



# Σχολή Ηλεκτρολόγων Μηχανικών & Μηχανικών Υπολογιστών ΕΜΠ

Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών

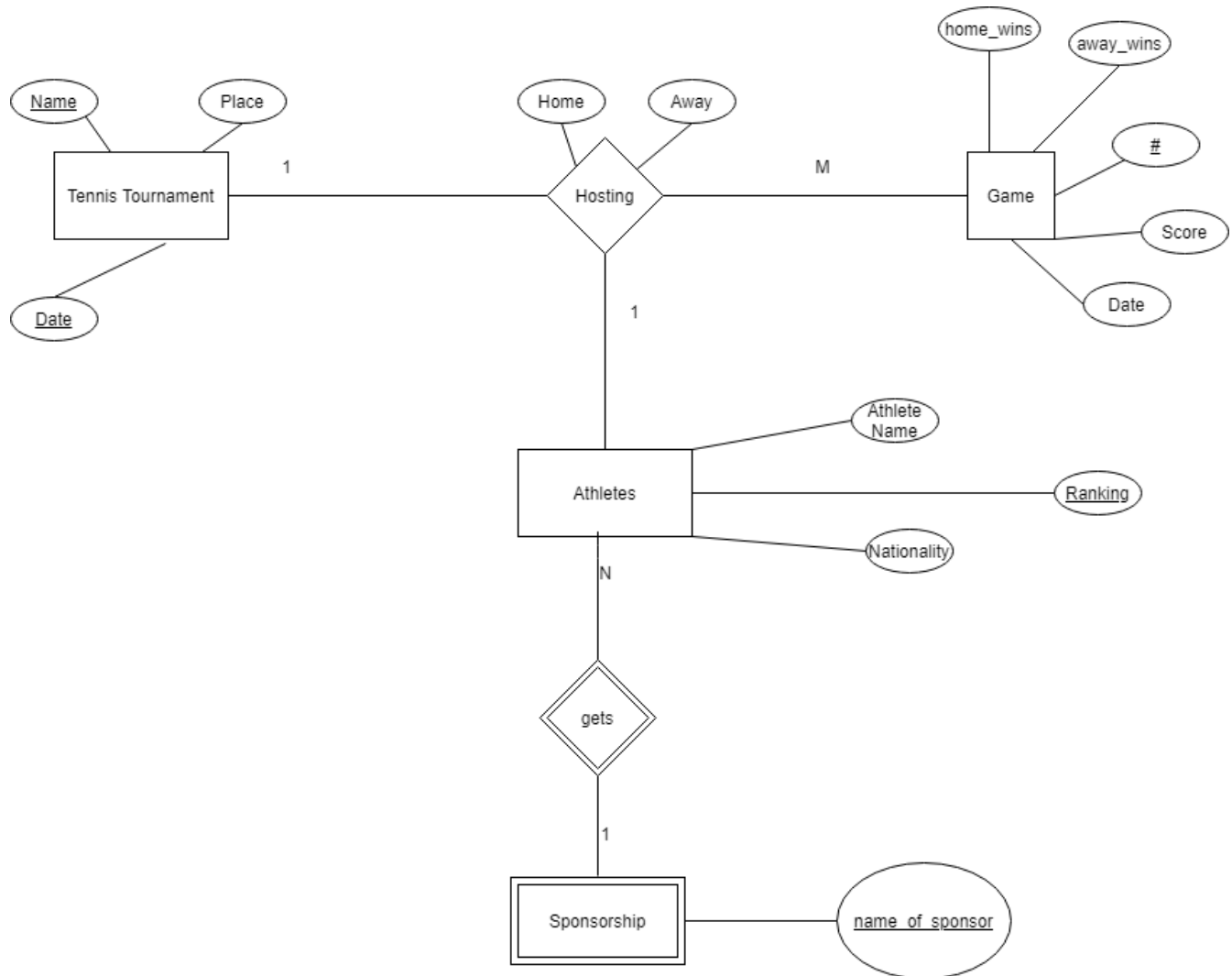
## Βάσεις Δεδομένων Σειρά Ασκήσεων

Συμμετέχοντες:

- 1) Παναγιώτα Μικαέλα Ευλιά – 03118859
- 2) Εμμανουήλ Αναστάσιος Σερλής – 03118125
- 3) Βασιλική Χανή – 03118213

## Άσκηση 1:

A) Το ζητούμενο ER είναι το κάτωθι:



## Σχόλια-Παρατηρήσεις:

1) Κάθε διαφορετικό τουρνουά μπορεί να λαμβάνει χώρα μέχρι και μία φορά τον χρόνο. Με αυτόν τον τρόπο, ο συνδυασμός των attributes name και date αποτελεί ικανό κλειδί του Tennis Tournament

2) Τα attributes home και away της σχέσης hosting είναι boolean, και λαμβάνουν την τιμή 1, αν ο αθλητής του εν λόγω παιχνιδιού βρίσκεται εντός ή εκτός έδρας αντίστοιχα. Διαφορετικά λαμβάνουν την τιμή 0

3) Ομοίως με την παρατήρηση (2), τα attributes home\_wins και away\_wins της οντότητας Games είναι boolean και λαμβάνουν την τιμή 1, αν νίκησε ο εντός ή ο εκτός έδρας παίκτης αντίστοιχα. Διαφορετικά λαμβάνουν την τιμή 0

4) Θεωρούμε ότι το ranking ενός αθλητή είναι ετήσιο, καθώς και ότι δεν γίνεται δύο διαφορετικοί αθλητές να έχουν το ίδιο ranking. Έτσι, το εν λόγω attribute αρκεί για κλειδί της οντότητας Athletes

5) Με βάση την παραπάνω υλοποίηση, η σχέση Hosting θα έχει 2 tuples ανά αγώνα, μιας και σε κάθε αγώνα συμμετέχουν 2 αθλητές

B) Το αντίστοιχο σχεσιακό σχήμα:

Tennis Tournament(Date,Name,Place)  
 Games(#,Date,Score,home\_wins,away\_wins)  
 Athletes(Ranking,Name,Nationality)  
 Sponshorship(name of sponsor)  
 Hosting(Tennis Tournament.Name,Tennis  
Tournament.Date,#,Ranking,Athletes.Name,home,away,score  
 )  
 Gets(Ranking,Athletes.name,name\_of\_sponsor)  
 Home\_Results(#,home\_wins)  
 Away\_Results(#,away\_wins)  
 Participants\_of\_Tournament(Tennis\_Tournament.Name,Tennis  
Tournament.Date,Ranking,Athletes.Name)

Σχόλια-Παρατηρήσεις:

- 1) Πέρα από τους πίνακες των οντοτήτων και των σχέσεων που φαίνονται στο ER του ερωτήματος Α, προστέθηκαν δύο ακόμα πίνακες, οι Home\_Results και Away\_Results, ώστε να αποτυπωθεί η επίδραση της έδρας στην έκβαση των αποτελεσμάτων, καθώς και ο πίνακας Participants για να υπάρχει η λίστα συμμετεχόντων αθλητών ανά τουρνουά
- 2) Θεωρούμε ότι ο εντός έδρας παίκτης έχει το πλεονέκτημα να παίζει σε γήπεδο της επιλογής του (πιθανόν στο κοντινότερο ως προς τον τόπο κατοικίας του, ώστε να μειωθεί ο χρόνος μετάβασης)

## Άσκηση 2:

Q1:

Google\_Employees  $\leftarrow$

$\sigma_{\text{Company.companyname}='Google'}[\text{Person} \bowtie \text{Company}]$

Facebook\_shareholders  $\leftarrow$

$\sigma_{\text{Company.companyname}='Facebook'}[\text{Shares} \bowtie \text{Company}]$

union  $\leftarrow$  Google\_Employees  $\times$  Facebook\_shareholders

result\_Q1  $\leftarrow$

$\Pi_{\text{pid}}[\sigma_{(\text{Google\_Employees.pid}=\text{Facebook\_shareholders.pid}) \wedge (\text{Facebook\_shareholders.sharenum} > 500)}$   
[union]

Q2:

result\_Q2  $\leftarrow$

$\Pi_{\text{person.pid}}[\sigma_{(\text{Person.managerid}=\text{Shares.pid}) \wedge (\text{Shares.cid}=\text{Person.cid}) \wedge (\text{Sharenum} > 0)}[\text{Person} \times \text{Shares}]]$

Q3:

$$\text{result\_Q3} \leftarrow \Pi_{\text{pid}}[\Pi_{\text{pid}}(\text{Person})_{(\text{Person.pid}=\text{Shares.pid})} \bowtie_{(\text{pid} \text{ } g_{(\text{count}(\text{cid}) \geq 3)} \\ (\sigma_{(\text{sharenum} > 0)}(\text{Shares})))}]$$

Q4:

$$\text{result\_Q4} \leftarrow \Pi_{\text{pid}}[[\Pi_{\text{pid,cid}} \text{Shares} / \Pi_{\text{cid}} \text{Company}] \\ \bowtie_{\text{Shares.pid}=\text{Company.pid}} \text{Person}]$$

### Άσκηση 3:

Q1:

Ερμηνεία: Βρες τα καταστήματα(όνομα και κωδικός) που είτε είναι στην Αθήνα είτε έχουν λιγότερους από 100 εργαζομένους

SQL:

```
SELECT storeid, sname
FROM Store
WHERE employee_number <= 100 OR city = 'Αθήνα'
```

Q2:

Ερμηνεία: Βρες τα ονόματα των καταστημάτων που διαθέτουν μολύβια

SQL:

```
SELECT st.sname
FROM Store as st, Goods as g, Supply as sup
WHERE [g.gname='μολύβι' AND g.gid=sup.gid AND
st.storeid=sup.storeid]
```

Q3:

Ερμηνεία: Βρες το όνομα και την πόλη των καταστημάτων που παρέχουν τα ίδια προϊόντα με το κατάστημα με gid='0808'

SQL:

D: #DIVISION

```
SELECT * FROM Supply
WHERE x not in ( SELECT x FROM (
(SELECT x , y FROM (select y from S ) as p cross join
(select distinct x from Supply) as sp)
EXCEPT
(SELECT x , y FROM Supply) ) AS r )
```

S: PARONOMASTIS

```
SELECT gid
FROM Supply
WHERE storeid='0808'
```

#natural join+result

```
SELECT sname,city
FROM D,Store
WHERE D.storeid=Store.storeid
```

B.

Q4:

```
SELECT DISTINCT TOP 5 MAX (quantity) #κράτα τα top 5 μαγαζία ως προς το πλήθος
προϊόντων
FROM ( SELECT A,B,count(*) as quantity #βρες πλήθος κάθε προϊόντος ανα κατάστημα
      FROM
        SELECT DISTINCT A,B #natural join με διαγραφή διπλών τιμών
        FROM
          SELECT st.name as A, st.storeid as B
          FROM Store as st,Supply as sup
          WHERE st.storeid=sup.storeid
      GROUP BY A,B )
ORDER BY quantity DESC
```

Q5:

```
SELECT D #θέλουμε κάθε max προϊόν να κάνει πάνω από 200 ευρώ
FROM
  SELECT D, MAX( C ) AS E #εύρεση max προϊόντος ανά πόλη
  FROM
    SELECT A,B,C,st.city as D #natural join 2
    FROM
      (SELECT sup.storeid as A, g.gid as B g.price as C #natural join 1
      FROM Goods as g, Supply as sup
      WHERE sup.gid=B) , Store as st
    WHERE st.storeid=A
  GROUP BY D
WHERE E ≥ 200
```



### Επιμέρους Βήματα:

```
SELECT D, MAX( C ) AS E #εύρεση max προϊόντος ανά πόλη
FROM
  SELECT A,B,C,st.city as D
  FROM
    (SELECT sup.storeid as A, g.gid as B g.price as C
     FROM Goods as g, Supply as sup
     WHERE sup.gid=B), Store as st
  WHERE st.storeid=A
GROUP BY D
```

```
SELECT A,B,C,st.city as D #natural join 2
FROM
  (SELECT sup.storeid as A, g.gid as B g.price as C
   FROM Goods as g, Supply as sup
   WHERE sup.gid=B), Store as st
WHERE st.storeid=A
```

```
SELECT sup.storeid as A, g.gid as B g.price as C #natural join 1
FROM Goods as g, Supply as sup
WHERE g.gid=sup.gid
```

Q6:

D: #DIVISION

```
SELECT * FROM R
WHERE x not in ( SELECT x FROM (
(SELECT x , y FROM (select y from S ) as p cross join
(select distinct x from R) as sp)
EXCEPT
(SELECT x , y FROM R) ) AS r )
```

R: #ARITHMITIS

```
SELECT storeid,gid
FROM Supply
```

S: #PARONOMASTIS

```
SELECT storeid
FROM Store
WHERE city='Athens'
```

#natural join + result

```
SELECT gid
FROM D,Goods
WHERE D.gid=goods.gid
```

Q7:

```
(
SELECT B #βρες τα προιοντα στα μαγαζια της πάτρας
FROM
    SELECT A,B,C,st.city as D #natural join 2
FROM
    SELECT sup.storeid as A, g.gid as B g.price as C #natural join 1
    FROM Goods as g, Supply as sup
    WHERE sup.gid=B ) , Store as st
WHERE st.storeid=A
WHERE D = 'Athens'
)
```

-

```
(
SELECT B #βρες τα προιοντα στα μαγαζια της πάτρας
FROM
    SELECT A,B,C,st.city as D #natural join 2
FROM
    SELECT sup.storeid as A, g.gid as B g.price as C #natural join 1
    FROM Goods as g, Supply as sup
    WHERE sup.gid=B ) , Store as st
WHERE st.storeid=A
WHERE D = 'Patras'
)
```

### Άσκηση 4:

A. Με βάση τον αλγόριθμο της Διαφάνειας  
47(Normalization) έχουμε:

1.  $\text{Set\_A} = \{B, C\} \rightarrow$  Γνωρίσματα που δεν βρίσκονται στο δεξί μέλος

2.  $\text{Set\_B} = \{A\} \rightarrow$  Γνωρίσματα που βρίσκονται στο δεξί αλλά όχι στο αριστερό μέλος

3.  $\text{Set\_A} += \{B, C, E, A, D\} = R \Rightarrow$  **Μοναδικό υποψήφιο κλειδί το  $\text{Set\_A} = \{B, C\}$**

B. Για την ελάχιστη κάλυψη:

Θέλουμε ένα μόνο γνώρισμα στο δεξί μέλος

Συνεπώς έχουμε:

$$B \rightarrow EA \Rightarrow \{B \rightarrow E, B \rightarrow A\}$$

Στην συνέχεια προσπαθούμε να εφαρμόσουμε απλοποιήσεις στο αριστερό μέλος των σχέσεων:

$$B \rightarrow E \Rightarrow (\text{Μένει ως έχει})$$

$$B \rightarrow A \Rightarrow (\text{Μένει ως έχει})$$

$$EBC \rightarrow D \Rightarrow BC \rightarrow D \text{ (αφού } B \rightarrow E)$$

$$BED \rightarrow A \Rightarrow BD \rightarrow A \text{ (αφού } B \rightarrow E)$$

Τέλος, διαγράφουμε εξαρτήσεις που μπορούν να προκύψουν από τις υπόλοιπες, στην προκειμένη περίπτωση την εξάρτηση  $BD \rightarrow A$  (αφού έχουμε ήδη  $B \rightarrow A$ )

Τελικά, η ελάχιστη κάλυψη του set R είναι:

**min\_cover = {**

**B → E**

**B → A**

**BC → D**

**}**

Για την κανονική κάλυψη:

Εύρεση extraneous γνωρισμάτων στην αριστερή πλευρά των εξαρτήσεων:

(I) Για  $EBC \rightarrow D$ , έχουμε:

$(BC)^+ = BEACD \rightarrow$  περιλαμβάνει το D  $\Rightarrow$  Το E είναι extraneous

$(EC)^+ = EC \rightarrow$  Δεν περιλαμβάνει το D  $\Rightarrow$  Το B δεν είναι extraneous

$(EB)^+ = EBA \rightarrow$  Δεν περιλαμβάνει το D  $\Rightarrow$  Το C δεν είναι extraneous

Διαγράφουμε τα extraneous γνωρίσματα, και έχουμε:

$$EBC \rightarrow D \Rightarrow BC \rightarrow D$$

(II) Για  $BED \rightarrow A$ , έχουμε:

$(BE)^+ = BEA \Rightarrow$  Περιλαμβάνει το  $A \Rightarrow$  Το  $D$  είναι extraneous

$(BD)^+ = BEAD \Rightarrow$  Περιλαμβάνει το  $A \Rightarrow$  Το  $E$  είναι extraneous

$(ED)^+ = ED \Rightarrow$  Δεν περιλαμβάνει το  $A \Rightarrow$  Το  $B$  δεν είναι εξωτερικό

Διαγράφουμε τα extraneous γνωρίσματα και, έχουμε:

$$BED \rightarrow A \Rightarrow B \rightarrow A$$

Μέχρι στιγμής, έχουμε τις κάτωθι εξαρτήσεις:

$$B \rightarrow EA$$

$$BC \rightarrow D$$

$$B \rightarrow A$$

Εύρεση extraneous γνωρισμάτων στο δεξί μέλος των εξαρτήσεων:

(III) Για  $B \rightarrow EA$ , το  $A$  είναι extraneous γνώρισμα γιατί προκύπτει άμεσα από την εξάρτηση  $B \rightarrow A$ . Ωστόσο, σε μία canonical form θέλουμε τα αριστερά μέλη κάθε σχέσης να

είναι διαφορετικά μεταξύ τους, γεγονός που μας οδηγεί σε διαγραφή της εξάρτησης  $B \rightarrow A$

Τελικά, παίρνουμε ως αποτέλεσμα:

**canonical\_form = {**  
**BC  $\rightarrow$  D**  
**B  $\rightarrow$  EA**  
**}**

C. Η σχέση R είναι σε 1NF. Ωστόσο, **δεν είναι σε 2NF** καθώς ένα υποψήφιο κλειδί είναι το  $\{B,C\}$  αλλά υπάρχει σύνολο μη πρωτεόντων γνωρισμάτων, τα οποία εξαρτώνται από υποσύνολο του υποψήφιου κλειδιού. Συγκεκριμένα, αναφερόμαστε στην εν λόγω εξάρτηση:

$B \rightarrow EA$

D. Με βάση την διαφάνεια 69/Normalization

Χρησιμοποιούμε την κανονική μορφή του ερωτήματος B για να εξάγουμε τις κάτωθι σχέσεις

R1(ABE)

R2(EBCD)

Παρατηρούμε ότι το υποψήφιο κλειδί  $\{B,C\}$  βρίσκεται την σχέση R2, οπότε δεν χρειάζεται να εισάγουμε κάποια περαιτέρω σχέση

Ελέγχουμε αν κάποια σχέση αποτελεί υποσύνολο κάποιας άλλης (ώστε να διαγραφεί). Κάτι τέτοιο δεν ισχύει, οπότε μία διάσπαση σε 3NF είναι η κάτωθι:

**(B,E,A)**

**(B,C,D)**

### Άσκηση 5:

A. Με βάση τον αλγόριθμο της Διαφάνειας  
47(Normalization) έχουμε:

1.  $\text{Set\_A} = \{B\} \rightarrow$  Γνωρίσματα που δεν βρίσκονται στο δεξί μέλος

2.  $\text{Set\_B} = \{D\} \rightarrow$  Γνωρίσματα που βρίσκονται στο δεξί αλλά όχι στο αριστερό μέλος

3.  $\text{Set\_A} += \{B,D\} \neq R$

4.  $R - \text{Set\_B} = \{A,B,C\} = X$

5.  $\text{Set\_A} \cup X$  δίνει τα κάτωθι σύνολα, των οποίων θα υπολογίσουμε την κλειστότητα:

(I)  $(BA) += (BACD) = R \rightarrow$  **BA Υποψήφιο κλειδί**

(II)  $(B)^+ = (BD) \neq R \rightarrow$  B όχι υποψήφιο κλειδί



(III)  $(BC)^+ = (BCAD)^+ = R \rightarrow BC$  Υποψήφιο κλειδί

B.

Ξεκινώντας από την εξάρτηση  $AB \rightarrow C$  αυτή είναι σε BCNF μιας και το AB είναι κλειδί της R

Από την άλλη η εξάρτηση  $B \rightarrow D$  παραβαίνει τον κανόνα αυτό (το B δεν είναι κλειδί) οπότε κάνουμε το εξής “σπάσιμο”:

$$(BD)(R - \{D\}) = (BD)(ABC)$$

Στην σχέση (BD) το μόνο υποψήφιο κλειδί είναι το B και εφαρμόζεται η εξάρτηση  $B \rightarrow D$  στην οποία το αριστερό μέλος είναι το υποψήφιο κλειδί. Συνεπώς, η (BD) είναι σε BCNF

Στην σχέση (ABC), έχουμε υποψήφια κλειδιά τα AB και AC. Στην πρώτη εφαρμοζόμενη εξάρτηση ( $AB \rightarrow C$ ) το αριστερό μέλος είναι υποψήφιο κλειδί, ωστόσο στην επόμενη ( $C \rightarrow A$ ) το αριστερό μέλος C δεν είναι υποψήφιο κλειδί. Συνεπώς, σπάμε την (ABC) σε  $(CA)((ABC) - (A)) = (CA)(BC)$

Η (CA) έχει ως υποψήφιο κλειδί το C και η μόνη εφαρμοζόμενη σχέση είναι η  $C \rightarrow A$  η οποία έχει κλειδί στην αριστερή πλευρά της.

Η σχέση (BC) έχει ως υποψήφιο κλειδί το (BC) και καμία εφαρμοζόμενη σχέση. Συνεπώς, είναι σε BCNF

Τελικά το σπάσιμο πραγματοποιείται στις κάτωθι BCNF σχέσεις:

**(BD)(CA)(BC)**

