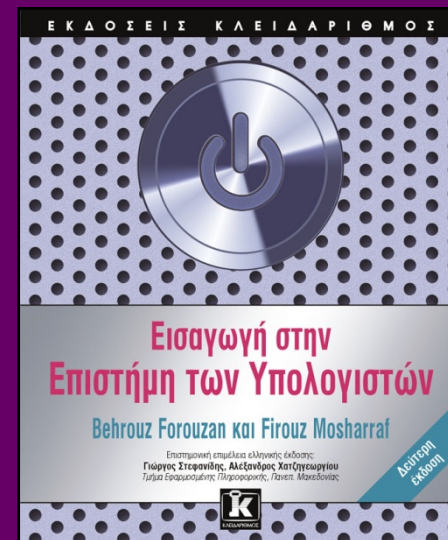


14



Βάσεις δεδομένων

Εισαγωγή στην Επιστήμη των Υπολογιστών ©
Εκδόσεις Κλειδάριθμος

Στόχοι

Μετά την ολοκλήρωση αυτού του κεφαλαίου, ο σπουδαστής θα είναι σε θέση:

- Να ορίζει μια βάση δεδομένων και ένα σύστημα διαχείρισης βάσεων δεδομένων (ΣΔΒΔ), καθώς και να περιγράφει τα συστατικά στοιχεία ενός ΣΔΒΔ.
- Να περιγράφει την αρχιτεκτονική ενός ΣΔΒΔ με βάση το πρότυπο ANSI/SPARC.
- Να ορίζει τα τρία παραδοσιακά μοντέλα βάσεων δεδομένων: το ιεραρχικό, το δικτυακό, και το σχεσιακό.
- Να περιγράφει το σχεσιακό μοντέλο και τις σχέσεις.
- Να κατανοεί τις λειτουργίες που εφαρμόζονται σε μια σχεσιακή βάση δεδομένων με διαταγές SQL.
- Να περιγράφει τα βήματα που ακολουθούνται στον σχεδιασμό βάσεων δεδομένων.
- Να ορίζει διαγράμματα ERM και E-R και να εξηγεί τις οντότητες και τις σχέσεις σε αυτό το μοντέλο.
- Να ορίζει τα ιεραρχικά επίπεδα της κανονικοποίησης και να κατανοεί τη λογική που εφαρμόζεται για την κανονικοποίηση των σχέσεων.
- Να περιγράφει άλλους τύπους βάσεων δεδομένων εκτός από το σχεσιακό μοντέλο.

14-1 ΕΙΣΑΓΩΓΗ

Παλαιότερα, για την αποθήκευση δεδομένων χρησιμοποιούνταν μόνο μεμονωμένα, μη σχετιζόμενα αρχεία, τα οποία ονομάζονται **επίπεδα αρχεία** (flat files). Έτσι, κάθε πρόγραμμα εφαρμογής σε μια εταιρεία χρησιμοποιούσε το δικό της αρχείο. Σε ένα πανεπιστήμιο, για παράδειγμα, κάθε τμήμα μπορεί να είχε το δικό του σύνολο αρχείων: η γραμματεία είχε ένα αρχείο με πληροφορίες και βαθμούς σπουδαστών, το τμήμα ανάθεσης μαθημάτων διατηρούσε ένα αρχείο με τα ονόματα των καθηγητών και τα μαθήματα που δίδασκαν, το τμήμα μισθοδοσίας είχε ένα δικό του αρχείο για ολόκληρο το προσωπικό, και ούτω καθεξής. Σήμερα, όμως, όλα αυτά τα επίπεδα αρχεία μπορούν να συνδυαστούν σε μία οντότητα, σε μια βάση δεδομένων για ολόκληρο το πανεπιστήμιο.

Ορισμός

Αν και είναι δύσκολο να δώσουμε έναν καθολικά αποδεκτό ορισμό τού τι είναι μια βάση δεδομένων, θα χρησιμοποιήσουμε τον παρακάτω συνηθισμένο ορισμό:



Ορισμός:

Μια βάση δεδομένων είναι μια συλλογή λογικά συσχετισμένων δεδομένων που χρησιμοποιούνται από τα προγράμματα εφαρμογών μιας εταιρείας.

Πλεονεκτήματα βάσεων δεδομένων

Αν συγκρίνουμε τις βάσεις δεδομένων με το σύστημα επίπεδων αρχείων, μπορούμε να αναφέρουμε πολλά πλεονεκτήματά τους.

Λιγότερος πλεονασμός

Σε ένα σύστημα επίπεδων αρχείων ο πλεονασμός (redundancy) είναι μεγάλος. Για παράδειγμα, στο σύστημα επίπεδων αρχείων ενός πανεπιστημίου, τα ονόματα των καθηγητών και των σπουδαστών αποθηκεύονται σε περισσότερα από ένα αρχεία.

Αποφυγή ασυνέπειας

Αν ένα τμήμα πληροφοριών είναι αποθηκευμένο σε περισσότερες από μία θέσεις, οι αλλαγές στα δεδομένα πρέπει να γίνονται σε όλες τις θέσεις.

Αποδοτικότητα

Μια βάση δεδομένων είναι συνήθως πιο αποδοτική από ένα σύστημα επίπεδων αρχείων, επειδή κάθε τμήμα πληροφοριών αποθηκεύεται σε λιγότερες θέσεις.

Ακεραιότητα δεδομένων

Η ακεραιότητα των δεδομένων (δείτε το Κεφάλαιο 16) διασφαλίζεται ευκολότερα σε μια βάση δεδομένων επειδή κάθε στοιχείο δεδομένων αποθηκεύεται σε λιγότερες θέσεις.

Εμπιστευτικότητα

Η εμπιστευτικότητα των δεδομένων διασφαλίζεται ευκολότερα όταν τα δεδομένα αποθηκεύονται σε μία κεντρική θέση.

14-2 ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

Ένα σύστημα διαχείρισης βάσεων δεδομένων, ή ΣΔΒΔ (database management system, DBMS), ορίζει, δημιουργεί, και συντηρεί βάσεις δεδομένων. Το ΣΔΒΔ παρέχει επίσης στους χρήστες ελεγχόμενη πρόσβαση στις βάσεις δεδομένων. Ένα ΣΔΒΔ είναι ένας συνδυασμός πέντε συστατικών στοιχείων: του υλικού, του λογισμικού, των δεδομένων, των χρηστών, και των διαδικασιών (Εικόνα 14.1).



Εικόνα 14.1 Συστατικά στοιχεία ενός ΣΔΒΔ

Υλικό

Το υλικό (hardware) είναι το φυσικό υπολογιστικό σύστημα το οποίο επιτρέπει την προσπέλαση των δεδομένων.

Λογισμικό

Το λογισμικό (software) είναι το πραγματικό πρόγραμμα που επιτρέπει στους χρήστες να προσπελάζουν, να συντηρούν, και να ενημερώνουν τα δεδομένα. Επίσης, το λογισμικό ελέγχει ποιοι χρήστες μπορούν να προσπελάζουν συγκεκριμένα τμήματα της βάσης δεδομένων.

Δεδομένα

Τα δεδομένα μιας βάσης δεδομένων αποθηκεύονται με φυσικό τρόπο σε αποθηκευτικές συσκευές. Σε μια βάση δεδομένων τα δεδομένα αποτελούν ξεχωριστή οντότητα από το λογισμικό που τα προσπελάζει.

Χρήστες

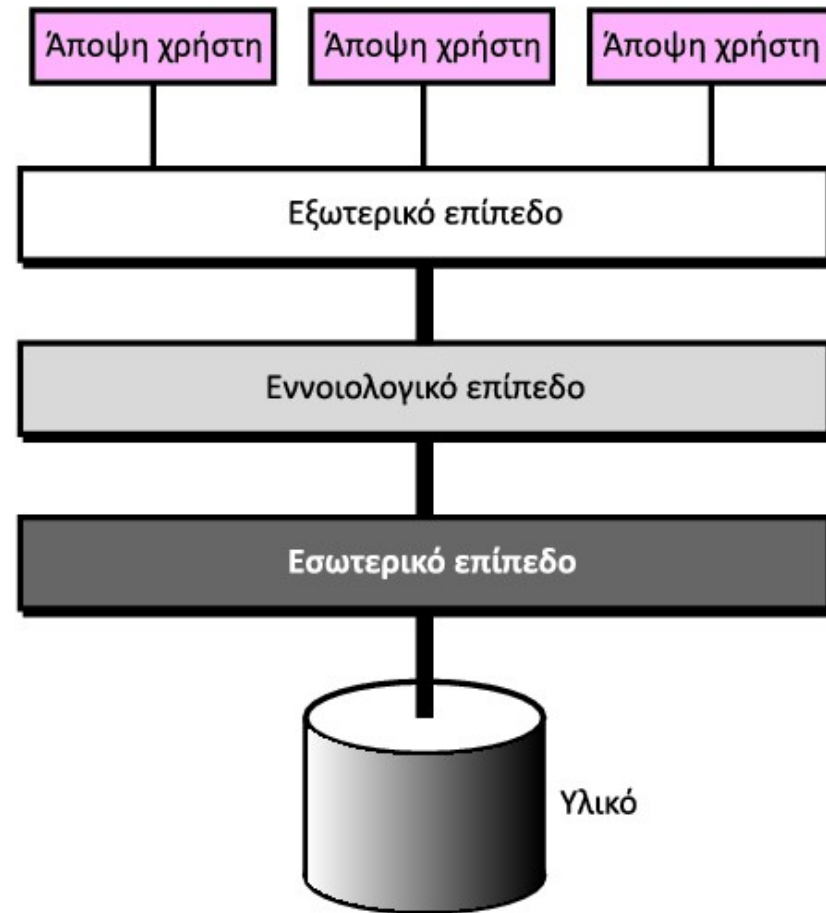
Ο όρος "χρήστες" σε ένα ΣΔΒΔ έχει ευρεία σημασία. Μπορούμε να χωρίσουμε τους χρήστες σε δύο κατηγορίες: τους **τελικούς χρήστες** και τα **προγράμματα εφαρμογών**.

Διαδικασίες

Το τελευταίο συστατικό στοιχείο ενός ΣΔΒΔ είναι ένα σύνολο διαδικασιών ή κανόνων που πρέπει να έχουν οριστεί με σαφή τρόπο και να ακολουθούνται από τους χρήστες της βάσης δεδομένων.

14-3 ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

Η Επιτροπή Σχεδιασμού και Απαιτήσεων Προτύπων (Standards Planning and Requirements Committee, ή SPARC) του Αμερικανικού Ινστιτούτου Εθνικών Προτύπων (ANSI) έχει καθιερώσει μια αρχιτεκτονική τριών επιπέδων για τα ΣΔΒΔ: το **εσωτερικό**, το **εννοιολογικό**, και το **εξωτερικό** (Εικόνα 14.2).



Εικόνα 14.2 Αρχιτεκτονική βάσεων δεδομένων

Εσωτερικό επίπεδο

Το **εσωτερικό επίπεδο** (internal level) προσδιορίζει το πού αποθηκεύονται τα δεδομένα στις αποθηκευτικές συσκευές. Αυτό το επίπεδο έχει να κάνει με μεθόδους προσπέλασης χαμηλού επιπέδου, καθώς και με τον τρόπο που μεταφέρονται τα byte από και προς τις αποθηκευτικές συσκευές. Με άλλα λόγια, το εσωτερικό επίπεδο αλληλεπιδρά απευθείας με το υλικό.

Εννοιολογικό επίπεδο

Το **εννοιολογικό επίπεδο** (conceptual level) καθορίζει τη λογική όψη των δεδομένων. Σε αυτό το επίπεδο ορίζονται το μοντέλο δεδομένων και τα διαγράμματα δομής, ενώ εκεί υπάρχουν και οι κύριες λειτουργίες του ΣΔΒΔ, όπως τα ερωτήματα. Το ΣΔΒΔ μετατρέπει την εσωτερική όψη των δεδομένων στην εξωτερική όψη που βλέπουν οι χρήστες. Το εννοιολογικό επίπεδο λειτουργεί ως μεσάζων και απαλλάσσει τους χρήστες από την ανάγκη να χειρίζονται απευθείας το εσωτερικό επίπεδο.

Εξωτερικό επίπεδο

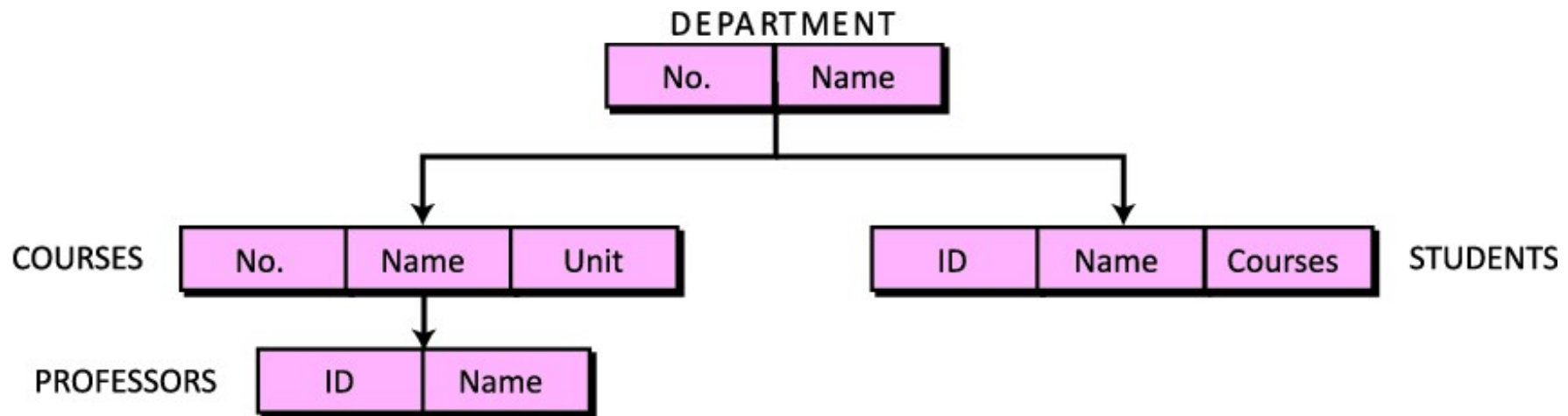
Το **εξωτερικό επίπεδο** (external level) αλληλεπιδρά απευθείας με τον χρήστη (τους τελικούς χρήστες ή τα προγράμματα εφαρμογών). Μετατρέπει τα δεδομένα που έρχονται από το εννοιολογικό επίπεδο σε μια μορφή και όψη που να είναι πιο οικείες στους χρήστες.

14-4 ΜΟΝΤΕΛΑ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

Ένα μοντέλο βάσεων δεδομένων καθορίζει τον λογικό σχεδιασμό των δεδομένων. Το μοντέλο περιγράφει επίσης τις σχέσεις μεταξύ των διαφόρων μερών των δεδομένων. Στην ιστορία του σχεδιασμού βάσεων δεδομένων έχουν χρησιμοποιηθεί τρία μοντέλα: το ιεραρχικό, το δικτυακό, και το σχεσιακό.

Ιεραρχικό μοντέλο βάσεων δεδομένων

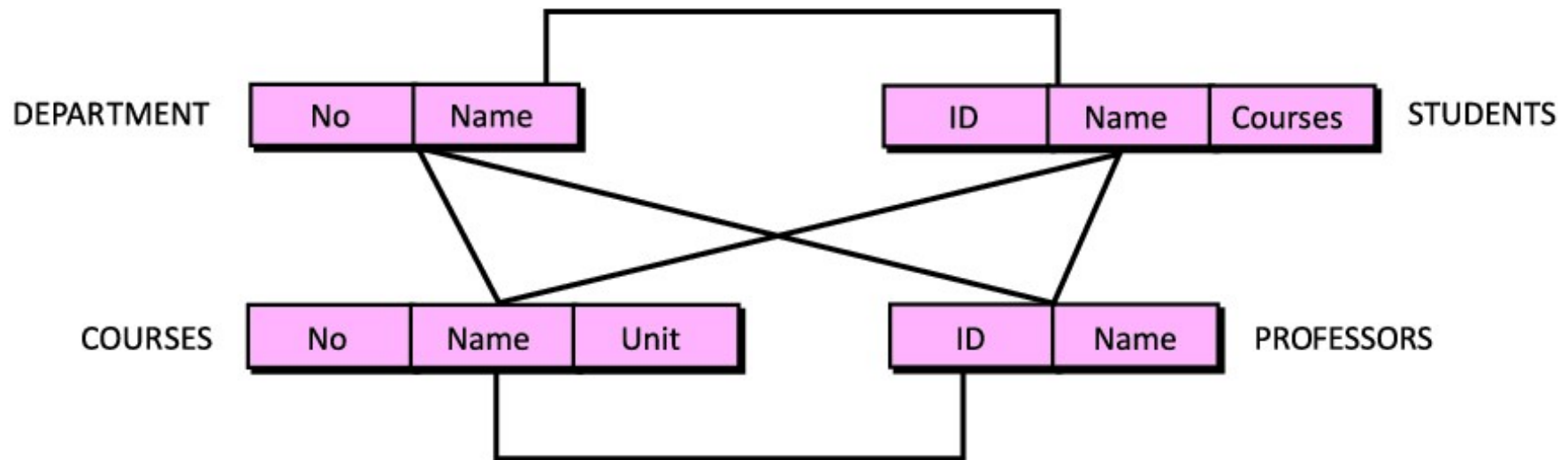
Στο ιεραρχικό μοντέλο (hierarchical model) τα δεδομένα οργανώνονται σε ένα δέντρο. Κάθε οντότητα έχει μόνο μία γονική, αλλά μπορεί να έχει πολλές θυγατρικές. Στην κορυφή της ιεραρχίας υπάρχει μια οντότητα η οποία ονομάζεται *ρίζα*.



Εικόνα 14.3 Παράδειγμα ιεραρχικού μοντέλου για την αναπαράσταση ενός πανεπιστημίου

Δικτυακό μοντέλο βάσεων δεδομένων

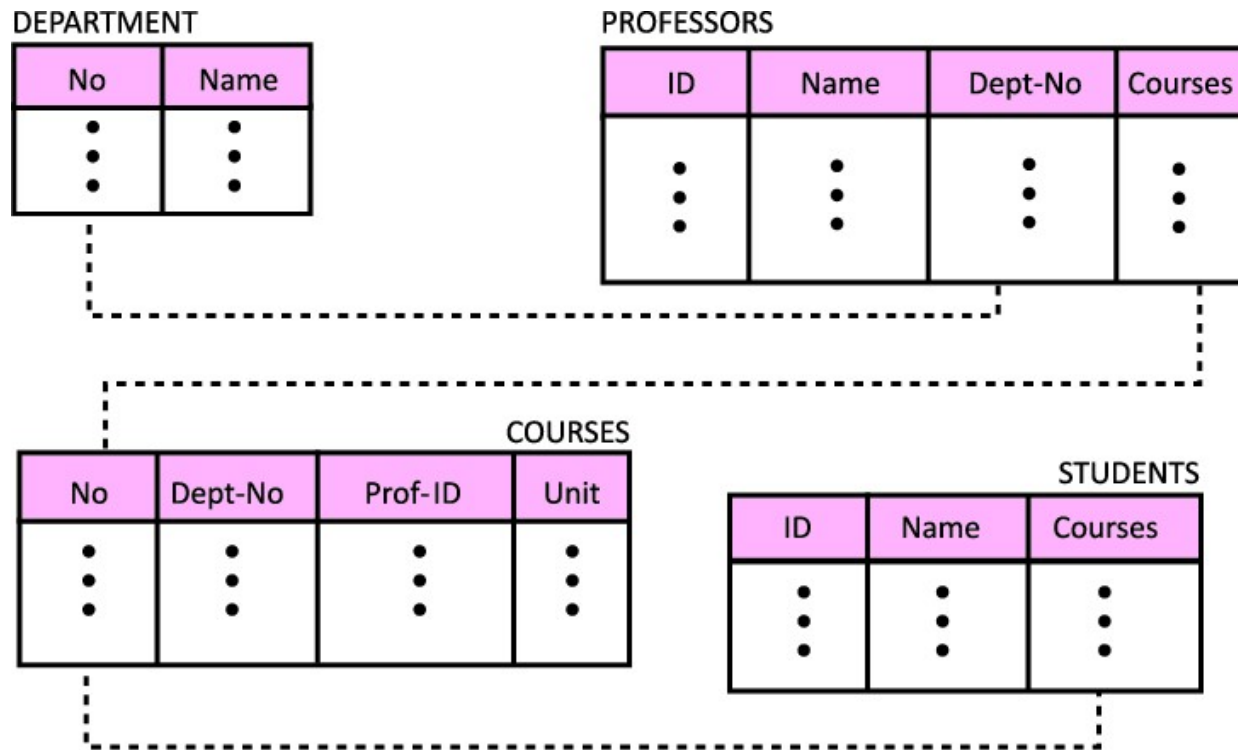
Στο δικτυακό μοντέλο (network model) οι οντότητες οργανώνονται σε έναν γράφο όπου κάποιες από αυτές μπορούν να προσπελαστούν μέσω διαφόρων διαδρομών (Εικόνα 14.4).



Εικόνα 14.4 Παράδειγμα δικτυακού μοντέλου για την αναπαράσταση ενός πανεπιστημίου

Σχεσιακό μοντέλο βάσεων δεδομένων

Στο σχεσιακό μοντέλο τα δεδομένα οργανώνονται σε διδιάστατους πίνακες που ονομάζονται σχέσεις (relations). Ωστόσο, οι πίνακες (ή σχέσεις) σχετίζονται μεταξύ τους, όπως θα δούμε σύντομα.



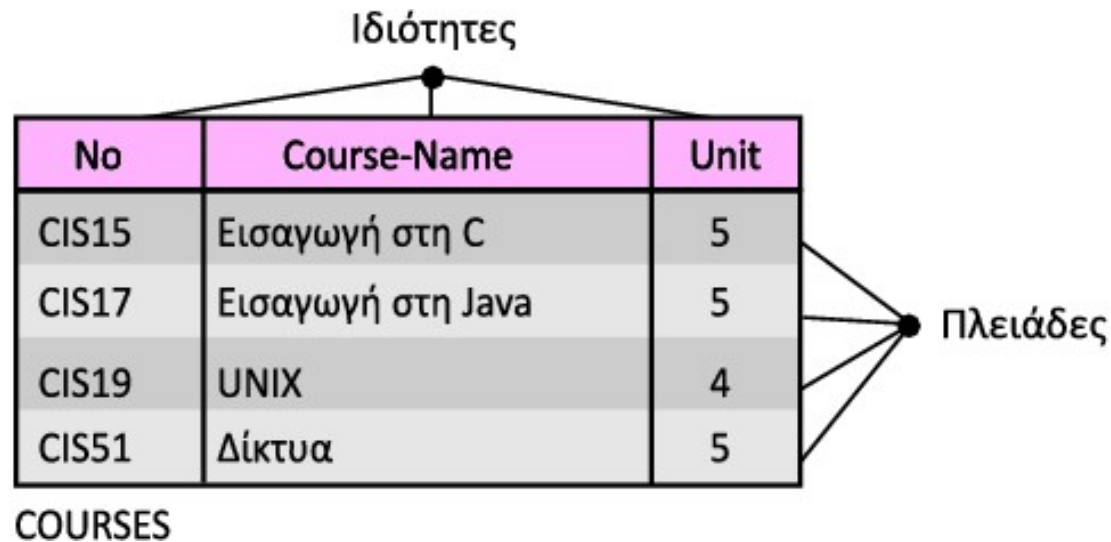
Εικόνα 14.5 Παράδειγμα σχεσιακού μοντέλου για την αναπαράσταση ενός πανεπιστημίου

14.5 ΤΟ ΣΧΕΣΙΑΚΟ ΜΟΝΤΕΛΟ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

Στο **σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων** ή **ΣΣΔΒΔ** (relational database management system, RDBMS) τα δεδομένα αναπαρίστανται ως ένα σύνολο **σχέσεων**.

Σχέσεις

Φαινομενικά μια **σχέση** (relation) είναι ένας διδιάστατος πίνακας. Το ΣΣΔΒΔ οργανώνει τα δεδομένα έτσι ώστε η εξωτερική όψη να είναι ένα σύνολο σχέσεων ή πινάκων. Αυτό δεν σημαίνει ότι τα δεδομένα είναι αποθηκευμένα σε πίνακες· ο φυσικός τρόπος αποθήκευσης των δεδομένων είναι ανεξάρτητος από τη λογική οργάνωσή τους.



Εικόνα 14.6 Ένα παράδειγμα σχέσης

Μια σχέση ενός ΣΣΔΒΔ έχει τα ακόλουθα χαρακτηριστικά:

- ❑ **Όνομα.** Κάθε σχέση σε μια σχεσιακή βάση δεδομένων πρέπει να έχει ένα όνομα το οποίο να είναι μοναδικό μεταξύ των άλλων σχέσεων.
- ❑ **Ιδιότητες.** Κάθε στήλη μιας σχέσης ονομάζεται ιδιότητα (attribute). Οι ιδιότητες είναι οι επικεφαλίδες των στηλών του πίνακα στην Εικόνα 14.6.
- ❑ **Πλειάδες.** Κάθε γραμμή μιας σχέσης ονομάζεται πλειάδα (tuple). Οι πλειάδες ορίζουν συλλογές από τιμές ιδιοτήτων. Το συνολικό πλήθος των γραμμών μιας σχέσης ονομάζεται πληθικότητα (cardinality) της σχέσης. Σημειώστε ότι η πληθικότητα μιας σχέσης αλλάζει αν προστεθούν ή διαγραφούν πλειάδες. Αυτό κάνει τη βάση δεδομένων δυναμική.

14-6 ΛΕΙΤΟΥΡΓΙΕΣ ΣΕ ΣΧΕΣΕΙΣ

Σε μια σχεσιακή βάση δεδομένων μπορούν να οριστούν διάφορες λειτουργίες για τη δημιουργία νέων σχέσεων από τις υπάρχουσες. Σε αυτή την ενότητα θα περιγράψουμε εννέα λειτουργίες: την εισαγωγή, τη διαγραφή, την ενημέρωση, την επιλογή, την προβολή, τη σύνδεση, την ένωση, την τομή, και τη διαφορά. Αντί να αναφερθούμε σε αυτές τις λειτουργίες αφηρημένα, θα περιγράψουμε κάθε μία με τον τρόπο που ορίζεται στη γλώσσα ερωτημάτων βάσεων δεδομένων SQL (Structured Query Language).

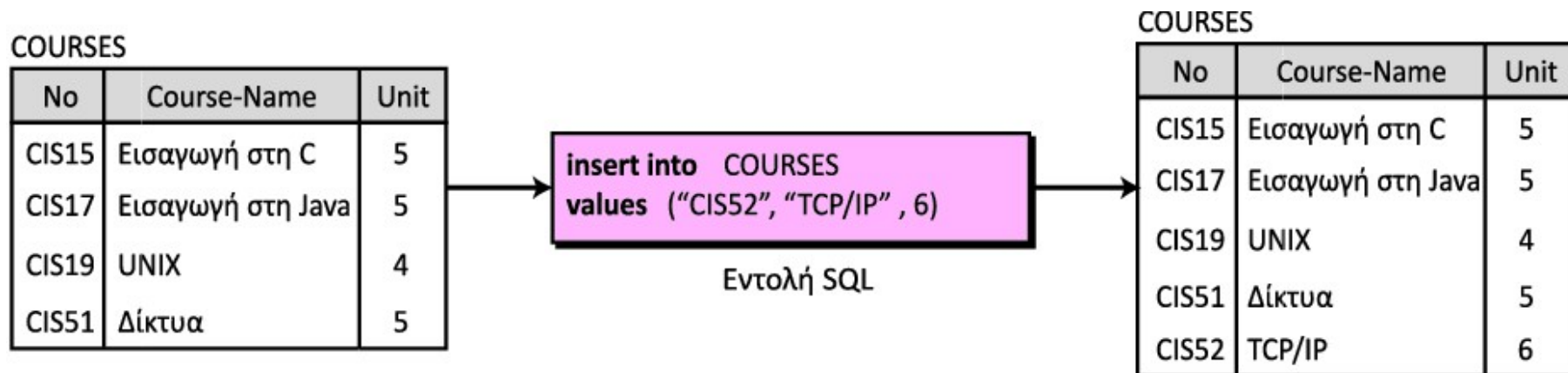
Δομημένη Γλώσσα Ερωτημάτων

Η **Δομημένη Γλώσσα Ερωτημάτων** (Structured Query Language, ή SQL) είναι μια γλώσσα που προτυποποιήθηκε από το Αμερικανικό Ινστιτούτο Εθνικών Προτύπων (ANSI) και τον Διεθνή Οργανισμό Προτυποποίησης (ISO) για χρήση σε σχεσιακές βάσεις δεδομένων. Είναι **δηλωτική** και όχι **διαδικασιακή** γλώσσα, κάτι που σημαίνει ότι ο χρήστης δηλώνει αυτό που θέλει χωρίς να είναι ανάγκη να το περιγράψει σε μια αναλυτική διαδικασία. Η γλώσσα SQL αναπτύχθηκε αρχικά από την Oracle Corporation το 1979, και από τότε έχουν κυκλοφορήσει διάφορες εκδόσεις της.

Εισαγωγή

Η *λειτουργία εισαγωγής* (insert) είναι μονομελής — δηλαδή, εφαρμόζεται σε μία μόνο σχέση. Η λειτουργία αυτή εισάγει μια νέα πλειάδα στη σχέση. Η λειτουργία εισαγωγής χρησιμοποιεί την εξής μορφή:

```
insert into ΟΝΟΜΑ-ΣΧΕΣΗΣ  
values (... , ... , ...)
```

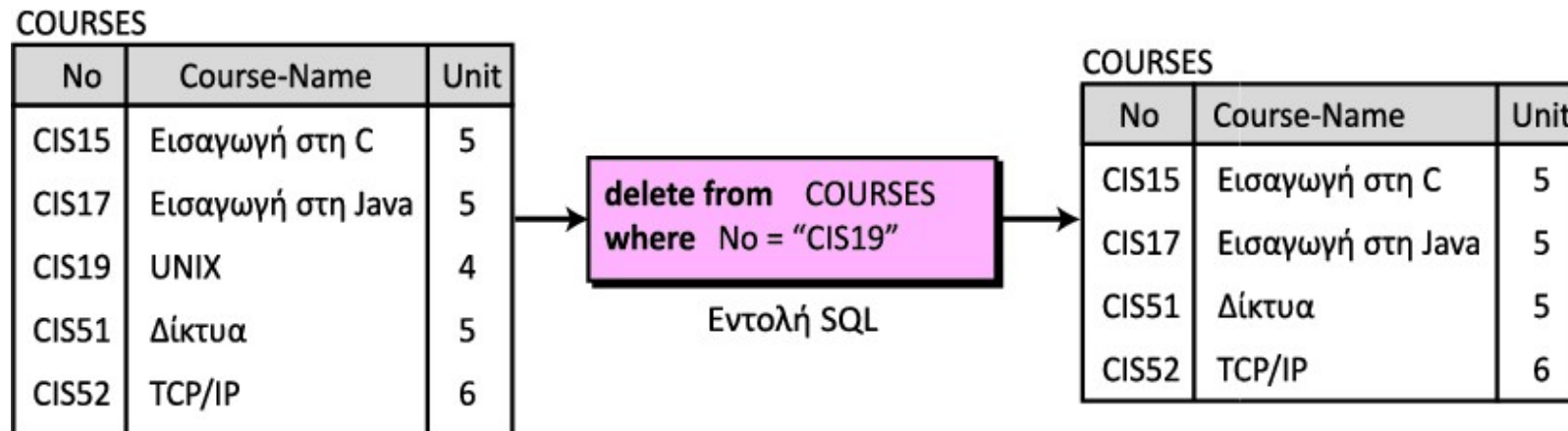


Εικόνα 14.7 Παράδειγμα λειτουργίας εισαγωγής

Διαγραφή

Η λειτουργία διαγραφής (delete) είναι επίσης μονομελής. Η λειτουργία αυτή διαγράφει μια πλειάδα η οποία ορίζεται από ένα κριτήριο στη σχέση. Η λειτουργία διαγραφής χρησιμοποιεί την εξής μορφή:

```
delete from ΟΝΟΜΑ-ΣΧΕΣΗΣ  
where κριτήρια
```

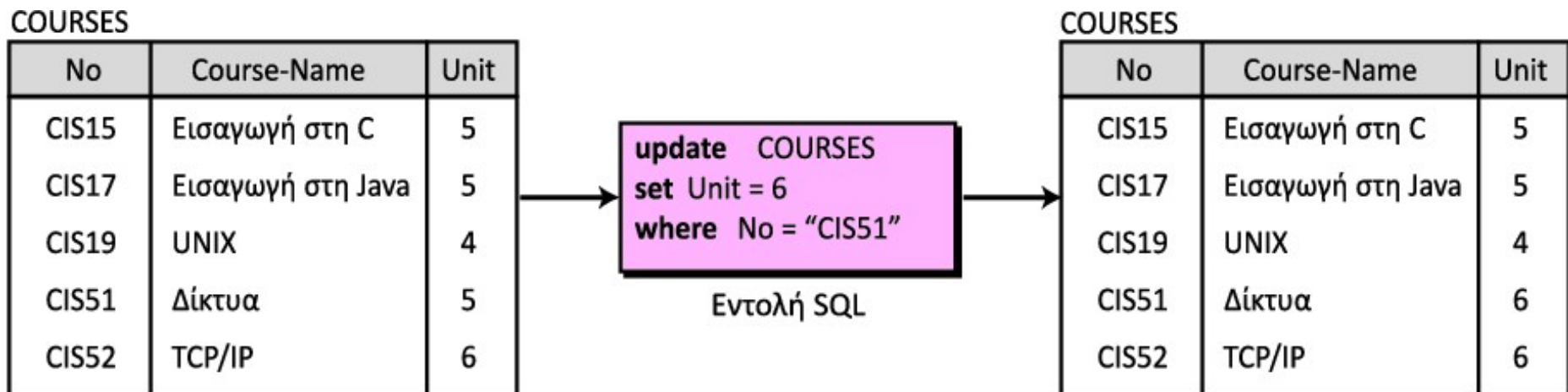


Εικόνα 14.8 Παράδειγμα λειτουργίας διαγραφής

Ενημέρωση

Η λειτουργία ενημέρωσης (update) είναι επίσης μονομελής, δηλαδή εφαρμόζεται σε μία μόνο σχέση. Η λειτουργία αυτή αλλάζει την τιμή κάποιων ιδιοτήτων σε μια πλειάδα. Η λειτουργία ενημέρωσης χρησιμοποιεί την εξής μορφή:

```
update ONOMA-ΣΧΕΣΗΣ  
set ιδιότητα1 = τιμή1, ιδιότητα2 = τιμή2, ...  
where κριτήρια
```



Εικόνα 14.9 Παράδειγμα λειτουργίας ενημέρωσης

Επιλογή

Η λειτουργία επιλογής (select) είναι και αυτή μονομελής. Οι πλειάδες (γραμμές) στη σχέση που προκύπτει αποτελούν υποσύνολο των πλειάδων στην αρχική σχέση.

```
select *  
from ΟΝΟΜΑ-ΣΧΕΣΗΣ  
where κριτήρια
```

COURSES

No	Course-Name	Unit
CIS15	Εισαγωγή στη C	5
CIS17	Εισαγωγή στη Java	5
CIS19	UNIX	4
CIS51	Δίκτυα	5
CIS52	TCP/IP	6

```
select *  
from COURSES  
where Unit =5
```

Εντολή SQL

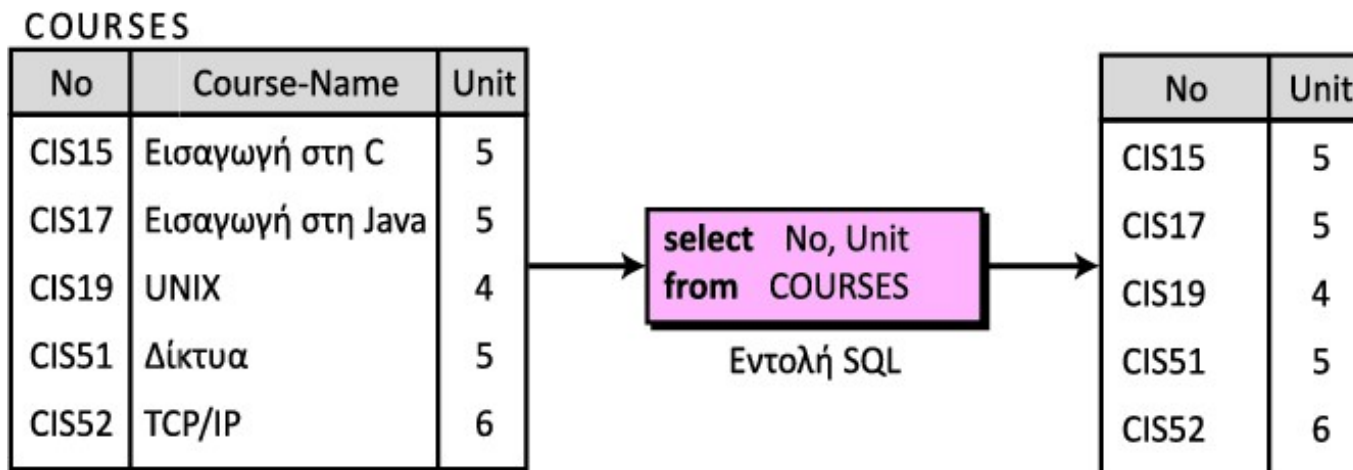
No	Course-Name	Unit
CIS15	Εισαγωγή στη C	5
CIS17	Εισαγωγή στη Java	5
CIS51	Δίκτυα	5

Εικόνα 14.10 Παράδειγμα λειτουργίας επιλογής

Προβολή

Η λειτουργία προβολής (project) είναι επίσης μονομελής, και δημιουργεί μια άλλη σχέση. Οι ιδιότητες (στήλες) στη σχέση που προκύπτει αποτελούν υποσύνολο των ιδιοτήτων στην αρχική σχέση.

```
select λίστα-ιδιοτήτων  
from ΟΝΟΜΑ-ΣΧΕΣΗΣ
```

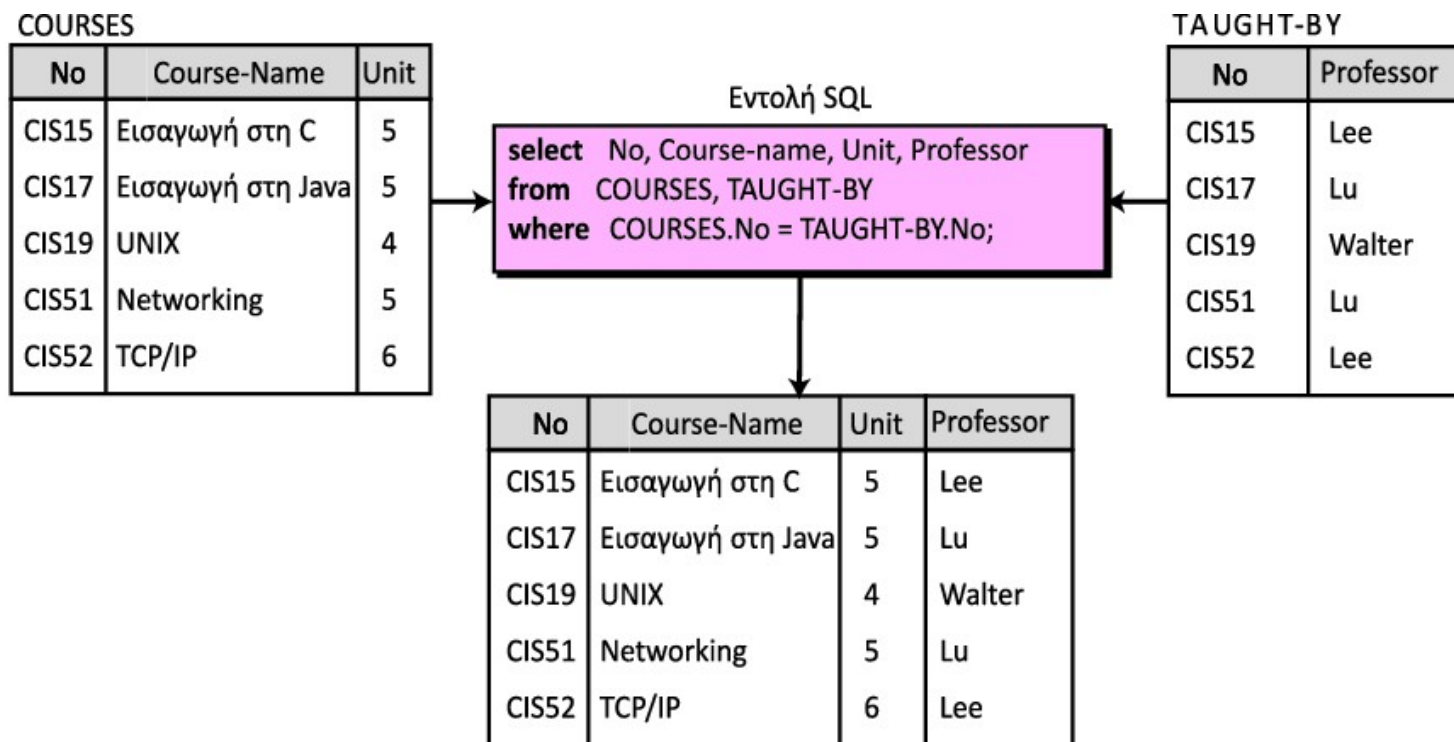


Εικόνα 14.11 Παράδειγμα λειτουργίας προβολής

Σύνδεση

Η λειτουργία σύνδεσης (join) είναι διμελής και συνδυάζει δύο σχέσεις με βάση κοινές ιδιότητες τους.

```
select λίστα-ιδιοτήτων  
from ΣΧΕΣΗ1, ΣΧΕΣΗ2  
where κριτήρια
```

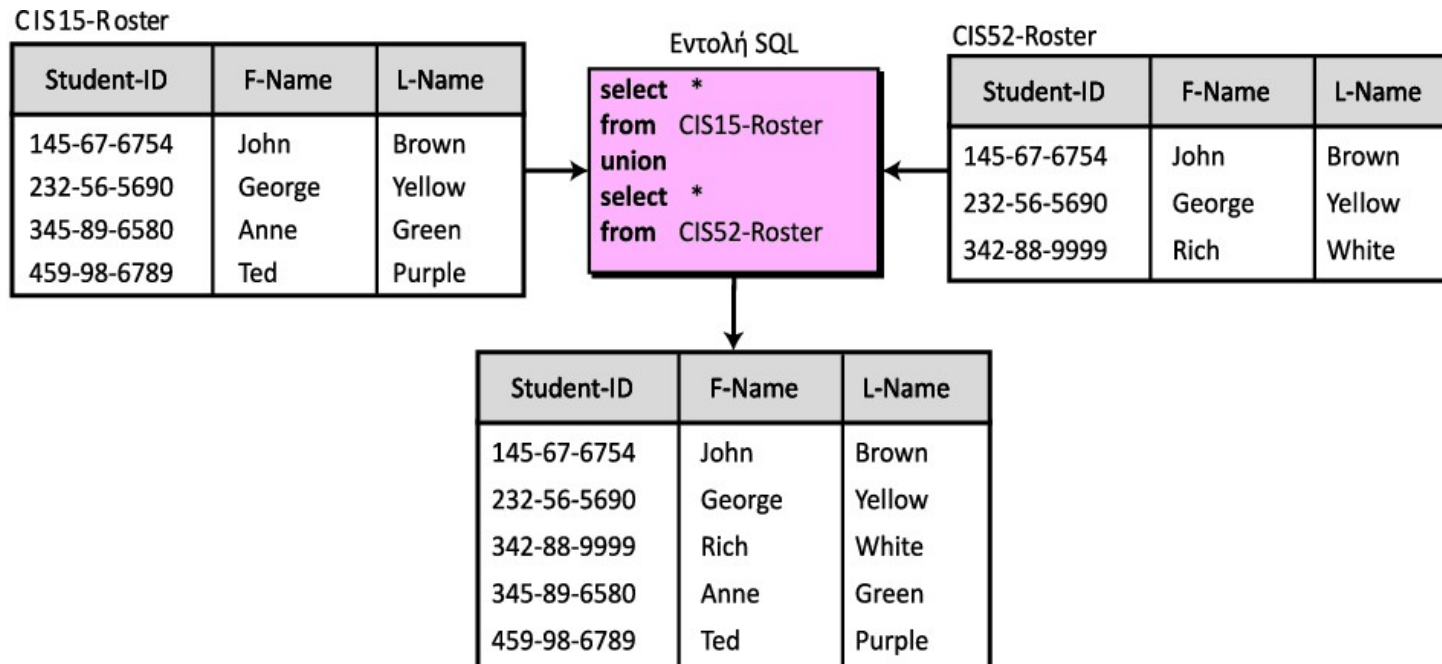


Εικόνα 14.12 Παράδειγμα λειτουργίας σύνδεσης

Ένωση

Η λειτουργία ένωσης (union) δέχεται δύο σχέσεις με το ίδιο σύνολο ιδιοτήτων.

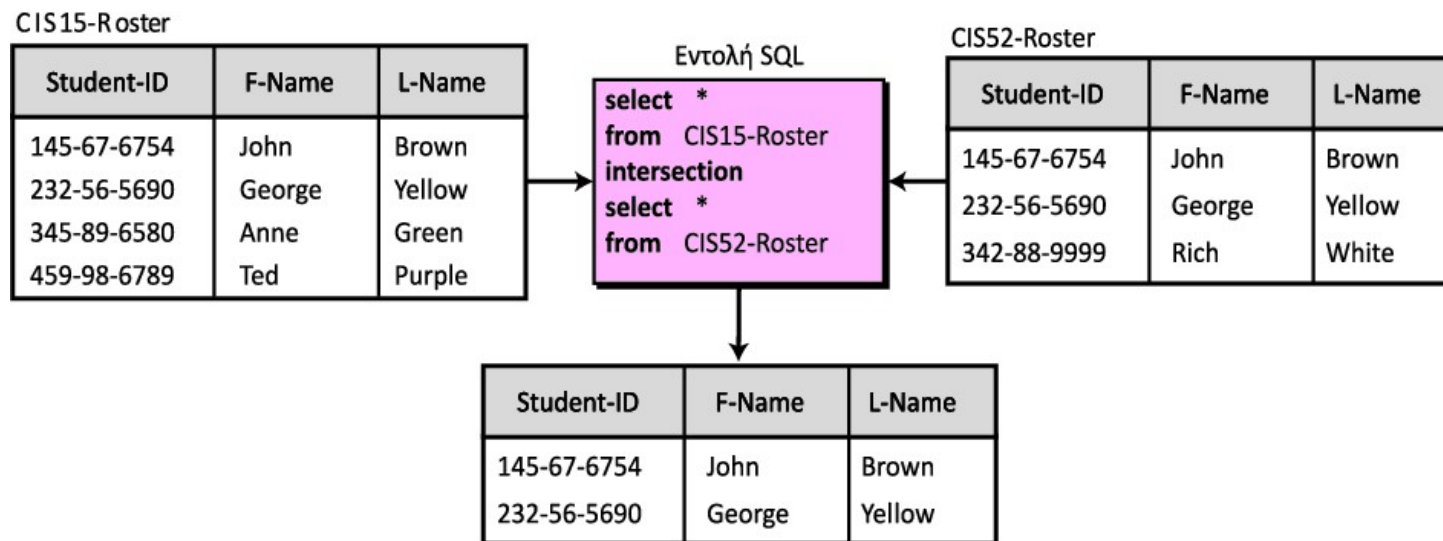
```
select *  
from ΣΧΕΣΗ1  
union  
select *  
from ΣΧΕΣΗ2
```



Τομή

Η λειτουργία τομής (intersection) δέχεται δύο σχέσεις και δημιουργεί μια νέα, η οποία αποτελεί την τομή τους.

```
select *  
from ΣΧΕΣΗ1  
intersection  
select *  
from ΣΧΕΣΗ2
```

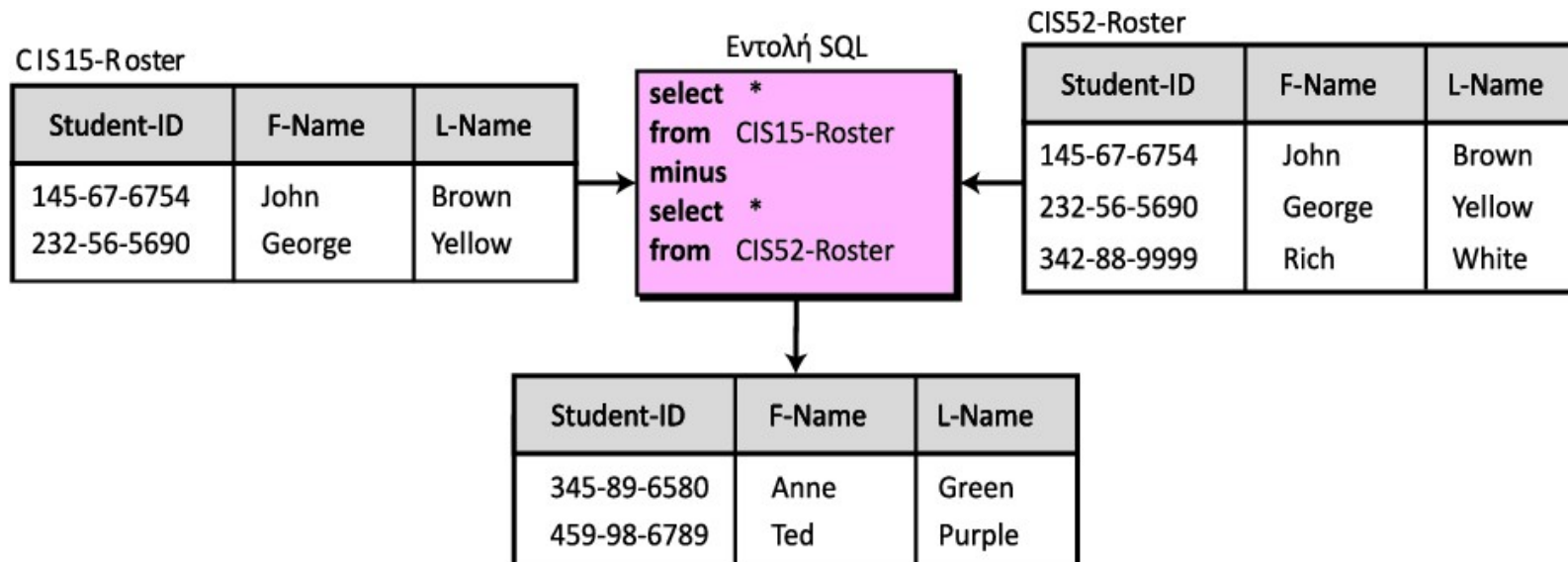


Εικόνα 14.14 Παράδειγμα λειτουργίας τομής

Διαφορά

Η λειτουργία διαφοράς (difference) εφαρμόζεται σε δύο σχέσεις με τις ίδιες ιδιότητες. Οι πλειάδες της προκύπτουσας σχέσης είναι αυτές που περιέχονται στην πρώτη σχέση αλλά όχι στη δεύτερη.

```
select *  
from ΣΧΕΣΗ1  
minus  
select *  
from ΣΧΕΣΗ2
```



Εικόνα 14.15 Παράδειγμα λειτουργίας διαφοράς

14-7 ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

Ο σχεδιασμός οποιασδήποτε βάσης δεδομένων είναι μια χρονοβόρα και δύσκολη εργασία που μπορεί να γίνει μόνο με μια διαδικασία βήμα προς βήμα. Το πρώτο βήμα συνήθως περιλαμβάνει πολλές συνεντεύξεις πιθανών χρηστών της βάσης δεδομένων. Το δεύτερο βήμα είναι η δόμηση ενός μοντέλου οντοτήτων-σχέσεων (**Entity-Relation Model, ERM**) που ορίζει τις οντότητες, τις ιδιότητες αυτών των οντοτήτων, καθώς και τις μεταξύ τους σχέσεις.

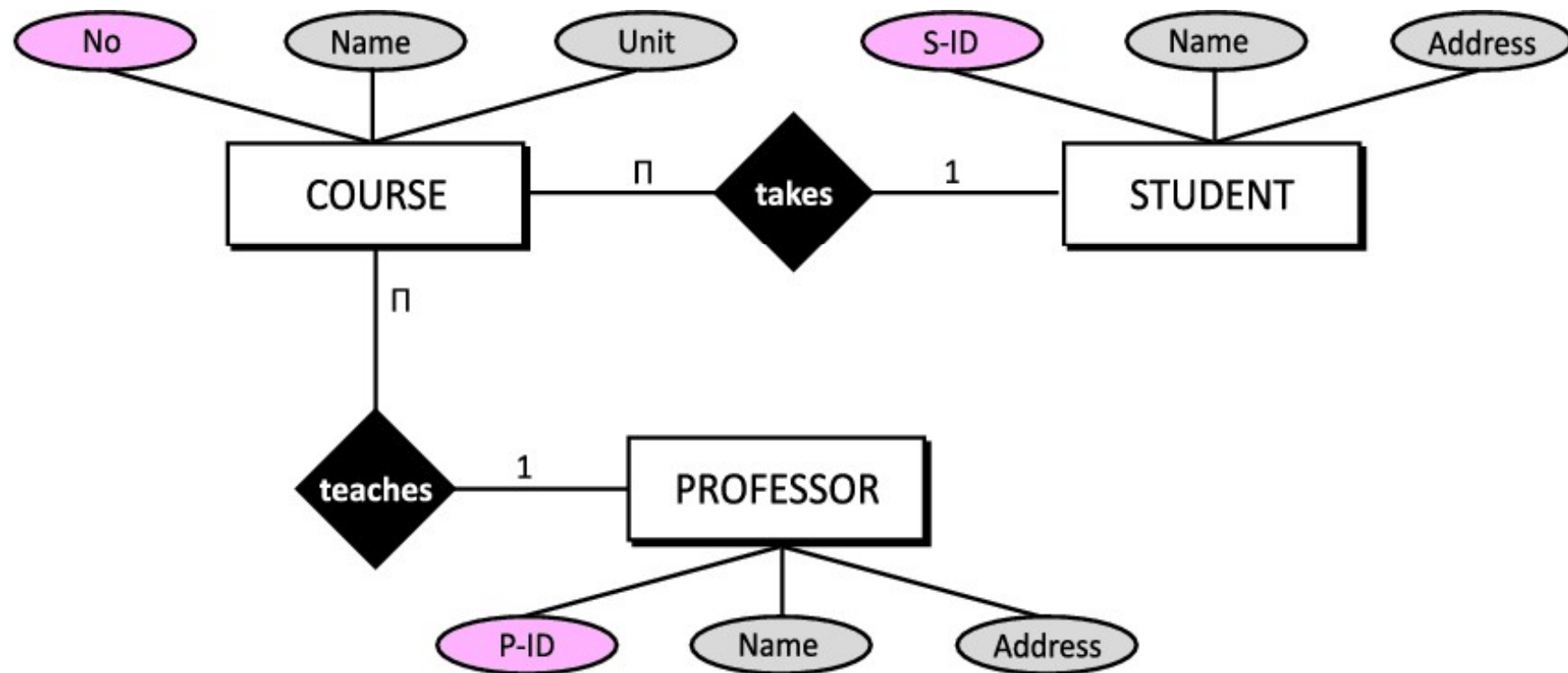
Μοντέλα οντοτήτων-σχέσεων

Σε αυτό το βήμα ο σχεδιαστής της βάσης δεδομένων δημιουργεί ένα διάγραμμα **οντοτήτων-σχέσεων** (Ο-Σ) για να απεικονίσει τις οντότητες για τις οποίες πρέπει να αποθηκεύονται πληροφορίες και τις συσχετίσεις (relationships) μεταξύ αυτών των οντοτήτων. Στα διαγράμματα Ο-Σ χρησιμοποιούνται αρκετά γεωμετρικά σχήματα, αν και εδώ χρησιμοποιούμε μόνο μερικά από αυτά:

- ❑ Τα ορθογώνια αντιπροσωπεύουν σύνολα οντοτήτων
- ❑ Οι ελλείψεις αντιπροσωπεύουν ιδιότητες
- ❑ Οι ρόμβοι αντιπροσωπεύουν σύνολα συσχετίσεων
- ❑ Οι γραμμές συνδέουν ιδιότητες με σύνολα οντοτήτων και σύνολα οντοτήτων με σύνολα συσχετίσεων

Παράδειγμα 14.1

Στην Εικόνα 14.16 φαίνεται ένα πολύ απλό διάγραμμα Ο-Σ με τρία σύνολα οντοτήτων, τις ιδιότητές τους, και τις συσχετίσεις μεταξύ αυτών των οντοτήτων.



Εικόνα 14.16 Οντότητες, ιδιότητες, και συσχετίσεις σε ένα διάγραμμα Ο-Σ

Από τα διαγράμματα Ο-Σ στις σχέσεις

Μετά την ολοκλήρωση του διαγράμματος Ο-Σ, μπορούν να δημιουργηθούν οι σχέσεις (πίνακες) της σχεσιακής βάσης δεδομένων.

Σχέσεις για σύνολα οντοτήτων

Για κάθε οντότητα που ορίζεται στο διάγραμμα Ο-Σ πρέπει να δημιουργηθεί μια σχέση (ένας πίνακας) στην οποία θα υπάρχουν n στήλες που σχετίζονται με n ιδιότητες του συγκεκριμένου συνόλου.

Παράδειγμα 14.2

Για κάθε σύνολο οντοτήτων που έχουμε ορίσει στην Εικόνα 14.16 μπορούμε να δημιουργήσουμε τρεις σχέσεις (πίνακες), όπως φαίνεται στην εικόνα 14.17.

COURSE		
No	Name	Unit
•	•	•
•	•	•
•	•	•

STUDENT		
S-ID	Name	Address
•	•	•
•	•	•
•	•	•

PROFESSOR		
P-ID	Name	Address
•	•	•
•	•	•
•	•	•

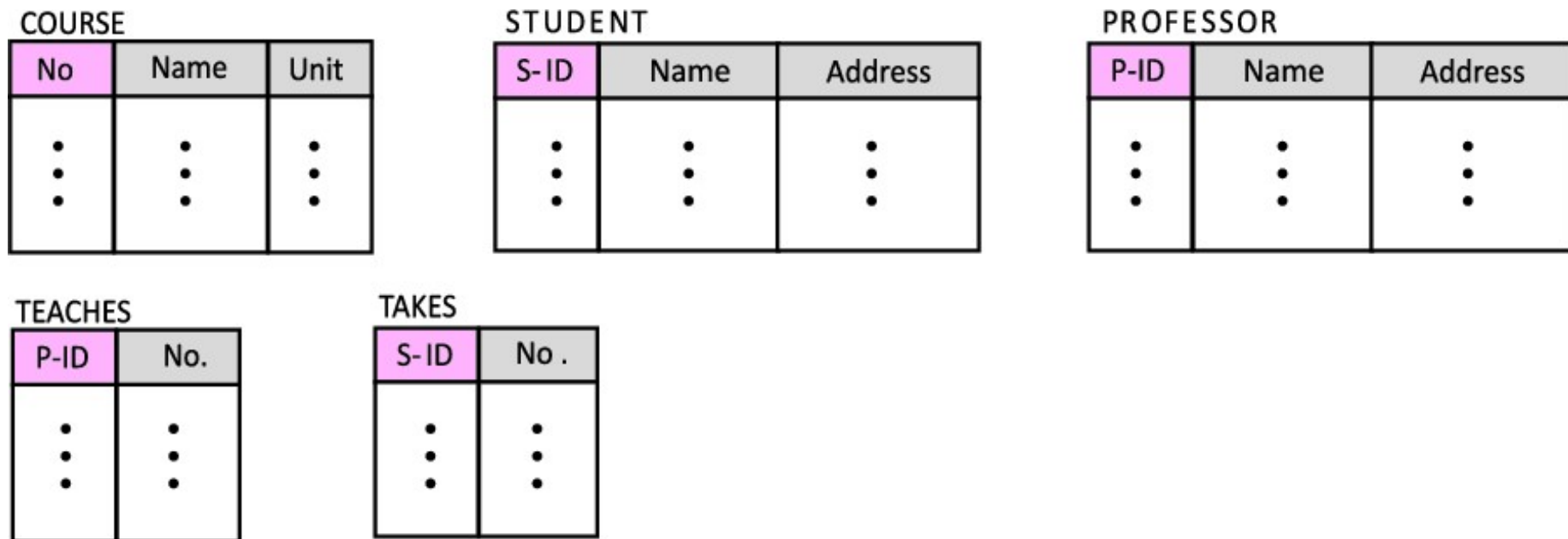
Εικόνα 14.17 Σχέσεις για το σύνολο οντοτήτων της Εικόνας 14.16

Σχέσεις για σύνολα συσχετίσεων

Για κάθε σύνολο συσχετίσεων (relationships) στο διάγραμμα Ο-Σ πρέπει να δημιουργηθεί μια σχέση (πίνακας). Αυτή η σχέση έχει μία στήλη για το κλειδί κάθε συνόλου οντοτήτων που εμπλέκεται σε αυτή τη συσχέτιση και μία στήλη για κάθε ιδιότητα της ίδιας της συσχέτισης, εφόσον βέβαια η συσχέτιση διαθέτει ιδιότητες (το οποίο δεν ισχύει στο παράδειγμά μας).

Παράδειγμα 14.3

Στην Εικόνα 14.16 υπάρχουν δύο σύνολα συσχετίσεων, TEACHES (διδάσκει) και TAKES (παίρνει), καθένα από τα οποία συνδέεται με δύο σύνολα οντοτήτων. Στην Εικόνα 14.18 φαίνεται η προσθήκη των σχέσεων (των πινάκων) για αυτά τα σύνολα συσχετίσεων στις σχέσεις που έχουμε ήδη δημιουργήσει.



Εικόνα 14.18 Οι σχέσεις για το διάγραμμα Ο-Σ της Εικόνας 14.16

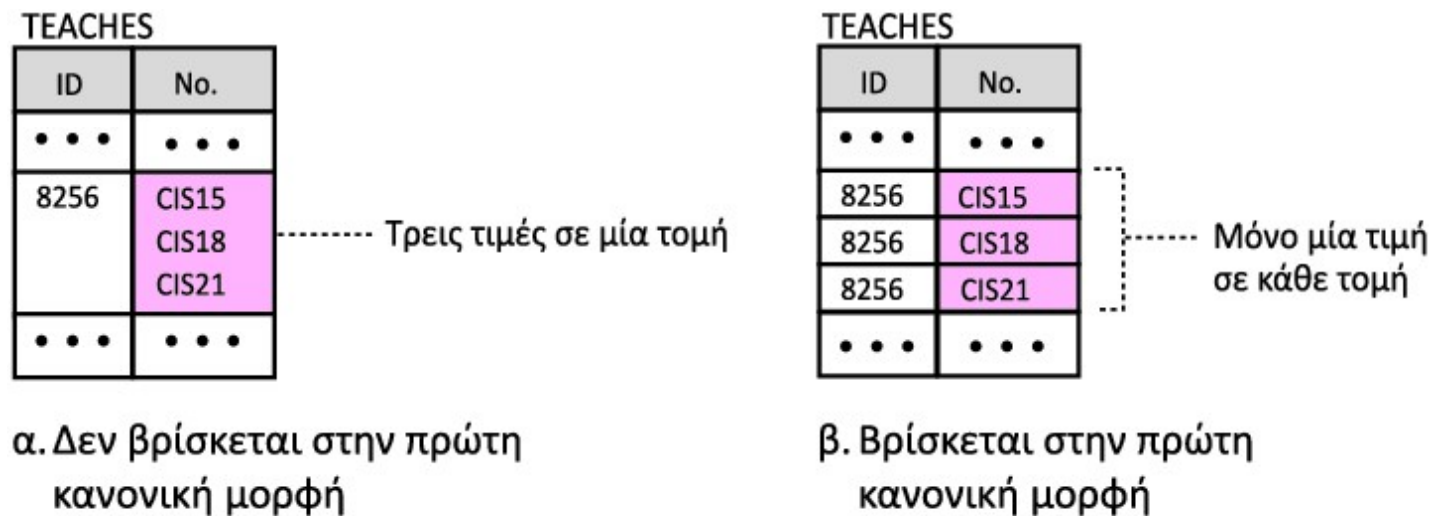
Κανονικοποίηση

Κανονικοποίηση ονομάζεται η διαδικασία κατά την οποία ένα καθορισμένο σύνολο σχέσεων (πινάκων) μετασχηματίζεται σε ένα νέο σύνολο σχέσεων με πιο "συμπαγή" δομή. Η κανονικοποίηση είναι απαραίτητη προκειμένου να είναι εφικτή η αναπαράσταση οποιασδήποτε σχέσης στη βάση δεδομένων, η χρήση γλωσσών όπως η SQL για την εφαρμογή ισχυρών λειτουργιών ανάκτησης που αποτελούνται από ατομικές λειτουργίες, η αποφυγή ανωμαλιών κατά την εισαγωγή, τη διαγραφή, και την ενημέρωση, και η εξάλειψη της ανάγκης αναδόμησης της βάσης δεδομένων κατά την προσθήκη νέων τύπων δεδομένων.

Κατά τη διαδικασία της κανονικοποίησης ορίζεται ένα σύνολο ιεραρχικών κανονικών μορφών (normal forms, NF). Έχουν προταθεί αρκετές κανονικές μορφές, όπως 1NF, 2NF, 3NF, BCNF (Boyce-Codd Normal Form), 4NF, PJNF (Projection/Joint Normal Form), 5NF, και ούτω καθεξής.

Πρώτη κανονική μορφή (1NF)

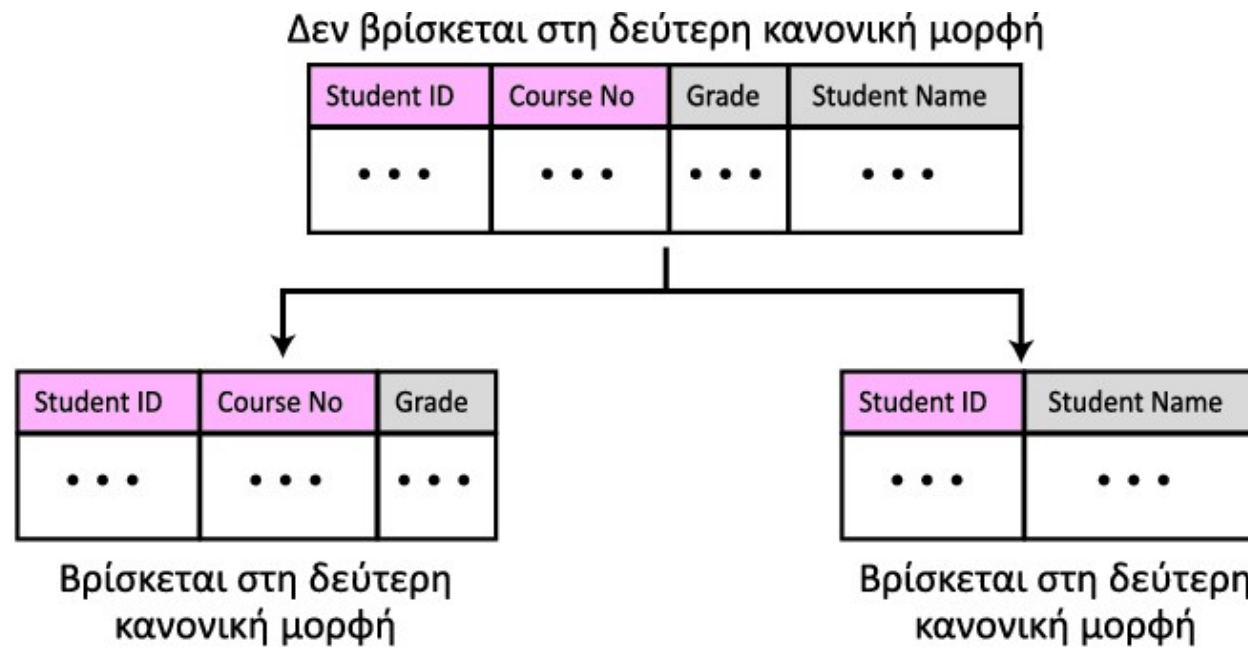
Κατά τον μετασχηματισμό οντοτήτων ή συσχετίσεων σε πίνακες, είναι πιθανό να προκύψουν μερικοί πίνακες στους οποίους υπάρχουν περισσότερες τιμές στην τομή μιας γραμμής ή στήλης.



Εικόνα 14.19 Παράδειγμα πρώτης κανονικής μορφής

Δεύτερη κανονική μορφή

Σε κάθε σχέση πρέπει να υπάρχει ένα κλειδί (που ονομάζεται πρωτεύον κλειδί) από το οποίο να εξαρτώνται όλες οι άλλες ιδιότητες (τιμές στηλών). Για παράδειγμα, όταν δίνεται το αναγνωριστικό (ID) ενός σπουδαστή, θα πρέπει να είμαστε σε θέση να εντοπίζουμε το όνομά του.



Εικόνα 14.20 Παράδειγμα δεύτερης κανονικής μορφής

Άλλες κανονικές μορφές

Στις άλλες κανονικές μορφές χρησιμοποιούνται πιο περίπλοκες εξαρτήσεις μεταξύ των ιδιοτήτων. Για τον λόγο αυτό αφήνουμε το συγκεκριμένο θέμα για βιβλία που ασχολούνται αποκλειστικά με βάσεις δεδομένων.

14-8 ΑΛΛΑ ΜΟΝΤΕΛΑ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ

Σήμερα το σχεσιακό μοντέλο βάσεων δεδομένων δεν είναι και το μοναδικό που χρησιμοποιείται. Δύο άλλα γνωστά μοντέλα βάσεων δεδομένων είναι το **κατανεμημένο** και το **αντικειμενοστρεφές**, τα οποία θα εξετάσουμε συνοπτικά σε αυτή την ενότητα.

Κατανεμημένες βάσεις δεδομένων

Το μοντέλο των κατανεμημένων βάσεων δεδομένων (distributed databases) στην πραγματικότητα δεν είναι καινούργιο, αλλά βασίζεται στο σχεσιακό μοντέλο. Ωστόσο, τα δεδομένα αποθηκεύονται σε πολλούς υπολογιστές που επικοινωνούν μέσω του Internet ή κάποιου ιδιωτικού δικτύου ευρείας περιοχής. Κάθε υπολογιστής (ή τοποθεσία, site) διατηρεί μέρος της βάσης δεδομένων ή και ολόκληρη τη βάση.

Καταμημένες κατανεμημένες βάσεις δεδομένων

Σε μια καταμημένη κατανεμημένη βάση δεδομένων τα δεδομένα είναι αποθηκευμένα τοπικά. Δηλαδή, τα δεδομένα είναι αποθηκευμένα στην τοποθεσία από την οποία χρησιμοποιούνται. Ωστόσο, αυτό δεν σημαίνει ότι μια τοποθεσία δεν μπορεί να προσπελάσει δεδομένα άλλων τοποθεσιών. Η προσπέλαση είναι κυρίως τοπική, αλλά και περιστασιακά καθολική.

Κατανεμημένες βάσεις δεδομένων αντιγράφων

Σε μια κατανεμημένη βάση δεδομένων αντιγράφων κάθε τοποθεσία διατηρεί ένα ακριβές αντίγραφο μιας άλλης τοποθεσίας. Κάθε τροποποίηση που γίνεται στα δεδομένα μιας τοποθεσίας επαναλαμβάνεται και στις υπόλοιπες τοποθεσίες. Ο λόγος ύπαρξης μιας τέτοιας βάσης δεδομένων είναι η ασφάλεια. Αν το σύστημα μιας τοποθεσίας παρουσιάσει πρόβλημα, οι χρήστες της μπορούν να προσπελάζουν τα δεδομένα μιας άλλης τοποθεσίας.

Αντικειμενοστρεφείς βάσεις δεδομένων

Μια αντικειμενοστρεφής βάση δεδομένων (object-oriented database) προσπαθεί να διατηρεί τα πλεονεκτήματα του σχεσιακού μοντέλου επιτρέποντας παράλληλα στις εφαρμογές να προσπελάζουν δομημένα δεδομένα. Σε μια αντικειμενοστρεφή βάση δεδομένων ορίζονται αντικείμενα και οι σχέσεις μεταξύ τους. Επιπλέον, κάθε αντικείμενο μπορεί να διαθέτει ιδιότητες οι οποίες εκφράζονται ως πεδία.

XML

Η γλώσσα ερωτημάτων που συνήθως χρησιμοποιείται για αντικειμενοστρεφείς βάσεις δεδομένων είναι η XML (Extensible Markup Language, Επεκτάσιμη Γλώσσα Σήμανσης). Όπως αναφέραμε στο Κεφάλαιο 6, η XML αρχικά σχεδιάστηκε για την προσθήκη πληροφοριών σήμανσης σε έγγραφα κειμένου, όμως εφαρμόζεται και ως γλώσσα ερωτημάτων σε βάσεις δεδομένων. Η XML μπορεί να αναπαριστά δεδομένα με ένθετη δομή.