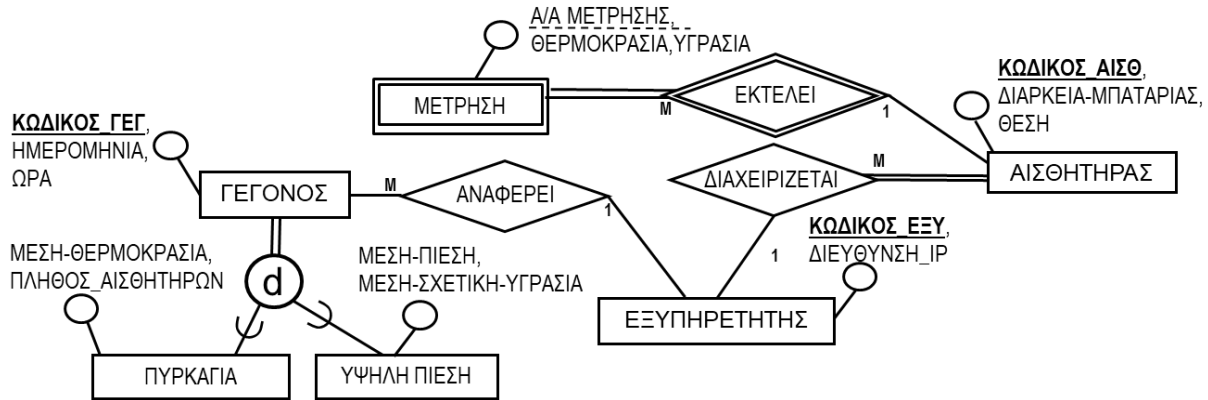


Εισαγωγή στις Βάσεις Δεδομένων

Ενότητα Α (Ερωτήματα θεωρίας και ασκήσεις πράξης)

Ερώτημα Α1 [20 βαθμοί]: Καταγράψετε το πλήρες script (της SQL) που ορίζει τους πίνακες που προκύπτουν μετά από τις απορροφήσεις στο παρακάτω Διάγραμμα Οντοτήτων-Συσχετίσεων (ΔΟΣ).



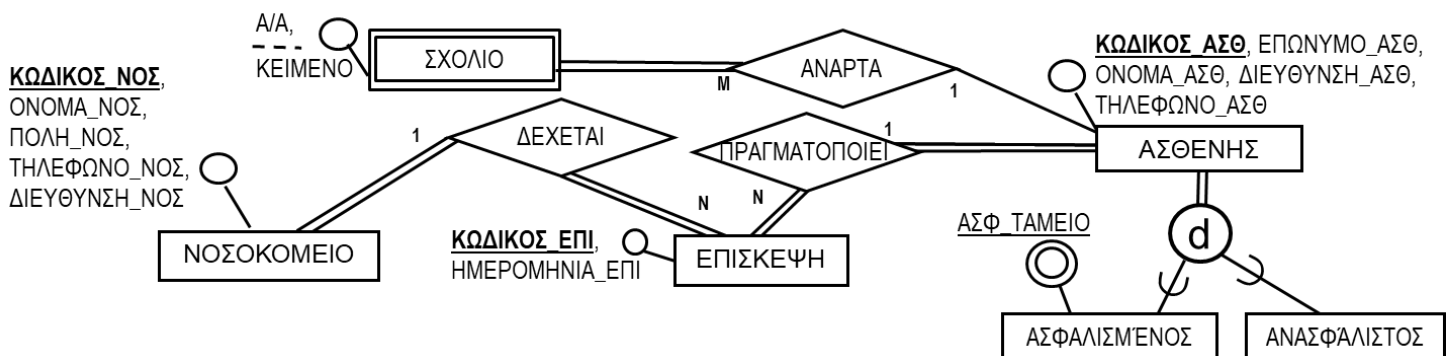
Ερώτημα Α2 [20 βαθμοί]: Με αναφορά στο ΔΟΣ του Ερωτήματος Α1 εξηγήστε παραθέστε ορισμούς και κατάλληλα παραδείγματα για τις έννοιες (α) ολική / μερική συμμετοχή οντότητας σε συσχέτιση [5 βαθμοί], (β) ασθενής οντότητα με σαφή αναφορά στην μητρική / προσδιορίζουσα οντότητα και το μερικό κλειδί [5 βαθμοί]. Ακολουθώντας τροποποιήστε το ΔΟΣ έτσι ώστε να περιλαμβάνεται ένα τουλάχιστον πλειότιμο γνώρισμα και μία αναδρομική συσχέτιση [5 βαθμοί] και εξηγήστε ποιες προσθήκες πινάκων πρέπει να γίνουν στο script του ερωτήματος Α1 ως αποτέλεσμα των παραπάνω [5 βαθμοί].

Ερώτημα Α3 [10 βαθμοί]:

Μια απλοποιημένη βάση δεδομένων ενός πληροφοριακού συστήματος που υποστηρίζει υποβολή αρχείων από φοιτητές περιλαμβάνει τον πίνακα UPLOAD (UPID, STUDENT_ID, FILE_NAME, FILE_TYPE, FILE_SIZE, TIMESTAMP), με γνωρίσματα τον κωδικό υποβολής (UPID), τον κωδικό φοιτητή που ανέβασε ένα αρχείο (STUDENT_ID), στοιχεία για το αρχείο όπως όνομα αρχείου (FILE_NAME), τύπος αρχείου (FILE_TYPE), μέγεθος αρχείου (FILE_SIZE) και τη χρονική στιγμή (TIMESTAMP) μιας υποβολής. Σημειώνεται ότι ένας φοιτητής μπορεί να υποβάλει πολλά αρχεία. Διατυπώστε σε SQL το ερώτημα 'Να εκτυπωθεί ο κωδικός του φοιτητή που ανέβασε το μικρότερο αρχείο σε μέγεθος' [5 βαθμοί]. Έστω ότι στη βάση δεδομένων προσθέτουμε τη σχέση STUDENT (STD_ID, STD_NAME, STD_SURNAME, EMAIL, PASSWORD) με τα στοιχεία των φοιτητών διατυπώστε τη δήλωση Σχεσιακής Άλγεβρας που απαντά στο ερώτημα 'Να βρεθούν τα e-mails όλων των φοιτητών που βρίσκονται στη βάση, οι οποίοι δεν έχουν ανεβάσει στη βάση αρχεία τύπου «ZIP»' [5 βαθμοί].

Ενότητα Β (Ερωτήματα από τη μελέτη περίπτωσης του εξαμήνου)

Ερώτημα Β1 [20 βαθμοί]: Με αναφορά στο ΔΟΣ που ακολουθεί: (Α) Εξηγήστε ποιοι πίνακες προκύπτουν μετά την εφαρμογή της διαδικασίας των απορροφήσεων [5] και ποιοι πίνακες δεσμεύονται με περιορισμούς αναφορικής ακεραιότητας [5 βαθμοί]. (Β) Διατυπώστε όλες τις δηλώσεις SQL που απαιτούνται έτσι ώστε ο κωδικός ενός ασθενή να αλλάξει από μια τιμή σε μια άλλη [10 βαθμοί].



Ερώτημα Β2 [30 βαθμοί]: Με βάση το script που προκύπτει από το παραπάνω ΔΟΣ, διατυπώστε δηλώσεις SQL για να διαγράψετε από τον πίνακα ΑΣΘΕΝΗΣ όλες τις πλειάδες (των ασθενών) που δεν έχουν καταχωρημένη κάποια επίσκεψη [10 βαθμοί]. Ακολουθώντας, διατυπώστε δηλώσεις SQL [10 βαθμοί] και Σχεσιακής Άλγεβρας [10 βαθμοί] για τα ερωτήματα:

- Να βρεθούν τα ονοματεπώνυμα των ασθενών που έκαναν επίσκεψη σε νοσοκομείο της αρεσκείας σας [2+2]
- Να βρεθούν το ονοματεπώνυμο των ασθενών που έχουν επισκεφθεί και το νοσοκομείο με όνομα Α και το νοσοκομείο με όνομα Β [4+4]
- Να βρεθούν τα νοσοκομεία που δεν έχουν δεχθεί καμία επίσκεψη [4+4]

Ενδεικτικές επιλύσεις

A1.

```
DROP TABLE IF EXISTS "ΕΞΥΠΗΡΕΤΗΣ" CASCADE;  
DROP TABLE IF EXISTS "ΑΙΣΘΗΤΗΡΑΣ" CASCADE;  
DROP TABLE IF EXISTS "ΓΕΓΟΝΟΣ" CASCADE;  
DROP TABLE IF EXISTS "ΜΕΤΡΗΣΗ" CASCADE;
```

-- Δημιουργία Πίνακα {ΕΞΥΠΗΡΕΤΗΣ}:

```
CREATE TABLE "ΕΞΥΠΗΡΕΤΗΣ"  
("κωδ_εξυπηρέτη" INT NOT NULL,  
"διεύθυνση_IP" VARCHAR(30) NOT NULL,  
PRIMARY KEY("κωδ_εξυπηρέτη"));
```

-- Δημιουργία του Πίνακα {ΑΙΣΘΗΤΗΡΑΣ}:

```
CREATE TABLE "ΑΙΣΘΗΤΗΡΑΣ"  
("κωδ_αισθητήρα" INT NOT NULL,  
"χρονικό_διάστημα" VARCHAR(30) NOT NULL,  
"διάρκεια_μπαταρίας" INT,  
"κωδ_εξυπηρέτη" INT,  
FOREIGN KEY ("κωδ_εξυπηρέτη") REFERENCES "ΕΞΥΠΗΡΕΤΗΣ" ("κωδ_εξυπηρέτη"),  
PRIMARY KEY("κωδ_αισθητήρα"));
```

-- Δημιουργία του Πίνακα {ΓΕΓΟΝΟΣ}:

```
CREATE TABLE "ΓΕΓΟΝΟΣ"  
("κωδ_γεγονότος" INT NOT NULL,  
"ημερομηνία" DATE NOT NULL,  
"ώρα" TEXT,  
"κωδ_εξυπηρέτη" INT,  
FOREIGN KEY ("κωδ_εξυπηρέτη") REFERENCES "ΕΞΥΠΗΡΕΤΗΣ" ("κωδ_εξυπηρέτη"),  
PRIMARY KEY("κωδ_γεγονότος"));
```

-- Δημιουργία του Πίνακα {ΜΕΤΡΗΣΗ}:

```
CREATE TABLE "ΜΕΤΡΗΣΗ"  
("κωδ_αισθητήρα" INT NOT NULL,  
"αα_μέτρησης" INT NOT NULL,  
"θερμοκρασία" TEXT,  
"υγρασία" TEXT,  
"πίεση" TEXT,  
FOREIGN KEY ("κωδ_αισθητήρα") REFERENCES "ΑΙΣΘΗΤΗΡΑΣ" ("κωδ_αισθητήρα"),  
PRIMARY KEY("κωδ_αισθητήρα", "αα_μέτρησης"));
```

-- Δημιουργία του Πίνακα {ΠΥΡΚΑΓΙΑ}:

```
CREATE TABLE "ΠΥΡΚΑΓΙΑ"  
("κωδ_γεγονότος" INT NOT NULL,  
"μέση_θερμοκρασία" INT,  
"πλήθος_αισθητήρων" INT,  
FOREIGN KEY ("κωδ_γεγονότος ") REFERENCES "ΓΕΓΟΝΟΣ" ("κωδ_γεγονότος"),  
PRIMARY KEY("κωδ_γεγονότος"));
```

-- Δημιουργία του Πίνακα {ΥΨΗΛΗ_ΠΙΕΣΗ}:

```
CREATE TABLE "ΥΨΗΛΗ_ΠΙΕΣΗ"  
("κωδ_γεγονότος" INT NOT NULL,  
"μέση_πίεση" INT,  
"μέση_σχετική_υγρασία" INT,  
FOREIGN KEY ("κωδ_γεγονότος ") REFERENCES "ΓΕΓΟΝΟΣ" ("κωδ_γεγονότος"),  
PRIMARY KEY("κωδ_γεγονότος"));
```

Σημείωση: Κατά τη βαθμολόγηση ή έμφαση είναι στο κατά πόσο καταγράφονται όλοι οι πίνακες, με σωστά γνωρίσματα, πρωτεύοντα κλειδιά και ξένα κλειδιά (δηλαδή αν εφαρμόζονται σωστά οι καντπονες απορρόφησης) καθώς και αν μετατρέπεται σωστά η ασθενής οντότητα και η ειδική συσχέτιση

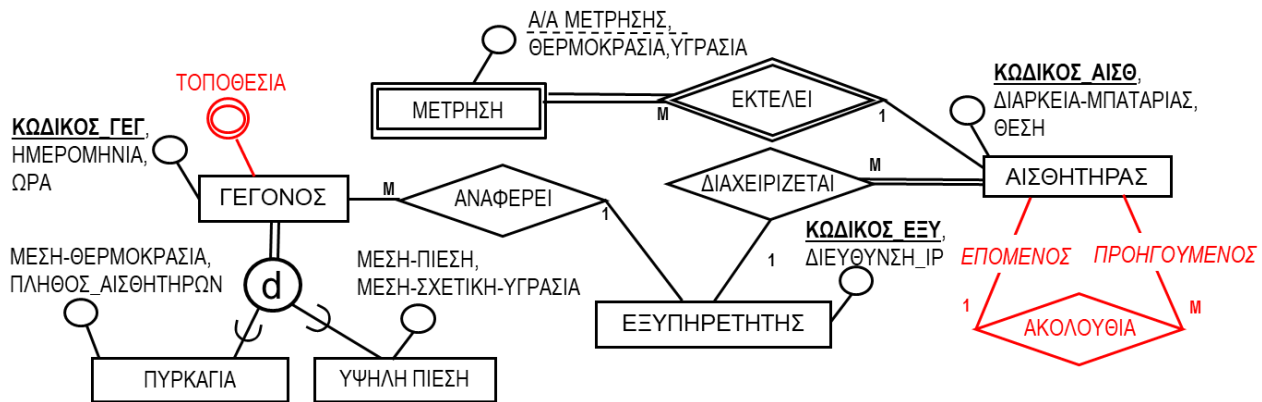
A2.

Ολική συμμετοχή (διπλή γραμμή) συνδέει την οντότητα ΑΙΣΘΗΤΗΤΑΣ με τη συσχέτιση ΔΙΑΧΕΙΡΙΖΕΤΑΙ έτσι ώστε κάθε παράσταση της οντότητας να σχετίζεται με κάποιο εξυπηρετητή

Μερική συμμετοχή (απλή γραμμή) συνδέει την οντότητα ΑΙΣΘΗΤΗΤΑΣ με τη συσχέτιση ΕΚΤΕΛΕΙ έτσι ώστε να υπάρχει παράσταση της οντότητας που να μην εκτελεί κάποια μέτρηση (περιστασιακά)

Ασθενής οντότητα είναι η οντότητα ΜΕΤΡΗΣΗ με μερικό κλειδί το γνώρισμα Α/Α ΜΕΤΡΗΣΗΣ που υπογραμμίζεται με διακεκομμένη γραμμή και μητρική/προσδιορίζουσα την οντότητα ΑΙΣΘΗΤΗΡΑΣ με τη οποία συνδέεται μέσω της συσχέτισης ΕΚΤΕΛΕΙ που έχει το διπλό περίγραμμα.

Τροποποιήσεις στο ΔΟΣ που θα περιλαμβάνουν ένα τουλάχιστον πλειότιμο γνώρισμα (δηλ. γνώρισμα για το οποίο ένα στιγμιότυπο μπορεί να πάρει περισσότερες από μία τιμές) και μία αναδρομική συσχέτιση (δηλ. συσχέτιση που συνδέει μια οντότητα με τον εαυτό της) φαίνονται στο αμέσως παρακάτω διάγραμμα με κόκκινο χρωματισμό. Συγκεκριμένα η προσθήκη πλειότιμου γνωρίσματος ΤΟΠΟΘΕΣΙΑ στην οντότητα ΓΕΓΟΝΟΣ σημαίνει ότι το εν λόγω γεγονός αφορά μια ή περισσότερες τοποθεσίες. Η αναδρομική συσχέτιση ΑΚΟΛΟΥΘΙΑ ουσιαστικά προσδιορίζει μια σειρά στους αισθητήρες με τους ρόλους ΕΠΟΜΕΝΟΣ και ΠΡΟΗΓΟΥΜΕΝΟΣ να σηματοδοτούν την κατεύθυνση.



Τα παραπάνω επιφέρουν αλλαγές και στο script του ερωτήματος A1 καθώς το πλειότιμο γνώρισμα θα δώσει νέο πίνακα με σύνθετο κλειδί (ΚΩΔΙΚΟΣ_ΓΕΓ, ΤΟΠΟΘΕΣΙΑ) που προσδιορίζει τις τοποθεσίες που επηρεάζονται από το γεγονός:

ΓΕΓΟΝΟΣ_ΕΚΤΑΣΗ (ΚΩΔΙΚΟΣ_ΓΕΓ, ΤΟΠΟΘΕΣΙΑ)

Στον παραπάνω πίνακα το γνώρισμα ΚΩΔΙΚΟΣ_ΓΕΓ πρέπει να δηλωθεί ως ξένο κλειδί και να δεσμεύεται από τις τιμές του "κωδ_γεγονότος" στον πίνακα ΓΕΓΟΝΟΣ. Επίσης, το γνώρισμα ΤΟΠΟΘΕΣΙΑ πρέπει να δηλωθεί ως NOT NULL καθώς είναι τμήμα του πρωτεύοντος κλειδιού.

Οπότε, ο νέος πίνακας είναι ως εξής:

-- Δημιουργία του Πίνακα { ΓΕΓΟΝΟΣ_ΕΚΤΑΣΗ }:

```
CREATE TABLE "ΓΕΓΟΝΟΣ_ΕΚΤΑΣΗ"
```

```
("ΚΩΔΙΚΟΣ_ΓΕΓ" INT NOT NULL,
```

```
"ΤΟΠΟΘΕΣΙΑ" TEXT NOT NULL,
```

```
FOREIGN KEY ("ΚΩΔΙΚΟΣ_ΓΕΓ") REFERENCES "ΓΕΓΟΝΟΣ" ("κωδ_γεγονότος"),
```

```
PRIMARY KEY("ΚΩΔΙΚΟΣ_ΓΕΓ", "ΤΟΠΟΘΕΣΙΑ"));
```

Η αναδρομική συσχέτιση ΔΕΝ θα δώσει νέο πίνακα καθώς ο λόγος πληθικότητας είναι 1:M που επιτρέπει απορρόφηση. Θα προκαλέσει όμως αλλαγή στον πίνακα ΑΙΣΘΗΤΗΡΑΣ (που την απορροφά) ως ακολούθως:

-- Δημιουργία του Πίνακα {ΑΙΣΘΗΤΗΡΑΣ}:

```
CREATE TABLE "ΑΙΣΘΗΤΗΡΑΣ"
```

```
("κωδ_αισθητήρα" INT NOT NULL,
```

```
"κωδ_αλλου_αισθητήρα" INT NULL,
```

```
"χρονικό_διάστημα" VARCHAR(30) NOT NULL,
```

```
"διάρκεια_μπαταρίας" INT,
```

```
"κωδ_εξυπηρετή" INT,
```

```
FOREIGN KEY ("κωδ_εξυπηρετή") REFERENCES "ΕΞΥΠΗΡΕΤΗΣ" ("κωδ_εξυπηρετή"),
```

```
FOREIGN KEY ("κωδ_αλλου_αισθητήρα") REFERENCES "ΑΙΣΘΗΤΗΡΑΣ" ("κωδ_αισθητήρα"),
```

```
PRIMARY KEY("κωδ_αισθητήρα"));
```

A3. Ο πίνακας που δίδεται είναι:

UPLOAD (UPID, STUDENT_ID, FILE_NAME, FILE_TYPE, FILE_SIZE, TIMESTAMP)

Για να υπολογίσουμε τον κωδικό του φοιτητή που ανέβασε το μικρότερο αρχείο σε μέγεθος μπορούμε να διατυπώσουμε τη δήλωση

```
SELECT DISTINCT STUDENT_ID  
FROM UPLOAD  
WHERE FILE_SIZE IN (SELECT MIN(FILE_SIZE) FROM UPLOAD)
```

Προσθέτοντας τον πίνακα STUDENT (STD_ID, STD_NAME, STD_SURNAME, EMAIL, PASSWORD) με τα στοιχεία των φοιτητών μπορούμε να υπολογίσουμε τα e-mails όλων των φοιτητών που βρίσκονται στη βάση, οι οποίοι δεν έχουν ανεβάσει στη βάση αρχεία τύπου «ZIP» με την ακόλουθη δήλωση Σχεσιακής Άλγεβρας:

$$\pi_{\text{EMAIL}}(\text{STUDENT}) - \pi_{\text{EMAIL}}(\sigma_{\text{FILE_TYPE}='ZIP'}(\text{STUDENT} \bowtie \text{UPLOAD}))$$

B1. Το πρώτο σκέλος του ερωτήματος σας έχει δοθεί ως ενδεικτική επίλυση ενώ οι πίνακες που δεσμεύονται με περιορισμούς αναφορικής ακεραιότητας είναι όσοι περιέχουν δήλωση FOREIGN KEY. Αναφορικά με την αλλαγή του κωδικού ενός ασθενή από μια τιμή σε μια άλλη, το ερώτημα είχε τεθεί στην Άσκηση 7, υπο-ερώτημα 5 όπου ο κωδικός ασθενή 36 θα άλλαζε σε 40. Ειδικότερα, είχαν παρουσιαστεί δύο λύσεις:

1^η Λύση (υπήρχε περιορισμός UNIQUE στον πίνακα "ΑΣΘΕΝΗΣ")

-- Dropping any UNIQUE constraint(s):

```
ALTER TABLE "ΑΣΘΕΝΗΣ"
```

```
DROP CONSTRAINT "ΑΣΘΕΝΗΣ_ΤΗΛΕΦΩΝΟ_ΑΣΘ_key";
```

-- Create a new record with the desired primary key value:

```
INSERT INTO "ΑΣΘΕΝΗΣ" VALUES (40, 'ΑΚΟΥΜΙΑΝΑΚΗΣ', 'ΔΗΜΟΣΘΕΝΗΣ', 'ΗΡΑΚΛΕΙΟ, ΚΡΗΤΗΣ', '6978122555');
```

-- Updating other tables to ensure referential constraints:

```
UPDATE "ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ"
```

```
SET "ΚΩΔΙΚΟΣ_ΑΣΘ" = 40;
```

-- Delete old tuple

```
DELETE FROM "ΑΣΘΕΝΗΣ"
```

```
WHERE "ΚΩΔΙΚΟΣ_ΑΣΘ" = 36;
```

-- Adding any UNIQUE constraint(s):

```
ALTER TABLE "ΑΣΘΕΝΗΣ"
```

```
ADD CONSTRAINT "ΤΗΛΕΦΩΝΟ_ΑΣΘ" UNIQUE ("ΤΗΛΕΦΩΝΟ_ΑΣΘ");
```

Σημείωση: Αν υποθεθεί ότι δεν υπάρχει UNIQUE constraint τότε απλά η λύση περιλαμβάνει τις υπόλοιπες δηλώσεις

2^η λύση (με BEGIN-COMMIT και εναλλαγή του session_replication_role)

```
BEGIN;
```

```
SET session_replication_role = 'replica';
```

```
UPDATE "ΑΣΘΕΝΗΣ" SET "ΚΩΔΙΚΟΣ_ΑΣΘ" = 40 WHERE "ΚΩΔΙΚΟΣ_ΑΣΘ" = 36;
```

```
UPDATE "ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ" SET "ΚΩΔΙΚΟΣ_ΑΣΘ" = 40 WHERE "ΚΩΔΙΚΟΣ_ΑΣΘ" = 36;
```

```
SET session_replication_role = 'origin';
```

```
COMMIT;
```

B2. Η δήλωση της SQL που διαγράφει από τον πίνακα ΑΣΘΕΝΗΣ όλες τις πλειάδες (των ασθενών) που δεν έχουν καταχωρημένη κάποια επίσκεψη είναι η ακόλουθη:

```
DELETE FROM "ΑΣΘΕΝΗΣ"
```

```
WHERE "ΚΩΔΙΚΟΣ_ΑΣΘ" NOT IN (  
    SELECT DISTINCT "ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ"."ΚΩΔΙΚΟΣ_ΑΣΘ"  
    FROM "ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ");
```

Σημείωση: Το ερώτημα αυτούσιο υπήρχε στην Άσκηση 7 (υπο-ερώτημα ερώτημα 4)

Ακολούθως, διατυπώστε δηλώσεις SQL [10 βαθμοί] και Σχεσιακής Άλγεβρας [10 βαθμοί] για τα ερωτήματα:

- Να βρεθούν τα ονοματεπώνυμα των ασθενών που έκαναν επίσκεψη σε νοσοκομείο της αρεσκείας σας [2+2]

Σημείωση: Το ερώτημα συμπεριλαμβάνεται στην Άσκηση 8, υπο-ερώτημα 5 για το νοσοκομείο ΑΓΙΟΣ ΑΝΔΡΕΑΣ

Δήλωση SQL

```
SELECT DISTINCT "ΕΠΩΝΥΜΟ_ΑΣΘ", "ΟΝΟΜΑ_ΑΣΘ", "ΔΙΕΥΘΥΝΣΗ_ΑΣΘ", "ΤΗΛΕΦΩΝΟ_ΑΣΘ"  
FROM "ΝΟΣΟΚΟΜΕΙΟ", "ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ", "ΑΣΘΕΝΗΣ"  
WHERE "ΑΣΘΕΝΗΣ"."ΚΩΔΙΚΟΣ_ΑΣΘ"="ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ"."ΚΩΔΙΚΟΣ_ΑΣΘ"  
AND "ΝΟΣΟΚΟΜΕΙΟ"."ΚΩΔΙΚΟΣ_ΝΟΣ"="ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ"."ΚΩΔΙΚΟΣ_ΝΟΣ"  
AND "ΝΟΣΟΚΟΜΕΙΟ"."ΟΝΟΜΑ_ΝΟΣ" = 'ΑΓΙΟΣ ΑΝΔΡΕΑΣ';
```

Δήλωση Σχεσιακής Άλγεβρας

```
π ("ΕΠΩΝΥΜΟ_ΑΣΘ", "ΟΝΟΜΑ_ΑΣΘ", "ΔΙΕΥΘΥΝΣΗ_ΑΣΘ", "ΤΗΛΕΦΩΝΟ_ΑΣΘ" (σ "ΟΝΟΜΑ_ΝΟΣ"='ΑΓΙΟΣ ΑΝΔΡΕΑΣ' ("ΝΟΣΟΚΟΜΕΙΟ" ⋈  
"ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ" ⋈ "ΑΣΘΕΝΗΣ"))
```

- Να βρεθούν το ονοματεπώνυμο των ασθενών που έχουν επισκεφθεί και το νοσοκομείο με όνομα Α και το νοσοκομείο με όνομα Β [4+4]

Σημείωση: Το ερώτημα συμπεριλαμβάνεται στην Άσκηση 9, υπο-ερώτημα 7 για τα νοσοκομεία ΑΧΕΠΑ και ΑΓΙΟΣ ΑΝΔΡΕΑΣ

Δήλωση SQL

```
SELECT "ΕΠΩΝΥΜΟ_ΑΣΘ", "ΟΝΟΜΑ_ΑΣΘ", "ΔΙΕΥΘΥΝΣΗ_ΑΣΘ", "ΤΗΛΕΦΩΝΟ_ΑΣΘ"
FROM "ΝΟΣΟΚΟΜΕΙΟ", "ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ", "ΑΣΘΕΝΗΣ"
WHERE "ΑΣΘΕΝΗΣ"."ΚΩΔΙΚΟΣ_ΑΣΘ"="ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ"."ΚΩΔΙΚΟΣ_ΑΣΘ"
AND "ΝΟΣΟΚΟΜΕΙΟ"."ΚΩΔΙΚΟΣ_ΝΟΣ"="ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ"."ΚΩΔΙΚΟΣ_ΝΟΣ"
AND "ΝΟΣΟΚΟΜΕΙΟ"."ΟΝΟΜΑ_ΝΟΣ" = 'ΑΧΕΠΑ'
INTERSECT
SELECT "ΕΠΩΝΥΜΟ_ΑΣΘ", "ΟΝΟΜΑ_ΑΣΘ", "ΔΙΕΥΘΥΝΣΗ_ΑΣΘ", "ΤΗΛΕΦΩΝΟ_ΑΣΘ"
FROM "ΝΟΣΟΚΟΜΕΙΟ", "ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ", "ΑΣΘΕΝΗΣ"
WHERE "ΑΣΘΕΝΗΣ"."ΚΩΔΙΚΟΣ_ΑΣΘ"="ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ"."ΚΩΔΙΚΟΣ_ΑΣΘ"
AND "ΝΟΣΟΚΟΜΕΙΟ"."ΚΩΔΙΚΟΣ_ΝΟΣ"="ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ"."ΚΩΔΙΚΟΣ_ΝΟΣ"
AND "ΝΟΣΟΚΟΜΕΙΟ"."ΟΝΟΜΑ_ΝΟΣ" = 'ΑΓΙΟΣ ΑΝΔΡΕΑΣ';
```

Δήλωση Σχεσιακής Άλγεβρας

π "ΕΠΩΝΥΜΟ_ΑΣΘ", "ΟΝΟΜΑ_ΑΣΘ", "ΔΙΕΥΘΥΝΣΗ_ΑΣΘ", "ΤΗΛΕΦΩΝΟ_ΑΣΘ" (σ "ΟΝΟΜΑ_ΝΟΣ"='Α' (ρ "ΝΟΣΟΚΟΜΕΙΟ" $\bowtie$ "ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ" $\bowtie$ "ΑΣΘΕΝΗΣ")) $\cap$

π "ΕΠΩΝΥΜΟ_ΑΣΘ", "ΟΝΟΜΑ_ΑΣΘ", "ΔΙΕΥΘΥΝΣΗ_ΑΣΘ", "ΤΗΛΕΦΩΝΟ_ΑΣΘ" (σ "ΟΝΟΜΑ_ΝΟΣ"='Β' (ρ "ΝΟΣΟΚΟΜΕΙΟ" $\bowtie$ "ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ" $\bowtie$ "ΑΣΘΕΝΗΣ"))

- Να βρεθούν τα νοσοκομεία που δεν έχουν δεχθεί καμία επίσκεψη [4+4]

Σημείωση: Το ερώτημα συμπεριλαμβάνεται αυτούσιο στην Άσκηση 9, υπο-ερώτημα 8

Δήλωση SQL

```
SELECT "ΝΟΣΟΚΟΜΕΙΟ"."ΚΩΔΙΚΟΣ_ΝΟΣ", "ΟΝΟΜΑ_ΝΟΣ"
FROM "ΝΟΣΟΚΟΜΕΙΟ"
EXCEPT
SELECT "ΝΟΣΟΚΟΜΕΙΟ"."ΚΩΔΙΚΟΣ_ΝΟΣ", "ΟΝΟΜΑ_ΝΟΣ"
FROM "ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ", "ΝΟΣΟΚΟΜΕΙΟ"
WHERE "ΝΟΣΟΚΟΜΕΙΟ"."ΚΩΔΙΚΟΣ_ΝΟΣ"="ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ"."ΚΩΔΙΚΟΣ_ΝΟΣ"
GROUP BY ΝΟΣΟΚΟΜΕΙΟ.ΚΩΔΙΚΟΣ_ΝΟΣ, ΟΝΟΜΑ_ΝΟΣ;
```

Δήλωση Σχεσιακής Άλγεβρας

π "ΚΩΔΙΚΟΣ_ΝΟΣ", "ΟΝΟΜΑ_ΝΟΣ" ("ΝΟΣΟΚΟΜΕΙΟ") $-$

π "ΚΩΔΙΚΟΣ_ΝΟΣ", "ΟΝΟΜΑ_ΝΟΣ" ("ΝΟΣΟΚΟΜΕΙΟ" $\bowtie$ "ΕΠΙΣΚΕΨΗ_ΑΣΘΕΝΗ")