

5η Εργασία: Κανονικοποίηση και Ερωτήματα Ομαδοποίησης και Συνάθροισης

Προθεσμία: 27/05/2021

Σκοπός:

Συνεχίζουμε με μερικές ακόμα αλλαγές κανονικοποίησης και έπειτα τρέχουμε ερωτήματα ομαδοποίησης και συνάθροισης.

Προαπαιτούμενα:

Θεωρούμε πως η βάση είναι στην κατάσταση που περιγράφεται στην 4η Εργασία.

Ζητούμενα Εργασίας:

Μέρος Α

Κανονικοποίηση Πινάκων

1. Θεωρείστε τις ακόλουθες συναρτησιακές εξαρτήσεις, οι οποίες επιβεβαιώθηκαν μέσω ερωτημάτων στην προηγούμενη άσκηση:
 - ο Στον πίνακα `Movie_Cast`, το γνώρισμα `person_id` καθορίζει συναρτησιακά τα γνώρισμα `name` και `gender`. Δηλαδή, δεν μπορούμε να έχουμε δύο ηθοποιούς με το ίδιο `person_id` και διαφορετικό `name` ή `gender`.
 - ο Στον πίνακα `Movie_Crew`, το γνώρισμα `person_id` καθορίζει συναρτησιακά τα γνώρισμα `name` και `gender`.

Βάσει των προηγούμενων συναρτησιακών εξαρτήσεων, να εξετάσετε αν η βάση δεδομένων βρίσκεται σε *BCNF μορφή*. Εφόσον αυτή δε βρίσκεται, να γίνουν τα απαραίτητα βήματα για την κανονικοποίηση της. Να περιγραφούν τα **βήματα της κανονικοποίησης** και να θεωρήσετε ότι οι πίνακες που θα προκύψουν από την κανονικοποίηση του `Movie_Cast` θα έχουν ονόματα: `Movie_Cast` (η νέα έκδοχή του αρχικού πίνακα) και `Actor`. Αντίστοιχα οι πίνακες που θα προκύψουν από την κανονικοποίηση του `Movie_Crew` θα έχουν ονόματα: `Movie_Crew` (η νέα έκδοχή του αρχικού πίνακα) και `CrewMember`.

ER διάγραμμα

2. Θεωρείστε ότι οι κλάσεις οντοτήτων (που προκύπτουν από τους προηγούμενους πίνακες) `Actor` και `CrewMember` είναι υποκλάσεις μίας νέας κλάσης `Person`. Να θεωρήσετε ότι η κλάση `Person` θα περιέχει όλα τα κοινά γνωρίσματα των `Actor` και `CrewMember`.
3. Να ανανεωθεί το ER διάγραμμα το οποίο να περιλαμβάνει τις κλάσεις οντοτήτων `Actor` και `CrewMember`, την υπερκλάση τους `Person` και να προστεθούν τα αντίστοιχα γνωρίσματα και κλειδιά, καθώς και οι συσχετίσεις των νέων κλάσεων με το υπόλοιπο διάγραμμα.

SQL

Με την βοήθεια της *BCNF* κανονικοποίησης και του ER διαγράμματος:

4. Να γραφτεί ο SQL κώδικας ο οποίος θα δημιουργήσει τους πίνακες `Person`, `Actor`, και `CrewMember` με τα αντίστοιχα πρωτογενή και ξένα κλειδιά.

5. Να γραφτεί το ερώτημα το οποίο να ελέγχει ποιες εγγραφές παραβιάζουν τις συναρτησιακές εξαρτήσεις `person_id → name` και `person_id → gender` για τον πίνακα που θα προκύψει από την ένωση των `Movie_Cast` και `Movie_Crew`. Το ερώτημα αυτό μπορεί να γίνει με `join` μεταξύ των δύο πινάκων.
Αφού βρεθεί κάποια εγγραφή που παραβιάζει τις συναρτησιακές εξαρτήσεις, να γίνουν οι αντίστοιχες αλλαγές για την εγγραφή χρησιμοποιώντας την εντολή `UPDATE`.
6. Να γραφτεί ο SQL κώδικας ο οποίος θα εισάγει τις κατάλληλες εγγραφές από τους πίνακες `Movie_Cast` και `Movie_Crew` στους πίνακες `Person`, `Actor`, και `CrewMember`. Να γίνει `INSERT` χρησιμοποιώντας το αποτέλεσμα ενός `SELECT` ερωτήματος με τα κατάλληλα πεδία.
 - ο Προσοχή έτσι ώστε τα `INSERT` να μην παραβιάζουν τις συνθήκες πρωτογενούς κλειδιού για τους 3 πίνακες. Να γίνει χρήση της εντολής `UNION` κατά το `INSERT` για τον πίνακα `Person`.
 - ο Προσοχή έτσι ώστε η σειρά των `INSERT` να μην παραβιάζει τις συνθήκες ξένου κλειδιού μεταξύ των 3 πινάκων.
7. Να γίνουν οι κατάλληλες προσθήκες ξένων κλειδιών και διαγραφές γνωρισμάτων για τους κανονικοποιημένους πίνακες `Movie_Cast` και `Movie_Crew`.
Πριν τη διαγραφή των γνωρισμάτων και την δημιουργία νέων ξένων κλειδιών για τους δύο πίνακες, να δημιουργηθούν τα αντίγραφα των πινάκων `Movie_Cast2` και `Movie_Crew2` τα οποία δε θα πρέπει να σβηστούν τόσο για λόγους backup, όσο και για να μπορούν οι εξεταστές να πραγματοποιούν τους κατάλληλους ελέγχους που σχετίζονται με τις προηγούμενες ασκήσεις.

Γράψτε τον SQL κώδικα για τα βήματα 4, 5, 6, 7 σε ένα αρχείο **part1.sql** μαζί με την αντίστοιχη περιγραφή σε σχόλια· την BCNF κανονικοποίηση σε ένα αρχείο **BCNF.docx** ή **BCNF.pdf** και το ER διάγραμμα στο αντίστοιχο **αρχείο εικόνας**.

Μέρος Β

Φτιάξτε 3 ερωτήματα συνάθροισης που να έχουν κάποιο νόημα.

- Όλα θα πρέπει να χρησιμοποιούν τον πίνακα **Person**.
- Όλα πρέπει να έχουν τουλάχιστον ένα `join`.
- Τουλάχιστον 2 από τα ερωτήματα θα πρέπει να έχουν ένα `WHERE`.
- Τουλάχιστον 2 από τα ερωτήματα θα πρέπει να έχουν ένα `GROUP BY`.
- Τουλάχιστον 2 από τα ερωτήματα θα πρέπει να έχουν ένα `HAVING`.

Γράψτε όλα τα ερωτήματα σε ένα αρχείο **part2.sql**. Προσθέστε τις σύντομες περιγραφές των ερωτημάτων και τα πλήθη των αποτελεσμάτων στο ίδιο αρχείο με τη μορφή σχολίων. Π.χ:

/ "Βρες μου τους τίτλους των ταινιών με μέσο όρο βαθμολογίας από χρήστες μεγαλύτερο του 4, μαζί με τον μέσο όρο βαθμολογίας τους"*

Output: 205 rows

**/*

```
SELECT m.title, avg(r.rating) as avgRating
FROM movie m
INNER JOIN ratings r
ON m.id = r.movie_id
GROUP BY m.id, m.title
HAVING avg(r.rating)>4
```

Εργαλεία:

- Postgres Cloud SQL instance
- Postgres psql ή/και PgAdmin
- [Draw.io](https://draw.io)

Συμβουλές για την υλοποίηση:

- Προτού αρχίσετε να κάνετε αλλαγές σε κάθε πίνακα θα μπορούσατε να φτιάχνετε ένα *αντίγραφο* αυτού. Εάν γίνει κάποιο λάθος και σας είναι δύσκολο να επιστρέψετε στην προηγούμενη κατάσταση που ήταν ο πίνακάς σας απλώς ξεκινάτε να δουλεύετε με το αντίγραφό του.
- Επιβεβαιώστε ότι ο χρήστης της βάσης του οποίου μας στέλνετε τα credentials όντως έχει πρόσβαση στη βάση σας και μπορεί να τρέχει SELECT queries.
- Τρέξτε και ελέγξτε κάθε ερώτημα στην βάση σας.
- Επιβεβαιώστε ότι κανένα ερώτημα δεν είναι άνευ ουσίας όσον αφορά την εξερεύνηση των δεδομένων με την έννοια ότι δεν είναι απλή εμφάνιση κάποιου πίνακα. Το ζητούμενο είναι να υπάρχει συνδυασμός κριτηρίων ώστε να εξαχεται κάποια γνώση.

Χρήσιμα links:

- DISTINCT: <https://www.postgresql.org/docs/9.6/static/sql-select.html#SQL-DISTINCT>
- ALTER TABLE: <https://www.postgresql.org/docs/9.6/static/sql-altertable.html>
- SELECT: <https://www.postgresql.org/docs/9.6/static/sql-select.html>
- UNION: <https://www.postgresql.org/docs/current/queries-union.html>
- Update: <https://www.postgresql.org/docs/current/sql-update.html>
- INSERT: <https://www.postgresql.org/docs/current/sql-insert.html>

Δείτε την χρήση της εντολής INSERT για την εισαγωγή δεδομένων που προκύπτουν από ένα SELECT ερώτημα.

Παραδοτέα:

- Δημιουργήστε ένα .txt αρχείο στο οποίο θα αναγράφονται το endpoint του Azure instance σας (Server name στο Overview tab του Azure), το όνομα της βάσης σας και το username και το password ενός χρήστη με read-only δικαιώματα, ώστε να μπορούμε να δούμε τους πίνακες της βάσης σας. Το .txt αρχείο θα πρέπει να έχει την παρακάτω μορφή:

Endpoint: <name_of_the_endpoint>
Username: <username>
Password: <password>
Database: <name_of_the_database>

- Βάλτε τα αρχεία part1.sql, BCNF.docx/pdf, part2.sql, το αρχείο εικόνας του ER και το αρχείο .txt σε ένα φάκελο. Το όνομα του φακέλου πρέπει να αποτελείται από τους αριθμούς μητρώου σας χωρισμένους με παύλα, δηλαδή *αριθμός_μητρώου_1-αριθμός_μητρώου_2*. Δημιουργήστε ένα .zip αρχείο αυτού του φακέλου, το οποίο θα έχει το ίδιο όνομα με τον φάκελο.
- Κάντε υποβολή το .zip αρχείο στο eclass στην ενότητα *Εργασίες / 5η Εργασία*.