Όνομα: Χαριτούδης Απόστολος-Ανδρέας

Α.Μ: Π17178

Μάθημα: Βάσεις Δεδομένων

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΑ (ΠΑΠΕΙ)

ΕΡΩΤΗΜΑ 1

Database: Airline Database

Η βάση δεδομένων περιέχει 8 πίνακες, οι οποίοι είναι: Bookings, Tickets, Tickets\_flights, Flights, Boarding\_passes, Seats, Aircrafts\_data και Airports\_data.

Το Bookings περιλαμβάνει τα: book\_ref, book\_date και total\_amount. Το book\_ref είναι τύπου character, μήκους 6 και είναι το primary key του πίνακα αυτού, καθώς χρησιμοποιείται για να κάνουμε μοναδικές συνδέσεις στα εισιτήρια και τους επιβάτες, καθώς και στην πτήση τους. Το book\_date είναι τύπου timestamp με timezone και οι ημερομηνίες που αναγράφονται σε αυτή την στήλη δεν είναι νωρίτερα από ένα μήνα από την πτήση τους. Τέλος το total\_amount είναι τύπου numeric (10,2) και χρησιμοποιείται για να δηλώσει το συνολικό ποσό που χρεώθηκε αυτός που έκανε την κράτηση. Να σημειωθεί πως όπως ζητήθηκε στην εργασία μια κράτηση μπορεί να περιλαμβάνει περισσότερα από ένα εισιτήρια και το total\_amount περιγράφει την συνολική τους αξία.

To Tickets περιλαμβάνει τα: ticket\_no, book\_ref, passenger\_id, passenger\_name και contact\_data. Το ticket\_no είναι το νούμερο του εισιτηρίου, τύπου character, μήκους 13 και λειτουργεί ως primary key, καθώς είναι unique (εδώ επειδή έβαλα καταλάθος διπλότυπες τιμές δεν μου επέτρεπε να το κάνω primary key). Εδώ το book\_ref αναφέρεται για να γίνει συσχέτιση με τα στοιχεία των επιβατών και το εισιτήριό τους. Λειτουργεί επίσης σαν ξένο κλειδί. Το passenger\_id είναι τύπου character varying, μήκους 20. Το passenger\_name είναι τύπου text και το contact\_data είναι jsonb.

Το Tickets\_flights περιλαμβάνει τα: ticket\_no, flight\_id, fare\_conditions και amount. Το ticket\_no που σε συνδυασμό με το flight\_id λειτουργεί ως primary key. Το flight\_id είναι τύπου integer και δεν επιτρέπεται να πάρει τιμή <= του 0. Επιπλέον, το flight\_id είναι και ξένο κλειδί. Το fare\_conditions είναι character varying, μήκους 10 και παίρνει τιμές Economy, First για first class και Business. Το amount περιγράφει την τιμή του εισιτηρίου. Σε αυτήν την περίπτωση εδώ περιλαμβάνεται μόνο η τιμή του ενός εισιτηρίου.

Το Flights περιλαμβάνει τα: flight\_id, flight\_no, scheduled\_departure, departure\_airport, arrival\_airport, status, aircraft\_code, actual\_departure, distance και actual\_arrival. Το flight\_id είναι το ίδιο με αυτό που αναφέρθηκε στο tickets\_flights. Το flight\_no είναι τύπου character, μήκους 6 και έχει δηλωθεί ως unique σε συνδυασμό με το scheduled\_departure, το οποίο είναι τύπου timestamp με timezone. Αυτός ο συνδυασμός χαρακτηρίστηκε unique, καθώς δεν γίνεται μια πτήση να φεύγει από διαφορετικά σημεία ταυτόχρονα. Το departure\_airport και arrival\_airport είναι τύπου character 3 και είναι η κωδική ονομασία των αεροδρομίων. Το status είναι character varying 20 και περιγράφει την κατάσταση της πτήσης, η οποία εξαρτάται από την function timesnap, η οποία έχει ένα time snapshot με ημερομηνία 30/06/2022 23:59 (αρχική ημερομηνία παράδοσης εργασίας), αλλά δεν χρησιμοποιείται κάπου, παρά σαν οδηγός για να καθοριστούν οι καταστάσεις των πτήσεων: On time ή delayed για πτήσεις που είναι να γίνουν την επόμενη μέρα, departed για πτήσεις που έχουν φύγει πριν από αυτήν την ημερομηνία, αλλά δεν έχουν φτάσει (δηλαδή η ημερομηνία άφιξης είναι μεταγενέστερη αυτής), arrived για όσες πτήσεις έχουν φτάσει στον προορισμό τους πριν από αυτήν την ημερομηνία, scheduled για όσες πτήσεις είναι να γίνουν εντός 1 μήνα με βάση αυτή την ημερομηνία και cancelled, η κατάσταση στην οποία η πτήση ακυρώνεται. Όμως, αυτή η κατάσταση δεν έχει κάποιο χρονικό περιθώριο καθώς μπορεί να γίνει για πτήσεις που είναι να πραγματοποιηθούν και για πτήσεις που ήταν να πραγματοποιηθούν, αλλά δεν πραγματοποιήθηκαν. Το aircraft\_code είναι ο 3ψήφιος κωδικός των αεροπλάνων και είναι μοναδικός και είναι τύπου character, μήκους 3 και είναι ξένο κλειδί. Το , actual\_departure και actual\_arrival είναι τύπου timestamp και περιγράφουν την ημερομηνία που τελικά πραγματοποιήθηκε κάποια πτήση. Τέλος το distance είναι τύπου integer και μας δίνει πληροφορίες για την απόσταση των 2 αεροδρομίων (αεροδρόμιο αναχώρησης, αεροδρόμιο άφιξης) σε km.

To Boarding\_passes περιλαμβάνει τα: ticket\_no, flight\_id, boarding\_no και seat\_no. Το ticket\_no είναι όπως αναφέρθηκε πριν ο αριθμός εισιτηρίου που είναι unique. Seat\_no είναι τυπου char var μηκους 4 και δηλώνει σε ποια σειρα και νουμερο καθεται ο επιβατης. Το boarding\_no είναι τυπου int και δειχνει προτεραιοτητα επιβιβασης.

Το Seats περιλαμβάνει τα: aircraft\_code, seat\_no και fare\_conditions. Το seats δείχνει ολες τις υπαρκτες θεσεις κάθε αεροπλανου που υπαρχεις την βαση, καθως και το κοστος και κατηγορια κάθε θεσης.

Το Aircrafts\_data περιλαμβάνει τα: aircraft\_code, model και range. Περιλαμβανει τον κωδικο του αεροπλανου, το μοντελο και την χιλιομετρικη αποσταση που μπορει να διανυσει.

Το Airports\_data περιλαμβάνει τα: airport\_code, airport\_name, city, coordinates και timezone. Περιλαμβανει τους κωδικους των αεροδρομιων, την γεωμετρικη τους τοποθεσια, το timezone στο οποιο ανηκουν και την πολη στην οποια ανηκουν.

BCNF

Όλοι οι πινακες είναι σε μορφη BCNF καθως κανενας πινακας και πιο σωστα κανενα στοιχειο του πινακα δεν εξαρταται από μη κλειδι attribute και δεν εχουμ διπλοτυπες στήλες.

Flights\_View και Routes\_View

Flights\_View

Το Flights\_View πραγματοποιειται ετσι:

CREATE OR REPLACE VIEW public.routes\_view

AS

WITH f3 AS (

SELECT f2.flight\_no,

f2.departure\_airport,

f2.arrival\_airport,

f2.aircraft\_code,

f2.duration,

array\_agg(f2.days\_of\_week) AS days\_of\_week

FROM ( SELECT f1.flight\_no,

f1.departure\_airport,

f1.arrival\_airport,

f1.aircraft\_code,

f1.duration,

f1.days\_of\_week

FROM ( SELECT flights.flight\_no,

flights.departure\_airport,

flights.arrival\_airport,

flights.aircraft\_code,

flights.scheduled\_arrival - flights.scheduled\_departure AS duration,

to\_char(flights.scheduled\_departure, 'ID'::text)::integer AS days\_of\_week

FROM flights) f1

GROUP BY f1.flight\_no, f1.departure\_airport, f1.arrival\_airport, f1.aircraft\_code, f1.duration, f1.days\_of\_week

ORDER BY f1.flight\_no, f1.departure\_airport, f1.arrival\_airport, f1.aircraft\_code, f1.duration, f1.days\_of\_week) f2

GROUP BY f2.flight\_no, f2.departure\_airport, f2.arrival\_airport, f2.aircraft\_code, f2.duration

)

SELECT f3.flight\_no,

f3.departure\_airport,

dep.airport\_name AS departure\_airport\_name,

dep.city AS departure\_city,

f3.arrival\_airport,

arr.airport\_name AS arrival\_airport\_name,

arr.city AS arrival\_city,

f3.aircraft\_code,

f3.duration,

f3.days\_of\_week

FROM f3,

airports dep,

airports arr

WHERE f3.departure\_airport = dep.airport\_code AND f3.arrival\_airport = arr.airport\_code;

ALTER TABLE public.routes\_view

OWNER TO postgres;

Routes\_View

Το Routes\_View

πραγματοποιειται ετσι:

CREATE OR REPLACE VIEW public.flights\_view

AS

SELECT f.flight\_id,

f.flight\_no,

f.scheduled\_departure,

timezone(dep.timezone, f.scheduled\_departure) AS scheduled\_departure\_local,

f.scheduled\_arrival,

timezone(arr.timezone, f.scheduled\_arrival) AS scheduled\_arrival\_local,

f.scheduled\_arrival - f.scheduled\_departure AS scheduled\_duration,

f.departure\_airport,

dep.airport\_name AS departure\_airport\_name,

dep.city AS departure\_city,

f.arrival\_airport,

arr.airport\_name AS arrival\_airport\_name,

arr.city AS arrival\_city,

f.status,

f.aircraft\_code,

f.actual\_departure,

timezone(dep.timezone, f.actual\_departure) AS actual\_departure\_local,

f.actual\_arrival,

timezone(arr.timezone, f.actual\_arrival) AS actual\_arrival\_local,

f.actual\_arrival - f.actual\_departure AS actual\_duration

FROM flights f,

airports dep,

airports arr

WHERE f.departure\_airport = dep.airport\_code AND f.arrival\_airport = arr.airport\_code;

ALTER TABLE public.flights\_view

OWNER TO postgres;

ΕΡΩΤΗΜΑ 2

Queries

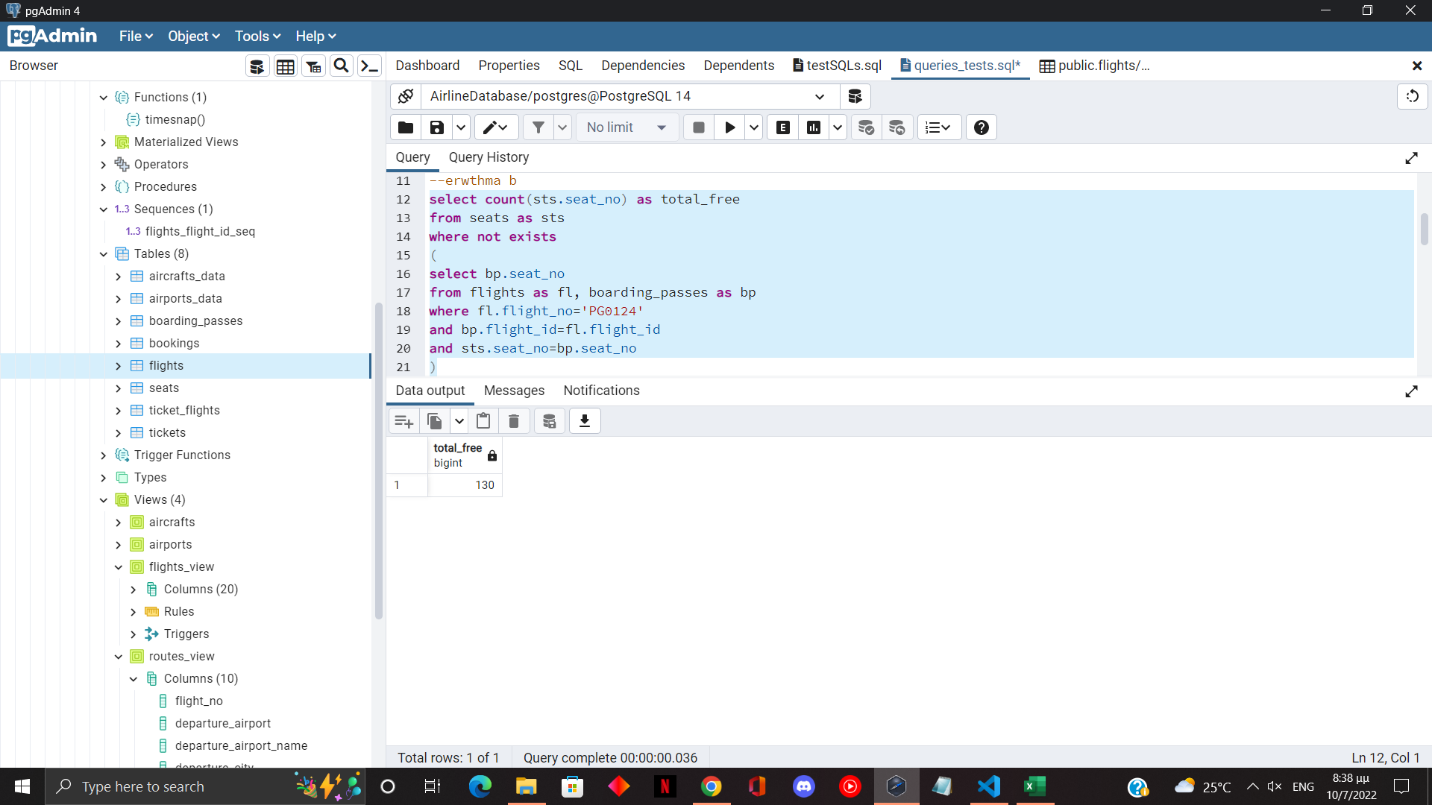
Όλα τα queries του ερωτήματος 2 βρίσκονται στο .sql αρχείο.

a. Για να επιτευχθεί το ζητούμενο της ερώτησης αυτής έγραψα το εξής query (δίνεται παρακάτω) αποτελέσματα

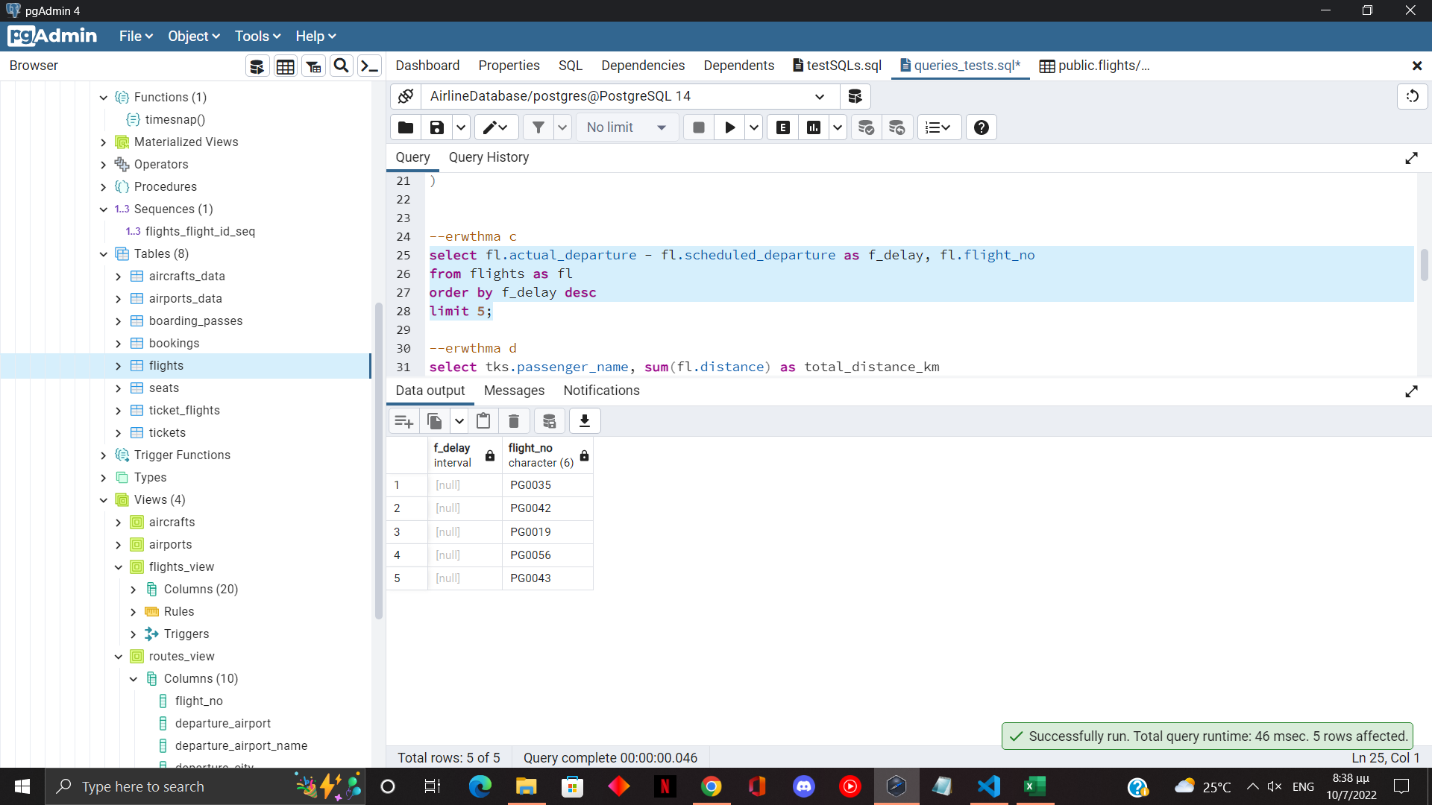
A screenshot of a computer

Description automatically generated

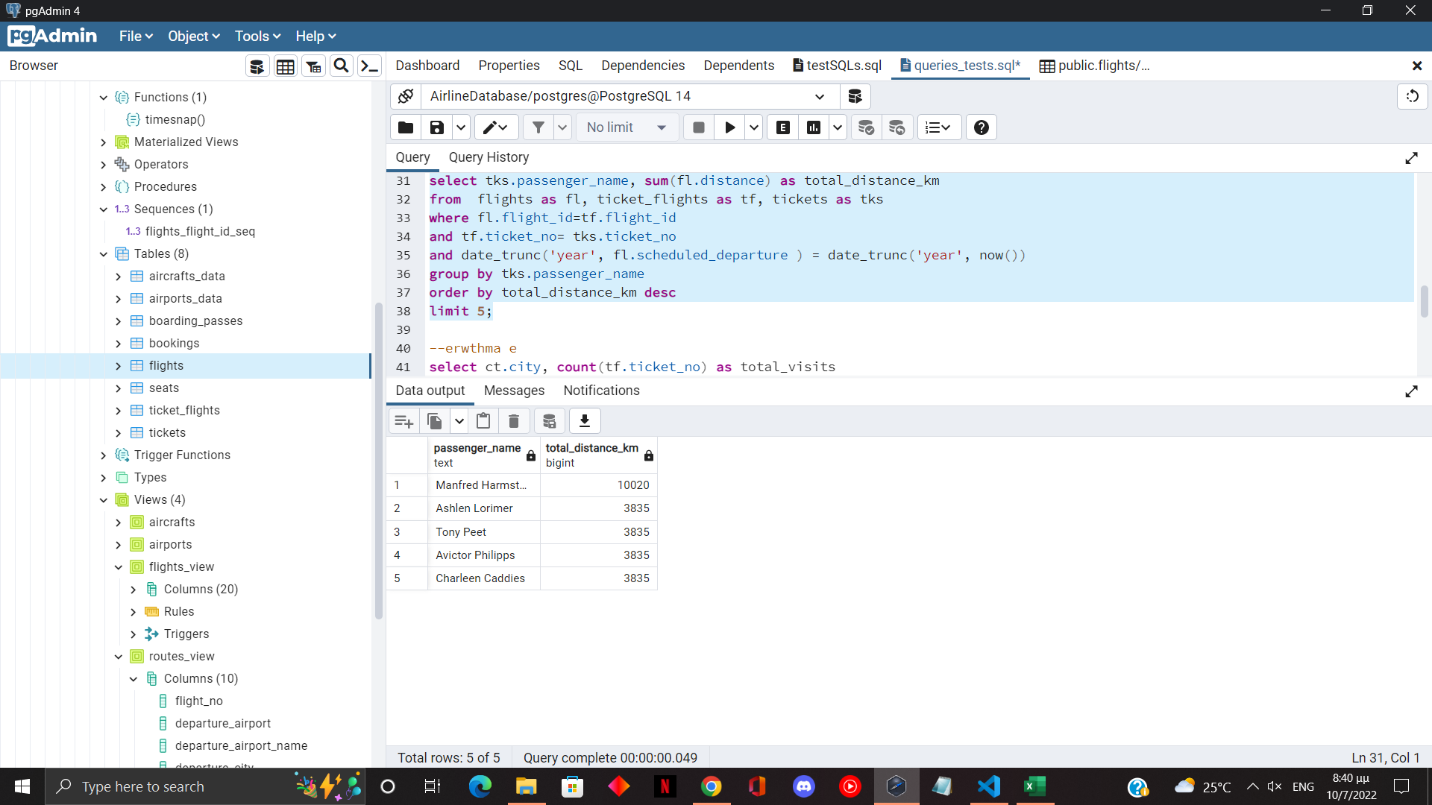
b. Για να επιτευχθεί το ζητούμενο της ερώτησης αυτής έγραψα το εξής query (δίνεται παρακάτω) αποτελέσματα. Επειδη στο πρωτο ερωτημα δεν εμφανιζεται καποια πτηση εβαλα μια τυχαια για να εμφανιστει



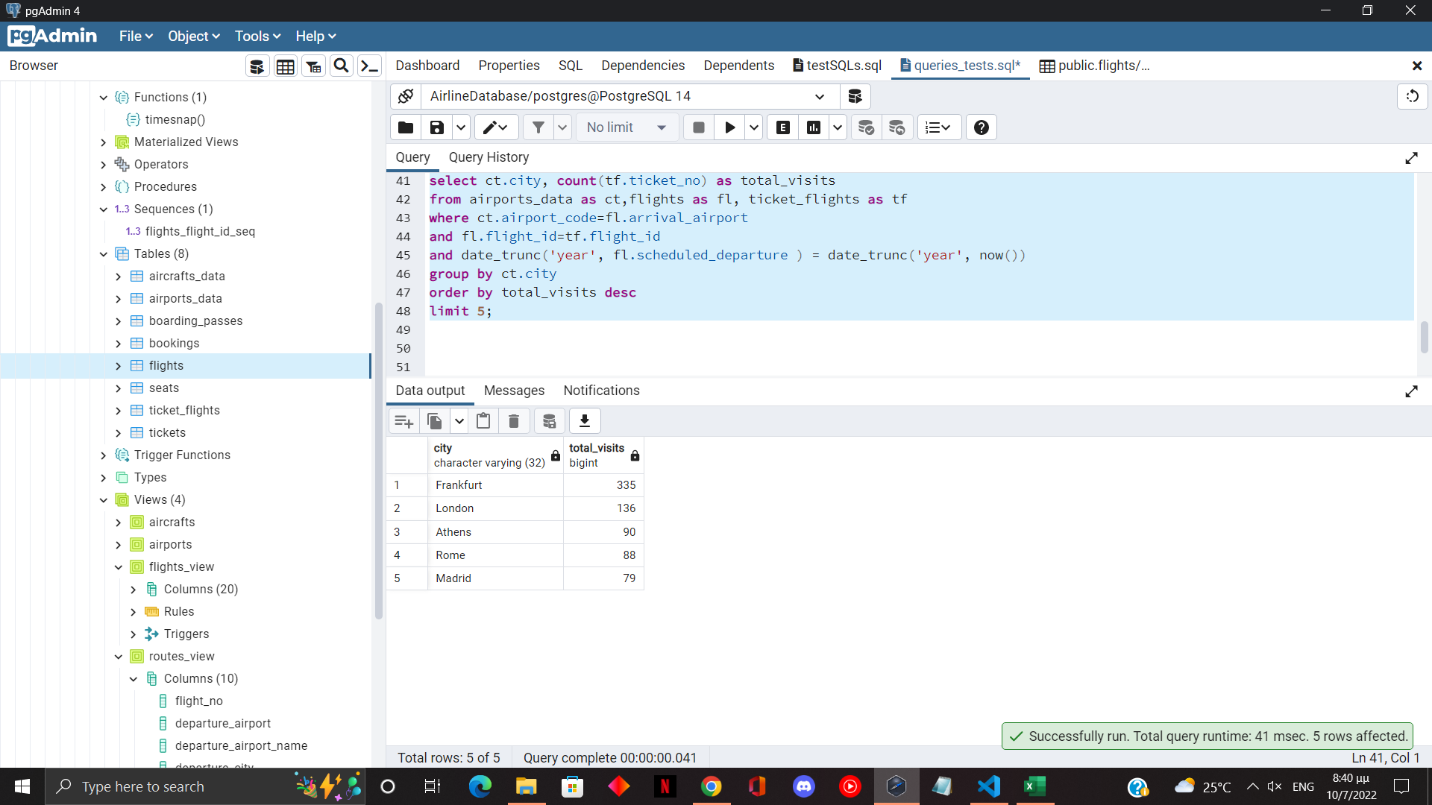
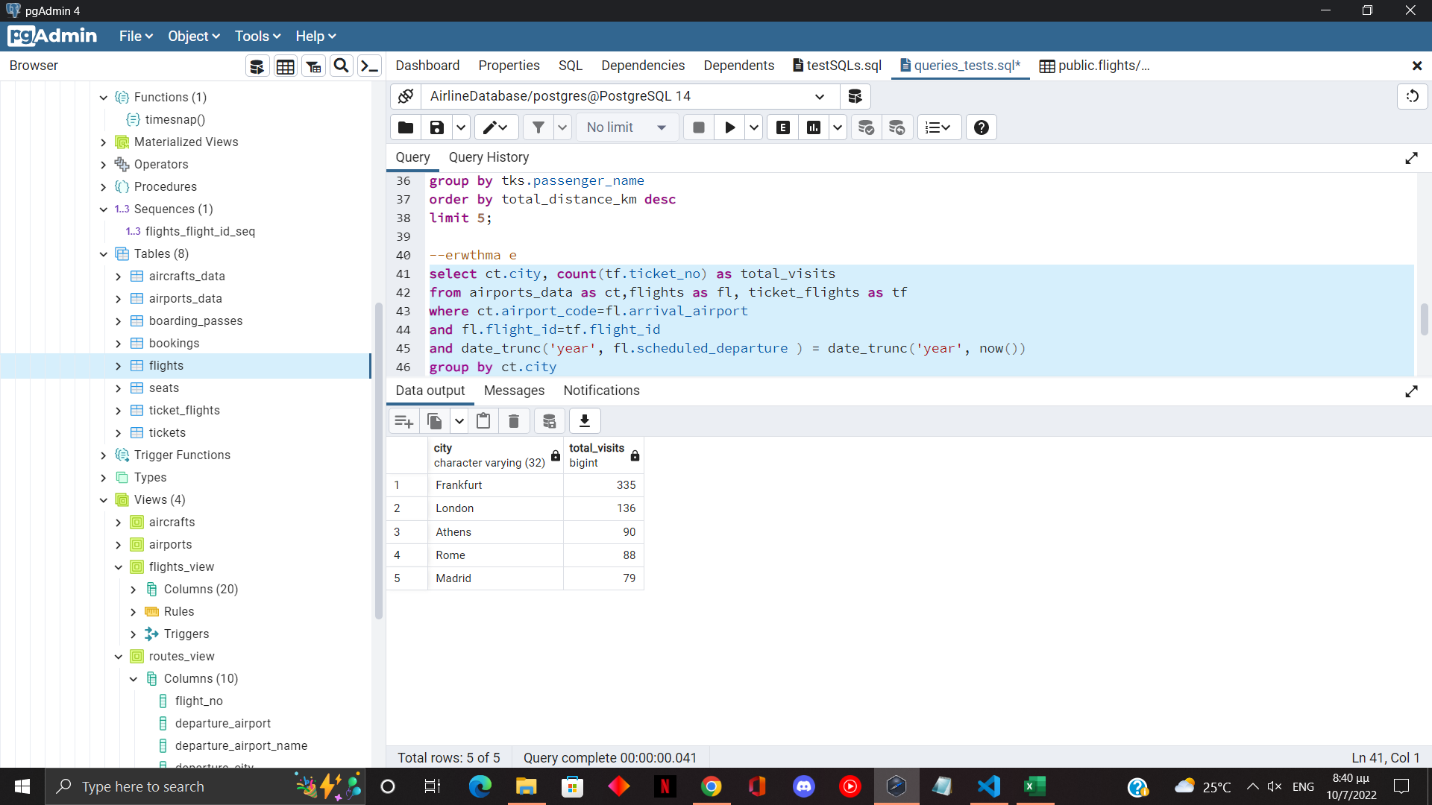
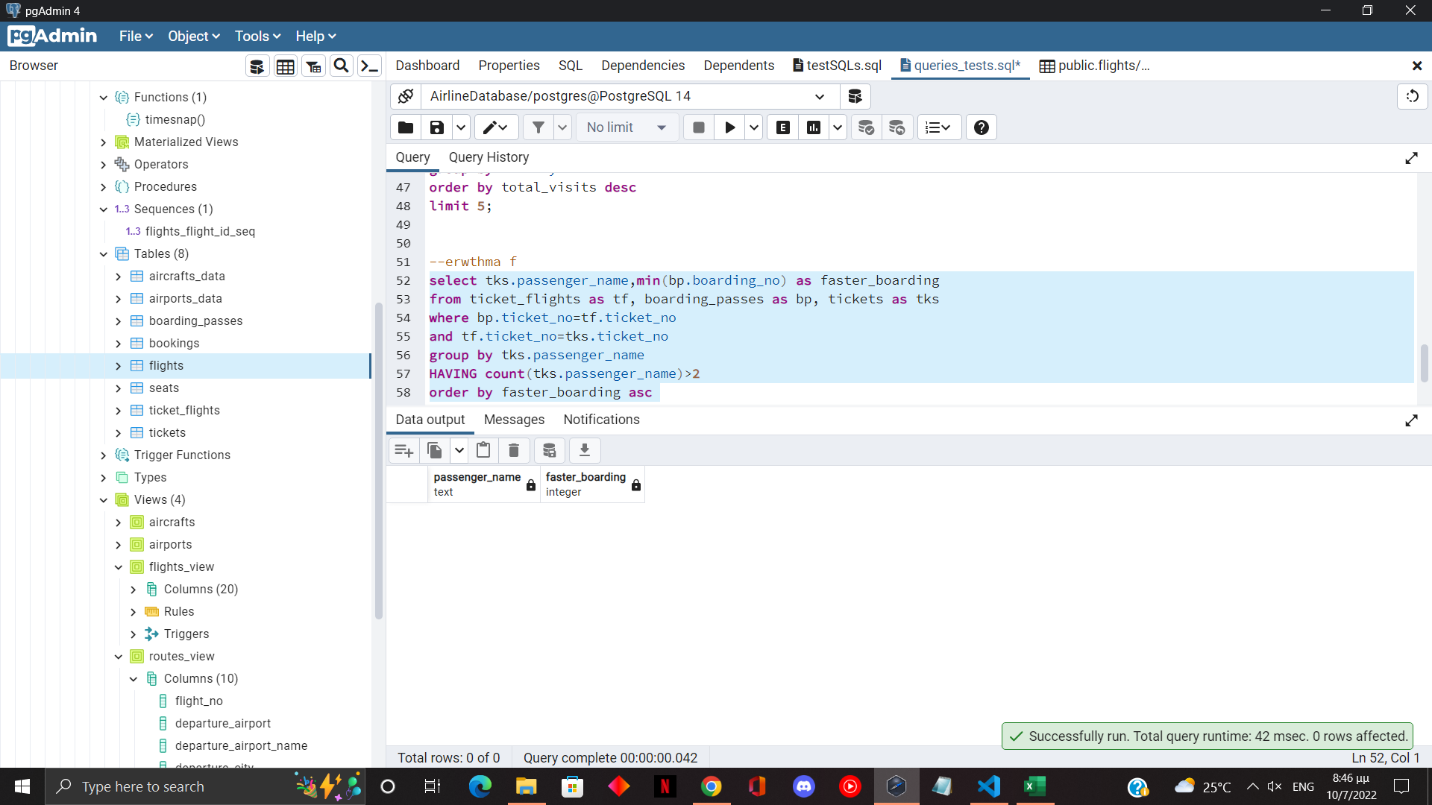
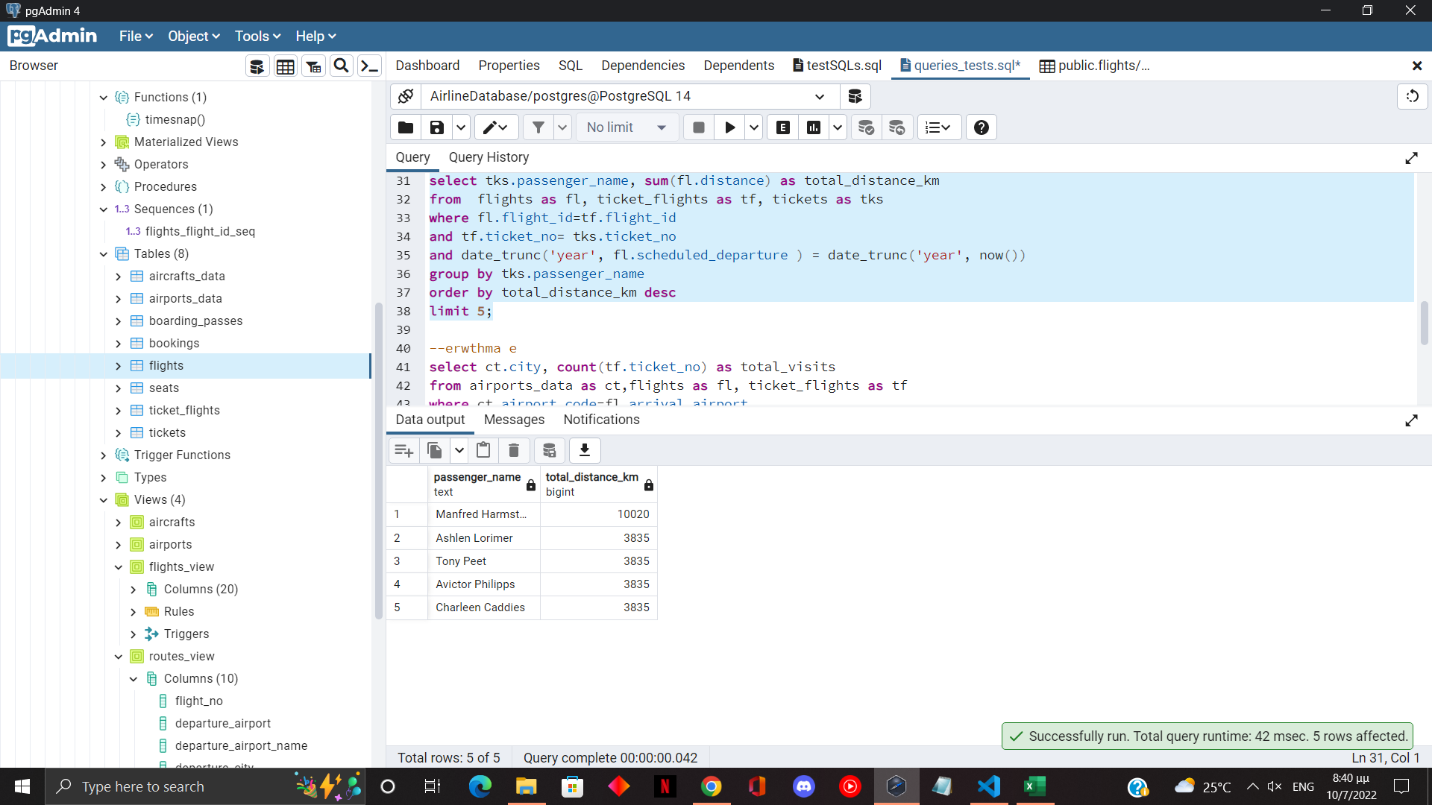
c. Για να επιτευχθεί το ζητούμενο της ερώτησης αυτής έγραψα το εξής query (δίνεται παρακάτω) αποτελέσματα



d. Για να επιτευχθεί το ζητούμενο της ερώτησης αυτής έγραψα το εξής query (δίνεται παρακάτω)



e. Για να επιτευχθεί το ζητούμενο της ερώτησης αυτής έγραψα το εξής query (δίνεται παρακάτω)



f. Για να επιτευχθεί το ζητούμενο της ερώτησης αυτής έγραψα το εξής query (δίνεται παρακάτω)