



# Βάσεις Δεδομένων I

## Ενότητα 2: Εισαγωγή στις βάσεις δεδομένων

Χ. Σκουρλάς



Το περιεχόμενο του μαθήματος διατίθεται με άδεια Creative Commons εκτός και αν αναφέρεται διαφορετικά



# Εισαγωγή στις βάσεις δεδομένων

## Πρέπει να μάθετε τις έννοιες:

Βάσεις δεδομένων

Συστήματα Βάσεων Δεδομένων

Συστήματα Διαχείρισης Βάσεων  
Δεδομένων

Μοντέλο Οντοτήτων Συσχετίσεων

Σχεσιακό μοντέλο δεδομένων

## Ιστορική αναδρομή

Ted Codd: Προτείνει το σχεσιακό μοντέλο

Peter Chen: Προτείνει το Μοντέλο Οντοτήτων  
Συσχετίσεων

# Περιγραφή Μαθήματος

- Σκοπός του μαθήματος είναι να παρουσιάσει εισαγωγικές έννοιες της τεχνολογίας των βάσεων δεδομένων και των συστημάτων βάσεων δεδομένων.
- Θα παρουσιασθούν οι παρακάτω έννοιες:  
Βάσεις δεδομένων, Συστήματα Βάσεων Δεδομένων, Αρχιτεκτονική συστημάτων βάσεων δεδομένων, Συστήματα Διαχείρισης Βάσεων Δεδομένων, Μοντέλα δεδομένων (ιστορικά μοντέλα, Ιεραρχικό, Δικτυωτό), Σχεσιακό μοντέλο

**Χ. Σκουρλάς**

# Στόχος Ενότητας

- Κύριος στόχος του μαθήματος είναι να εφοδιάσει τους φοιτητές με τις απαραίτητες γνώσεις έτσι ώστε να είναι ικανοί να κατανοήσουν σε επόμενα μαθήματα βασικές έννοιες σχεδιασμού και υλοποίησης βάσεων δεδομένων και συστημάτων βάσεων δεδομένων.

# Λέξεις κλειδιά

- Βάσεις δεδομένων
- Συστήματα Βάσεων Δεδομένων
- Συστήματα Διαχείρισης Βάσεων Δεδομένων
- Αρχιτεκτονική συστημάτων βάσεων δεδομένων
- Ιεραρχικό μοντέλο
- Δικτυωτό μοντέλο
- Σχεσιακό μοντέλο δεδομένων

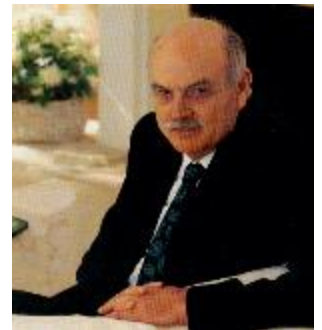
# Δύο ιστορικοί σταθμοί

1. Codd, E. F. (1970), A relational model for large shared data banks. Comm. ACM , 13:6 , pp 377-387.
2. Peter Chen (March 1976), The Entity-Relationship Model - Toward a Unified View of Data. ACM Transactions on Database Systems 1 (1): pp 9–36.

# Η εισαγωγή των σχεσιακών βάσεων δεδομένων οφείλεται στον Ted Codd

“Following a famous paper (πρόκειται για το Codd , E. F., A relational model for large shared data banks, Comm. ACM, 13:6, pp. 377-387) written by Ted Codd in 1970 database system changed significantly. Codd proposed the Database systems should present the user with a view of data organized as tables called relations” (p. 4)

(απόσπασμα από το σύγγραμμα των Ullman - Widrow)



# Relational Model

**Το 1970, ορίστηκε το Σχεσιακό Μοντέλο - Relational Model στο άρθρο: Codd, “A Relational Model of Data for Large Shared Data Banks”, Comm. ACM, 13:6, pp. 377-387**

- “It provides a means of describing data with its natural structure only--that is, without superimposing any additional structure for machine representation purposes. Accordingly, it provides a basis for a high level data language which will yield maximal independence between programs on the one hand and machine representation on the other.”(Codd 1970)



# Η εισαγωγή του Μοντέλου Οντοτήτων Συσχετίσεων οφείλεται στον Peter Chen

βλέπε άρθρο:

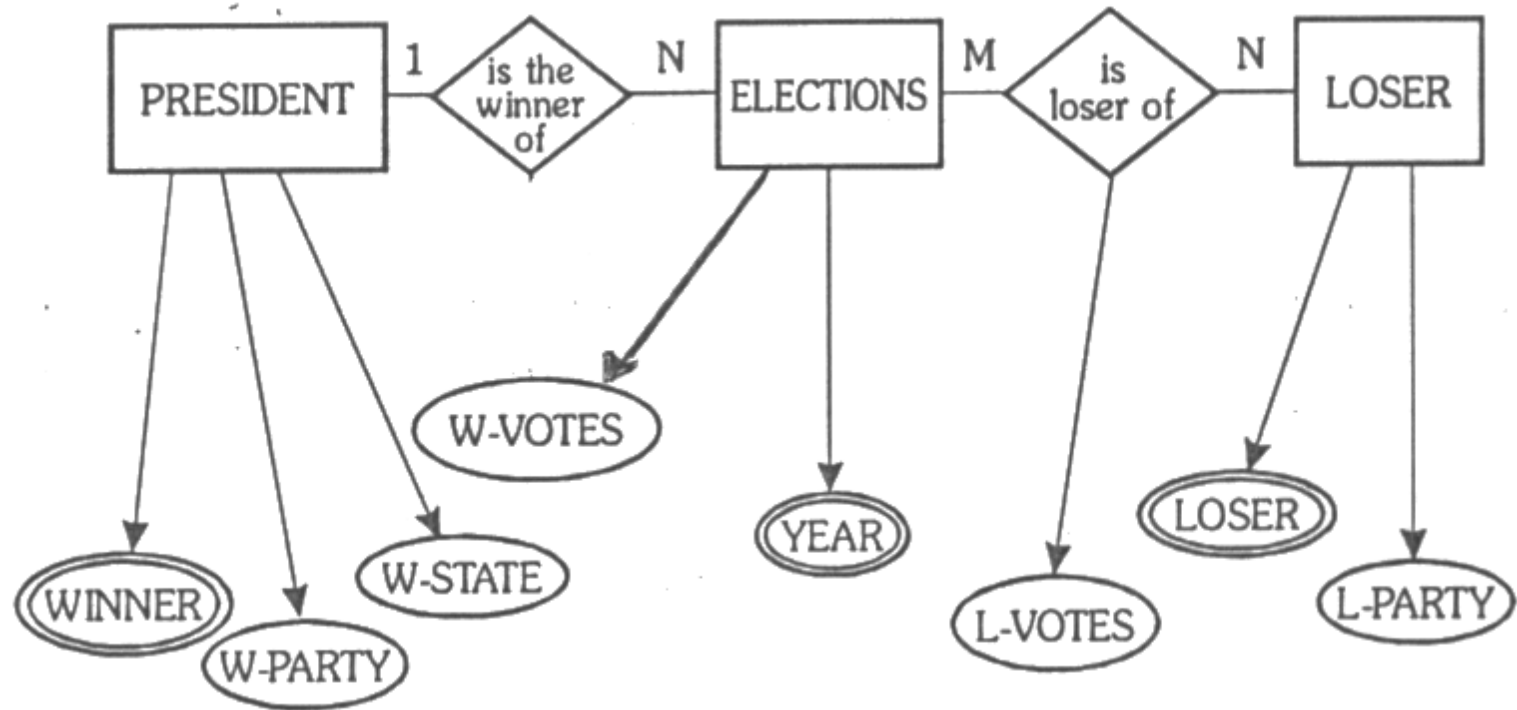
Peter Chen (March 1976), “The Entity-Relationship Model – Toward a Unified View of Data”, ACM Transactions on Database Systems, 1:1, pp. 9–36.

Στη συνέχεια βλέπουμε κάποια σύμβολα που χρησιμοποίησε ο Chen.



# Συμβολισμός Peter Chen

## Παράδειγμα: American Elections



# Συνοπτική ιστορία των βάσεων δεδομένων

- **Απαρχές:** Οι ρίζες πάνε πίσω σε βιβλιοθήκες, κυβερνητικά, επιχειρηματικά και ιατρικά αρχεία. Υπάρχει μια πολύ μακρά ιστορία αποθήκευσης πληροφοριών, ευρετηρίασης και ανάκτησης.
- **1960's:** Οι Υπολογιστές γίνονται οικονομικά αποδοτικοί. Δύο βασικά μοντέλα δεδομένων αναπτύχθηκαν: το **μοντέλο του δικτύου (network model)** **CODASYL** και το **ιεραρχικό (hierarchical) IMS**. Η πρόσβαση στη βάση δεδομένων είναι μέσω λειτουργιών χαμηλού επιπέδου δεικτών που συνδέουν τις εγγραφές (“Access to database is through low-level pointer operations linking records”).
- **1970-72:** E.F. Codd προτείνει το σχεσιακό μοντέλο (**relational model**)
- **1970's:** Πώς η θεωρία οδηγεί σε βέλτιστες πρακτικές.  
**Ingres:** Αναπτύχθηκε στην UCB. Το σύστημα χρησιμοποιεί QUEL ως γλώσσα ερωτημάτων. **Σύστημα R:** Αναπτύχθηκε στην IBM. Το σύστημα χρησιμοποιεί SEQUEL ως γλώσσα ερωτημάτων.
- **1976:** P. Chen προτείνει το **Entity-Relationship (ER) model** για σχεδίαση βάσεων (database design)

John Vaughn, “A Short Database History”

[math.hws.edu/vaughn/cpsc/343/2003/history.html](http://math.hws.edu/vaughn/cpsc/343/2003/history.html)

# Συνοπτική ιστορία των βάσεων δεδομένων

- **Early 1980's:** Σχεσιακά προϊόντα διαχείρισης βάσεων
- **Mid-1980's:** SQL (Structured Query Language). Το πρότυπο ("standard").
- **Early 1990's:** PowerBuilder (Sybase), Oracle Developer, VB (Microsoft), ... Client-server model. Excel/Access (MS). ODBC. Object Database Management Systems (ODBMS) prototypes.
- **Mid-1990's:** Internet/WWW. «Εκτίναξη» του Client-server. Web/DB εκθετική ανάπτυξη.
- **Late-1990's:** Active Server Pages, Front Page, Java Servlets, JDBC, Enterprise Java Beans, ColdFusion, Dream Weaver, Oracle Developer 2000, ....
- Open source: Apache, MySQL, ...
- Online Transaction processing (OLTP), online analytic processing (OLAP).
- **Early 21st century:** Συνεχής αύξηση εφαρμογών των βάσεων. Χρήση PDAs, POS transactions, ...
- Μεγάλοι παίκτες στην αγορά βάσεων (DB market): IBM, Microsoft, Oracle.
- **Future trends:** ???

# Τρεις βασικές έννοιες

- Δεδομένα (Data)
- Πληροφορία (information)
- Γνώση (knowledge)



# Στις βάσεις δεδομένων διαχειριζόμαστε δεδομένα

## Πρέπει να μάθετε τις έννοιες:

Δεδομένα (data). Τα λέμε και στοιχεία.

Μοντέλο δεδομένων

## Επεξήγηση με παράδειγμα

Θα δείτε τα Δεδομένα μιας τραπεζικής εφαρμογής

Θα τα δείτε τα δεδομένα οργανωμένα σε Σχεσιακό μοντέλο, δηλαδή σε πίνακες. Ο πίνακας «Πελάτης» έχει τα στοιχεία του πελάτη. Ο πίνακας «Κινήσεις\_Πελάτη» έχει τις αγορές των πελατών με πιστωτική κάρτα.

## CUSTOMER

| C_CODE | C_NAME | AGE | CREDIT_CARD_N0 | ... |
|--------|--------|-----|----------------|-----|
| 12345  | ULMAN  | 78  | 123456789      | ... |
| 23456  | DATE   | 37  | 987654321      | ... |
| ...    | ...    | ... | ...            | ... |

## CUSTOMER\_TRANSACTIONS

| TRANSACTION_NO | C_N0  | AMOUNT  | RATIONALE                  | T_DATE   | ... |
|----------------|-------|---------|----------------------------|----------|-----|
| ...            | ...   | ...     | ...                        | ...      | ... |
| 1010010        | 12345 | 35.40   | CLOTHS                     | 10.10.18 | ... |
| 1010020        | 12345 | 26.70   | DVDs                       | 20.10.18 | ... |
| 1010030        | 12345 | 1250.10 | CONFERENCE<br>REGISTRATION | 01.11.18 | ... |
| 1010040        | 12345 | 23.70   | CLOTHS                     | 08.11.18 | ... |
| ...            | ...   | ...     | ...                        | ...      | ... |

# Πληροφορία

- Ο υπάλληλος που ελέγχει τις κινήσεις εκτυπώνει καταστάσεις σε μηνιαία βάση (ή και συχνότερα) ανά πελάτη.
- Επιπλέον, έχει συγκριτικά στοιχεία κίνησης κάρτας ανά μήνα.
- Αυτές οι καταστάσεις και τα στοιχεία που περιλαμβάνονται σε αυτές είναι η πληροφορία που έχει ανά πελάτη.



# Γνώση

Ο υπάλληλος έχει μία λίστα με καταγεγραμμένα τα σημεία που πρέπει να προσέξει.

## Παράδειγμα

«Αν ο πελάτης έχει ασυνήθιστα μεγάλη χρέωση τον τρέχοντα μήνα σε σχέση με τους προηγούμενους τότε πρέπει να εξετάσεις το ζήτημα! Κάτι συμβαίνει!».

**Αυτή η λίστα περιλαμβάνει την καταγεγραμμένη (ρητή) γνώση.**

- Ένας έμπειρος υπάλληλος διαπιστώνει ότι ο κάτοχος της Κάρτας είναι απόφοιτος πρωτοβάθμιας εκπαίδευσης, ηλικίας 78 ετών και πληρώνει με κάρτα επιστημονικό συνέδριο.
- Αν και η λίστα δεν περιλαμβάνει κάτι σχετικό ο υπάλληλος γνωρίζει ότι η πληροφορία αυτή πρέπει να διερευνηθεί.

**Άρρητη γνώση που πρέπει να καταγραφεί.**



# Τι είναι οι βάσεις δεδομένων

## Πρέπει να μάθετε τις έννοιες:

Βάσεις Δεδομένων (databases)

Σχεσιακές βάσεις δεδομένων  
(relational databases)

## Επεξήγηση με παράδειγμα

Θα πρέπει να σκεφθείτε μια εκπαιδευτική βάση δεδομένων. Στο σχεσιακό μοντέλο η βάση δεδομένων είναι πίνακες που έχουν τα στοιχεία των σπουδαστών, τα στοιχεία των καθηγητών, τις βαθμολογίες, τις εγγραφές των σπουδαστών κ.λπ.

# Βάση δεδομένων («απλή» προσέγγιση)

Μία βάση δεδομένων είναι ένα είδος ηλεκτρονικής αρχειοθέτησης δεδομένων (στοιχείων) - data ενός οργανισμού ή μιας επιχείρησης ή ακόμη και ενός φυσικού προσώπου, π.χ. ενός μεμονωμένου επαγγελματία.

Μαζί με τα στοιχεία υπάρχει το σύνολο των εφαρμογών που επιτρέπουν στους χρήστες της βάσης να καταχωρήσουν και να ανακτήσουν τα στοιχεία αυτά.

Σε μία εκπαιδευτική βάση παράδειγμα Δεδομένων είναι τα στοιχεία σπουδαστών και καθηγητών, οι βαθμολογίες, οι δηλώσεις των μαθημάτων κατά την εγγραφή του σπουδαστή κ.λπ.

Παράδειγμα εφαρμογής είναι τα προγράμματα που «αναλαμβάνουν» τη διαχείριση των βαθμολογιών των σπουδαστών.

# Σύστημα Βάσης Δεδομένων

Ένα Σύστημα Βάσης Δεδομένων είναι ένα σύστημα καταχώρησης, ενημέρωσης και ανάκτησης δεδομένων βασιζόμενο σε υπολογιστή και αποτελείται από τέσσερις συνιστώσες:

- Δεδομένα (Data). Προσοχή στη διαφορά των εννοιών: Δεδομένα, Πληροφορία και Γνώση.
- Υλικό (Hardware)
- Λογισμικό (Software) , με κυριότερο στοιχείο του το Σύστημα Διαχείρισης Βάσεων Δεδομένων (π.χ., Oracle, mySQL) και βέβαια τις εφαρμογές για τον τελικό χρήστη.
- Χρήστες (Users)

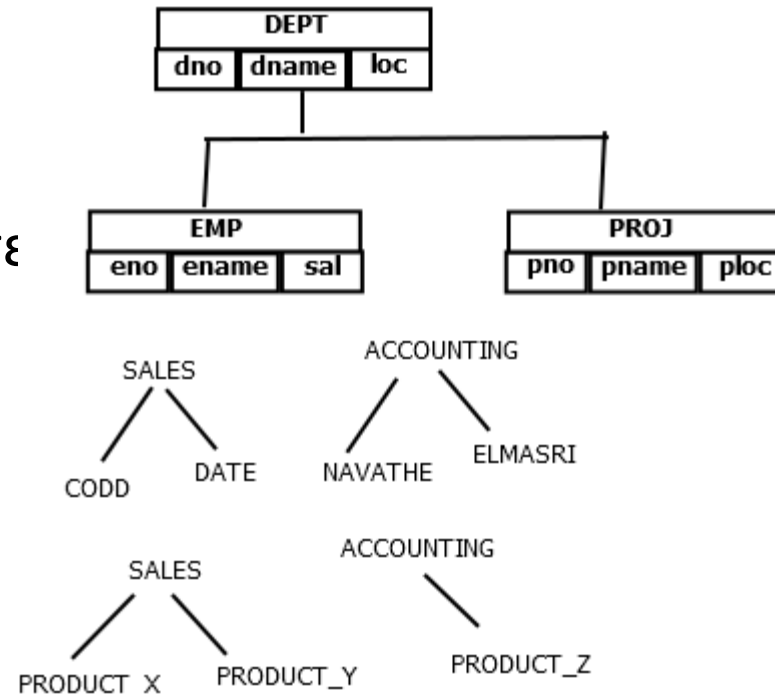
# Γιατί χρησιμοποιούμε βάση δεδομένων

Ο βασικός λόγος χρησιμοποίησης συστημάτων βάσεων δεδομένων σε μία επιχείρηση ή ένα οργανισμό είναι το γεγονός ότι ένα τέτοιο σύστημα οργάνωσης και ανάκτησης δεδομένων, “οικοδομείται” με την εποπτεία και την καθοδήγηση του οργανισμού ή της επιχείρησης ώστε να καλύψει τις ανάγκες του συνόλου των τμημάτων και εφοδιάζει τον οργανισμό ή την επιχείρηση με Κεντρικό έλεγχο (centralized control) των λειτουργικών στοιχείων (operational data) του.

**Τι είναι βάση δεδομένων στα διάφορα  
μοντέλα δεδομένων**

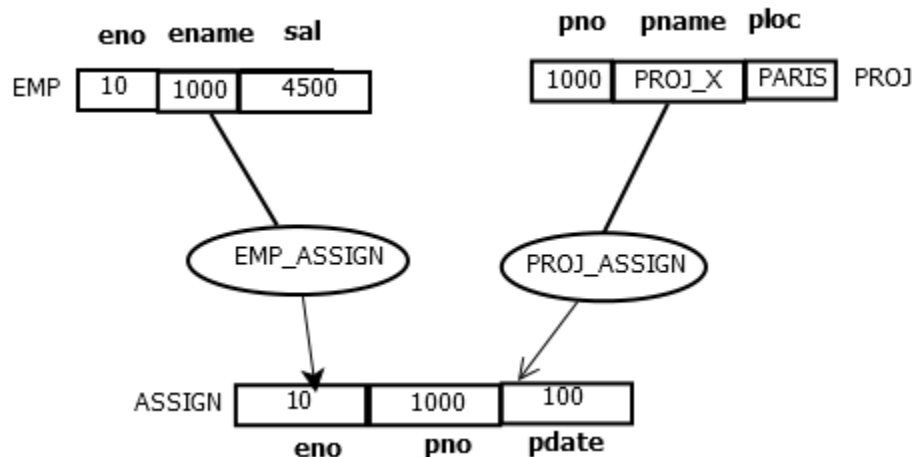
# Ιεραρχική βάση δεδομένων

- Μία ιεραρχική βάση δεδομένων είναι μία βάση που έχει “οικοδομηθεί” σαν ένα δέντρο συσχετισμένων τύπων εγγραφών (record types).
- Όλες οι ιεραρχικές βάσεις δεδομένων έχουν μία και μόνο μία “ρίζα”. Αυτή η “ρίζα” - τύπος δεδομένων μπορεί να συμμετέχει σε 1:M συσχετίσεις με οποιοδήποτε αριθμό τύπων δεδομένων - “παιδιών”. Κάθε “παιδί” μπορεί επίσης να έχει “παιδιά” - τύπους δεδομένων.



# Δικτυωτή βάση δεδομένων

- Μια δικτυωτή βάση δεδομένων αποτελείται από μία συλλογή τύπων εγγραφών. Οι τύποι εγγραφών συνδέονται μεταξύ τους με ειδικές συνδέσεις (links).
- Μία σύνδεση αναπαριστά μία συσχέτιση 1:M ανάμεσα σε μία εγγραφή ενός τύπου (που ονομάζεται Ιδιοκτήτης - “owner”) και σε ένα σύνολο εγγραφών (set of records) ενός άλλου τύπου (που ονομάζονται μέλη - “the members”), π.χ. Assign είναι μέλος και EMP, PROJ ιδιοκτήτες.





# Διαφορά

- Η κύρια διαφορά μιας δικτυωτής βάσης δεδομένων από την ιεραρχική βάση συνίσταται στο γεγονός ότι δεν υπάρχει περιορισμός στον αριθμό και την κατεύθυνση των συνδέσεων που μπορούμε να ορίσουμε ανάμεσα σε τύπους εγγραφών. Επίσης, δεν υπάρχει ανάγκη για τύπο εγγραφής “ρίζα”.



# Σχεσιακό μοντέλο

## Σχεσιακή βάση δεδομένων

### Πρέπει να μάθετε τις έννοιες:

Σχεσιακό μοντέλο. Όλα τα βλέπουμε σαν πίνακες  
Σχεσιακή βάση δεδομένων. Πρέπει να γράψουμε προγράμματα για να κατασκευάσουμε τους πίνακες και να γράφουμε τα δεδομένα-στοιχεία σε αυτούς

### Επεξήγηση με παράδειγμα

Θα δείτε μια βάση εφοδιασμού ανταλλακτικών από προμηθευτές. Θα δούμε 4 προμηθευτές και 4 ανταλλακτικά. Θα δούμε και 6 εφοδιασμούς.

# Σχεσιακό μοντέλο βάση δεδομένων - Παράδειγμα

## Προμηθευτές (SUPPLIERS)

| Κωδικός(SNO) | Επωνυμία(SNAME) | έδρα(SCITY) |
|--------------|-----------------|-------------|
| S010         | FIAT            | ΛΟΝΔΙΝΟ     |
| S020         | OPEL            | ΠΑΡΙΣΙ      |
| S030         | FORD            | ΠΑΡΙΣΙ      |
| S040         | PORCHE          | ΑΘΗΝΑ       |

## Ανταλλακτικά (PARTS)

| PNO   | PNAME  | PCOLOR  | PCITY   |
|-------|--------|---------|---------|
| P0100 | ΒΙΔΑ Α | ΚΟΚΚΙΝΗ | ΛΟΝΔΙΝΟ |
| P0200 | ΒΙΔΑ Β | ΚΟΚΚΙΝΗ | ΠΑΡΙΣΙ  |
| P0300 | ΒΙΔΑ C | ΚΟΚΚΙΝΗ | ΡΩΜΗ    |
| P0400 | ΒΙΔΑ C | ΚΙΤΡΙΝΗ | ΛΟΝΔΙΝΟ |

## Εφοδιασμός (SHIPMENT)

| SNO  | PNO   | QTY | SDATE      |
|------|-------|-----|------------|
| S010 | P0100 | 300 | 16/10/2014 |
| S010 | P0200 | 200 | 16/10/2014 |
| S010 | P0300 | 400 | 17/10/2014 |
| S020 | P0100 | 300 | 17/10/2014 |
| S020 | P0200 | 400 | 17/10/2014 |
| S030 | P0200 | 200 | 17/10/2014 |

# Έννοιες σχεσιακών Β.Δ.

Σκοπός μας είναι να παρουσιάσουμε τώρα κάποιες πρώτες έννοιες σχεσιακών βάσεων δεδομένων (relational database) ώστε να κατανοήσουμε αυτά που θα συζητήσουμε στη συνέχεια:

- Αρχιτεκτονική βάσεων δεδομένων,
- Data Definition Language,
- Data Manipulation Language,
- Data Control Language,
- Όψεις – views,
- Ευρετήρια – indexes.

Θα δοθούν παραδείγματα με χρήση γλώσσας SQL ώστε οι φοιτητές να κατανοήσουν και να εμβαθύνουν στις έννοιες.



# Σχεσιακό μοντέλο

## Σχεσιακή βάση δεδομένων

**Πρέπει να μάθετε καλά την έννοια:**

Σχεσιακή βάση δεδομένων.

Θυμηθείτε ότι τελικά θα πρέπει να γράψουμε προγράμματα για να κατασκευάσουμε τους πίνακες και να γράφουμε τα δεδομένα-στοιχεία σε αυτούς

### **Επεξήγηση με παράδειγμα**

Θα δείτε μια βάση προσωπικού με 2 πίνακες. Πίνακα με δεδομένα των τμημάτων μιας εταιρείας και πίνακα των υπαλλήλων της εταιρείας.

- Ο πίνακας dept έχει στήλες deptno, dname, loc και κάθε γραμμή του αποθηκεύει τα στοιχεία ενός τμήματος. Το πρώτο τμήμα έχει κωδικό (deptno)10, όνομα (dname) ACCOUNTING και έδρα (loc) ATHENS. Ο πίνακας emp έχει στήλες empno, ename, job, hiredate, mgr, sal, comm, deptno και κάθε γραμμή του αποθηκεύει τα στοιχεία ενός υπαλλήλου. Παρατηρήστε ότι οι δύο πίνακες είναι «συνδεδεμένοι» με τιμές των στηλών deptno που περιλαμβάνουν. Οι δύο πίνακες ανήκουν σε μία σχεσιακή βάση δεδομένων.

| Empno | Ename   | Job        | Hiredate | Mgr | Sal  | Comm | Deptno |
|-------|---------|------------|----------|-----|------|------|--------|
| 10    | CODD    | ANALYST    | 1/1/89   | 15  | 3000 |      | 10     |
| 15    | ELMASRI | ANALYST    | 2/5/95   | 15  | 1200 | 150  | 10     |
| 20    | NAVATHE | SALESMAN   | 7/7/77   | 20  | 2000 |      | 20     |
| 30    | DATE    | PROGRAMMER | 4/5/04   | 15  | 1800 | 200  | 10     |

emp (πίνακας υπαλλήλων)

dept (πίνακας τμημάτων)

| Deptno | Dname      | Loc    |
|--------|------------|--------|
| 10     | ACCOUNTING | ATHENS |
| 20     | SALES      | LONDON |
| 30     | RESEARCH   | ATHENS |
| 40     | PAYROLL    | LONDON |

# **Εμβόλιμη συζήτηση: Διαχείριση σχεσιακής βάσης δεδομένων με Προϊόντα Διαχείρισης Βάσης Δεδομένων**

Εισαγωγή στη χρήση του προϊόντος `mysql`



# Πως ο προγραμματιστής δημιουργεί μια Σχεσιακή βάση δεδομένων

**Πρέπει να μάθετε καλά τις δηλώσεις:**

CREATE TABLE, για να

δημιουργείτε τους πίνακες

INSERT INTO, για να γράφετε

γραμμές στους πίνακες

SELECT, για να βλέπετε γραμμές των πινάκων

## **Επεξήγηση με παράδειγμα**

Θα χρησιμοποιήσουμε τη γλώσσα SQL που

έχει το προϊόν MySQL για να γράψουμε

δηλώσεις. Με τις δηλώσεις θα διαχειριστούμε

μια σχεσιακή βάση προσωπικού με 2 πίνακες.

Πίνακα με δεδομένα των τμημάτων μιας

εταιρείας και πίνακα των υπαλλήλων της.



### emp (πίνακας υπαλλήλων)

| Empno | Ename   | Job        | Hiredate | Mgr | Sal  | Comm | Deptno |
|-------|---------|------------|----------|-----|------|------|--------|
| 10    | CODD    | ANALYST    | 1/1/89   | 15  | 3000 |      | 10     |
| 15    | ELMASRI | ANALYST    | 2/5/95   | 15  | 1200 | 150  | 10     |
| 20    | NAVATHE | SALESMAN   | 7/7/77   | 20  | 2000 |      | 20     |
| 30    | DATE    | PROGRAMMER | 4/5/04   | 15  | 1800 | 200  | 10     |

### dept (πίνακας τμημάτων)

| Deptno | Dname      | Loc    |
|--------|------------|--------|
| 10     | ACCOUNTING | ATHENS |
| 20     | SALES      | LONDON |
| 30     | RESEARCH   | ATHENS |

# Δημιουργήστε τη βάση και τους δύο πίνακες και εισάγετε στοιχεία

```
DROP DATABASE NEW_PERSONNEL;
CREATE DATABASE new_personnel;
USE new_personnel;
CREATE TABLE DEPT(DEPTNO INT(2) NOT NULL,
                   DNAME VARCHAR(14), LOC VARCHAR(14));
CREATE TABLE EMP(EMPNO INT(4) NOT NULL,
                   ENAME VARCHAR(10), JOB VARCHAR(25),
                   HIREDATE DATE, MGR INT(4), SAL FLOAT(7,2),
                   COMM FLOAT(7,2), DEPTNO INT(2));
SHOW TABLES;
INSERT INTO DEPT(DEPTNO, DNAME, LOC)
VALUES (10, 'ACCOUNTING', 'NEW YORK');
INSERT INTO EMP VALUES (10, 'CODD', 'ANALYST',
'1989/01/01', 15, 3000, NULL, 10);
```

# Δημιουργήστε τη βάση και τους δύο πίνακες και εισάγετε στοιχεία

**Δείτε τα στοιχεία των πινάκων της βάσης**

```
SELECT * FROM EMP;
```

```
SELECT * FROM DEPT;
```



# Πως ο προγραμματιστής δημιουργεί μια Σχεσιακή βάση δεδομένων – χρήση Oracle

**Πρέπει να μάθετε καλά τις εντολές:**

CREATE TABLE, για να δημιουργείτε  
τους πίνακες

INSERT INTO, για να γράφετε  
γραμμές στους πίνακες

SELECT, για να βλέπετε γραμμές των πινάκων

**Επεξήγηση με παράδειγμα**

Θα χρησιμοποιήσουμε τη γλώσσα SQL που  
έχει το προϊόν Oracle για να γράψουμε εντολές.

Με τις εντολές θα διαχειριστούμε την ίδια  
σχεσιακή βάση προσωπικού με 2 πίνακες.

Πίνακα με δεδομένα των τμημάτων μιας  
εταιρείας και πίνακα των υπαλλήλων της.

# **Διαχείριση σχεσιακής βάσης δεδομένων με Προϊόντα Διαχείρισης Βάσης Δεδομένων**

Εισαγωγή στη χρήση του προϊόντος της Oracle

# Δημιουργήστε τη βάση και τους δύο πίνακες και εισάγετε στοιχεία

```
CREATE TABLE DEPT (DEPTNO NUMBER(2) NOT NULL,  
    DNAME VARCHAR2(14), LOC VARCHAR2(14));  
CREATE TABLE EMP (EMPNO NUMBER(4) NOT NULL,  
    ENAME VARCHAR2(10), JOB VARCHAR2(25),  
    HIREDATE DATE, MGR NUMBER(4), SAL NUMBER(7,2),  
    COMM NUMBER(7,2), DEPTNO NUMBER(2));  
INSERT INTO DEPT (DEPTNO, DNAME, LOC)  
    VALUES (10, 'ACCOUNTING', 'NEW YORK');  
INSERT INTO EMP  
    VALUES (10, 'CODD', 'ANALYST',  
    '01/01/1989', 15, 3000, NULL, 10);
```

# Δημιουργήστε τη βάση και τους δύο πίνακες και εισάγετε στοιχεία

- **Δείτε τα στοιχεία των πινάκων της βάσης.**

```
SELECT * FROM EMP;
```

```
SELECT * FROM DEPT;
```

- **Πως θα διαγράψετε τους πίνακές σας**

```
DROP TABLE EMP;
```

```
DROP TABLE DEPT;
```



# Υπάρχουν μικροδιαφορές στον προγραμματισμό ανάλογα με το προϊόν που χρησιμοποιούμε

Πρέπει να μάθετε καλά τις εντολές σε Oracle και MySQL:

CREATE TABLE, για να δημιουργείτε

τους πίνακες

INSERT INTO, για να γράφετε

γραμμές στους πίνακες

SELECT, για να βλέπετε γραμμές

των πινάκων

## Επεξήγηση με παράδειγμα

Δείτε τον πίνακα διαφορών που ακολουθεί.



# Συγκριτικός Πίνακας διαφορών

| mySQL                                                                                                                                                                     | Oracle                                                                                                                                                                           |
|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| CREATE DATABASE new_personnel;                                                                                                                                            |                                                                                                                                                                                  |
| USE new_personnel;                                                                                                                                                        |                                                                                                                                                                                  |
| CREATE TABLE DEPT(DEPTNO INT(2) NOT NULL,<br>DNAME VARCHAR(14), LOC VARCHAR(14));                                                                                         | CREATE TABLE DEPT(DEPTNO NUMBER(2) NOT NULL,<br>DNAME VARCHAR2(14), LOC VARCHAR2(14));                                                                                           |
| CREATE TABLE EMP(EMPNO INT(4) NOT NULL,<br>ENAME VARCHAR(10), JOB VARCHAR(25),<br>HIREDATE DATE, MGR INT(4),<br>SAL FLOAT(7,2), COMM FLOAT(7,2),<br>DEPTNO INT(2));       | CREATE TABLE EMP(EMPNO NUMBER(4) NOT NULL,<br>ENAME VARCHAR2(10), JOB VARCHAR2(25),<br>HIREDATE DATE, MGR NUMBER(4),<br>SAL NUMBER(7,2), COMM NUMBER(7,2),<br>DEPTNO NUMBER(2)); |
| INSERT INTO DEPT(DEPTNO, DNAME, LOC)<br>VALUES (10, 'ACCOUNTING', 'NEW YORK');<br>INSERT INTO EMP<br>VALUES (10, 'CODD', 'ANALYST', '1989/01/01', 15,<br>3000, NULL, 10); | INSERT INTO DEPT(DEPTNO, DNAME, LOC)<br>VALUES (10, 'ACCOUNTING', 'NEW YORK');<br>INSERT INTO EMP<br>VALUES (10, 'CODD', 'ANALYST', '01/01/1989', 15,<br>3000, NULL, 10);        |
| SELECT * FROM EMP;<br>SELECT * FROM DEPT;                                                                                                                                 | SELECT * FROM EMP;<br>SELECT * FROM DEPT;                                                                                                                                        |
| DROP TABLE EMP;<br>DROP TABLE DEPT;                                                                                                                                       | DROP TABLE EMP;<br>DROP TABLE DEPT;                                                                                                                                              |
| DROP DATABASE NEW_PERSONNEL;                                                                                                                                              |                                                                                                                                                                                  |
| SHOW TABLES;                                                                                                                                                              | SELECT * FROM Tab;                                                                                                                                                               |

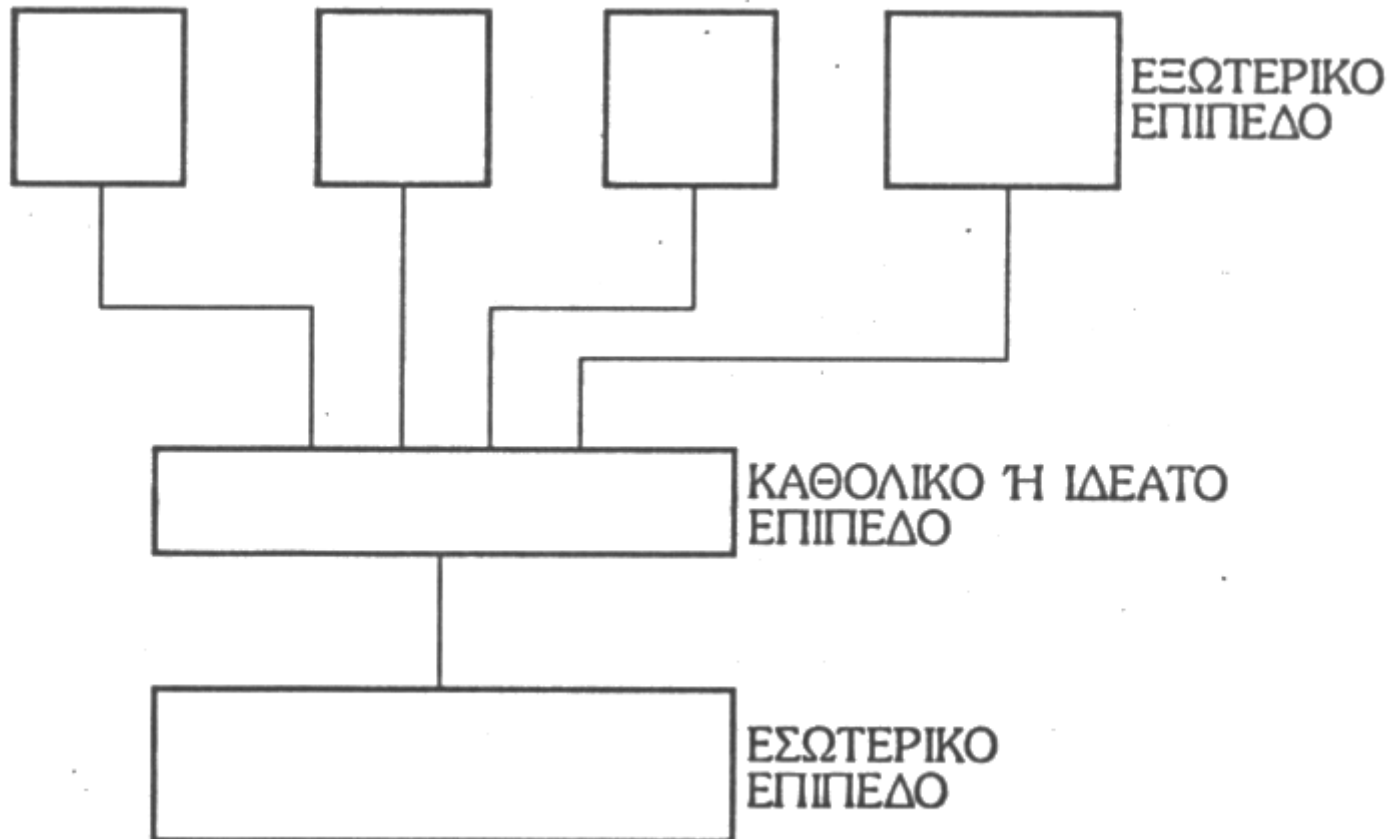
# Έννοιες & Παραδείγματα

Σκοπός μας, στη συνέχεια, είναι να παρουσιάσουμε τις έννοιες:

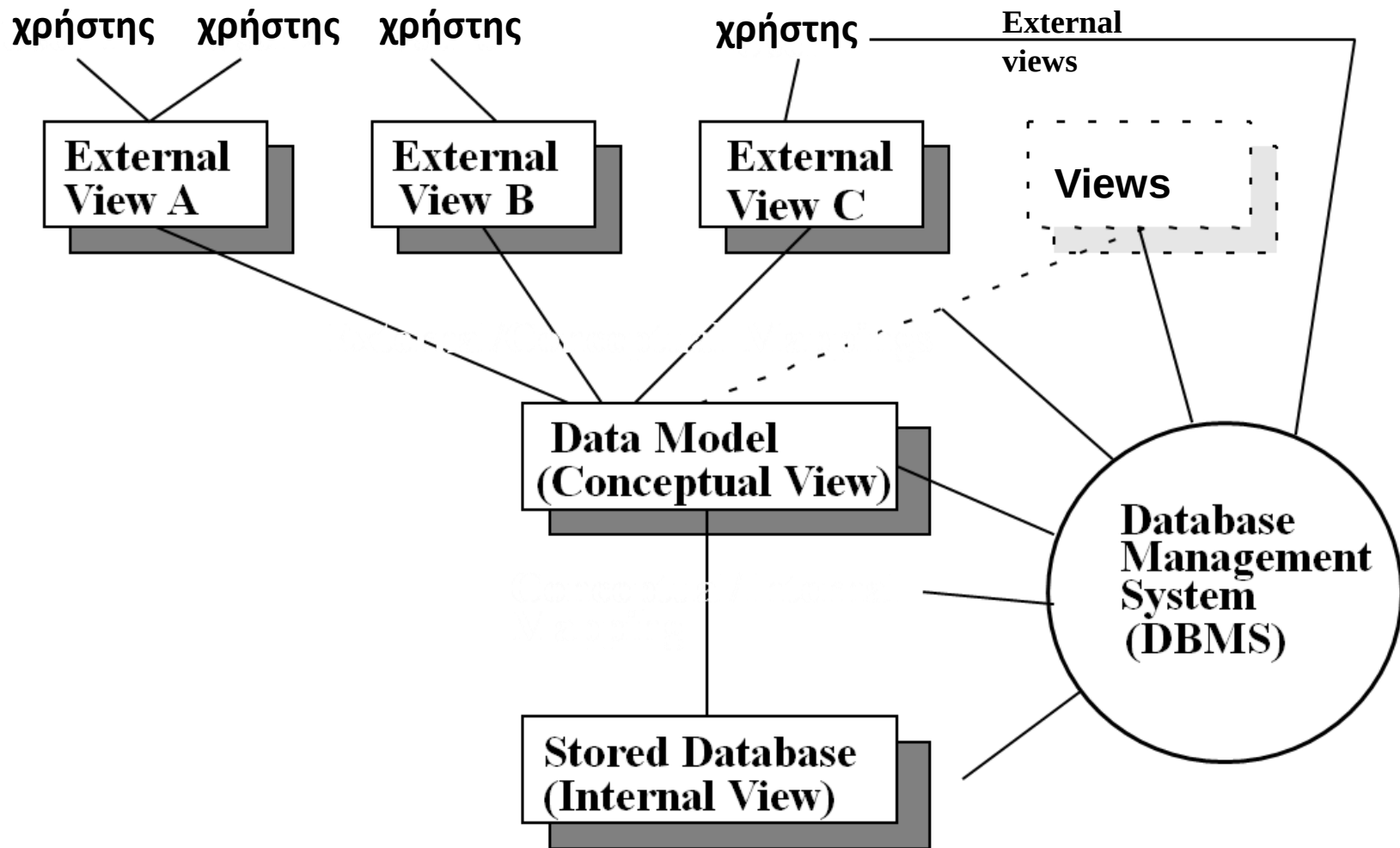
- Αρχιτεκτονική βάσεων δεδομένων,
- Γλώσσα Ελέγχου Δεδομένων – Data Control Language,
- Όψεις (views),
- Ευρετήρια (index)

και μία σειρά παραδειγμάτων ώστε οι φοιτητές να κατανοήσουν και να εμβαθύνουν στις έννοιες αυτές.

# Αρχιτεκτονική ANSI/SPARC



# Σύστημα Διαχείρισης Β.Δ.



# Βασικές έννοιες αρχιτεκτονικής ANSI/SPARC

1. **Εσωτερικό επίπεδο** (internal level)  
οπτική γωνία αυτού που ασχολείται με την αποθήκευση (οργάνωση) των δεδομένων (data). Αυτό δε σημαίνει ότι η εσωτερική άποψη προσεγγίζει το επίπεδο των περιορισμών του υλικού (hardware)!
2. **Καθολικό ή εννοιολογικό επίπεδο** (conceptual level)  
οπτική γωνία αυτού που έχει καθολική εικόνα της Βάσης

## **Διαχειριστής Βάσεως Δεδομένων** (Data Base Administrator-DBA)

Ο DBA έχει την ευθύνη της δημιουργίας του πλαισίου (framework) όπου οι χρήστες (προγραμματιστές και τελικοί χρήστες) επεξεργάζονται δεδομένα, "χτίζει" τη βάση δεδομένων (data base) της επιχείρησης, ορίζει και τροποποιεί τους τύπους δεδομένων (data types) των στηλών κ.λπ.

# Βασικές έννοιες αρχιτεκτονικής ANSI/SPARC

## 3. Εξωτερικό επίπεδο (external level)

οπτική γωνία των χρηστών. Οι χρήστες ανήκουν στις παρακάτω κατηγορίες:

- **προγραμματιστές** που έχουν γλώσσες προγραμματισμού στη διάθεσή τους (π.χ. Java, C++) και το Σ.Δ.Β.Δ. για να δημιουργήσουν και να υποστηρίξουν τη λειτουργία της Β.Δ.
- **τελικοί χρήστες** (end-users) **που έχουν εξειδικευμένες γνώσεις** και μπορούν να δημιουργήσουν κάποιες απλές εφαρμογές. Αυτοί χρησιμοποιούν ειδικές γλώσσες φτιαγμένες στα μέτρα τους ή και προγράμματα κατασκευής εφαρμογών που αποτελούν τμήμα του Σ.Δ.Β.Δ ή είναι αυτοτελείς συνιστώσες και συνεργάζονται με αυτό. Συνήθως οι γεννήτριες εφαρμογών και εκτυπώσεων (application generators , report generators) υπάγονται σε αυτήν την κατηγορία.

# Βασικές έννοιες αρχιτεκτονικής ANSI/SPARC

- *τελικοί χρήστες που χρησιμοποιούν προγράμματα εφαρμογής* φτιαγμένα από τους προγραμματιστές για εισαγωγή και αναζήτηση στοιχείων από τη Β.Δ.
- Ολοι αυτοί οι χρήστες δεν ενδιαφέρονται για ολόκληρη τη βάση δεδομένων αλλά για τμήματά της. Δηλαδή, βλέπουν τη βάση σαν το σύνολο των δεδομένων που τους ενδιαφέρει.
- Συχνά η βάση δεδομένων σε εξωτερικό επίπεδο (external level) λέγεται λογική (logical data base ) και σε Καθολικό επίπεδο (conceptual level) φυσική ( physical or stored data base).

# Υπογλώσσα Δεδομένων

Η ΥΔ είναι μία γλώσσα ορισμού και διαχείρισης βάσης δεδομένων και αποτελείται από δύο συνιστώσες:

1. **Γλώσσα Ορισμού Δεδομένων-ΓΟΔ** (DDL-data definition language) που μας επιτρέπει να ορίζουμε μία βάση δεδομένων, τους πίνακές της κ.λπ.
2. **Γλώσσα Χειρισμού (ή Επεξεργασίας) Δεδομένων -ΓΧΔ** (data manipulation language - DML) που μας επιτρέπει να διαχειριζόμαστε τα στοιχεία (των πινάκων, των ευρετηρίων κ.λπ. ) της βάσης δεδομένων που δημιουργήσαμε με τη ΓΟΔ.

Στην υπογλώσσα δεδομένων πρέπει να προσθέσουμε και τη

3. **Γλώσσα Ελέγχου Δεδομένων - ΓΕΔ** (DCL - data control language) που μας επιτρέπει να ελέγχουμε και να εξασφαλίζουμε την ασφάλεια ενός συστήματος βάσεων δεδομένων.



# Καθολικό σχήμα, εσωτερικό σχήμα και εξωτερικές όψεις της σχεσιακής βάσης δεδομένων



## Πρέπει να μάθετε καλά τις έννοιες:

Καθολικό σχήμα, όλοι οι πίνακες της  
βάσης μας

Εξωτερικά σχήματα ή όψεις, κάθε  
χρήστης βλέπει τους «δικούς» του  
πίνακες, όχι όλη τη βάση

Εσωτερικό σχήμα, ευρετήρια κ.λπ. που μας  
βοηθούν να βρίσκουμε γρηγορότερα τα δεδομένα  
που μας ενδιαφέρουν (επιταχύνουν την εκτέλεση  
των εντολών SELECT)

## Επεξήγηση με παράδειγμα

Δείτε τις εντολές που δημιουργούν τα 3 σχήματα.

# Εννοιολογικό ή Καθολικό σχήμα

```
CREATE TABLE DEPT (DEPTNO NUMBER(2) ,  
                    DNAME CHAR(14) ,  
                    CHAR(13)) ;  
  
CREATE TABLE EMP (EMPNO NUMBER(4) NOT NULL ,  
                   ENAME CHAR(10) ,  
                   JOB CHAR(9) ,  
                   MGR NUMBER(4) ,  
                   HIREDATE DATE ,  
                   SAL NUMBER(7,2) ,  
                   COMM NUMBER(7,2) ,  
                   DEPTNO NUMBER(2) ,  
                   PROJNO NUMBER(3)) ;  
  
CREATE TABLE PROJ (PROJNO NUMBER(3) NOT NULL ,  
                    PNAME CHAR(5) ,  
                    BUDGET NUMBER (7,2)) ;
```

# Εξωτερικό σχήμα

- Εξωτερικό σχήμα υπαλλήλων με επικεφαλής τον Ανδρέου

```
CREATE VIEW PROJMGR (EMPLOYEE, SALARY)
      AS SELECT  ENAME, SAL
      FROM      EMP
      WHERE     EMP.MGR = 'ΑΝΔΡΕΟΥ Ν.';
```

- Εξωτερικό σχήμα των υπαλλήλων του έργου 100.

```
CREATE VIEW PROJSTAFF
      (EMPLOYEE, PROJECT, PROJECT_NUMBER)
      AS  SELECT ENAME, PNAME, EMP.PROJNO
      FROM EMP, PROJ
      WHERE EMP.PROJNO = PROJ.PROJNO
      AND    PROJ.PROJNO = 100;
```

# Εξωτερικό σχήμα

- Εξωτερικό σχήμα των υπαλλήλων του Λονδίνου

```
CREATE VIEW LONDON_PROJECTS  
  (PROJECT, EMPLOYEE, EMP_NUMBER, LOCATION)  
  AS      SELECT  PNAME, ENAMEE, EMPNO, LOC  
            FROM    PROJ, EMP, DEPT  
            WHERE   EMP.DEPTNO = DEPT.DEPTNO  
            AND     EMP.PROJNO = PROJ.PROJNO  
            AND     DEPT.LOC = 'ΛΟΝΔΙΝΟ';
```

# Εσωτερικό σχήμα

Ο Διαχειριστής ορίζει τους απαραίτητους δείκτες, συστάδες κτλ ώστε να βελτιωθεί ο χρόνος αναζήτησης στοιχείων από τη βάση για συγκεκριμένες αναζητήσεις που ενδιαφέρουν ιδιαίτερα.

## Ορισμός δεικτών

- Η δήλωση

```
CREATE INDEX INAME ON EMP (ENAME) ;
```

επιταχύνει την αναζήτηση με κριτήριο το όνομα του υπαλλήλου.

- Η δήλωση

```
CREATE INDEX ISAL ON EMP (SAL) ;
```

επιταχύνει αναζητήσεις με κριτήριο το μισθό.



# Σύστημα Διαχείρισης Βάσης Δεδομένων

## Διαχειριστής βάσης δεδομένων

**Πρέπει να μάθετε καλά 2 συστήματα (προϊόντα):**

Oracle, για μεγάλες εφαρμογές

MySQL, κυρίως για δυναμικές βάσεις στο διαδίκτυο

### **Ο διαχειριστής**

Πρέπει να μάθετε τα καθήκοντά του.

Είναι ο μόνος που βλέπει όλα τα σχήματα της βάσης (καθολικό κ.λπ.)

# Σύστημα Διαχείρισης Βάσης Δεδομένων Database Management System (DBMS)

Το DBMS είναι το λογισμικό που διαχειρίζεται κάθε πρόσβαση (access) στη βάση δεδομένων και είναι υπεύθυνο για την εφαρμογή των ελέγχων εξουσιοδοτημένης πρόσβασης (authorisation checks) και των διαδικασιών ορθότητας (validation procedures).

# Σύστημα Διαχείρισης Βάσης Δεδομένων Database Management System (DBMS)

Εννοιολογικά:

- 1) Ο χρήστης αιτείται πρόσβαση (access request) χρησιμοποιώντας Γλώσσα DML
- 2) Το DBMS δέχεται την αίτηση και τη διερμηνεύει.
- 3) Το DBMS ελέγχει το εξωτερικό σχήμα,
- 4) την αντιστοιχία με το εννοιολογικό σχήμα (the external/conceptual mapping) και
- 5) την αντιστοιχία εννοιολογικού / εσωτερικού σχήματος (the conceptual internal mapping) και τον ορισμό της δομής αποθήκευσης (storage structure).

Τέλος, το DBMS εκτελεί τις αναγκαίες εργασίες στην αποθηκευμένη βάση δεδομένων.

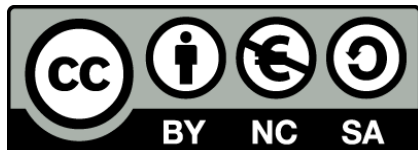


# Ο ρόλος του ΔΒΔ

- Μεγάλη σημασία σε ένα σύστημα Β.Δ. έχει η δραστηριότητα του διαχειριστή-ΔΒΔ (Data Base Administrator-DBA), δηλαδή του προσώπου ή της ομάδας προσώπων που έχουν την ευθύνη και το συνολικό έλεγχο του συστήματος
- Ο DBA συμμετέχει στη σχεδίαση της βάσης δεδομένων και των εφαρμογών, παρέχει στους προγραμματιστές την τεχνική υποστήριξη για την υλοποίηση των εφαρμογών και έχει την ευθύνη της αποδοτικής λειτουργίας του συστήματος.

# Τέλος Ενότητας

Ερωτήσεις;



**Σημειώματα**

# Σημείωμα Αναφοράς

Copyright Πανεπιστήμιο Δυτικής Αττικής, Χ. Σκουρλάς 2018.

Χ. Σκουρλάς. «Βάσεις Δεδομένων Ι. Ενότητα 2: Εισαγωγή στις βάσεις δεδομένων», Έκδοση: 1.0. Αθήνα 2018. Διαθέσιμο από τη δικτυακή διεύθυνση: [pyles.teiath.gr](http://pyles.teiath.gr).

# Σημείωμα Χρήσης Έργων Τρίτων

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».

**έργα τρίτων**

# Σημείωμα Αδειοδότησης

Το παρόν υλικό διατίθεται με τους όρους της άδειας χρήσης Creative Commons Αναφορά, Μη Εμπορική Χρήση Παρόμοια Διανομή 4.0 [1] ή μεταγενέστερη, Διεθνής Έκδοση. Εξαιρούνται τα αυτοτελή έργα τρίτων π.χ. φωτογραφίες, διαγράμματα κ.λ.π., τα οποία εμπεριέχονται σε αυτό και τα οποία αναφέρονται μαζί με τους όρους χρήσης τους στο «Σημείωμα Χρήσης Έργων Τρίτων».



[1] <http://creativecommons.org/licenses/by-nc-sa/4.0/>

Ως **Μη Εμπορική** ορίζεται η χρήση:

- που δεν περιλαμβάνει άμεσο ή έμμεσο οικονομικό όφελος από την χρήση του έργου, για το διανομέα του έργου και αδειοδόχο
- που δεν περιλαμβάνει οικονομική συναλλαγή ως προϋπόθεση για τη χρήση ή πρόσβαση στο έργο
- που δεν προσπορίζει στο διανομέα του έργου και αδειοδόχο έμμεσο οικονομικό όφελος (π.χ. διαφημίσεις) από την προβολή του έργου σε διαδικτυακό τόπο

Ο δικαιούχος μπορεί να παρέχει στον αδειοδόχο ξεχωριστή άδεια να χρησιμοποιεί το έργο για εμπορική χρήση, εφόσον αυτό του ζητηθεί.

# Διατήρηση Σημειωμάτων

Οποιαδήποτε αναπαραγωγή ή διασκευή του υλικού θα πρέπει να συμπεριλαμβάνει:

- το Σημείωμα Αναφοράς
- το Σημείωμα Αδειοδότησης
- τη δήλωση Διατήρησης Σημειωμάτων
- το Σημείωμα Χρήσης Έργων Τρίτων (εφόσον υπάρχει)

μαζί με τους συνοδευόμενους υπερσυνδέσμους.