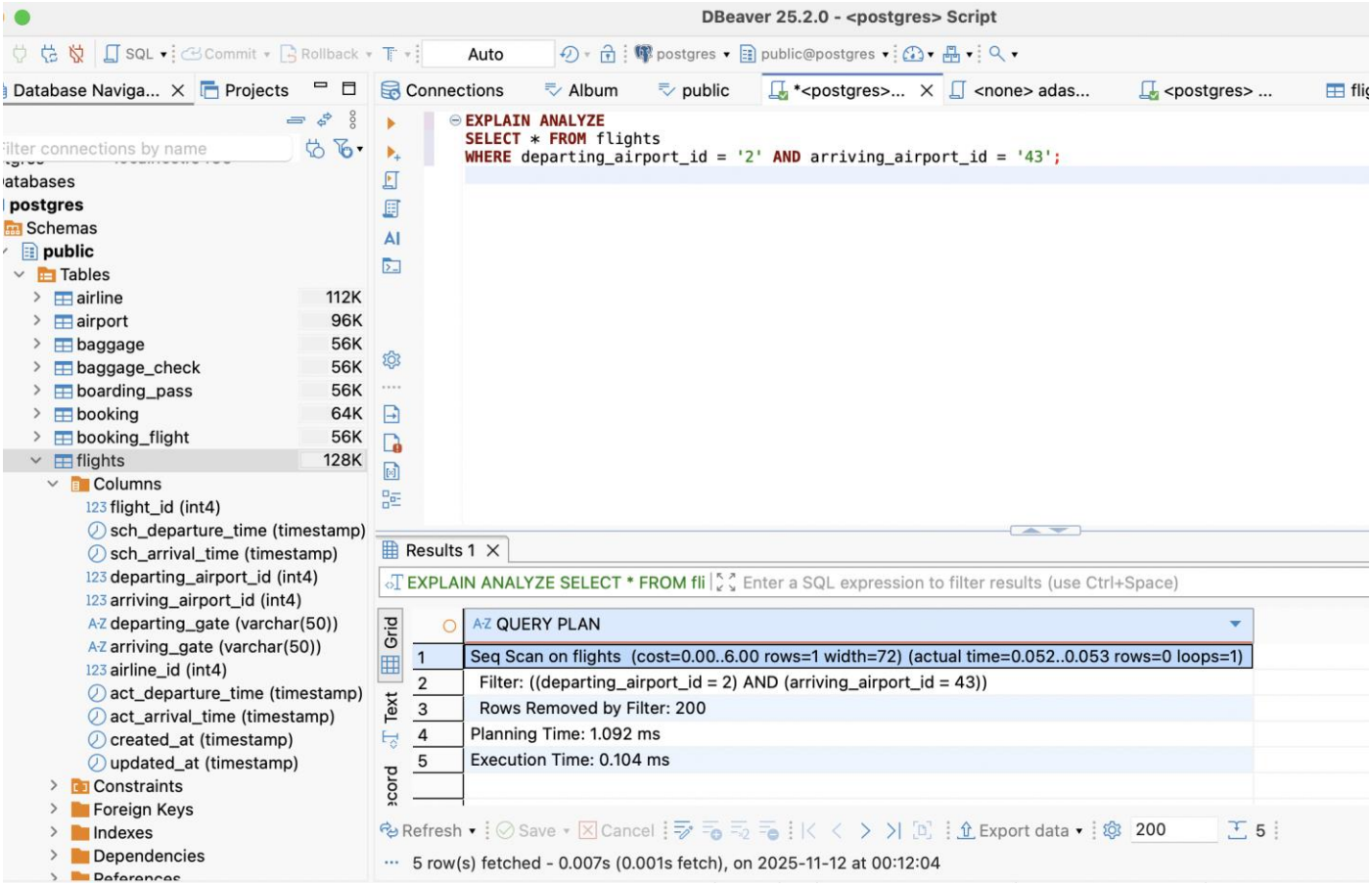
3. Create a composite index on the departure_airport_id and arrival_airport_id columns.



4. Evaluate the difference in query performance with and without indexes. Measure performance differences.

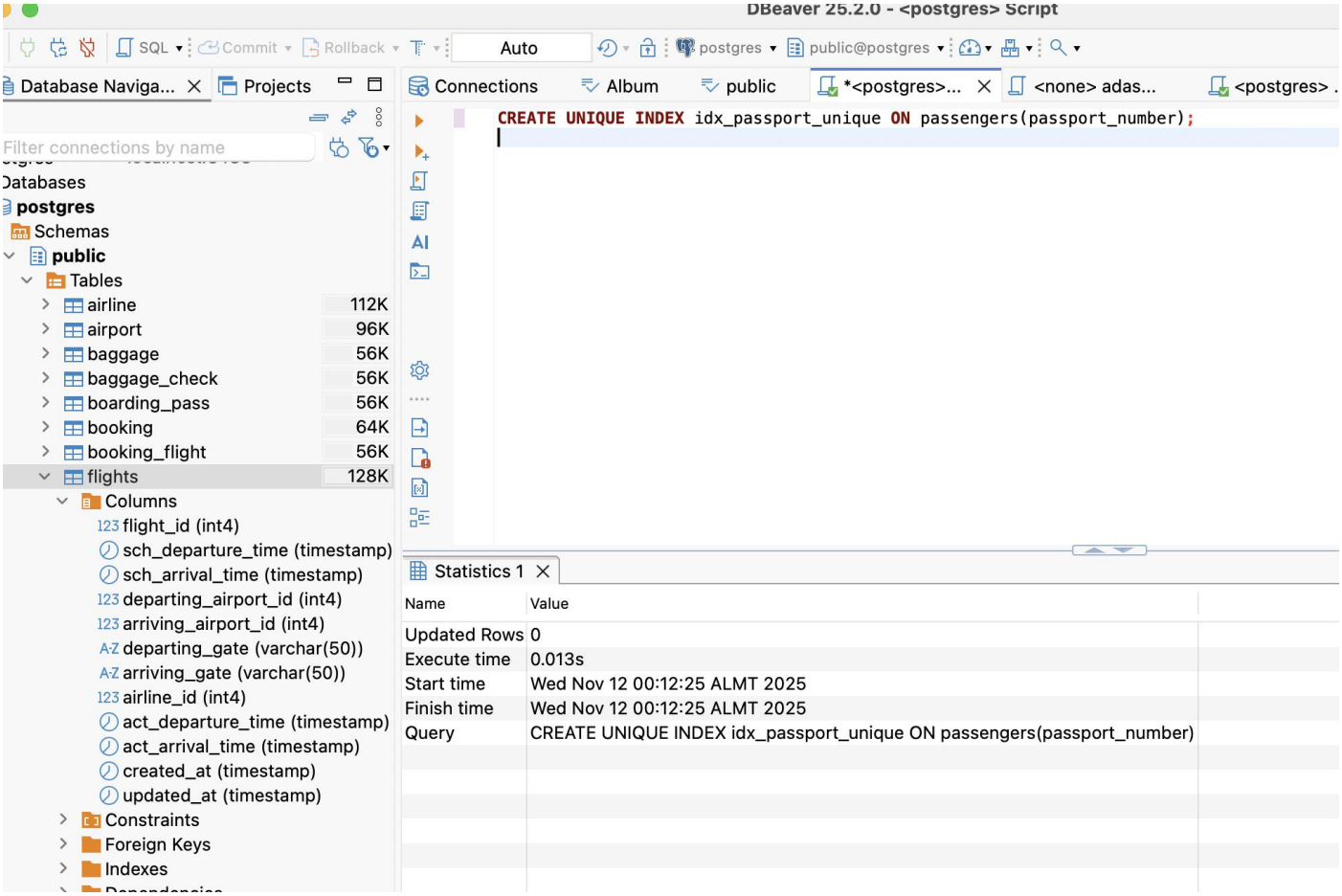


5. Use EXPLAIN ANALYZE to check index usage in a query filtering by departure_airport and arrival_airport.

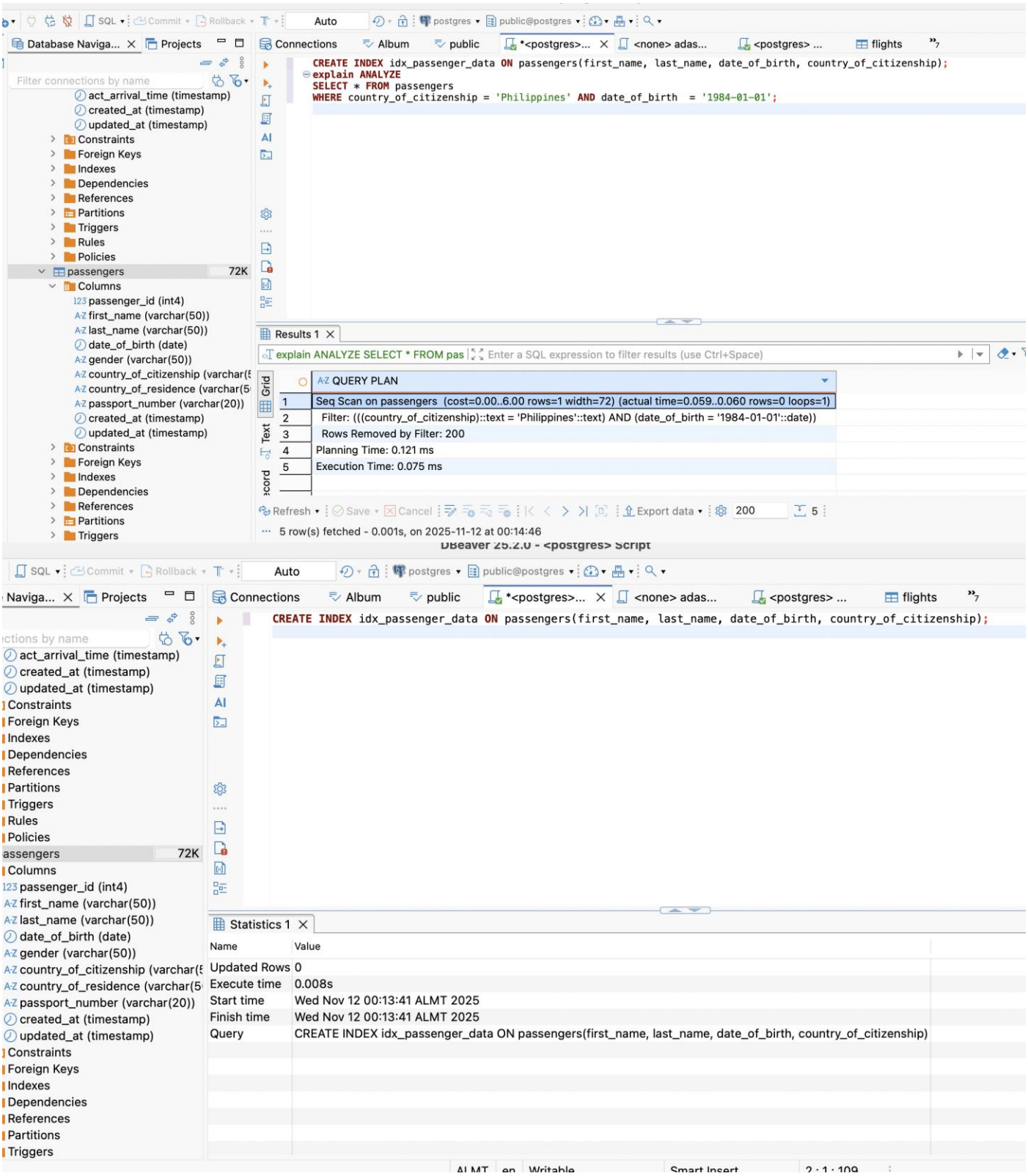


6. Create a unique index for the passport_number of the Passengers table. Check if the index was created or not. Insert into the table two new passengers.

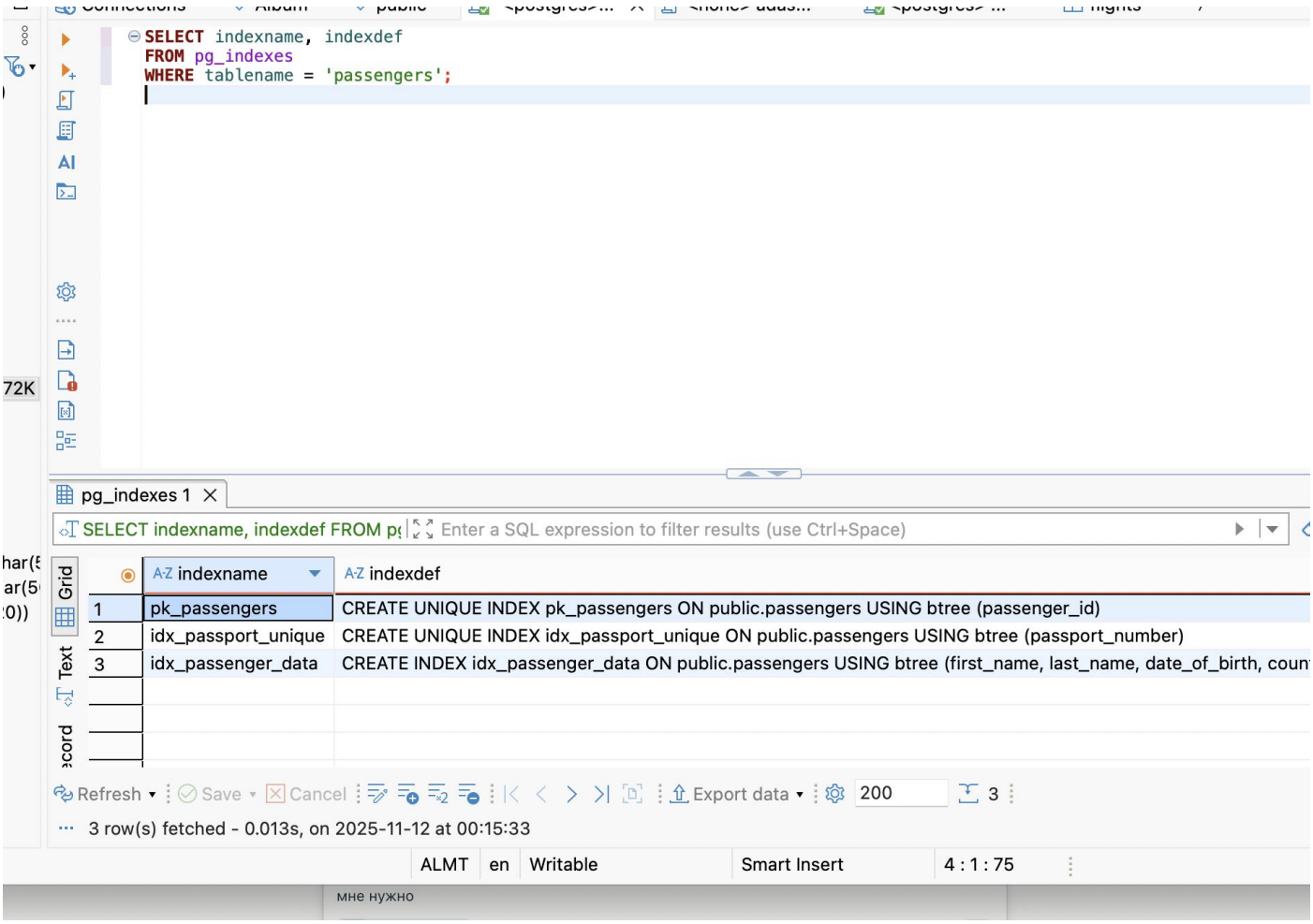
Explain in your own words what is going on in the output?



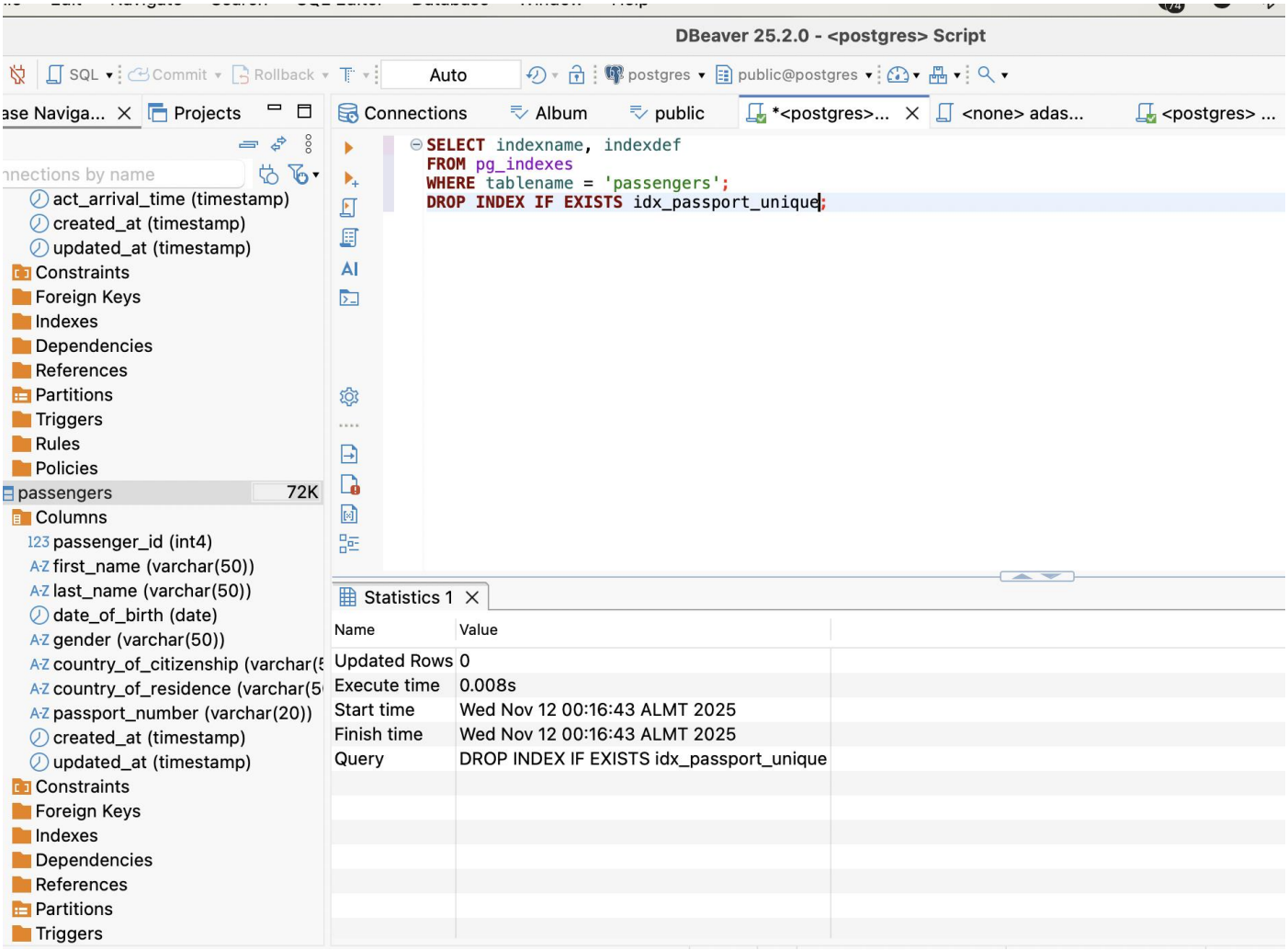
7. Create an index for the Passengers table. Use for that first name, last name, date of birth and country of citizenship. Then, write a SQL query to find a passenger who was born in Philippines and was born in 1984 and check if the query uses indexes or not. Give the explanation of the results.



8. Write a SQL query to list indexes for table Passengers. After delete the created indexes.


SELECT indexname, indexdef
FROM pg_indexes
WHERE tablename = 'passengers';
pg_indexes 1
SELECT indexname, indexdef FROM pg
Enter a SQL expression to filter results (use Ctrl+Space)
indexname
indexdef
1 pk_passengers CREATE UNIQUE INDEX pk_passengers ON public.passengers USING btree (passenger_id)
2 idx_passport_unique CREATE UNIQUE INDEX idx_passport_unique ON public.passengers USING btree (passport_number)
3 idx_passenger_data CREATE INDEX idx_passenger_data ON public.passengers USING btree (first_name, last_name, date_of_birth, coun
Refresh
Save
Cancel
Export data
200
3 row(s) fetched - 0.013s, on 2025-11-12 at 00:15:33
ALMT
en
Writable
Smart Insert
4 : 1 : 75

мне нужно


DBeaver 25.2.0 - <postgres> Script
SELECT indexname, indexdef
FROM pg_indexes
WHERE tablename = 'passengers';
DROP INDEX IF EXISTS idx_passport_unique;
act_arrival_time (timestamp)
created_at (timestamp)
updated_at (timestamp)
Constraints
Foreign Keys
Indexes
Dependencies
References
Partitions
Triggers
Rules
Policies
passengers 72K
Columns
passenger_id (int4)
first_name (varchar(50))
last_name (varchar(50))
date_of_birth (date)
gender (varchar(50))
passport_number (varchar(20))
Statistics 1
Name Value
Updated Rows 0
Execute time 0.008s
Start time Wed Nov 12 00:16:43 ALMT 2025
Finish time Wed Nov 12 00:16:43 ALMT 2025
Query DROP INDEX IF EXISTS idx_passport_unique