

**Πανεπιστήμιο Πειραιώς
Τμήμα Πληροφορικής
Έτος: 2021 - 2022**



**Μάθημα:
«ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ»
Εργασία: Γενική Εργασία Μαθήματος
Εξάμηνο: 4ο**

Ομάδα εργασίας:

Αιμιλιανός Κουρπάς Δανάς Π20100,
Απόστολος Σιαμπάνης Π20173,
Θεόδωρος Κοζάνογλου Π20094

Ερώτημα 1. Σχεδιακή Βάση Δεδομένων

Εκφώνηση:

a. Με βάση τα παραπάνω στοιχεία, σχεδιάστε το σχεσιακό σχήμα της ΒΔ, να το υλοποιήσετε (εντολές CREATE TABLE) στο ΣΔΒΔ PostgreSQL και φορτώστε με δεδομένα τους πίνακες. Επιπλέον, καλείστε να τεκμηριώσετε τους περιορισμούς ακεραιότητας των πινάκων (και να δηλώσετε τυχόν περιορισμούς που προκύπτουν από την εκφώνηση αλλά δεν μπορέσατε να υποστηρίξετε μέσα από τους περιορισμούς ακεραιότητας των πινάκων). Το παραδοτέο του υποερωτήματος είναι το σχεσιακό σχήμα της ΒΔ, οι εντολές CREATE TABLE και τα αρχεία τα οποία θα εισάγετε στους πίνακες. Οδηγία: για την ευκολότερη παραγωγή αληθοφανών δεδομένων προτείνεται να χρησιμοποιήσετε κάποιο εργαλείο παραγωγής δεδομένων (data generator) (π.χ. www.mockaroo.com, <https://faker.readthedocs.io/en/master/>, <https://devskiller.github.io/jfairy/>).

b. Εφαρμόστε τη θεωρία της κανονικοποίησης πάνω στο σχεσιακό σχήμα της ΒΔ που σχεδιάσατε και ελέγξτε τον κάθε πίνακα εάν ακολουθεί την BCNF. Σε αντίθετη περίπτωση, αποσυνθέστε τους προβληματικούς πίνακες ώστε όλη η ΒΔ να είναι σε BCNF.

c. Πάνω στο τελικό σχήμα της ΒΔ υλοποιήστε 2 προβολές/όψεις (views):

- **Flights_View**. Μια προβολή που θα αφορά μια συγκεκριμένη ημερομηνία (π.χ. 30/5/2022) και θα περιλαμβάνει τις «δυναμικές» πληροφορίες των πτήσεων εκείνης της ημέρας: αριθμός πτήσης, αεροδρόμιο (κωδικός – όνομα) και πόλη αναχώρησης, αεροδρόμιο (κωδικός – όνομα) και πόλη άφιξης, τοπική ημερομηνία και ώρα αναχώρησης προγραμματισμένη & πραγματική, τοπική ημερομηνία και ώρα άφιξης (προγραμματισμένη & πραγματική), διάρκεια πτήσης (προγραμματισμένη & πραγματική).
- **Routes_View**. Μια προβολή που θα αφορά μια συγκεκριμένη εβδομάδα (π.χ. Δε. 30/5/2022 – Κυ. 5/6/2022) και θα περιλαμβάνει τις «στατικές» πληροφορίες των πτήσεων εκείνης της εβδομάδας: αριθμός πτήσης, αεροδρόμιο (κωδικός – όνομα) και πόλη αναχώρησης, αεροδρόμιο (κωδικός – όνομα) και πόλη άφιξης, μοντέλο αεροσκάφους, προγραμματισμένη διάρκεια πτήσης και ένα αλφαριθμητικό «days_of_week» το οποίο εμφανίζει τις ημέρες της εβδομάδας κατά τις οποίες εκτελείται η συγκεκριμένη πτήση (π.χ. το αλφαριθμητικό "Mo -- We -- -- Sa --" αντιστοιχεί σε μια πτήση που εκτελείται Δευτέρα, Τετάρτη και Σάββατο).

Απάντηση:

A.

Ανάλυση της αρχικής υλοποίησης της Βάσης Δεδομένων.

Βασιζόμενοι στην περιγραφή της βάσης από την εκφώνηση, ακολουθεί μια λεπτομερέστερη ανάλυση των σχέσεων:

- Για την κράτηση δημιουργήθηκε η σχέση **“booking”**, ο οποίος περιέχει τα ακόλουθα χαρακτηριστικά:
 - **book_ref**: Αριθμός Κράτησης
 - **book_date**: Ημερομηνία Κράτησης
 - **total_amount**: Συνολική Αξία Εισιτηρίων της Κράτησης
 - **departure_date**: Ημερομηνία Αναχώρησης
 - **flight_code**: Κωδικός Πτήσης

Το **Primary Key** της σχέσης **“booking”** είναι το χαρακτηριστικό **book_ref**, ενώ τα **Foreign Keys** της σχέσης είναι τα χαρακτηριστικά **departure_date** και **flight_code** όπου αναφέρονται στη σχέση **“flight”** (η οποία θα περιγραφεί παρακάτω).

- Για το εισιτήριο δημιουργήθηκε η σχέση **“ticket”**, η οποία περιέχει τα ακόλουθα χαρακτηριστικά:
 - **ticket_no**: Αριθμός Εισιτηρίου
 - **passenger_id**: Αριθμός Εγγράφου Ταυτότητας επιβάτη
 - **passenger_lastname**: Επώνυμο επιβάτη
 - **passenger_firstname**: Όνομα επιβάτη
 - **passenger_phone**: Αριθμός Τηλεφώνου επιβάτη
 - **passenger_email**: E-mail επιβάτη
 - **book_ref**: Αριθμός Κράτησης
 - **flight_code**: Αριθμός Πτήσης

- **departure_date:** Ημερομηνία Αναχώρησης
- **amount:** Τιμή Εισιτηρίου
- **flight_class:** Κατηγορία Ταξιδιού

Το **Primary Key** της σχέσης “**ticket**” είναι ο συνδυασμός χαρακτηριστικών (**ticket_no**, **flight_code** και **departure_date**), ενώ τα **Foreign Key** της σχέσης είναι το χαρακτηριστικό **book_ref** όπου αναφέρεται στη σχέση “**booking**” και τα χαρακτηριστικά **flight_code** και **departure_date** τα οποία αναφέρονται στη σχέση “**flight**” (ο οποίος θα περιγραφεί παρακάτω).

- Για τη πτήση δημιουργήθηκε ο πίνακας “**flight**”, η οποία περιέχει τα ακόλουθα χαρακτηριστικά:
 - **flight_code:** Αριθμός Πτήσης
 - **departure_date:** Ημερομηνία Αναχώρησης
 - **departure_airport:** Αεροδρόμιο Αναχώρησης
 - **arrival_airport:** Αεροδρόμιο Άφιξης
 - **distance:** Χιλιομετρική Απόσταση
 - **scheduled_departure_time:** Προγραμματισμένη Ώρα Αναχώρησης
 - **scheduled_arrival_time:** Προγραμματισμένη Ώρα Άφιξης
 - **scheduled_duration:** Προγραμματισμένη Διάρκεια Πτήσης
 - **actual_departure_time:** Πραγματική Ώρα Αναχώρησης
 - **actual_arrival_time:** Πραγματική Ώρα Άφιξης
 - **flight_status:** Κατάσταση Πτήσης
 - **aircraft_code:** Μοντέλο Αεροσκάφους

Το **Primary Key** της σχέσης “**flight**” είναι ο συνδυασμός των χαρακτηριστικών **flight_code** και **departure_date**, ενώ τα **Foreign Keys** του πίνακα είναι τα χαρακτηριστικά **departure_airport** και **arrival_airport** όπου αναφέρονται το κάθε ένα ξεχωριστά στη σχέση “**airport**” (ο οποίος θα περιγραφεί παρακάτω) και το χαρακτηριστικό **aircraft_code** όπου αναφέρεται στη σχέση “**aircraft**” (ο οποίος θα περιγραφεί παρακάτω).

- Για το αεροδρόμιο δημιουργήθηκε η σχέση “**airport**”, η οποία περιέχει τα ακόλουθα χαρακτηριστικά:
 - **airport_code:** Κωδικός αεροδρομίου
 - **airport_name:** Ονομασία αεροδρομίου
 - **airport_city:** Πόλη αεροδρομίου
 - **airport_timezone:** Ζώνης ώρας που ανήκει το αεροδρόμιο

Το **Primary Key** του πίνακα “**airport**” είναι το χαρακτηριστικό **airport_code**.

- Για το αεροσκάφος δημιουργήθηκε η σχέση “**aircraft**”, το οποίο περιέχει τα ακόλουθα χαρακτηριστικά:
 - **aircraft_code:** Μοντέλο Αεροσκάφους
 - **aircraft_name:** Ονομασία Αεροσκάφους
 - **aircraft_capacity:** Χωρητικότητα Αεροσκάφους
 - **aircraft_range:** Μέγιστη Χιλιομετρική Απόσταση που μπορεί να διανύσει το αεροσκάφος

Το **Primary Key** της σχέσης “**aircraft**” είναι το χαρακτηριστικό **aircraft_code**.

- Για την κάρτα επιβίβαση δημιουργήθηκε η σχέση “**boarding_pass**”, η οποία περιέχει τα ακόλουθα χαρακτηριστικά:
 - **ticket_no:** Αριθμός Εισιτηρίου
 - **flight_code:** Κωδικός Πτήσης
 - **departure_date:** Ημερομηνία Αναχώρησης
 - **seat_no:** Θέση Αεροπλάνου
 - **boarding_no:** Θέση Επιβίβασης

Το **Primary Key** της σχέσης “**boarding_pass**” είναι ο συνδυασμός των χαρακτηριστικών **seat_no**, **flight_code** και **departure_date**.

B.

Για να είναι ένας τύπος σχέσης **R** σε **Boyce-Codd** Κανονική Μορφή, θα πρέπει όλες οι συναρτησιακές εξαρτήσεις **a→b** που ισχύουν στον **R**, το **a** να αποτελεί υποψήφιο κλειδί του **R**.

Σε αυτό το μέρος οι σχέσεις θα αναφέρονται ως πίνακες και τα χαρακτηριστικά ως πεδία.

Ακολουθούν οι έλεγχοι **BCNF** για τους πίνακες που δημιουργήθηκαν (με την σειρά που αναγράφονται στο πρώτο μέρος):

1. Για την σχέση “**booking**”, οι συναρτησιακές εξαρτήσεις είναι οι ακόλουθες:

FD1: **book_ref** → flight_code, departure_date, book_date & total_amount

Υποψήφιο κλειδί είναι το **book_ref**.

Είναι **1NF**, γιατί τα πεδία τιμών όλων των χαρακτηριστικών του πίνακα “**booking**” είναι ατομικά.

Είναι **2NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υποψήφιο κλειδί **book_ref** δεν προσδιορίζεται από τμήμα ενός υποψήφιου κλειδιού.

Είναι **3NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υποψήφιο κλειδί **book_ref** δεν προσδιορίζεται μεταβατικά (δηλαδή μέσω κάποιου άλλου χαρακτηριστικού το οποίο δεν συμμετέχει σε υποψήφιο κλειδί) από ένα υποψήφιο κλειδί.

Είναι **BCNF**, γιατί το **book_ref** της συναρτησιακής εξάρτησης **FD1** αποτελεί υποψήφιο κλειδί και κατ’ επέκταση **πρωτεύον κλειδί** του πίνακα “**booking**”.

2. Για τον πίνακα “**ticket**”, οι συναρτησιακές εξαρτήσεις είναι οι ακόλουθες:

FD1: **ticket_no** → book_ref, passenger_id, flight_code, departure_date, amount, flight_class, passenger_firstname, passenger_lastname, passenger_phone, passenger_email

FD2: **passenger_id** → passenger_firstname, passenger_lastname, passenger_phone, passenger_email

FD3: **ticket_no, passenger_id** → book_ref, passenger_id, passenger_firstname, passenger_lastname, passenger_phone, passenger_email, flight_code, departure_date, amount, flight_class

Υποψήφια κλειδιά είναι το **ticket_no** και ο συνδυασμός των χαρακτηριστικών (**ticket_no, passenger_id**)

Είναι **1NF**, γιατί τα πεδία τιμών όλων των χαρακτηριστικών του πίνακα “**ticket**” είναι ατομικά.

Δεν είναι **2NF**, γιατί υπάρχει χαρακτηριστικό (**passenger_firstname**) που προσδιορίζεται από μέρος ενός υποψήφιου κλειδιού, που σε αυτή την περίπτωση είναι το **passenger_id**.

Άρα, κατ’ επέκταση δεν είναι **3NF** και **BCNF**.

Καθώς, στον ίδιο αριθμό εισιτηρίου περιλαμβάνονται περισσότερα από ένα τμήματα πτήσεων και για την αποφυγή της επαναληπτικότητας πληροφορίας, θα χρειαστεί να γίνει διάσπαση στον πίνακα “**ticket**” σε τρεις πίνακες: “**ticket**”, “**more_flights**” και “**passenger**”.

Οι νέοι πίνακες θα είναι:

- Ο πίνακας “**ticket**” περιέχει τα ακόλουθα πεδία:
 - **ticket_no**: Αριθμός Εισιτηρίου
 - **passenger_id**: Αριθμός Εγγράφου Ταυτότητας επιβάτη
 - **book_ref**: Αριθμός Κράτησης

Για τον νέο πίνακα “**ticket**”, οι συναρτησιακές εξαρτήσεις είναι οι ακόλουθες:

FD1: **ticket_no** → passenger_id, book_ref

Υποψήφιο κλειδί είναι το **ticket_no**.

Είναι **1NF**, γιατί τα πεδία τιμών όλων των χαρακτηριστικών του πίνακα “**ticket**” είναι ατομικά.

Είναι **2NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υπονήφιο κλειδί **ticket_no** δεν προσδιορίζεται από τμήμα ενός υπονήφιου κλειδιού.

Είναι **3NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υπονήφιο κλειδί **ticket_no** δεν προσδιορίζεται μεταβατικά (δηλαδή μέσω κάποιου άλλου χαρακτηριστικού το οποίο δεν συμμετέχει σε υπονήφιο κλειδί) από ένα υπονήφιο κλειδί.

Είναι **BCNF**, γιατί το **ticket_no** της συναρτησιακής εξάρτησης **FD1** αποτελεί μοναδικό υπονήφιο κλειδί και κατ' επέκταση **πρωτεύον κλειδί** του πίνακα "**ticket**".

- Ο πίνακας "**more_flights**" περιέχει τα ακόλουθα πεδία:
 - **ticket_no**: Αριθμός Εισιτηρίου
 - **flight_code**: Αριθμός Πτήσης
 - **departure_date**: Ημερομηνία Αναχώρησης
 - **amount**: Τιμή Εισιτηρίου
 - **fare**: Κατηγορία Ταξιδιού

Για τον νέο πίνακα "**more_flights**", οι συναρτησιακές εξαρτήσεις είναι οι ακόλουθες:

FD1: **ticket_no**, **flight_code**, **departure_date** → amount, fare

Υπονήφιο κλειδί είναι ο συνδυασμός των πεδίων **ticket_no**, **departure_date** και **flight_code**.

Είναι **1NF**, γιατί τα πεδία τιμών όλων των χαρακτηριστικών του πίνακα "**more_flights**" είναι ατομικά.

Είναι **2NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υπονήφιο κλειδί **ticket_no**, **departure_date**, **flight_code** δεν προσδιορίζεται από τμήμα ενός υπονήφιου κλειδιού.

Είναι **3NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υπονήφιο κλειδί **ticket_no**, **departure_date**, **flight_code** δεν προσδιορίζεται μεταβατικά (δηλαδή μέσω κάποιου άλλου χαρακτηριστικού το οποίο δεν συμμετέχει σε υπονήφιο κλειδί) από ένα υπονήφιο κλειδί.

Είναι **BCNF**, γιατί το **ticket_no**, **departure_date**, **flight_code** της συναρτησιακής εξάρτησης **FD1** αποτελεί μοναδικό υπονήφιο κλειδί και κατ' επέκταση **πρωτεύον κλειδί** του πίνακα "**more_flights**".

- Ο πίνακας "**passenger**" περιέχει τα ακόλουθα πεδία:
 - **passenger_id**: Αριθμός Εγγράφου Ταυτότητας επιβάτη
 - **passenger_firstname**: Όνομα επιβάτη
 - **passenger_lastname**: Επώνυμο επιβάτη
 - **passenger_phone**: Αριθμός Τηλεφώνου επιβάτη
 - **passenger_email**: E-mail επιβάτη

Για τον νέο πίνακα "**passenger**", οι συναρτησιακές εξαρτήσεις είναι οι ακόλουθες:

FD1: **passenger_id** → passenger_firstname, passenger_lastname, passenger_phone, passenger_email

FD2: **passenger_phone** → passenger_id, passenger_firstname, passenger_lastname, passenger_email

FD3: **passenger_email** → passenger_id, passenger_firstname, passenger_lastname, passenger_phone

Υπονήφια κλειδιά είναι τα **passenger_id**, **passenger_phone** και **passenger_email**.

Είναι **1NF**, γιατί τα πεδία τιμών όλων των χαρακτηριστικών του πίνακα "**passenger**" είναι ατομικά.

Είναι **2NF**, γιατί κανένα χαρακτηριστικό από αυτά που συμμετέχουν στο υπονήφιο κλειδί **passenger_id** δεν προσδιορίζεται από τμήμα ενός υπονήφιου κλειδιού.

Είναι **3NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υπονήφιο κλειδί **passenger_id** δεν προσδιορίζεται μεταβατικά (δηλαδή μέσω κάποιου άλλου χαρακτηριστικού το οποίο δεν συμμετέχει σε υπονήφιο κλειδί) από ένα υπονήφιο κλειδί.

Είναι **BCNF**, γιατί τα **passenger_id**, **passenger_phone** και **passenger_email** των συναρτησιακών εξαρτήσεων **FD1**, **FD2** και **FD3** αντίστοιχα αποτελούν υποψήφια κλειδιά.

Ως **πρωτεύον κλειδί** του πίνακα “**passenger**” ορίζουμε το **passenger_id**.

3. Για τον πίνακα “**flight**”, οι συναρτησιακές εξαρτήσεις είναι οι ακόλουθες:

FD1: **departure_date**, **flight_code** → **departure_airport**, **arrival_airport**, **distance**, **scheduled_departure_time**, **scheduled_arrival_time**, **scheduled_duration**, **actual_departure_time**, **actual_arrival_time**, **flight_status**, **aircraft_code**

Υποψήφιο κλειδί είναι ο συνδυασμός των πεδίων **departure_date** και **flight_code**.

Είναι **1NF**, γιατί τα πεδία τιμών όλων των χαρακτηριστικών του πίνακα “**flight**” είναι ατομικά.

Είναι **2NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υποψήφιο κλειδί **departure_date**, **flight_code** δεν προσδιορίζεται από τμήμα ενός υποψήφιου κλειδιού.

Είναι **3NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υποψήφιο κλειδί **departure_date**, **flight_code** δεν προσδιορίζεται μεταβατικά (δηλαδή μέσω κάποιου άλλου χαρακτηριστικού το οποίο δεν συμμετέχει σε υποψήφιο κλειδί) από ένα υποψήφιο κλειδί.

Είναι **BCNF**, γιατί το **departure_date**, **flight_code** της συναρτησιακής εξάρτησης **FD1** αποτελεί υποψήφιο κλειδί και κατ’ επέκταση **πρωτεύον κλειδί** του πίνακα “**flight**”.

4. Για τον πίνακα “**airport**”, οι συναρτησιακές εξαρτήσεις είναι οι ακόλουθες:

FD1: **airport_code** → **airport_name**, **airport_city**, **airport_timezone**

FD2: **airport_name** → **airport_code**, **airport_city**, **airport_timezone**

FD3: **airport_city** → **airport_timezone**

Τα υποψήφια κλειδιά είναι τα **airport_code**, **airport_name**.

Είναι **1NF**, γιατί τα πεδία τιμών όλων των χαρακτηριστικών του πίνακα “**airport**” είναι ατομικά.

Δεν είναι **2NF**, γιατί χαρακτηριστικό, το **airport_timezone**, το οποίο δεν προσδιορίζεται από χαρακτηριστικό το οποίο δεν ανήκει σε υποψήφιο κλειδί.

Άρα, δεν είναι **3NF**, ούτε **BCNF**.

Επομένως, θα χρειαστεί να γίνει διάσπαση στον πίνακα “**airport**” σε δύο πίνακες: “**airport**” και “**city**”.

Οι νέοι πίνακες θα είναι:

- Ο πίνακας “**airport**” περιέχει τα ακόλουθα πεδία:
 - **airport_code**: Κωδικός αεροδρομίου
 - **airport_name**: Ονομασία αεροδρομίου
 - **airport_city**: Πόλη αεροδρομίου

Για τον νέο πίνακα “**airport**”, οι συναρτησιακές εξαρτήσεις είναι οι ακόλουθες:

FD1: **airport_code** → **airport_name**, **airport_city**

FD2: **airport_name** → **airport_code**, **airport_city**

Τα υποψήφια κλειδιά είναι τα **airport_name** και **airport_code**.

Είναι **1NF**, γιατί τα πεδία τιμών όλων των χαρακτηριστικών του πίνακα “**airport**” είναι ατομικά.

Είναι **2NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στα υποψήφια κλειδιά **airport_name** και **airport_code** δεν προσδιορίζεται από τμήμα ενός υποψήφιου κλειδιού.

Είναι **3NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στα υπονήφια κλειδιά **airport_name** και **airport_code** δεν προσδιορίζεται μεταβατικά (δηλαδή μέσω κάποιου άλλου χαρακτηριστικού το οποίο δεν συμμετέχει σε υπονήφιο κλειδί) από ένα υπονήφιο κλειδί.

Είναι **BCNF**, γιατί τα **airport_name** και **airport_code** των συναρτησιακών εξαρτήσεων **FD1** και **FD2** αντίστοιχα αποτελούν υπονήφια κλειδιά.

Ως **πρωτεύον κλειδί** του πίνακα “**airport**” ορίζουμε το **airport_code**.

- Ο πίνακας “**city**” περιέχει τα ακόλουθα πεδία:
 - **city_name**: Όνομα πόλης
 - **city_timezone**: Ζώνης ώρας της πόλης.

FD1: **city_name** → city_timezone

Υπονήφιο κλειδί είναι το **city_name**.

Είναι **1NF**, γιατί τα πεδία τιμών όλων των χαρακτηριστικών του πίνακα “**city**” είναι ατομικά.

Είναι **2NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υπονήφιο κλειδί **city_name** δεν προσδιορίζεται από τμήμα ενός υπονήφιου κλειδιού.

Είναι **3NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υπονήφιο κλειδί **city_name** δεν προσδιορίζεται μεταβατικά (δηλαδή μέσω κάποιου άλλου χαρακτηριστικού το οποίο δεν συμμετέχει σε υπονήφιο κλειδί) από ένα υπονήφιο κλειδί.

Είναι **BCNF**, γιατί το **city_name** της συναρτησιακής εξάρτησης **FD1** αποτελεί υπονήφιο κλειδί και κατ’ επέκταση **πρωτεύον κλειδί** του πίνακα “**city**”.

5. Για τον πίνακα “**aircraft**”, οι συναρτησιακές εξαρτήσεις είναι οι ακόλουθες:

FD1: **aircraft_code** → aircraft_name, aircraft_capacity, aircraft_range

FD2: **aircraft_name** → aircraft_code, aircraft_capacity, aircraft_range

Υπονήφιο κλειδί είναι τα **aircraft_code** και **aircraft_name**.

Είναι **1NF**, γιατί τα πεδία τιμών όλων των χαρακτηριστικών του πίνακα “**aircraft**” είναι ατομικά.

Είναι **2NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στα υπονήφια κλειδιά **aircraft_code** δεν προσδιορίζεται από τμήμα ενός υπονήφιου κλειδιού.

Είναι **3NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υπονήφιο κλειδί **aircraft_code** δεν προσδιορίζεται μεταβατικά (δηλαδή μέσω κάποιου άλλου χαρακτηριστικού το οποίο δεν συμμετέχει σε υπονήφιο κλειδί) από ένα υπονήφιο κλειδί.

Είναι **BCNF**, γιατί το **aircraft_code** της συναρτησιακής εξάρτησης **FD1** και το **aircraft_name** της συναρτησιακής εξάρτησης **FD2**, αποτελούν υπονήφια κλειδιά.

Επιλέγουμε ως **πρωτεύον κλειδί** του πίνακα “**aircraft**” το υπονήφιο κλειδί **aircraft_code**.

6. Για τον πίνακα “**boarding_pass**”, οι συναρτησιακές εξαρτήσεις είναι οι ακόλουθες:

FD1: **flight_code, departure_date, seat_no** → ticket_no, boarding_no

Υπονήφιο κλειδί είναι ο συνδυασμός των πεδίων **flight_code, departure_date** και **seat_no**.

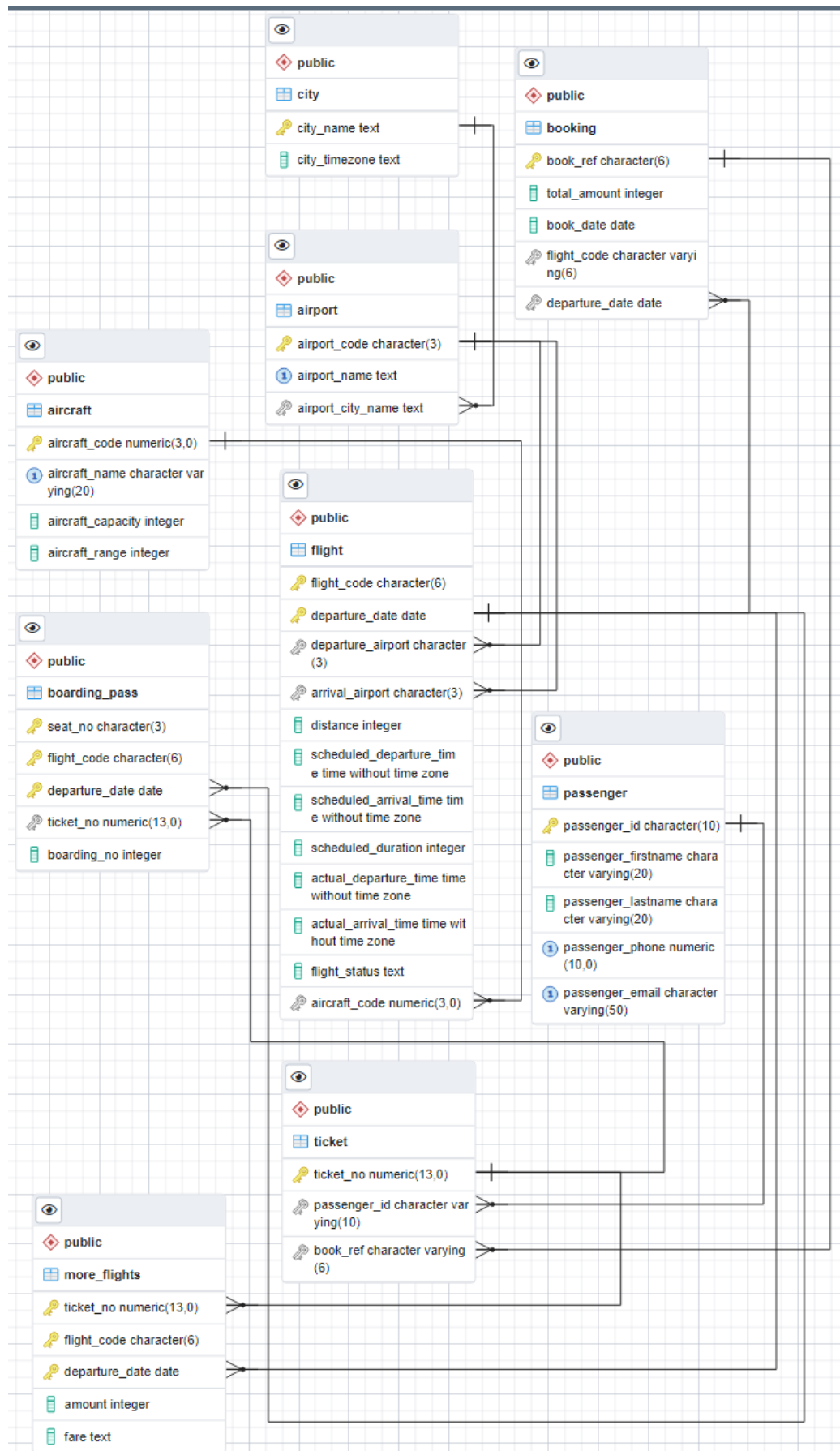
Είναι **1NF**, γιατί τα πεδία τιμών όλων των χαρακτηριστικών του πίνακα “**boarding_pass**” είναι ατομικά.

Είναι **2NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στο υπονήφιο κλειδί, **flight_code, departure_date** και **seat_no** δεν προσδιορίζεται από τμήμα ενός υπονήφιου κλειδιού.

Είναι **3NF**, γιατί κανένα χαρακτηριστικό από αυτά που δεν συμμετέχουν στα υπονήφια κλειδιά **ticket_no** δεν προσδιορίζεται μεταβατικά (δηλαδή μέσω κάποιου άλλου χαρακτηριστικού το οποίο δεν συμμετέχει σε υπονήφιο κλειδί) από ένα υπονήφιο κλειδί.

Είναι **BCNF**, γιατί ο συνδυασμός χαρακτηριστικών του πρώτου μέρους της συναρτησιακής εξάρτησης **FD1**, (**flight_code**, **departure_date**, **seat_no**), αποτελεί υπονήφιο κλειδί.

Το πρωτεύον κλειδί του πίνακα “**boarding_pass**” είναι το **flight_code**, **departure_date**, **seat_no**.



C.

Flights_View

```
WITH deair AS (SELECT airport_code AS departure_airport_code, airport_name AS departure_airport_name,
airport_city_name AS departure_airport_city, city_timezone AS departure_airport_timezone FROM airport INNER
JOIN city ON airport_city_name = city_name),
```

```
arair AS (SELECT airport_code AS arrival_airport_code, airport_name AS arrival_airport_name, airport_city_name
AS arrival_airport_city, city_timezone AS arrival_airport_timezone FROM airport INNER JOIN city ON
airport_city_name = city_name),
```

```
fl AS (SELECT flight_code,departure_airport, arrival_airport, departure_date, scheduled_departure_time,
actual_departure_time, scheduled_arrival_time, actual_arrival_time, scheduled_duration FROM flight WHERE
departure_date = '2022-06-16')
```

```
SELECT flight_code, departure_airport_code, departure_airport_name, departure_airport_city, arrival_airport_code,
arrival_airport_name, arrival_airport_city, date(departure_date AT TIME ZONE departure_airport_timezone) AS
departure_date, actual_departure_time AT TIME ZONE departure_airport_timezone AS actual_departure_time,
(date(departure_date AT TIME ZONE arrival_airport_timezone ) + make_interval(0,0,0,0,scheduled_duration)) AS
arrival_date, scheduled_arrival_time AT TIME ZONE arrival_airport_timezone as scheduled_arrival_time,
actual_arrival_time AT TIME ZONE arrival_airport_timezone as actual_arrival_time,
make_interval(0,0,0,0,scheduled_duration) AS scheduled_duration, (actual_arrival_time - actual_departure_time)
AS actual_duration FROM (fl INNER JOIN deair ON departure_airport = departure_airport_code) INNER JOIN
arair ON arrival_airport = arrival_airport_code;
```

	flight_code character (6)	departure_airport_code character (3)	departure_airport_name text	departure_airport_city text	arrival_airport_code character (3)	arrival_airport_name text	arrival_airport_city text	departure_date date	actual_departure_time time with time zone	arrival_date timestamp without time zone	scheduled_arrival_time time with time zone	actual_arrival_time time with time zone	scheduled_duration interval	actual_duration interval
1	PG5391	YFI	Ainsworth Airport	Turangi	LND	Lanham Field	Wringingulih	2022-06-16	15:12:00+12:00	2022-06-16 12:34:00	09:36:00+07:00	10:16:00+07:00	12:34:00	00:04:00
2	PG4027	WRA	Dutch Gap Airstrip	Bendari	SPB	Tembec Heliport	Goragorskiy	2022-06-16	09:29:00+07:00	2022-06-16 09:45:00	05:08:00+03:00	06:06:00+03:00	09:45:00	00:37:00
3	PG9895	TWX	Midwest Heliport	Malveira	WML	Jecan Airport	Metz	2022-06-15	02:01:00+01:00	2022-06-15 03:30:00	03:03:00+02:00	03:46:00+02:00	03:30:00	00:45:00
4	PG8346	LBI	Breezecroft Airport	Kanlagay	OYJ	Bondair Airport	Tembau	2022-06-16	00:58:00+08:00	2022-06-16 12:46:00	01:03:00+08:00	01:36:00+08:00	12:46:00	00:38:00
5	PG2702	EUN	Fisher Airport	Sasnovy Bor	PEB	Ozark Skies Airpark	Kotovo	2022-06-16	08:06:00+03:00	2022-06-16 12:13:00	08:20:00+03:00	08:58:00+03:00	12:13:00	00:52:00
6	PG6442	OYJ	Bondair Airport	Tembau	AHI	Hardy Airport	Priargunsk	2022-06-16	21:02:00+08:00	2022-06-16 02:08:00	21:58:00+09:00	22:59:00+09:00	02:08:00	00:57:00
7	PG1948	XZQ	Campbell Field	Pointe-à-Pitre	UCR	Randall Airport	San-Pédro	2022-06-15	08:35:00-04:00	2022-06-15 09:05:00	11:33:00+00:00	12:38:00+00:00	09:05:00	00:03:00
8	PG2371	AHI	Hardy Airport	Priargunsk	LFN	Drisko Airport	Habartov	2022-06-16	09:48:00+09:00	2022-06-15 10:57:00	02:52:00+02:00	03:18:00+02:00	10:57:00	00:30:00

Routes_View

```
WITH deair AS (SELECT airport_code AS departure_airport_code, airport_name AS departure_airport_name,
airport_city_name AS departure_airport_city FROM airport INNER JOIN city ON airport_city_name = city_name),
arair AS (SELECT airport_code AS arrival_airport_code, airport_name AS arrival_airport_name, airport_city_name
AS arrival_airport_city FROM airport INNER JOIN city ON airport_city_name = city_name), fl_air AS (SELECT
flight_code,departure_airport, arrival_airport, aircraft.aircraft_code, scheduled_duration, aircraft_name,
departure_date FROM flight INNER JOIN aircraft ON aircraft.aircraft_code = flight.aircraft_code WHERE
departure_date BETWEEN '2022-05-30' AND '2022-06-05') SELECT flight_code, departure_airport_code,
departure_airport_name, departure_airport_city, arrival_airport_code, arrival_airport_name, arrival_airport_city,
aircraft_name AS aircraft_code, make_interval(0,0,0,0,scheduled_duration) AS scheduled_duration FROM (fl_air
INNER JOIN deair ON departure_airport = departure_airport_code) INNER JOIN arair ON arrival_airport =
arrival_airport_code ORDER BY departure_date ASC;
```

	flight_code character (6)	departure_airport_code character (3)	departure_airport_name text	departure_airport_city text	arrival_airport_code character (3)	arrival_airport_name text	arrival_airport_city text	aircraft_code character varying (20)	scheduled_duration interval
1	PG2535	OYJ	Bondair Airport	Tembau	PEB	Ozark Skies Airpark	Kotovo	Superjet 246-036	05:43:00
2	PG4084	VKP	Hummingbird Heliport	Ebu	NKH	Hawk Haven Airport	Har Nur	Boeing 397-053	07:23:00
3	PG5294	HLI	Pocock Airport	Sasnovy Bor	ZOE	Richards Field	Bluff	Airbus 661-424	02:18:00
4	PG1320	ABW	Aeleron Airport	Malveira	POS	Hensley Airpark	Tungelsta	Airbus 633-537	07:26:00
5	PG7504	LPW	Tackaberry Airport	Wringinputih	PEB	Ozark Skies Airpark	Kotovo	Boeing 217-517	02:01:00
6	PG8845	SPB	Tembec Heliport	Goragorskiy	YDH	Flat Creek Field	Sulop	Airbus 704-001	09:49:00
7	PG3756	UBR	Folmar Airport	Amos	KQI	Teufel Heliport	Yong'an	Boeing 573-195	11:51:00
8	PG9164	PZK	Grawunder Field	Krousón	MSA	Jones Landing	Douliu	Superjet 856-961	10:42:00
9	PG8818	YFI	Ainsworth Airport	Turangi	LPW	Tackaberry Airport	Wringinputih	Airbus 889-993	05:30:00
10	PG4807	OYJ	Bondair Airport	Tembau	YFI	Ainsworth Airport	Turangi	Airbus 456-457	05:33:00
11	PG5259	SPB	Tembec Heliport	Goragorskiy	YZN	Kistler Ranch Airport	Ebu	Airbus 469-891	12:59:00
12	PG6355	PEB	Ozark Skies Airpark	Kotovo	LBI	Breezcroft Airport	Kanlagay	Airbus 936-763	01:27:00
13	PG0765	UBR	Folmar Airport	Amos	ZOE	Richards Field	Bluff	Airbus 436-631	03:30:00
14	PG0240	ABW	Aeleron Airport	Malveira	CTO	Mc Kay Airport	Ormeta	Airbus 554-607	13:19:00
15	PG5341	VZY	Oljato Airport	Đà Lạt	LBI	Breezcroft Airport	Kanlagay	Boeing 529-716	08:36:00
Total rows: 15 of 15 Query complete 00:00:00.048									

Ερώτημα 2. Εκτελέστε τις παρακάτω ερωτήσεις (queries) στη ΒΔ (εντολές SELECT)

Εκφώνηση:

- Ποιος ταξίδεψε με μια συγκεκριμένη πτήση (π.χ. PG0404) χθες στη θέση 1A και πότε έγινε η κράτηση του εισιτηρίου; Η χθεσινή ημερομηνία θα υπολογίζεται με βάση την ημερομηνία εκτέλεσης του ερωτήματος.
- Πόσες θέσεις παρέμειναν ελεύθερες στην ανωτέρω πτήση;
- Ποιες πτήσεις είχαν τις μεγαλύτερες καθυστερήσεις μέσα στο 2022; Εμφανίστε τη λίστα των 5 πρώτων.
- Βρείτε τους 5 πιο συχνούς ταξιδιώτες (frequent travelers) μέσα στο 2022, δηλαδή αυτούς που έκαναν τα περισσότερα χιλιόμετρα πτήσεων.
- Βρείτε τους 5 πιο δημοφιλείς προορισμούς μέσα στο 2022, δηλαδή τις πόλεις προς τις οποίες ταξίδεψαν οι περισσότεροι επιβάτες.
- Βρείτε τους πιο «γρήγορους» επιβάτες, δηλαδή αυτούς που έκαναν check-in πρώτοι για όλες τις πτήσεις τους. Λάβετε υπόψη μόνο τους επιβάτες που έκαναν τουλάχιστον δύο πτήσεις.

Απάντηση:

A.

```
SELECT t.passenger_id, passenger_firstname, passenger_lastname, seat_no, bk.book_date AS ticket_book_date
FROM ((ticket AS t INNER JOIN passenger AS p ON t.passenger_id = p.passenger_id) INNER JOIN boarding_pass
AS b ON t.ticket_no = b.ticket_no) INNER JOIN flight AS f ON b.departure_date = f.departure_date INNER JOIN
booking AS bk ON f.departure_date = bk.departure_date WHERE f.flight_code = 'PG0878' AND seat_no = '52V'
AND f.departure_date= CURRENT_DATE - INTERVAL '1 DAY';
```

	passenger_id character varying (10)	passenger_firstname character varying (20)	passenger_lastname character varying (20)	seat_no character (3)	ticket_book_date date
1	QU-424087	Jayme	Burnes	52V	2022-06-23

(Ημερομηνία εκτέλεσης του query: 2/7/2022)

(Το αποτέλεσμα του παραπάνω query θα είναι διαφορετικό ανάλογα την ημέρα και πιθανώς να μην εμφανίσει αποτέλεσμα είτε επειδή δεν θα εκτελεστεί την προηγούμενη μέρα μία πτήση με κωδική ονομασία 'PG0878').

B.

```
WITH ts AS (SELECT aircraft_capacity AS total_seats_no FROM flight AS f INNER JOIN aircraft AS air ON
f.aircraft_code = air.aircraft_code WHERE f.flight_code = 'PG6590' AND f.departure_date = CURRENT_DATE -
INTERVAL '1 DAY'), bs AS (SELECT COUNT(seat_no) AS booked_seats_no FROM ((flight AS fl INNER JOIN
boarding_pass AS bp ON fl.flight_code = bp.flight_code AND fl.departure_date = bp.departure_date) INNER JOIN
```

ticket AS t ON bp.ticket_no = t.ticket_no) INNER JOIN booking AS b ON b.book_ref = t.book_ref WHERE fl.flight_code = 'PG6590' AND fl.departure_date = CURRENT_DATE - INTERVAL '1 DAY') SELECT ts.total_seats_no AS total_aircraft_seats, bs.booked_seats_no AS total_booked_seats, (ts.total_seats_no - bs.booked_seats_no) AS free_seats FROM ts, bs;

	total_aircraft_seats integer	total_booked_seats bigint	free_seats bigint
1	70	1	69

(Ημερομηνία εκτέλεσης του query: 2/7/2022)

(Εκτυπώνει τις στήλες total_aircraft_seats και total_booked_seats για να φανεί η πράξη: όλες οι θέσεις μείον οι μη-διαθέσιμες)

C.

SELECT flight_code, departure_date, actual_departure_time - scheduled_departure_time AS delay FROM flight WHERE departure_date >= '2022-01-01' and departure_date <= '2022-12-31' AND flight_status = 'Arrived' ORDER BY delay DESC LIMIT 5;

	flight_code [PK] character (6)	departure_date [PK] date	delay interval
1	PG2717	2022-09-06	01:11:00
2	PG0894	2022-07-31	01:11:00
3	PG9283	2022-11-18	01:10:00
4	PG5841	2022-07-13	01:10:00
5	PG6705	2022-09-11	01:09:00

(Δεν συμπεριλήφθηκε η συνθήκη flight_status = 'Arrived' AND flight_status = 'Delayed' γιατί σύμφωνα με την εκφώνηση ζητούνται τα δρομολόγια που έχουν γίνει σε όλο το 2022)

D.

SELECT pa.passenger_id AS passenger_id, pa.passenger_firstname AS firstname, pa.passenger_lastname AS lastname FROM flight AS f INNER JOIN more_flights AS mf ON (f.flight_code = mf.flight_code AND f.departure_date = mf.departure_date) INNER JOIN ticket AS ti ON mf.ticket_no = ti.ticket_no INNER JOIN passenger AS pa ON pa.passenger_id = ti.passenger_id WHERE flight_status = 'Arrived' AND (f.departure_date >= '2022-01-01' AND f.departure_date <= '2022-12-31') GROUP BY pa.passenger_id ORDER BY SUM(f.distance) DESC LIMIT 5;

	passenger_id [PK] character (10)	firstname character varying (20)	lastname character varying (20)
1	NY-790223	Bink	Grabb
2	FN-180406	Winnie	Mantz
3	BM-601970	Kimberli	Sigmund
4	DR-357842	Charla	Schuh
5	RA-840352	Melvin	Hegarty

E.

SELECT city_name AS popular_destinations_in_2022 FROM flight AS f INNER JOIN airport AS a ON f.arrival_airport = a.airport_code INNER JOIN city AS c ON a.airport_city_name = c.city_name WHERE

f.flight_status = 'Arrived' AND (f.departure_date >= '2022-01-01' AND f.departure_date <= '2022-12-31') GROUP BY city_name ORDER BY COUNT(f.arrival_airport) DESC LIMIT 5;

	popular_destinations_in_2022 text
1	Wringinputih
2	Priargunsk
3	Kotovo
4	Tungelsta
5	Yong'an

F.

WITH fp AS (SELECT pa.passenger_id AS passenger_id, pa.passenger_firstname AS firstname, pa.passenger_lastname AS lastname, COUNT(fl.flight_code) AS times FROM passenger AS pa INNER JOIN ticket as ti ON pa.passenger_id = ti.passenger_id INNER JOIN boarding_pass AS bp ON ti.ticket_no = bp.ticket_no INNER JOIN flight AS fl ON bp.departure_date = fl.departure_date AND bp.flight_code = fl.flight_code INNER JOIN more_flights AS mf ON fl.departure_date = mf.departure_date AND fl.flight_code = mf.flight_code WHERE flight_status != 'Scheduled' GROUP BY pa.passenger_id)

SELECT passenger_id, firstname, lastname FROM fp WHERE times >= 2 ORDER BY times DESC LIMIT 5;

	passenger_id [PK] character (10)	firstname character varying (20)	lastname character varying (20)
1	HR-067250	Christen	Matveiko
2	TL-212595	Gabrila	Toghill
3	EG-886358	Rosalyn	Swales
4	MH-170209	Vidovic	Cornewell
5	QU-424087	Jayme	Burnes

Ερώτημα 3. Υλοποίηση triggers, cursors

Εκφώνηση:

a. Φτιάξτε έναν trigger ο οποίος κρατά/γεμίζει ένα πίνακα-ιστορικό. Ο trigger ενεργοποιείται στις μεταβολές διαγραφή και ενημέρωση του κύριου πίνακα. Όταν διαγράφονται με επιτυχία γραμμές από τον πίνακα booking (π.χ. διαγράφονται όλες οι εγγραφές με ημερομηνία μικρότερη του 1970-01-01 00:00:00), τότε οι διαγραμμένες γραμμές εισάγονται αυτόματα στον πίνακα booking-log. Όταν πραγματοποιείται μια ενημέρωση μιας τιμής στον κύριο πίνακα booking, τότε ο trigger ενεργοποιείται και καταγράφει την παλιά εγγραφή στον πίνακα booking-log. Ιδανικά μαζί με τη μεταβαλλόμενη εγγραφή καταγράφει και την ημερομηνία και ώρα της αλλαγής καθώς και το είδος της μεταβολής (d-για delete και u-για update).

b. Βρείτε και εμφανίστε τα στοιχεία ημερομηνία αναχώρησης, ημερομηνία άφιξης, όνομα αεροδρομίου, κωδικός πτήσης, όνομα επιβάτη, ομαδοποιημένα ανά όνομα επιβάτη και ημερομηνία αναχώρησης. Χρησιμοποιείστε cursors ώστε να εμφανίσετε τις γραμμές σε ομάδες των 15.

Απάντηση:

A.

Κώδικας Trigger:

```

CREATE OR REPLACE FUNCTION backup_booking()
RETURNS TRIGGER AS $edit_booking$
BEGIN
    CREATE TABLE IF NOT EXISTS booking_log
    (
        id SERIAL PRIMARY KEY,
        book_ref CHAR(6) NOT NULL,
        total_amount INTEGER NOT NULL,
        book_date DATE NOT NULL,
        flight_code VARCHAR(6) NOT NULL,
        departure_date DATE NOT NULL,
        delete_time TIMESTAMP NOT NULL,
        function_type CHAR(1) NOT NULL
    );
    IF (TG_OP = 'DELETE') THEN
        INSERT INTO booking_log(book_ref,total_amount,book_date,flight_code,departure_date, delete_time, function_type)
        VALUES (OLD.book_ref, OLD.total_amount, OLD.book_date, OLD.flight_code, OLD.departure_date, current_timestamp, 'd');
        RETURN OLD;
    ELSIF (TG_OP = 'UPDATE') THEN
        INSERT INTO booking_log(book_ref,total_amount,book_date,flight_code,departure_date, delete_time, function_type)
        VALUES (OLD.book_ref, OLD.total_amount, OLD.book_date, OLD.flight_code, OLD.departure_date, current_timestamp, 'u');
        RETURN NEW;
    END IF;
END;
$edit_booking$ LANGUAGE plpgsql;
CREATE TRIGGER edit_booking
BEFORE DELETE OR UPDATE ON booking
FOR EACH ROW
EXECUTE PROCEDURE backup_booking();

```

Ενεργοποίηση του trigger με την εκτέλεση ενός DELETE query στο booking:

9	DELETE FROM booking WHERE book_date < '2022-01-03';	
Data output	Messages	Notifications
NOTICE: relation "booking_log" already exists, skipping DELETE 16		
Query returned successfully in 57 msec.		
Total rows: 1000 of 1000	Query complete 00:00:00.057	

Ενεργοποίηση του trigger με την εκτέλεση ενός UPDATE query στο booking:

9	UPDATE booking SET total_amount = '2000' WHERE book_date = '2022-01-04';	
Data output	Messages	Notifications
NOTICE: relation "booking_log" already exists, skipping NOTICE: relation "booking_log" already exists, skipping UPDATE 2		
Total rows: 1000 of 1000	Query complete 00:00:00.036	

Προβολή πίνακα booking_log μετά τις δύο παραπάνω ενέργειες στο booking πίνακα:

12

SELECT * FROM booking_log;

Data output

Messages

Notifications

≡+

📄

▼

📋

🗑

🔄

⬇

	id [PK] integer	book_ref character (6)	total_amount integer	book_date date	flight_code character varying (6)	departure_date date	delete_time timestamp without time zone	function_type character (1)
1	1	d9D466	74386	2021-12-26	PG1764	2022-01-22	2022-07-10 21:00:56.680833	d
2	2	U9XbFh	43168	2021-12-21	PG5916	2022-01-12	2022-07-10 21:00:56.680833	d
3	3	1WV1k8	51617	2021-12-26	PG5617	2022-01-10	2022-07-10 21:00:56.680833	d
4	4	C24833	35523	2021-12-31	PG8485	2022-01-17	2022-07-10 21:00:56.680833	d
5	5	t51m46	86348	2021-12-30	PG6650	2022-01-04	2022-07-10 21:00:56.680833	d
6	6	z4B10g	12738	2021-12-30	PG4606	2022-01-16	2022-07-10 21:00:56.680833	d
7	7	9rYb5o	21676	2021-12-31	PG3590	2022-01-23	2022-07-10 21:00:56.680833	d
8	8	P080rZ	88542	2022-01-02	PG4741	2022-01-22	2022-07-10 21:00:56.680833	d
9	9	3s5v0Z	91675	2021-12-27	PG2741	2022-01-08	2022-07-10 21:00:56.680833	d
10	10	1cCI00	67042	2021-12-27	PG9996	2022-01-14	2022-07-10 21:00:56.680833	d
11	11	7XHc85	72677	2021-12-26	PG5556	2022-01-22	2022-07-10 21:00:56.680833	d
12	12	u2VNo9	34626	2021-12-30	PG7850	2022-01-19	2022-07-10 21:00:56.680833	d
13	13	l2bP92	61610	2021-12-27	PG9993	2022-01-24	2022-07-10 21:00:56.680833	d
14	14	A54S89	70011	2021-12-27	PG3045	2022-01-01	2022-07-10 21:00:56.680833	d
15	15	i0cNoJ	73953	2022-01-01	PG2651	2022-01-08	2022-07-10 21:00:56.680833	d
16	16	6S7Su9	8147	2022-01-02	PG6542	2022-01-08	2022-07-10 21:00:56.680833	d
17	17	6VO70P	26275	2022-01-04	PG0799	2022-01-16	2022-07-10 21:08:33.210886	u
18	18	R8I35W	64065	2022-01-04	PG2718	2022-01-12	2022-07-10 21:08:33.210886	u

Total rows: 18 of 18

Query complete 00:00:00.048

B.

Κώδικας Cursor:

```
CREATE OR REPLACE FUNCTION fun()
RETURNS INT
AS
$$
DECLARE

    my_cursor cursor for(
        WITH deair AS
            (SELECT airport_code AS departure_airport_code, airport_name AS departure_airport_name, city_timezone AS departure_airport_timezone
            FROM airport INNER JOIN city ON airport_city_name = city_name),
            arair AS
            (SELECT airport_code AS arrival_airport_code, airport_name AS arrival_airport_name, city_timezone AS arrival_airport_timezone
            FROM airport INNER JOIN city ON airport_city_name = city_name),
            fl_pass AS
            (SELECT fl.flight_code, fl.departure_date, fl.scheduled_duration, fl.departure_airport, fl.arrival_airport, pa.passenger_firstname, pa.passenger_lastname
            FROM flight AS fl INNER JOIN more_flights AS mf ON (fl.flight_code = mf.flight_code AND fl.departure_date = mf.departure_date)
            INNER JOIN ticket AS ti ON mf.ticket_no = ti.ticket_no
            INNER JOIN passenger AS pa ON ti.passenger_id = pa.passenger_id),
            all_data AS
            (SELECT departure_date,
            date(departure_date + make_interval(0,0,0,0,0,scheduled_duration)) AS arrival_date,
            departure_airport_name, arrival_airport_name,
            flight_code,
            passenger_firstname, passenger_lastname
            FROM (fl_pass INNER JOIN deair ON departure_airport = departure_airport_code) INNER JOIN arair ON arrival_airport = arrival_airport_code
            GROUP BY passenger_lastname, passenger_firstname, departure_date, scheduled_duration, departure_airport_name, arrival_airport_name, flight_code, passenger_firstname, passenger_lastname)

        SELECT * FROM all_data);

    rec all_data%rowtype;
    counter int:=0;
    cnt int:=0;
BEGIN
    open my_cursor;
    loop
        WHILE(counter <=14)
            FETCH my_cursor into rec;
            EXIT WHEN NOT FOUND;
            counter:=counter+1;
            RAISE NOTICE '%000000%',rec.departure_date,rec.arrival_date,rec.departure_airport_name,rec.arrival_airport_name,rec.flight_code,rec.passenger_firstname,rec.passenger_lastname;
        END;
        counter:=0;
        cnt:=cnt+1;
    end loop;
    close my_cursor;
    RETURN cnt;
END;
$$
LANGUAGE plpgsql;
```

Ερώτημα 4. Σύνδεση ΒΔ με Application Programming Interface (API)

Εκφώνηση:

Υλοποιήστε προγραμματιστικά έναν client σε οποιαδήποτε γλώσσα προγραμματισμού γνωρίζετε (π.χ. Python, Java, C) χρησιμοποιώντας την κατάλληλη βιβλιοθήκη σύνδεσης με την PostgreSQL (π.χ. psycopg2, JDBC, ODBC). Ο client θα συνδέεται στο ΣΔΒΔ της PostgreSQL, θα εκτελεί τα queries του Ερωτήματος 2, και θα εμφανίζει τα αποτελέσματα στον χρήστη (είτε σε terminal είτε με γραφικά).

Απάντηση:

Το API δημιουργήθηκε με την γλώσσα προγραμματισμού Python.

Το αρχείο που περιέχει τον πρόγραμμα είναι το thema4.py και συμπεριλαμβάνεται στον αποσυμπίεσμένο φάκελο που βρίσκεται η τελική εργασία.

Στιγμιότυπα εκτέλεσης του προγράμματος:

Ερώτημα 2a:

```
Please Choose A Query :
1)2a
2)2b
3)2c
4)2d
5)2e
6)2f
[QUERY:1
Flight_code:PG8955
[Seat:12Z
--QUERY-A--
+-----+-----+-----+-----+-----+
| Passenger_ID | Firstname | Lastname | seat_no | ticket_book_date |
+-----+-----+-----+-----+-----+
| JZ-492786    | Merle    | Broadwell | 12Z     | 2022-06-13       |
| JZ-492786    | Merle    | Broadwell | 12Z     | 2022-06-23       |
| JZ-492786    | Merle    | Broadwell | 12Z     | 2022-07-06       |
+-----+-----+-----+-----+-----+
Do you want to choose another query? Y/N Y
```

Ερώτημα 2b:

```
Flight_code:PG8955
--QUERY-B--
+-----+-----+-----+
| Total_aircraft_seats | total_booked_seats | free_seats |
+-----+-----+-----+
| 134                  | 1                  | 133        |
+-----+-----+-----+
Do you want to choose another query? Y/N Y
QUERY:3
--QUERY-C--
+-----+-----+-----+
| Flight_code | departure_date | Delay |
+-----+-----+-----+
| PG2717      | 2022-09-06     | 1:11:00 |
| PG0894      | 2022-07-31     | 1:11:00 |
| PG9283      | 2022-11-18     | 1:10:00 |
| PG6809      | 2022-01-05     | 1:09:00 |
| PG6705      | 2022-09-11     | 1:09:00 |
+-----+-----+-----+
```


Ερώτημα 2c:

```
QUERY:3
--QUERY-C--
+-----+-----+-----+
| Flight_code | departure_date | Delay |
+-----+-----+-----+
| PG2717      | 2022-09-06     | 1:11:00 |
| PG0894      | 2022-07-31     | 1:11:00 |
| PG9283      | 2022-11-18     | 1:10:00 |
| PG6809      | 2022-01-05     | 1:09:00 |
| PG6705      | 2022-09-11     | 1:09:00 |
+-----+-----+-----+
```

Ερώτημα 2d:

```
--QUERY-D--
+-----+-----+-----+
| Passenger_ID | Firstname | Lastname |
+-----+-----+-----+
| NY-790223    | Bink      | Grabb    |
| FN-180406    | Winnie    | Mantz    |
| BM-601970    | Kimberli  | Sigmund  |
| DR-357842    | Charla    | Schuh    |
| RA-840352    | Melvin    | Hegarty  |
+-----+-----+-----+
```

Ερώτημα 2e:

```
QUERY:5
--QUERY E--
+-----+
| popular_destinations_in_2022 |
+-----+
| Wringinputih                  |
| Priargunsk                    |
| Kotovo                        |
| Tungelsta                     |
| Sasnovy Bor                   |
+-----+
```

Ερώτημα 2f:

```
QUERY:6
--QUERY F--
+-----+-----+-----+
| Passenger_ID | Firstname | Lastname |
+-----+-----+-----+
| HR-067250    | Christen | Matveiko |
| TL-212595    | Gabrila  | Toghill  |
| EG-886358    | Rosalyn  | Swales   |
| MH-170209    | Vidovic  | Cornewell |
| QU-424087    | Jayme    | Burnes   |
+-----+-----+-----+
Do you want to choose another query? Y/N n
PROGRAM TERMINATED
```

Σημειώσεις:

Στον συμπίεσμένο φάκελο βρίσκονται τα αρχεία:

- **airlineCreateTables.sql:** που δημιουργεί τους πίνακες της βάσης.
- **Db_csv.zip:** που περιλαμβάνει όλα τα δεδομένα της βάσης.
- **airlineDBRestore.sql:** αρχείο που περιλαμβάνει όλους τους πίνακες, τα αντίστοιχα δεδομένα τους και τα triggers και cursors του ερωτήματος 3.
- **Thema4.py:** που περιλαμβάνει το πρόγραμμα του ερωτήματος 4.

Το πρόγραμμα python για να εκτελεστεί σωστά θα πρέπει:

- 1) Να εγκατασταθούν οι βιβλιοθήκες:
 - **Psycopg2**
 - **Sys**
 - **PrettyTable**
- 2) Στην γραμμή 8 του αρχείου να εισάγεται τα κατάλληλα στοιχεία του server στον οποίο έχετε περάσει την βάση δεδομένων (την οποία μπορείτε να φορτώσετε κατευθείαν αφού δημιουργήσετε πρώτα μια βάση δεδομένων με όνομα airline και μέσω της λειτουργίας Restore από το pgAdmin4 - μέσα στον συμπίεσμένο φάκελο βρίσκεται το αρχείο airlineDBRestore.sql που δημιουργεί την βάση δεδομένων της άσκησης.)