

1	
2	
3	
ΣΥΝ	

ΟΝΟΜΑ \_\_\_\_\_

Αρ. Μητρώου \_\_\_\_\_

ΕΘΝΙΚΟ ΜΕΤΣΟΒΙΟ ΠΟΛΥΤΕΧΝΕΙΟ  
ΤΜΗΜΑ ΗΛΕΚΤΡΟΛΟΓΩΝ ΜΗΧΑΝΙΚΩΝ ΚΑΙ ΜΗΧΑΝΙΚΩΝ ΥΠΟΛΟΓΙΣΤΩΝ  
ΤΟΜΕΑΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

## ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ ΕΞΕΤΑΣΗ 2011-2012 (Σεπτεμβρίου)

**ΣΗΜΕΙΩΣΗ:** Οι απαντήσεις δίνονται στα κενά μεταξύ ερωτήσεων !

Διδάσκων: Ι. Βασιλείου

### ΘΕΜΑ 1.- ΣΧΕΣΙΑΚΟ ΜΟΝΤΕΛΟ ΚΑΙ ΓΛΩΣΣΕΣ [30]

Θεωρήστε την παρακάτω βάση δεδομένων

ΠΙΛΟΤΟΣ(Όνομα, Πόλη, Ώρες-Πτήσης)  
ΠΕΤΑΕΙ(Όνομα, Αεροσκάφος, Μισθός)  
ΑΝΗΚΕΙ(Αεροσκάφος, Αεροπορική-Εταιρεία, Πόλη-Εγγραφής)

#### (α) Γράψτε στη γλώσσα SQL τα ερωτήματα (queries). [10]

- i.- Βρες τα ονόματα των πιλότων που πετούνε αεροσκάφη με πόλη εγγραφής την ίδια με αυτή που μένουν.
- ii.- Βρες τα ονόματα των πιλότων που έχουν μεγαλύτερο μισθό από κάθε πιλότο της εταιρείας BLA

#### ΛΥΣΗ:

i.

```
SELECT      P.Όνομα
FROM        ΠΙΛΟΤΟΣ as P, ΠΕΤΑΕΙ as Q, ΑΝΗΚΕΙ as A
WHERE       (P.Όνομα = Q.Όνομα) AND (Q.Αεροσκάφος = A. Αεροσκάφος) AND
            (P.Πόλη = A.Πόλη-Εγγραφής)
```

ii.

```
SELECT DISTINCT      P.Όνομα
FROM        ΠΕΤΑΕΙ as P
WHERE       P.Μισθός > (SELECT      max(R.Μισθός)
                        FROM        ΠΕΤΑΕΙ as R, ΑΝΗΚΕΙ as A
                        WHERE       (R.Αεροσκάφος = A. Αεροσκάφος)
                        AND         (A.Αεροπορική-Εταιρεία = 'BLA')
                        GROUYP BY   R. Όνομα
                        )
```

//Αν κάθε ΠΙΛΟΤΟΣ πετάει περισσότερα από 1 Αεροπλάνα και για καθένα παίρνει διαφορετικό μισθό... θέλουμε το άθροισμα; //

```
SELECT DISTINCT P.Όνομα
FROM ΠΕΤΑΕΙ as P
GROUP BY P.Όνομα
HAVING sum(Q.Μισθός) > all(SELECT sum(R.Μισθός) as TotalM
                           FROM ΠΕΤΑΕΙ as R, ΑΝΗΚΕΙ as A
                           WHERE (R.Αεροσκάφος = A. Αεροσκάφος)
                           AND (A.Αεροπορική-Εταιρία = 'BLA')
                           GROYP BY R. Όνομα
                           )
```

// ή

```
SELECT DISTINCT P.Όνομα
FROM ΠΕΤΑΕΙ as P
GROUP BY P.Όνομα
HAVING sum(Q.Μισθός) > (SELECT max(TotalM)
                        FROM (SELECT sum(R.Μισθός) as TotalM
                              FROM ΠΕΤΑΕΙ as R, ΑΝΗΚΕΙ as A
                              WHERE (R.Αεροσκάφος = A. Αεροσκάφος)
                              AND (A.Αεροπορική-Εταιρία = 'BLA')
                              GROYP BY R. Όνομα
                              )
                        )
```

(β) [10] Γράψτε σε σχεσιακή άλγεβρα τα ερωτήματα (queries).

- i.- Βρες τα ονόματα των πιλότων που πετούνε αεροσκάφη με πόλη εγγραφής την ίδια με αυτή που μένουν.
- ii.- Βρες τα ονόματα των πιλότων που έχουν μεγαλύτερο μισθό από 200000 και πετούνε το αεροσκάφος ALB.

---

ΛΥΣΗ:

i.

$\pi_{\text{ΠΙΛΟΤΟΣ.Όνομα}} ( \sigma_{(\text{ΠΙΛΟΤΟΣ.Πόλη} = \text{ΑΝΗΚΕΙ.Πόλη-Εγγραφής})} ( \text{ΠΙΛΟΤΟΣ} \bowtie \text{Όνομα ΠΕΤΑΕΙ} \bowtie \text{Αεροσκάφος ΑΝΗΚΕΙ} ) )$

ii.

$\delta ( \pi_{\text{ΠΕΤΑΕΙ.Όνομα}} ( \sigma_{(\text{ΑΝΗΚΕΙ.Εταιρία} = \text{'BLA'}) \wedge (\text{ΠΕΤΑΕΙ.Μισθός} > 200000)} ( \text{ΠΕΤΑΕΙ} \bowtie \text{Αεροσκάφος ΑΝΗΚΕΙ} ) ) )$

//ή με division, αντί για select distinct:

$( \pi_{\text{ΠΕΤΑΕΙ.Όνομα}} ( \sigma_{(\text{ΠΕΤΑΕΙ.Μισθός} > 200000)} \text{ΠΕΤΑΕΙ} ) ) \div ( \pi_{\text{ΑΝΗΚΕΙ.Αεροσκάφος}} ( \sigma_{(\text{ΑΝΗΚΕΙ.Εταιρία} = \text{'BLA'})} \text{ΑΝΗΚΕΙ} ) )$

**(γ) [10] Θεωρείστε την παραπάνω Βάση Δεδομένων:**

- i. Δώστε ένα παράδειγμα μίας χρήσιμης όψης / προβολής (VIEW) στη σχέση ΠΙΛΟΤΟΣ (χρησιμοποιώντας την SQL), στην οποία επιτρέπεται να γίνουν μεταβολές (ενημερώσεις, διαγραφές, εισαγωγές).
  - ii. Δώστε ένα παράδειγμα μίας χρήσιμης όψης στη σχέση ΠΙΛΟΤΟΣ, στην οποία ΔΕΝ επιτρέπεται να γίνουν ενημερώσεις σε κάποιο γνώρισμα.
- 

**ΛΥΣΗ:**

i.

**CREATE VIEW V1 as**

```
SELECT      P.Όνομα, P.Ώρες-Πτήσης  
FROM        ΠΙΛΟΤΟΣ as P  
WHERE       (P.Ώρες-Πτήσης > 50)
```

ii.

**CREATE VIEW V2 as**

```
SELECT      P.Όνομα, P.Πόλη, count(Q.Αεροσκάφος), sum(Q.Μισθός)  
FROM        ΠΙΛΟΤΟΣ as P, ΠΕΤΑΕΙ as Q  
WHERE       (P.Όνομα = Q.Όνομα) AND (Q.Μισθός > 5000)  
GROUP BY   P.Όνομα, P.Πόλη
```

**ΘΕΜΑ 2 – ΓΕΝΙΚΕΣ ΕΡΩΤΗΣΕΙΣ -ΚΑΝΟΝΙΚΟΠΟΙΗΣΗ-ΦΥΣΙΚΟΣ ΣΧΕΔΙΑΣΜΟΣ -  
ΒΕΛΤΙΣΤΟΠΟΙΗΣΗ [35]**

**(α) [12] Κανονικοποίηση**

Θεωρείστε το Σχήμα R με τα γνωρίσματα (attributes): A, B, C, D, E, G, H και τις λειτουργικές εξαρτήσεις (functional dependencies) που ισχύουν στο R :

$$F = \{ AB \rightarrow C, AC \rightarrow B, AD \rightarrow E, B \rightarrow D, BC \rightarrow A, E \rightarrow G \}$$

Για κάθε ένα από τα παρακάτω σύνολα γνωρισμάτων

- (i) {A, B, C }
- (ii) {A, B, C, E, G}
- (iii) {A, C, E, H }

να κάνετε **τρία πράγματα**:

- (1) Υπολογίστε το σύνολο των εξαρτήσεων που ισχύουν για το συγκεκριμένο σύνολο γνωρισμάτων και διατυπώστε μια (ελάχιστη) κανονική κάλυψη του
- (2) Αναγνωρίστε αν το σύνολο γνωρισμάτων ικανοποιεί την κανονική μορφή BCNF,
- (3) αν το σύνολο δεν είναι σε BCNF, τότε επιδιώξτε μια αποσύνθεση σε ένα σύνολο από BCNF σχήματα που διατηρούν τις εξαρτήσεις.

Υπενθυμίζεται ο ορισμός της κανονικής μορφής BCNF (εάν και μόνο εάν για κάθε  $X \rightarrow Y$  που ισχύει τότε ένα εκ των δύο συμβαίνει: είτε το  $X \rightarrow Y$  είναι τετριμμένη λειτουργική εξάρτηση ή το X είναι υπερκλειδί)

---

**ΛΥΣΗ**

- (i) {A, B, C }**

- 1.  $AB \rightarrow C, AC \rightarrow B, BC \rightarrow A$  - ήδη είναι μια ελάχιστη κάλυψη**
- 2. Είναι σε BCNF διότι τα AB, AC και BC είναι όλα υποψήφια κλειδιά στη σχέση.**
- 3. Είναι ήδη BCNF.**

- (ii) {A, B, C, E, G}**

- 1.  $AB \rightarrow C, AC \rightarrow B, BC \rightarrow A, E \rightarrow G$ . - ήδη είναι μια ελάχιστη κάλυψη.**
- 2. Τα ABE, ACE και BCE είναι όλα υποψήφια κλειδιά. Αλλά δεν είναι ούτε 2NF γιατί το E είναι υποσύνολο των κλειδίων και ισχύει η  $E \rightarrow G$ . Είναι σε 1NF.**
- 3. Η αποσύνθεση σε ABC, ABE και EG είναι BCNF.**

(iii) {A, C, E, H}

1. Δεν υπάρχουν λειτουργικές εξαρτήσεις – ήδη μια ελάχιστη κάλυψη.
2. Το κλειδί είναι η ίδια η ACEH
3. Είναι σε μορφή BCNF.

(β) [13] Σχεδίαση Φυσικού Σχήματος

Θεωρήστε ένα σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων που περιέχει τις εξής δύο σχέσεις: **Athlete** (αθλητές) και **Team** (ομάδες).

**Athlete**(id, name, age, salary, tname)

**Team**(tname, league)

Ο πίνακας **Athlete** έχει 1000 πλειάδες και κάθε πλειάδα έχει σταθερό μήκος 500 bytes. Το πρωτεύον κλειδί “id” έχει μήκος 40 bytes. Ο πίνακας **Team** έχει 100 πλειάδες και κάθε πλειάδα έχει μήκος 200 bytes. Το πρωτεύον κλειδί “tname” έχει μήκος 20 bytes. Κάθε μπλοκ έχει μέγεθος 8K (8192) bytes, από τα οποία τα 192 bytes δεσμεύονται για την κεφαλίδα του μπλοκ. Μια εγγραφή δε μπορεί να επεκτείνεται σε περισσότερα του ενός μπλοκ. Όλες οι εγγραφές του πίνακα **Athlete** αποθηκεύονται σειριακά στο δίσκο, βάσει του id. Ομοίως, όλες οι εγγραφές του πίνακα **Team** αποθηκεύονται σειριακά στο δίσκο, βάσει του tname. Υποθέστε τα εξής:

- Κάθε δείκτης σε μπλοκ έχει μέγεθος 6 bytes.
- Κάθε δείκτης σε εγγραφή έχει μέγεθος 8 bytes.
- Μια εγγραφή ευρετηρίου δεν επεκτείνεται σε περισσότερα του ενός μπλοκ.

(β1) (9 μονάδες) Υποθέστε ότι θέλουμε να δημιουργήσουμε ένα πρωτεύον ευρετήριο INDEX1 στο Athlete.id.

(i) Χρησιμοποιώντας ένα πυκνό (dense) ευρετήριο, ποιος είναι ο ελάχιστος αριθμός μπλοκ που απαιτούνται για το INDEX1;

(ii) Χρησιμοποιώντας ένα αραιό (sparse) ευρετήριο, ποιος είναι ο ελάχιστος αριθμός μπλοκ που απαιτούνται για το INDEX1;

**Λύση (β1):**

(i) Κάθε εγγραφή του πυκνού ευρετηρίου καταλαμβάνει  $40+8=48$  bytes. Άρα σε ένα μπλοκ χωράνε  $\text{floor}((8192-192)/48)=166$  εγγραφές ευρετηρίου και συνεπώς ο ελάχιστος αριθμός μπλοκ που απαιτούνται για το ευρετήριο είναι  $\text{ceiling}(1000/166) = 7$  μπλοκ.

(ιι) Στην περίπτωση αραιού ευρετηρίου, κάθε εγγραφή καταλαμβάνει  $40+6=46$  bytes. Άρα σε ένα μπλοκ χωράνε  $\text{floor}((8192-192)/46)=173$  εγγραφές ευρετηρίου.

Επειδή έχουμε αραιό ευρετήριο, θα έχουμε μια εγγραφή ευρετηρίου για κάθε μπλοκ της σχέσης Athlete. Σε ένα μπλοκ χωράνε  $(8192-192)/500=16$  πλειάδες της σχέσης Athlete. Άρα, για να αποθηκεύσουμε ολόκληρη τη σχέση Athlete, χρειαζόμαστε  $\text{ceiling}(1000/16)=63$  μπλοκ. Συνεπώς ο ελάχιστος αριθμός μπλοκ που απαιτούνται για το αραιό ευρετήριο είναι  $\text{ceiling}(63/173)=1$  μπλοκ.

(β2) (4 μονάδες) Υποθέστε ότι θέλουμε να δημιουργήσουμε ένα δευτερεύον ευρετήριο INDEX2 στο Team. league.

(ι) Θα είχε νόημα να είναι το INDEX2 πυκνό (dense) ευρετήριο; Εξηγήστε την απάντησή σας.

(ιι) Θα είχε νόημα να είναι το INDEX2 αραιό (sparse) ευρετήριο; Εξηγήστε την απάντησή σας.

### Λύση (β2):

(ι) Ναι, δεδομένου ότι πρόκειται για δευτερεύον ευρετήριο σε μη πρωτεύον γνώρισμα, θα χρειαζούμαστε ένα δείκτη ευρετηρίου για κάθε πλειάδα της σχέσης.

(ιι) Όχι, διότι οι εγγραφές του πίνακα Team δεν είναι αποθηκευμένες στο δίσκο σειριακά βάσει του γνωρίσματος Team.league, αλλά βάσει του Team.tname.

### (γ) [10] Βελτιστοποίηση Ερωτημάτων

Θεωρείστε το παρακάτω σχεσιακό σχήμα (με τα πρωτεύοντα κλειδιά υπογραμμισμένα):

**MUSICIAN** (mid, name, instrument, city, age, company)

**SONG** (sid, title, artist, composer, genre)

**PARTICIPATES** (mid, sid, recording\_date)

Όπου τα *mid* και *sid* είναι ξένα κλειδιά αναφερόμενα στις **MUSICIAN** και **SONG** αντίστοιχα.

Αυτές οι σχέσεις έχουν τα ακόλουθα χαρακτηριστικά:

- Η σχέση **MUSICIAN** έχει 20.000 πλειάδες με 40 πλειάδες σε κάθε σελίδα δίσκου.
- $NKeys(instrument)$  για τη σχέση **MUSICIAN** είναι 200 (δηλαδή υπάρχουν 200 διακριτές τιμές για το γνώρισμα instrument).
- $NKeys(age)$  για τη **MUSICIAN** είναι 50.
- Η σχέση **SONG** έχει 1.000 πλειάδες με 80 πλειάδες ανά σελίδα.
- Η σχέση **PARTICIPATES** έχει 80.000 πλειάδες με 100 πλειάδες ανά σελίδα.

Έχουμε ορίσει τα παρακάτω ευρετήρια:

- Ένα ευρετήριο κατακερματισμού στο *mid* της **MUSICIAN**.
- Ένα συσταδοποιημένο B+ δεντρικό ευρετήριο στο *instrument* της **MUSICIAN**.
- Ένα μη-συσταδοποιημένο B+ δεντρικό ευρετήριο στο *age* της **MUSICIAN**.

Υποθέστε ότι μια καταχώρηση ευρετηρίου έχει το μισό μέγεθος μιας πλειάδας δεδομένων.

**(γ1) (5 μονάδες)** Δώστε (i) το **πιο αποδοτικό πλάνο** εκτέλεσης του παρακάτω ερωτήματος και (ii) το κόστος σε αριθμό σελίδων που προσπελούνται για την εκτέλεση του πλάνου. Στην απάντησή σας διευκρινίστε εάν και ποιο(-α) ευρετήριο(-α) χρησιμοποιείται(-ούνται). Περιγράψτε λεπτομερώς όποια επιπλέον υπόθεση κάνετε.

Query a:       SELECT \*  
                  FROM MUSICIAN  
                  WHERE instrument = 'Guitar' AND age = 30

### Λύση (γ1):

Υποθέστε μια ομοιόμορφη κατανομή των μουσικών σε μουσικά όργανα και σε ηλικιακές ομάδες. Η σχέση **MUSICIAN** έχει  $\text{ceiling}(20.000/40)=500$  σελίδες. Η επιλογή στο γνώρισμα *instrument* έχει την μεγαλύτερη επιλεκτικότητα ( $1/200$ ) και αφού το αντίστοιχο ευρετήριο είναι συσταδοποιημένο αρκεί να διαβάσουμε ένα πολύ μικρό μέρος των πλειάδων της σχέσης  $\text{ceiling}(500/200)$ , συν το κόστος να διατρέξουμε το δεντρικό ευρετήριο από την ρίζα και να προσπελάσουμε κάποιες από τις σελίδες-φύλλα του.

Πλάνο: σάρωση του ευρετηρίου (index scan) στο *instrument* με «on the fly» φιλτράρισμα στο *age*.

Μέγεθος του B+ δέντρου στο *instrument*: Επειδή έχουμε συσταδοποιημένο ευρετήριο, η **MUSICIAN** αναγκαστικά πρέπει να είναι ταξινομημένη ως προς το *instrument*. Έτσι το B+ δεντρικό ευρετήριο έχει 200 δείκτες προς τις σελίδες της σχέσης (ένα δείκτη ευρετηρίου προς κάθε ξεχωριστή τιμή του γνωρίσματος). Δεδομένου ότι μια καταχώρηση ευρετηρίου έχει το μισό μέγεθος μιας πλειάδας δεδομένων, σε κάθε σελίδα χωράνε 80 Index entries. Επομένως το B+ δένδρο θα έχει  $\text{ceiling}(200/80) = 3$  φύλλα και 1 ρίζα.

Total cost:  $2 + \text{ceiling}(500/200) = 5$ .

**(γ2) (5 μονάδες)** Υποθέτοντας ομοιόμορφη κατανομή των τιμών και ανεξαρτησία

των στηλών, εκτιμήστε τον αριθμό των πλειάδων που επιστρέφονται από το παρακάτω ερώτημα.

Query b:       SELECT \*  
                  FROM MUSICIAN M, SONG S, PARTICIPATES P  
                  WHERE M.mid = P.mid AND S.sid = P.sid

### Λύση (γ2):

Η μέγιστη πληθικότητα της επερώτησης είναι  $|M| \times |S| \times |P|$   
 $= 20.000 \times 1.000 \times 80.000 = 16 \times 10^{11}$

Reduction Factors:  $1/20000$  for  $M.mid = P.mid$ ,  $1/1000$  for  $S.sid = P.sid$

Άρα, ο αριθμός των επιστρεφόμενων πλειάδων είναι  $(1/20.000)(1/1.000)(16 \times 10^{11})$   
 $= 80.000$ .

Μια απλούστερη λύση είναι να θεωρήσουμε μια σύζευξη βασισμένη σε εξωτερικά κλειδιά: για κάθε πλειάδα στη PARTICIPATES, υπάρχει μόνο μια πλειάδα που ταιριάζει από τη MUSICIAN και μια από τη SONG. Άρα το μέγεθος του αποτελέσματος είναι **80.000**.



### ΘΕΜΑ 3. -- ΜΟΝΤΕΛΛΟΠΟΙΗΣΗ [35]

Θέλετε να δημιουργήσετε μια βάση δεδομένων για την Βιομηχανία του ΚΙΝΗΜΑΤΟΓΡΑΦΟΥ. Θα πρέπει να αποθηκευτούν πληροφορίες για:

- τις ΤΑΝΙΕΣ που δημιουργούνται (π.χ., Τίτλος, Έτος Παραγωγής, Είδος Ταινίας, Διάρκεια σε λεπτά, κλπ.)
- τους ΗΘΟΠΟΙΟΥΣ (π.χ., όνομα, διεύθυνση, έτος γέννησης,, κλπ)
- τις ΕΤΑΙΡΕΙΕΣ ή ΣΤΟΥΝΤΙΟ στις οποίες ανήκουν οι ταινίες (π.χ., Όνομα, Διεύθυνση, Πρόεδρος, κλπ.)

- i. Δώστε το διάγραμμα οντοτήτων-συσχετίσεων για τη βάση αυτή. Ποια είναι τα κλειδιά για τη κάθε οντότητα?
- ii. Επεκτείνοντας τις πληροφορίες για τη Βάση Δεδομένων, θεωρείστε την έννοια του ΣΥΜΒΟΛΑΙΟΥ μιας εταιρείας με έναν ηθοποιό για μια ταινία . Πως θα το παρουσιάσετε στο παραπάνω διάγραμμα? Σαν μια οντότητα ή μια συσχέτιση (μεταξύ ποιών οντοτήτων)? Πως γίνεται τώρα το διάγραμμα?
- iii. Προσπαθήσετε να επεκτείνετε ακόμη περισσότερο τη Βάση Δεδομένων εισάγοντας και νέες (λογικές και χρήσιμες) έννοιες. Για παράδειγμα, πως παρίστανται οι «συνέχειες» (sequel) των ταινιών (π.χ., Άρχοντας των Δακτυλιδιών 1 μετά το 2, κλπ.) Πως μπορούμε να χρησιμοποιήσουμε ISA ιεραρχίες (π.χ., είδη ταινιών), κλπ.
- iv. Δώστε την αντίστοιχη περιγραφή (schema) ολόκληρης της Βάσης Δεδομένων στο σχεσιακό μοντέλο.

Να γραφούν οι οποιεσδήποτε παραδοχές που κάνετε για τα παραπάνω (π.χ., “μια ταινία ανήκει το πολύ σε ένα στούντιο ”, κλπ.)

---

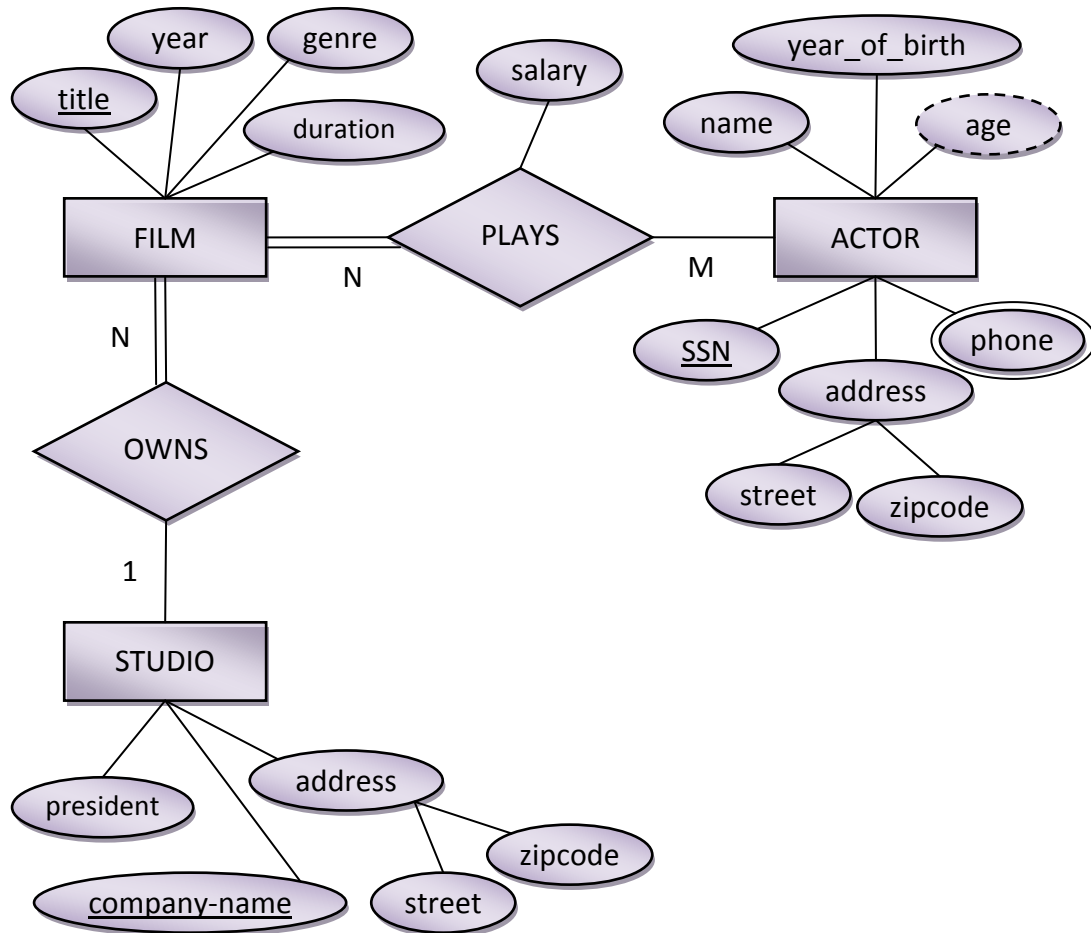
### ι. E-R

Τα κλειδιά είναι

FILM: title

ACTOR: ssn

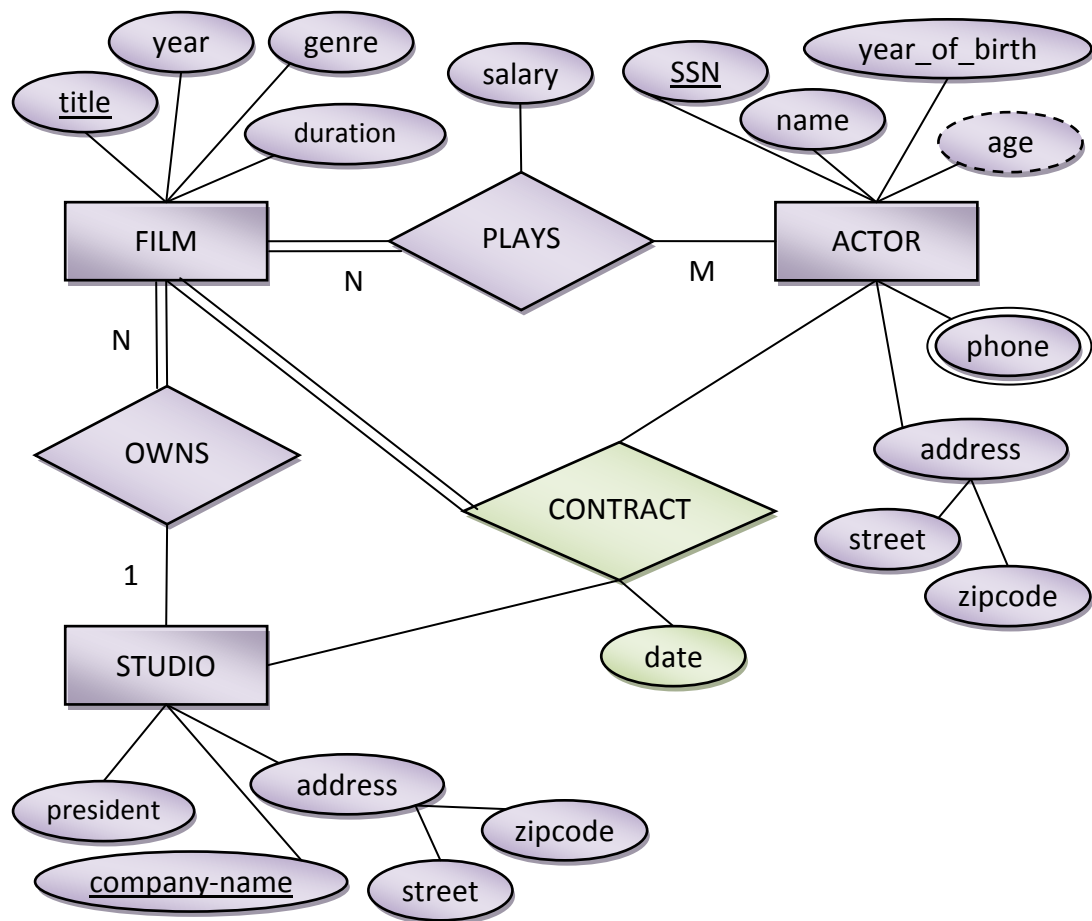
STUDIO: company-name.



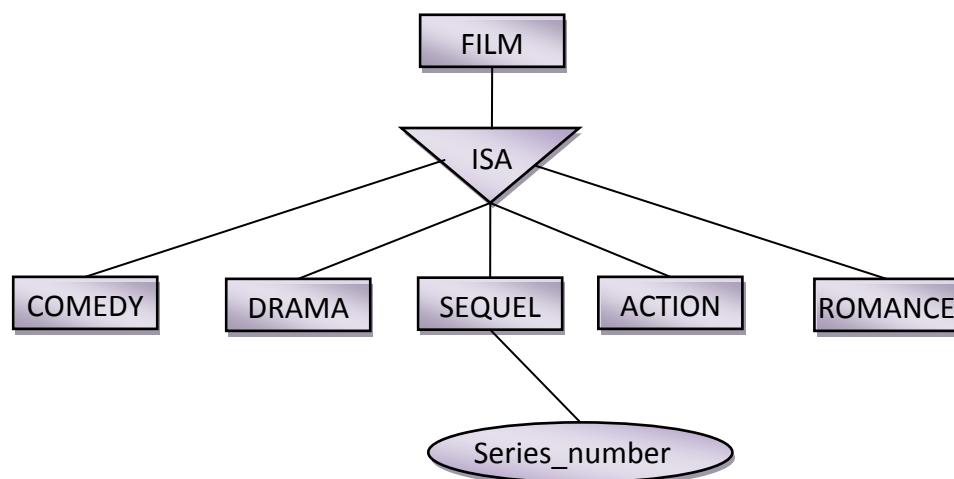
### ΠΑΡΑΔΟΧΕΣ:

- Σε όλες τις ταινίες παίζουν ηθοποιοί.
- Κάθε ταινία ανήκει σε 1 εταιρία (studio).
- Υπάρχουν ηθοποιοί που δεν έχουν παίξει ακόμα σε ταινία.
- Ένας ηθοποιός μπορεί να έχει παίξει σε πολλές ταινίες.
- Σε μια ταινία μπορούν να παίζουν πολλοί ηθοποιοί.
- Ο τίτλος κάθε ταινίας είναι μοναδικός.
- Το όνομα εταιρίας είναι μοναδικό.

ii. Συσχέτιση μεταξύ FILM, ACTOR, STUDIO.



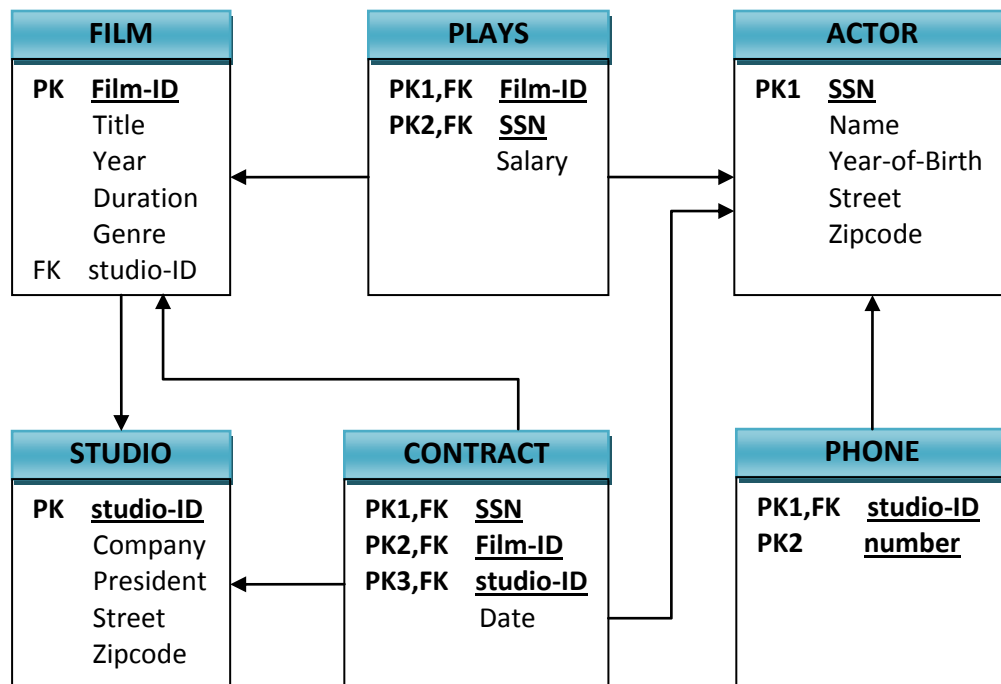
iii. ISA για το είδος των ταινιών:



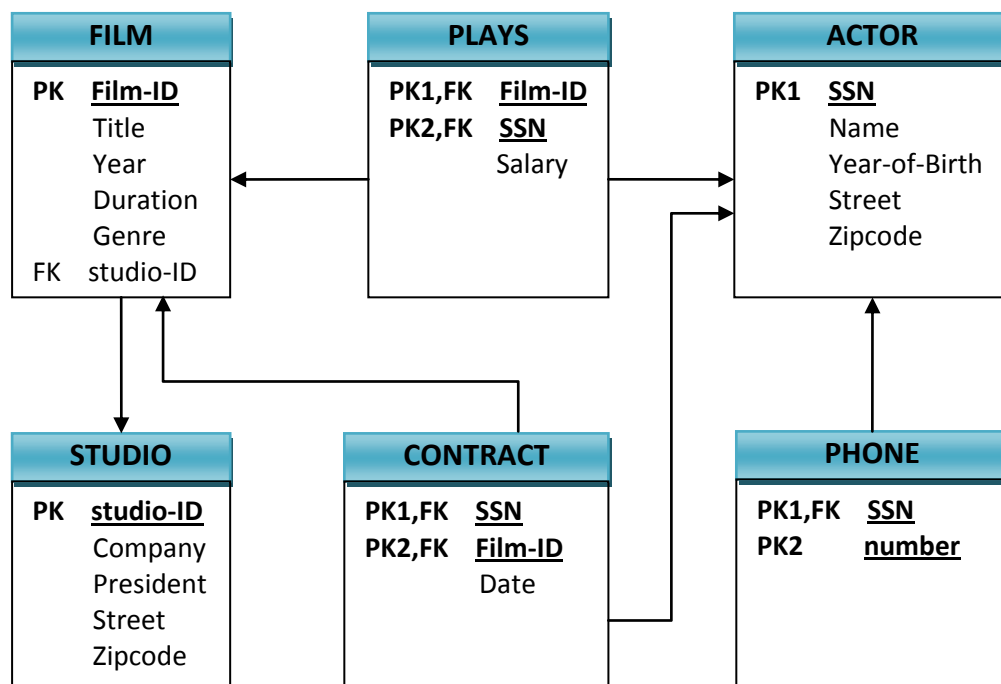
ΠΑΡΑΔΟΧΕΣ:

- Η συμμετοχή του FILM στην ISA είναι ολική.
- Οι κατηγορίες δεν είναι ξένες μεταξύ τους (disjoint). Μια ταινία μπορεί να ανήκει σε περισσότερες από μία (π.χ. SEQUEL και ACTION).

#### iv. RELATIONAL



*Or: since each FILM belongs to **only one** STUDIO:*



Thus, “**PLAYS**” and “**CONTRACT**” could be combined in one relation:

