



# ΕΙΣΑΓΩΓΗ ΣΤΗΝ ΕΠΙΣΤΗΜΗ ΤΩΝ ΥΠΟΛΟΓΙΣΤΩΝ

<http://eclass.aueb.gr/courses/INF511/>

## Συστήματα Βάσεων Δεδομένων (ΚΕΦΑΛΑΙΟ 9)

Αλκμήνη Σγουρίτσα

Κοδριγκτώνος 12, 2<sup>ος</sup> όροφος

E-mail: [alkmini@aeub.gr](mailto:alkmini@aeub.gr)

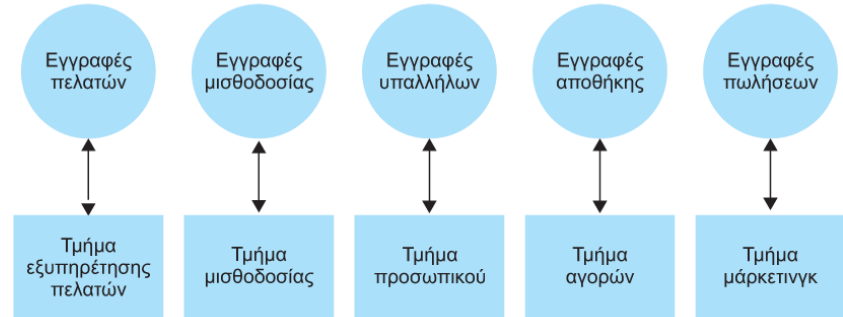
# Κεφάλαιο 9: Συστήματα Βάσεων Δεδομένων /Εισαγωγή στην Μηχανική Μάθηση

- Σχεσιακό μοντέλο βάσης δεδομένων
  - Σχεσιακές λειτουργίες
  - Εισαγωγή στην γλώσσα SQL
  - Ακεραιότητα βάσεων δεδομένων
  - Μηχανική μάθηση
    - παλινδρόμηση
    - κατηγοριοποίηση
    - συσταδοποίηση
- Ιδιωτικότητα

# Σύγκριση: Αρχείο και Βάση Δεδομένων (ΒΔ)

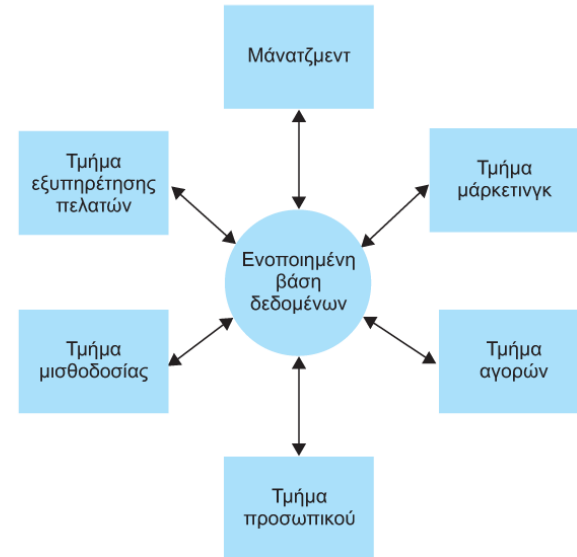
- **Οργάνωση αρχείου (file system)** είναι ένα μονοδιάστατο σύστημα αποθήκευσης (παριστάνει την πληροφορία από **μια οπτική γωνία**)
  - Π.χ. λίστα από συνθέτες και τα κομμάτια τους

α. Πληροφοριακό σύστημα με αρχεία



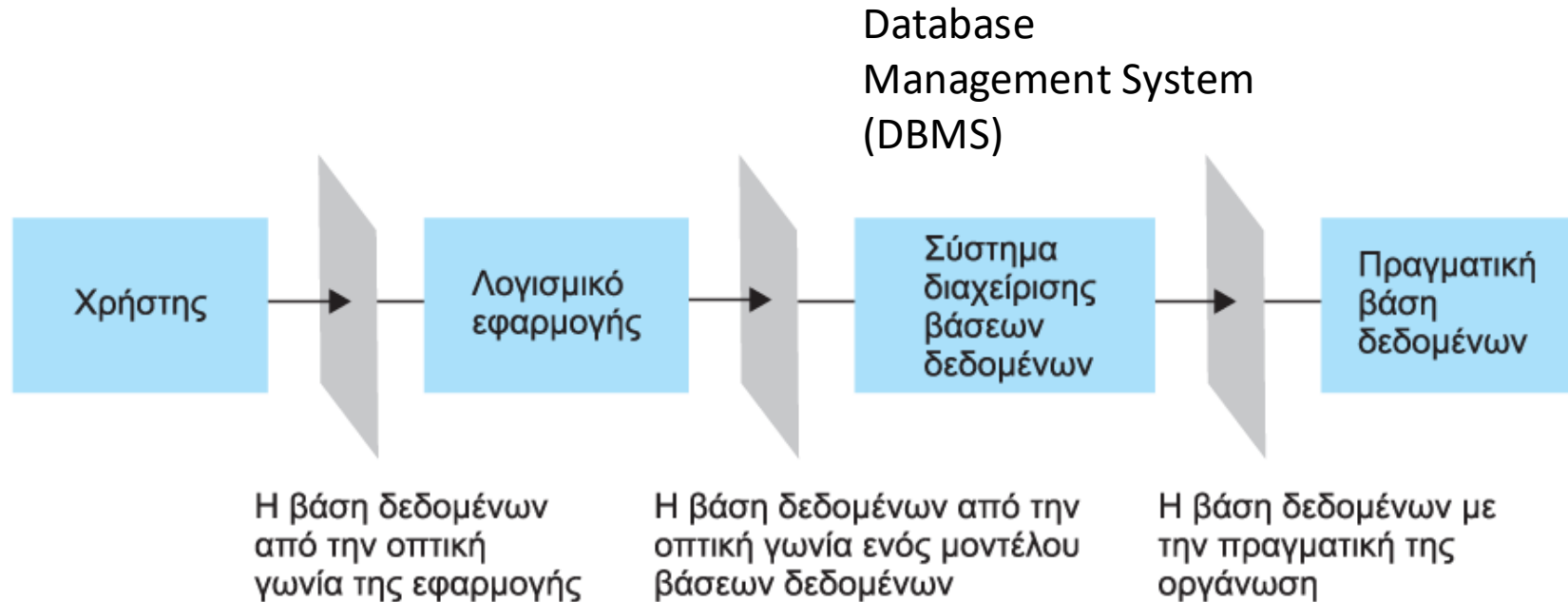
- **Βάση δεδομένων (database):** Συλλογή δεδομένων που είναι πολυδιάστατη, δηλ. **οι εσωτερικοί σύνδεσμοι** μεταξύ των καταχωρήσεών της κάνουν τις πληροφορίες προσπελάσιμες από **πολλές οπτικές γωνίες**
  - Π.χ. οι δουλειές ενός συνθέτη, οι συνθέτες για ένα είδος μουσικής, κλπ

β. Πληροφοριακό σύστημα με βάση δεδομένων

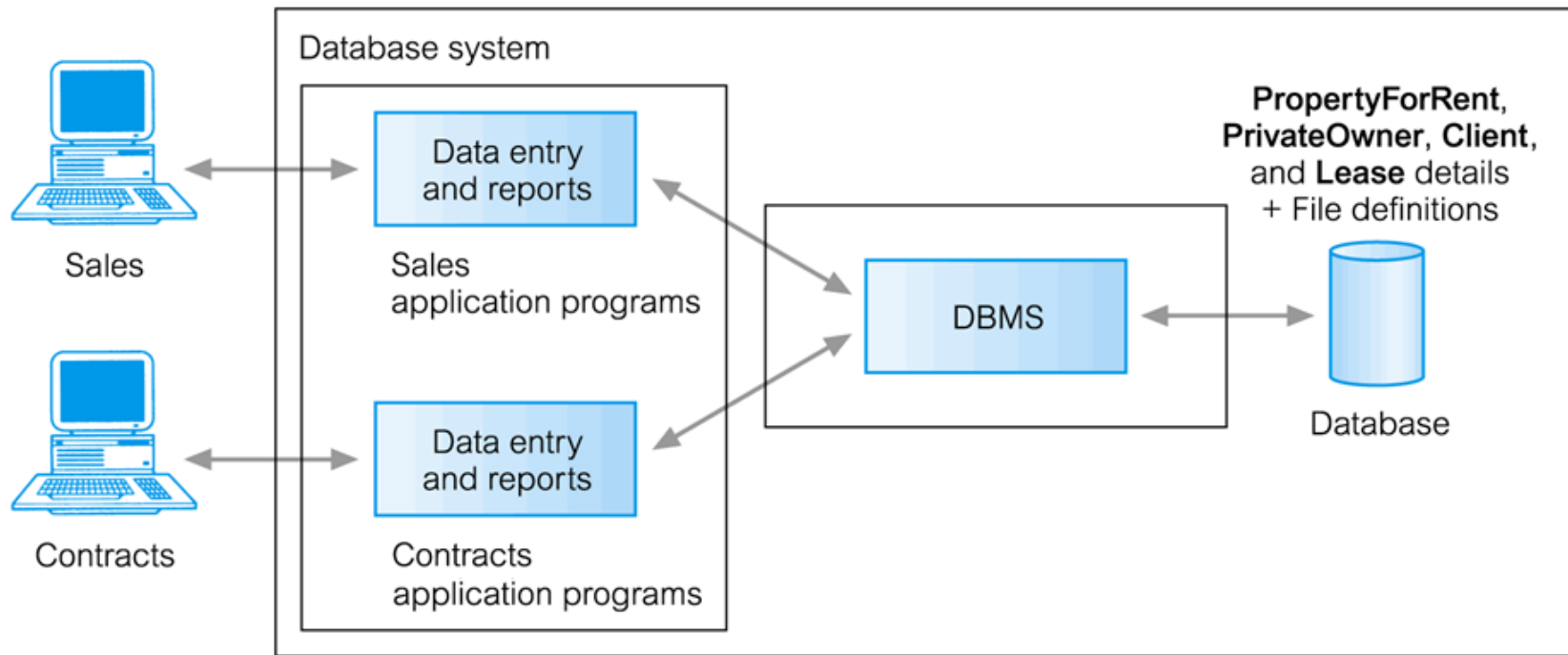


Η βάση δεδομένων παρέχει ενοποιημένη πληροφορία

# Τα νοητικά επίπεδα της υλοποίησης μιας Βάσης Δεδομένων



# Τα νοητικά επίπεδα της υλοποίησης μιας Βάσης Δεδομένων- Παράδειγμα



**PropertyForRent** (propertyNo, street, city, postcode, type, rooms, rent, ownerNo)

**PrivateOwner** (ownerNo, fName, lName, address, telNo)

**Client** (clientNo, fName, lName, address, telNo, prefType, maxRent)

**Lease** (leaseNo, propertyNo, clientNo, paymentMethod, deposit, paid, rentStart, rentFinish)

# Συστήματα Διαχείρισης Βάσεων Δεδομένων

- **Σύστημα Διαχείρισης Βάσεων Δεδομένων** – ΣΔΒΔ (**Database Management System** – DBMS): Λογισμικό που χειρίζεται μια Βάση Δεδομένων σε απάντηση των αιτήσεων (queries) που έρχονται από τις εφαρμογές
  - Όταν το λογισμικό εφαρμογής καθορίσει τι ζητά ο χρήστης, **χρησιμοποιεί το ΣΔΒΔ για να πάρει τα αποτελέσματα**
  - Το ΣΔΒΔ κρατά τις λεπτομέρειες αποθήκευσης και οργάνωσης της Βάσης **μακριά από το χρήστη** (αποκρύπτει πολυπλοκότητες της εσωτερικής δομής της Βάσης, επιτρέπει στον χρήστη να φαντάζεται την αποθηκευμένη πληροφορία με απλό τρόπο).
    - **Έλέγχει την πρόσβαση στα δεδομένα**
- **Κατανεμημένη (distributed) βάση δεδομένων**: Βάση Δεδομένων αποθηκευμένη σε πολλά μηχανήματα (συσκευές)

# Σχεσιακό Μοντέλο Βάσης Δεδομένων

- **Σχέση (relation):** Ένας ορθογώνιος πίνακας (κάτι σαν του excel)
  - **Ιδιότητα (attribute):** Μια στήλη του πίνακα, ορίζει ένα χαρακτηριστικό
  - **Πλειάδα (tuple):** Μια γραμμή του πίνακα, εδώ περιέχει πληροφορία για έναν υπάλληλο

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

# Σχεσιακό Μοντέλο Βάσης Δεδομένων

- Σημαντικό σημείο σχεδιασμού είναι ο **καθορισμός** των σχέσεων
- Αποφυγή πολλαπλών εννοιών μέσα σε μια σχέση
  - Μπορεί να οδηγήσει σε **πλεονάζοντα δεδομένα**
  - Μπορεί να προκαλέσει **προβλήματα όταν ανανεώνουμε** τα δεδομένα
    - **Διαγραφή** δεδομένων
    - **Εισαγωγή** δεδομένων
    - **Αλλαγή** δεδομένων



# Μια σχέση με πλεονάζουσες πληροφορίες

EmpId (Κωδ. υπάλληλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)	Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)	Start Date (Ημερ. έναρξης)	Term Date (Ημερ. τέλους)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333	F5	Εποπτεύων ορόφου	FM3	Πωλήσεων	1-9-2002	30-9-2003
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333	D7	Επικεφαλής τμήματος	K2	Πωλήσεων	1-10-2003	*
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999	F5	Εποπτεύων ορόφου	FM3	Πωλήσεων	1-10-2002	*
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555	S25X	Γραμματέας	T5	Προσωπικού	1-3-1999	30-4-2001
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555	S26Z	Γραμματέας	T6	Λογιστήριο	1-5-2001	*
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•

- **Πλεονάζουσες** πληροφορίες στην βάση (επαναλαμβάνονται όποτε κάποιος υπάλληλος έχει προϋπηρεσία ή όταν σε μια θέση εργάζονται πάνω από 1 υπάλληλοι).
- Σπατάλη χώρου.

# Πρόβλημα διαγραφής δεδομένων

Empl Id (Κωδ. υπάλληλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)	Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)	Start Date (Ημερ. έναρξης)	Term Date (Ημερ. τέλους)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333	F5	Εποπτεύων ορόφου	FM3	Πωλήσεων	1-9-2002	30-9-2003
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333	D7	Επικεφαλής τμήματος	K2	Πωλήσεων	1-10-2003	*
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999	F5	Εποπτεύων ορόφου	FM3	Πωλήσεων	1-10-2002	*
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555	S25X	Γραμματέας	T5	Προσωπικού	1-3-1999	30-4-2001
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555	S26Z	Γραμματέας	T6	Λογιστήριο	1-5-2001	*
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•

- Υποθέστε ο Γιάννης Αθανασίου παραιτείται και πρέπει να τον **διαγράψουμε** από τη βάση.
- Υποθέστε επίσης ότι **δεν υπάρχει** άλλος υπάλληλος με κωδικό θέσης D7
- Ποιο είναι το πρόβλημα;
- Η διαγραφή των δεδομένων που αφορούν τον εν λόγω υπάλληλο θα οδηγήσει στην **πλήρη διαγραφή της θέσης D7**.

# Πρόβλημα εισαγωγής δεδομένων (1)

Empl Id (Κωδ. υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)	Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)	Start Date (Ημερ. έναρξης)	Term Date (Ημερ. τέλους)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333	F5	Εποπτεύων ορόφου	FM3	Πωλήσεων	1-9-2002	30-9-2003
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333	D7	Επικεφαλής τμήματος	K2	Πωλήσεων	1-10-2003	*
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999	F5	Εποπτεύων ορόφου	FM3	Πωλήσεων	1-10-2002	*
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555	S25X	Γραμματέας	T5	Προσωπικού	1-3-1999	30-4-2001
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555	S26Z	Γραμματέας	T6	Λογιστήριο	1-5-2001	*
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•

- Υποθέστε ότι η Μαρία Γεωργιάδου πήρε προαγωγή και πρέπει να **προσθέσουμε** στη βάση ότι από 1-12-2005 έγινε επικεφαλής τμήματος με Job Id D7.
- Τι πληροφορία θα πρέπει να προσθέσουμε;
- Πρέπει να προσθέσουμε **ξανά όλα τα προσωπικά της στοιχεία και τα στοιχεία της θέσης**.
  - Πρέπει να προστεθούν **σωστά για να υπάρχει συμβατότητα**.
  - **Σπατάλη χώρου και χρόνου**.

# Πρόβλημα εισαγωγής δεδομένων (2)

Empl Id (Κωδ. υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)	Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)	Start Date (Ημερ. έναρξης)	Term Date (Ημερ. τέλους)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333	F5	Εποπτείων ορόφου	FM3	Πωλήσεων	1-9-2002	30-9-2003
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333	D7	Επικεφαλής τμήματος	K2	Πωλήσεων	1-10-2003	*
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999	F5	Εποπτείων ορόφου	FM3	Πωλήσεων	1-10-2002	*
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555	S25X	Γραμματέας	T5	Προσωπικού	1-3-1999	30-4-2001
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555	S26Z	Γραμματέας	T6	Λογιστήριο	1-5-2001	*
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•

- Υποθέστε ότι έγινε πρόσληψη ενός καινούργιου υπαλλήλου χωρίς όμως να του έχουν ανατεθεί τα καθήκοντά του.
- Τι πληροφορία θα πρέπει να προσθέσουμε;
- Πρέπει να προσθέσουμε μία καινούργια πλειάδα (tuple) με τα στοιχεία του υπαλλήλου μόνο για τις 4 πρώτες ιδιότητες. Οι υπόλοιπες ιδιότητες θα μείνουν κενές.
  - **Σπατάλη χώρου και δυσκολία διαχείρισης** (πολλά κενά στη βάση)

# Πρόβλημα αλλαγής δεδομένων

Empl Id (Κωδ. υπάλληλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)	Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)	Start Date (Ημερ. έναρξης)	Term Date (Ημερ. τέλους)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333	F5	Εποπτείων ορόφου	FM3	Πωλήσεων	1-9-2002	30-9-2003
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333	D7	Επικεφαλής τμήματος	K2	Πωλήσεων	1-10-2003	*
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999	F5	Εποπτείων ορόφου	FM3	Πωλήσεων	1-10-2002	*
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555	S25X	Γραμματέας	T5	Προσωπικού	1-3-1999	30-4-2001
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555	S26Z	Γραμματέας	T6	Λογιστήριο	1-5-2001	*
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•	•	•

- Υποθέστε ότι ο Στέλιος Μπέλλος μετακόμισε και θέλουμε να **ανανεώσουμε** τη βάση με τη νέα του διεύθυνση σε κάθε πλειάδα (tuple) που εμφανίζεται ο εν λόγω υπάλληλος.
- Τι αλλαγές πρέπει να γίνουν;
- Πρέπει να αντικαταστήσουμε την παλιά διεύθυνση με την καινούργια σε **όλες** τις πλειάδες που εμφανίζεται ο εν λόγω υπάλληλος.
  - Πρέπει να προστεθούν **σωστά για να υπάρχει συμβατότητα**.
  - **Σπατάλη χώρου και χρόνου**

# Βελτίωση Σχεσιακού Σχεδιασμού

- **Διαίρεση (decomposition):**
  - Κατανομή των στηλών μιας σχέσης σε δύο ή περισσότερες σχέσεις
  - Επανάληψη των στηλών που απαιτούνται για διατήρηση των συσχετίσεων μεταξύ των σχέσεων
  - **Μη απωλεστική** (non-loss ή lossless) διαίρεση: Μια “σωστή” διαίρεση που δεν οδηγεί σε απώλεια πληροφοριών
  - Χρειάζεται προσοχή στον καθορισμό των σωστών σχέσεων.
- Είναι πολύ σημαντικός ο **σωστός σχεδιασμός** των βάσεων δεδομένων για να είναι **εύχρηστες και αποδοτικές**.

# Παράδειγμα Βάσης Δεδομένων που αποτελείται από τρεις σχέσεις

**Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)**

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

**Σχέση JOB (ΘΕΣΗ)**

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτεΰων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

**Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)**

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

# Μη απωλεστική διαίρεση

**Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)**

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

**Σχέση JOB (ΘΕΣΗ)**

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτεύων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

Ανήκουν στα τμήματα προσωπικού και λογιστηρίου

**Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)**

Οι θέσεις που κατείχε ο υπάλληλος 23Y34

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•



# Παράδειγμα Βάσης Δεδομένων που αποτελείται από τρεις σχέσεις

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Η πληροφορία που επαναλαμβάνεται είναι **απαραίτητη για τη συσχέτιση** των σχέσεων.

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτεΐων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

Πώς θα αλλάζατε τη βάση αν

- Ένας υπάλληλος παραιτηθεί;
- Ένας υπάλληλος πάρει προαγωγή;
- Ένας καινούργιος υπάλληλος προσληφθεί;
- Ένας υπάλληλος αλλάξει διεύθυνση;

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

# Primary Key

Κάθε σχέση πρέπει να έχει ένα **μοναδικό primary key**.

**Primary key**: Μία ιδιότητα (attribute), ή ένα σύνολο από ιδιότητες, της σχέσης που προσδιορίζουν μοναδικά μία πλειάδα (tuple).

- Το primary key πρέπει να είναι **irreducible**, δηλαδή **κανένα γνήσιο υποσύνολο** από ιδιότητες δε προσδιορίζει μοναδικά μία πλειάδα.
- Το primary key **δεν μπορεί να μην παίρνει τιμή**, δηλδ δεν μπορεί να είναι **null (entity integrity)**
- Συνήθως διαλέγουμε σαν primary key τις ιδιότητες που είναι **πιο δύσκολο να αλλάξουν**, και είναι πιο ασφαλείς στο να προσδιορίζουν μοναδικά τις πλειάδες.

# Παράδειγμα Primary Keys

**Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)**

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Θα μπορούσε να χρησιμοποιήσει ως PK αν δεν υπήρχε το Empl Id

**Σχέση JOB (ΘΕΣΗ)**

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτεύων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

**Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)**

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

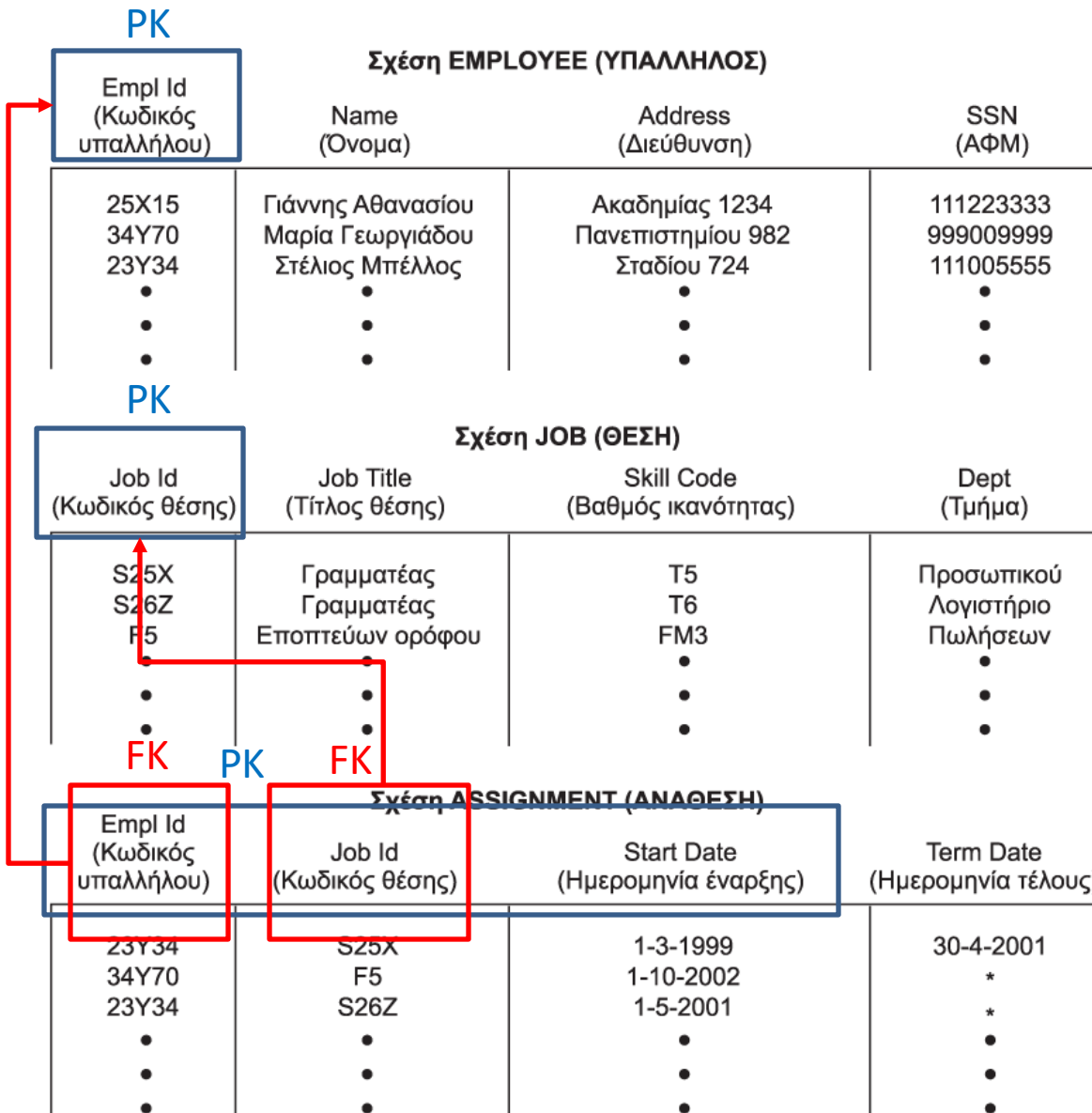
Τι γίνεται αν κάποιος υπάλληλος μπορεί να έχει την ίδια θέση πάνω από μία φορά;  
**Τότε το PK πρέπει να συμπεριλάβει και το Start Date**

# Foreign Key

**Foreign key:** Μία ιδιότητα (attribute), ή ένα σύνολο από ιδιότητες, της σχέσης που ταυτίζονται με το primary key μίας άλλης (ή ακόμα και της ίδιας).

- **Referential integrity:** Κάθε τιμή του foreign key πρέπει
  - είτε να **ταυτίζεται** με μία τιμή του Primary key με το οποίο σχετίζεται
  - είτε να είναι **κενό** (null)

# Παράδειγμα Foreign Keys



Foreign keys στη σχέση ASSIGNMENT:

FK: Empl Id references EMPLOYEE(Empl Id)

FK: Job Id references JOB(Job Id)

# Παράδειγμα: Εντοπισμός τμημάτων στα οποία έχει εργαστεί ο υπάλληλος 23Y34

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

PK

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτευών ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

Ανήκουν στα τμήματα προσωπικού και λογιστηρίου

FK

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

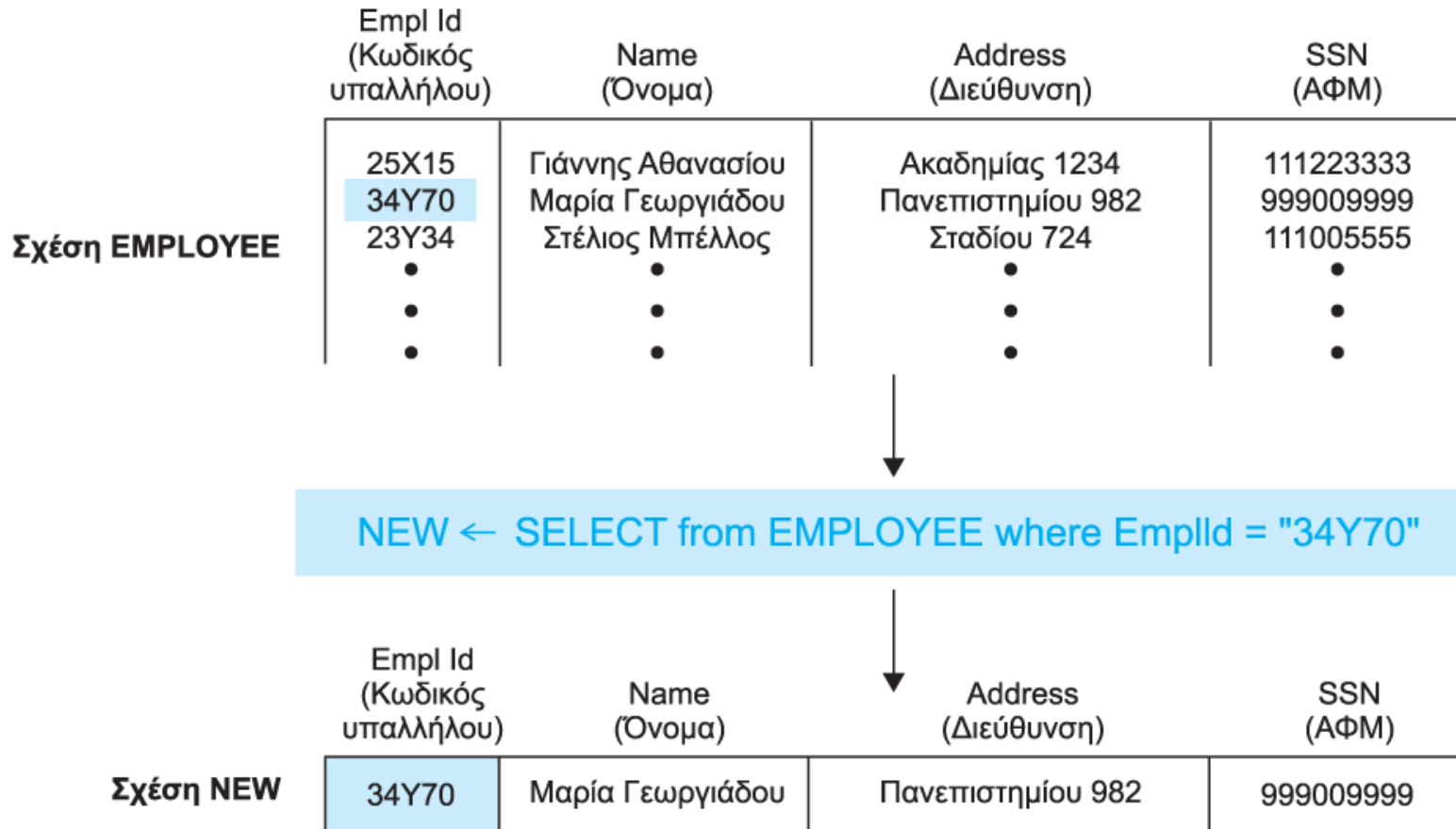
Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

Οι θέσεις που κατείχε ο υπάλληλος 23Y34

# Σχεσιακές Λειτουργίες (Relational Operations)

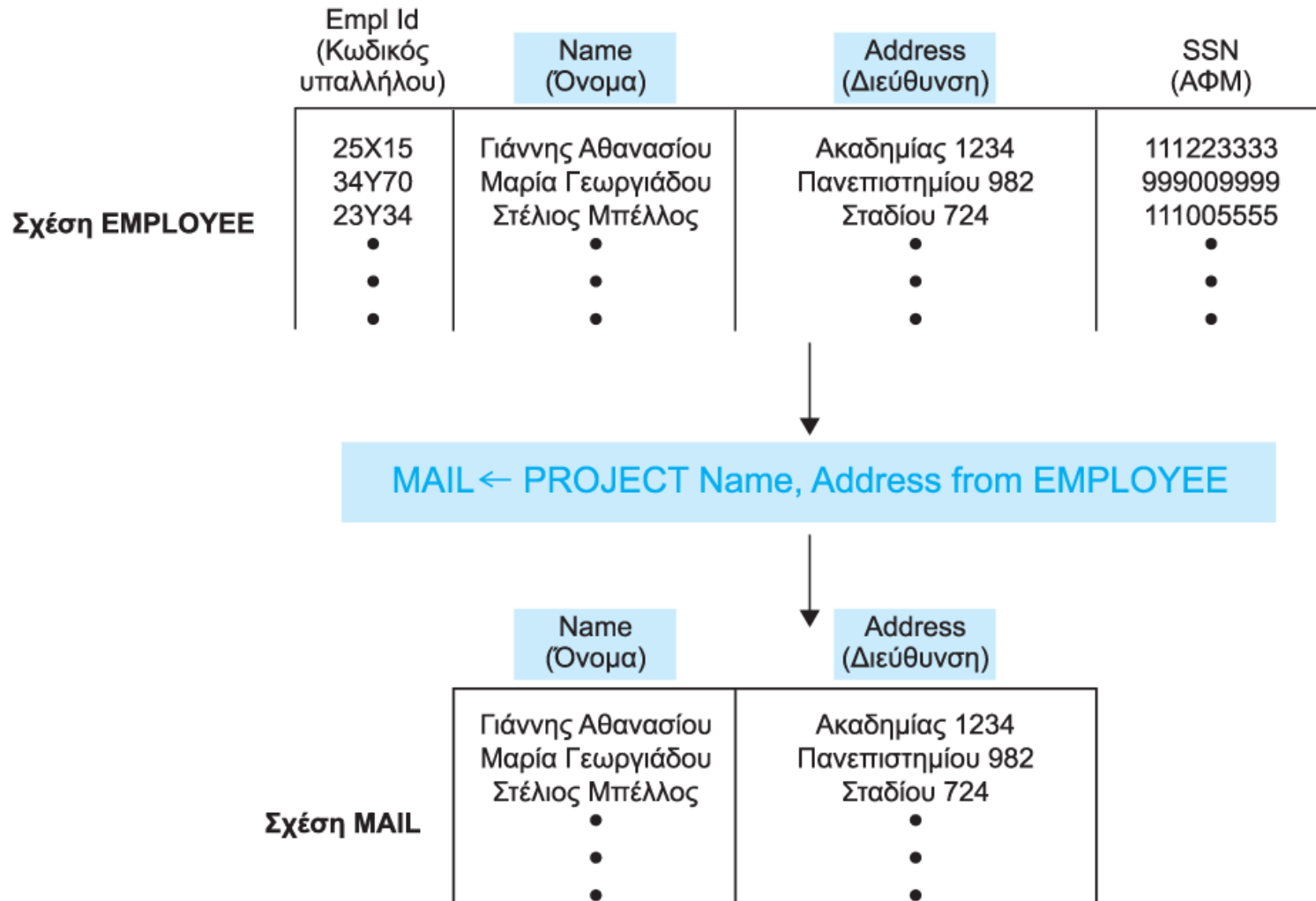
- **Select (Επιλογή):** Επιλογή γραμμών
- **Project (Προβολή):** Επιλογή στηλών
- **Join (Σύνδεση):** Συγκέντρωση πληροφοριών από δύο ή περισσότερες σχέσεις

# Λειτουργία SELECT (Επιλογή)

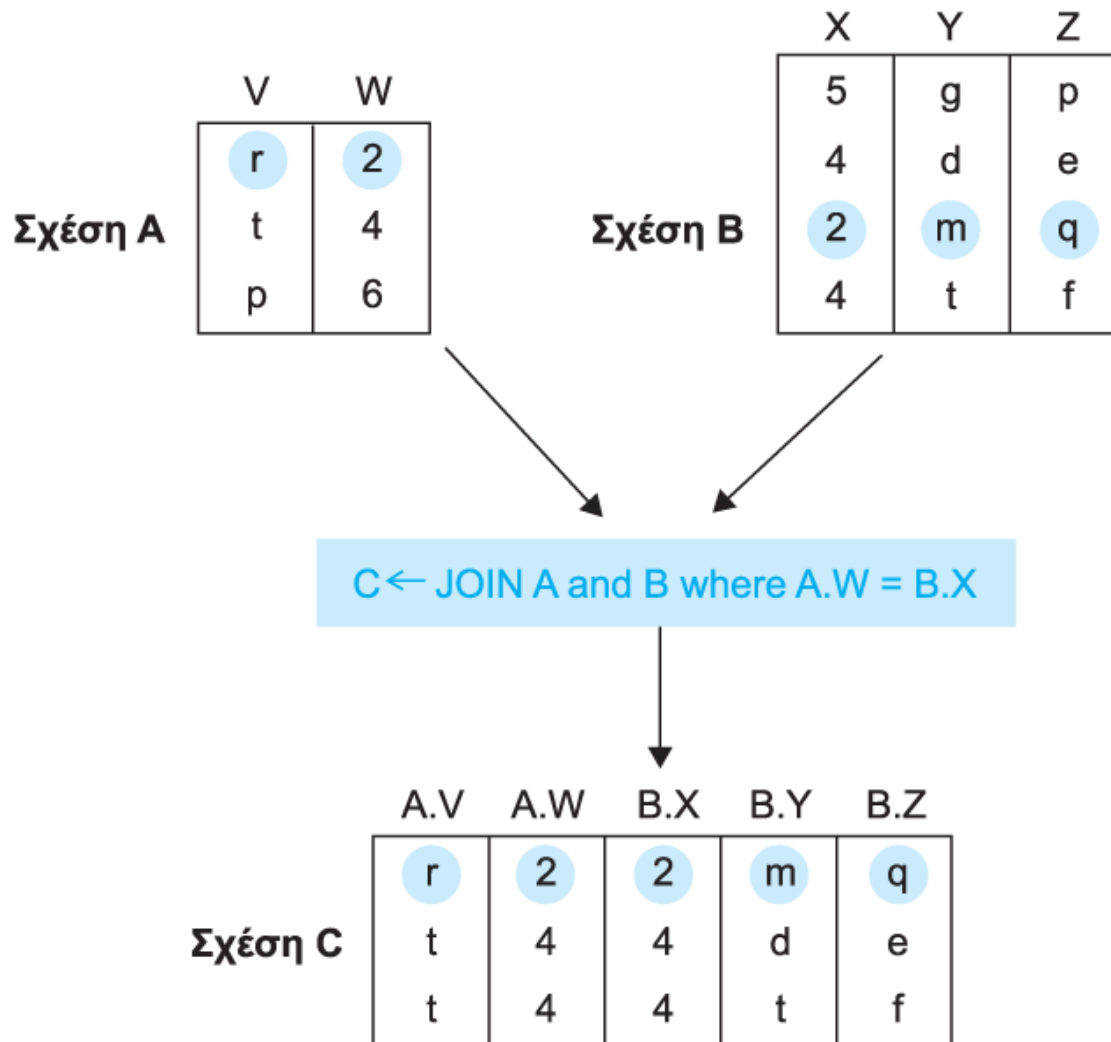




# Λειτουργία PROJECT (Προβολή)



# Λειτουργία JOIN (Σύνδεση)



# Παράδειγμα

Ερώτηση: να βρεθούν τα Id υπαλλήλων και το τμήμα όπου εργάζεται και έχει εργαστεί κάθε υπάλληλος

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Σχέση JOB (ΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)	Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
23Y34	S25X	1-3-1999	30-4-2001	S25X	Γραμματέας	T5	Προσωπικού
34Y70	F5	1-10-2002	*	S26Z	Γραμματέας	T6	Λογιστήριο
23Y34	S26Z	1-5-2001	*	F5	Εποπτεύων ορόφου	FM3	Πωλήσεων
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•

NEW1 ← JOIN ASSIGNMENT and JOB where ASSIGNMENT.JobId = JOB.JobId

Σχέση NEW1

ASSIGNMENT Empl Id	ASSIGNMENT Job Id	ASSIGNMENT Start Date	ASSIGNMENT Term Date	JOB Job Id	JOB Job Title	JOB Skill Code	JOB Dept
23Y34	S25X	1-3-1999	30-4-2001	S25X	Γραμματέας	T5	Προσωπικού
34Y70	F5	1-10-2002	*	F5	Εποπτεύων ορόφου	FM3	Πωλήσεων
23Y34	S26Z	1-5-2001	*	S26Z	Γραμματέας	T6	Λογιστήριο
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•

# Δομημένη Γλώσσα Επερωτήσεων (SQL)

- Ερωτήματα ή Επερωτήσεις (queries) σε ΒΔ: βασικός τρόπος αλληλεπίδρασης του χρήστη με μια ΒΔ
- SQL: Structured Query Language
  - Γλώσσα στην οποία υλοποιούνται ερωτήματα σε ΒΔ με **σχεσιακό μοντέλο**
  - MySQL: σχεσιακό σύστημα διαχείρισης ΒΔ
  - Η Oracle χρησιμοποιεί SQL
- Λειτουργίες στην SQL:
  - select, project, join
  - insert
  - update
  - delete

# Παραδείγματα SQL (1)

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Ποιες ιδιότητες

```
select Name, Address
from EMPLOYEE
```

Από ποια σχέση

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτεΰων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

Όλες οι ιδιότητες

```
select *
from EMPLOYEE
where Name = 'Στέλιος Μπέλλος'
```

Συνθήκη (κράτα τις πλειάδες που ικανοποιούν τη συνθήκη)

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

```
select Name, Address, SSN
from EMPLOYEE
where Name = 'Γιάννης Αθανασίου'
```

# Παραδείγματα SQL (2)

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Σχέση JOB (ΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτευών ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

```
select *
from ASSIGNMENT, JOB
where ASSIGNMENT.JobId = JOB.JobId
```

ASSIGNMENT Empl Id	ASSIGNMENT Job Id	ASSIGNMENT Start Date	ASSIGNMENT Term Date	JOB Job Id	JOB Job Title	JOB Skill Code	JOB Dept
23Y34	S25X	1-3-1999	30-4-2001	S25X	Γραμματέας	T5	Προσωπικού
34Y70	F5	1-10-2002	*	F5	Εποπτευών ορόφου	FM3	Πωλήσεων
23Y34	S26Z	1-5-2001	*	S26Z	Γραμματέας	T6	Λογιστήριο
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•
•	•	•	•	•	•	•	•

```
select JOB.JobTitle, ASSIGNMENT.StartDate
from ASSIGNMENT, JOB
where ASSIGNMENT.JobId = JOB.JobId AND JOB.SkillCode='FM3'
```

Προαιρετικά γιατί είναι μοναδικές ιδιότητες του καινούργιου πίνακα

# Παράδειγμα SQL (3)

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτείων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•

```
select EmplId, Dept #attributes
from ASSIGNMENT, JOB #relations
where ASSIGNMENT.JobId = JOB.JobId
and TermDate IS NULL #conditions
```

EmplId	Dept
34Y70	Πωλήσεων
23Y34	Λογιστήριο
•	•
•	•

# Εναλλακτικές join

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτεύων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•

```
select EmplId, Dept
from ASSIGNMENT, JOB
where ASSIGNMENT.JobId = JOB.JobId
```

```
select EmplId, Dept
from ASSIGNMENT natural join JOB
```

```
select EmplId, Dept
from ASSIGNMENT join JOB using (JobId)
```

```
select EmplId, Dept
from ASSIGNMENT join JOB on ASSIGNMENT.JobId = JOB.JobId
```



# Renaming tables

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτεΰων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•

```
select A.*
from ASSIGNMENT A, JOB J
where A.JobId = J.JobId and J.SkillCode='FM3'
```

**Σημαντικότητα του Renaming:** Παράδειγμα όταν θέλουμε να κάνουμε join μία σχέση με τον εαυτό της.

Π.χ. Βρες τα Job Titles που εμφανίζονται σε **περισσότερα από έναν** Job Id

```
select J1.JobTitle
from JOB J1, JOB J2
where J1.JobTitle=J2.JobTitle and J1.JobId<>J2.JobId
```

# Παράδειγμα SQL (4)

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Βρες τα **ονόματα των υπαλλήλων** όπου κάποια θέση που κατείχαν/κατέχουν έχει **skill code 'T5'**

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτείων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

Που βρίσκεται η πληροφορία που ζητάμε;  
Που βρίσκονται και ποιες είναι οι συνθήκες;  
Πώς σχετίζονται οι σχέσεις;

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

select **Name**  
from **EMPLOYEE E**, **JOB J**, **ASSIGNMENT A**  
where **E.EmplId=A.EmplId** and **J.JobId=A.JobId**  
and **SkillCode='T5'**

# Παράδειγμα SQL (5)

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Βρες τα **Empl Id** των υπαλλήλων όπου κάποια θέση που κατείχαν/κατέχουν έχει **skill code** 'T5'

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτείων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

Που βρίσκεται η πληροφορία που ζητάμε;  
Που βρίσκονται και ποιες είναι οι συνθήκες;  
Πώς σχετίζονται οι σχέσεις;

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

select **EmplId**  
from **JOB J**, **ASSIGNMENT A**  
where **J.JobId=A.JobId** and **SkillCode='T5'**

Το EmplId υπάρχει και στο ASSIGNMENT άρα το να συμπεριλάβουμε και τη σχέση EMPLOYEE είναι περιττό και **μη αποδοτικό**.

# Παράδειγμα SQL (6)

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Βρες τα **Empl Id** των υπαλλήλων που τους έχει ανατεθεί τώρα κάποια θέση.

```
select EmplId  
from ASSIGNMENT  
where TermDate IS NULL
```

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id	Job Id	Start Date	Term Date
25X15	S25X	1-3-2015	1-12-2023
<del>34Y70</del>	S26Z	20-4-2016	30-6-2020
34Y70	F5	1-7-2020	NULL

Βρες τα **Empl Id** των υπαλλήλων που **δεν** τους έχει ανατεθεί τώρα κάποια θέση.

```
select EmplId  
from ASSIGNMENT  
where TermDate IS NOT NULL
```

Λάθος γιατί

- 1) συμπεριλαμβάνει υπαλλήλους που κατείχαν προηγούμενες θέσεις και κατέχουν και τώρα κάποια θέση
- 2) Δεν συμπεριλαμβάνει τους υπαλλήλους που δεν κατείχαν ποτέ κάποια θέση.

# Nested queries (IN) - 1

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτεΰων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

Βρες τα **Empl Id** των υπαλλήλων όπου κάποια θέση που κατείχαν/κατέχουν έχει **skill code 'T5'**

```
select EmplId
from JOB J, ASSIGNMENT A
where J.JobId=A.JobId and SkillCode='T5'
```

Εναλλακτικά:

```
select EmplId
from ASSIGNMENT A
where A.JobId IN (select J.JobId
from JOB J
where SkillCode='T5')
```

Ψάξε το A.JobId στο αποτέλεσμα του subquery

# Nested queries (IN) - 2

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτείων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

Βρες τα ονόματα των υπαλλήλων όπου κάποια θέση που κατείχαν/κατέχουν έχει skill code 'T5'

```
select Name
from EMPLOYEE E, JOB J, ASSIGNMENT A
where E.EmplId=A.EmplId and J.JobId=A.JobId
and SkillCode='T5'
```

Εναλλακτικά (1):

```
select Name
from EMPLOYEE E, ASSIGNMENT A
where E.EmplId=A.EmplId
and A.JobId IN (select J.JobId
from JOB J
where SkillCode='T5')
```

# Nested queries (IN) - 3

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτείων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

Βρες τα ονόματα των υπαλλήλων όπου κάποια θέση που κατείχαν/κατέχουν έχει skill code 'T5'

```
select Name
from EMPLOYEE E, JOB J, ASSIGNMENT A
where E.EmplId=A.EmplId and J.JobId=A.JobId
and SkillCode='T5'
```

Εναλλακτικά (2):

```
select Name
from EMPLOYEE E,
where E.EmplId IN (select A.EmplId
from JOB J, ASSIGNMENT A
where J.JobId=A.JobId and
SkillCode='T5')
```

# Nested queries (IN) - 4

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτεΐων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

Βρες τα ονόματα των υπαλλήλων όπου κάποια θέση που κατείχαν/κατέχουν έχει skill code 'T5'

```
select Name
from EMPLOYEE E, JOB J, ASSIGNMENT A
where E.EmplId=A.EmplId and J.JobId=A.JobId
and SkillCode='T5'
```

Εναλλακτικά (3):

```
select Name
from EMPLOYEE E,
where E.EmplId IN (select A.EmplId
from ASSIGNMENT A
where A.JobId IN (select J.JobId
from JOB J
where SkillCode='T5'))
```



# Nested queries (NOT IN) - 1

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
⋮	⋮	⋮	⋮

Βρες τα **Empl Id** των υπαλλήλων που τους έχει ανατεθεί τώρα κάποια θέση.

```
select EmplId  
from ASSIGNMENT  
where TermDate IS NULL
```

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id	Job Id	Start Date	Term Date
25X15	S25X	1-3-2015	1-12-2023
34Y70	S26Z	20-4-2016	30-6-2020
34Y70	F5	1-7-2020	NULL

Βρες τα **Empl Id** των υπαλλήλων που **δεν** τους έχει ανατεθεί τώρα κάποια θέση.

Θέλουμε όλα τα Empl Id εκτός από τα μπλε.

```
Select EmplId  
From EMPLOYEE E  
Where E.EmplId NOT IN (select A.EmplId  
from ASSIGNMENT A  
where TermDate IS NULL)
```

# Nested queries (NOT IN) - 2

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	<del>Μαρία Γεωργιάδου</del>	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Βρες τα **Empl Id** των υπαλλήλων που τους έχει ανατεθεί τώρα κάποια θέση.

```
select EmplId  
from ASSIGNMENT  
where TermDate IS NULL
```

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id	Job Id	Start Date	Term Date
25X15	S25X	1-3-2015	1-12-2023
34Y70	S26Z	20-4-2016	30-6-2020
34Y70	F5	1-7-2020	NULL

Βρες τα **ονόματα των υπαλλήλων** των υπαλλήλων που **δεν** τους έχει ανατεθεί τώρα κάποια θέση.

```
Select Name  
From EMPLOYEE E  
Where E.EmplId NOT IN (select A.EmplId  
from ASSIGNMENT A  
where TermDate IS NULL)
```

# Ερωτήσεις για το σπίτι

**Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)**

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

**Σχέση JOB (ΘΕΣΗ)**

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτείων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

**Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)**

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

1) Βρες όλα τα τμήματα (Dept) που έχει δουλέψει η Μαρία Γεωργιάδου

2) Βρες όλα τα τα τμήματα (Dept) που έχει δουλέψει ο υπάλληλος με κωδικό (EmplId) 23Y34

3) Βρες τα ΑΦΜ (SSN) των υπαλλήλων που έχουν δουλέψει ως γραμματείς.

4) Βρες τους κωδικούς των υπαλλήλων (EmplId) που έχουν δουλέψει ως γραμματείς.

# Απαντήσεις 1

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτεύων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

1) Βρες όλα τα τμήματα (Dept) που έχει δουλέψει η Μαρία Γεωργιάδου

```
select Dept
from EEMPLOYEE E, JOB J, ASSIGNMENT A
where E.EmplId=A.EmplId and J.JobId=A.JobId
and Name = 'Μαρία Γεωργιάδου'
```

2) Βρες όλα τα τα τμήματα (Dept) που έχει δουλέψει ο υπάλληλος με κωδικό (EmplId) 23Y34

```
select Dept
from JOB J, ASSIGNMENT A
where J.JobId=A.JobId and EmplId = '23Y34'
```

# Απαντήσεις 2

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Σχέση JOB (ΘΕΣΗ)

Job Id (Κωδικός θέσης)	Job Title (Τίτλος θέσης)	Skill Code (Βαθμός ικανότητας)	Dept (Τμήμα)
S25X	Γραμματέας	T5	Προσωπικού
S26Z	Γραμματέας	T6	Λογιστήριο
F5	Εποπτείων ορόφου	FM3	Πωλήσεων
•	•	•	•
•	•	•	•
•	•	•	•

Σχέση ASSIGNMENT (ΑΝΑΘΕΣΗ)

Empl Id (Κωδικός υπαλλήλου)	Job Id (Κωδικός θέσης)	Start Date (Ημερομηνία έναρξης)	Term Date (Ημερομηνία τέλους)
23Y34	S25X	1-3-1999	30-4-2001
34Y70	F5	1-10-2002	*
23Y34	S26Z	1-5-2001	*
•	•	•	•
•	•	•	•
•	•	•	•

3) Βρες τα ΑΦΜ (SSN) των υπαλλήλων που έχουν δουλέψει ως γραμματείς.

```
select SSN
from EMPLOYEE E, JOB J, ASSIGNMENT A
where E.EmplId=A.EmplId and J.JobId=A.JobId
and JobTitle = 'Γραμματέας'
```

4) Βρες τους κωδικούς των υπαλλήλων (EmplId) που έχουν δουλέψει ως γραμματείς.

```
select EmplId
from JOB J, ASSIGNMENT A
where J.JobId=A.JobId and JobTitle =
'Γραμματέας'
```

# Matching strings (1)

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

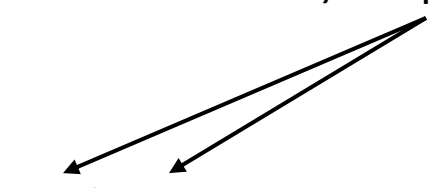
Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Η SQL έχει δύο ειδικούς χαρακτήρες για σύγκριση με συμβολοσειρές χαρακτήρων.

- **percent (%)**: ταυτίζεται με οποιαδήποτε συμβολοσειρά
- **underscore (\_)**: ταυτίζεται με οποιοδήποτε χαρακτήρα
- Π.χ. Βρες τα ονόματα των υπαλλήλων που τα EmplId τους περιέχουν τη συμβολοσειρά '34'

Ταυτίζεται με οποιοδήποτε string

```
select Name
from EMPLOYEE
where EmplId like '%34%'
```



# Matching strings (2)

Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

Η SQL έχει δύο ειδικούς χαρακτήρες για σύγκριση με συμβολοσειρές χαρακτήρων.

- **percent (%)**: ταυτίζεται με οποιαδήποτε συμβολοσειρά
- **underscore (\_)**: ταυτίζεται με οποιοδήποτε χαρακτήρα
- Π.χ. Βρες τα ονόματα των υπαλλήλων που τα EmpId ξεκινάνε από '2', τελειώνουν σε '5' και έχουν **τουλάχιστον** 3 χαρακτήρες.

```
select Name  
from EMPLOYEE  
where EmpId like '2_ %5'
```

Ταυτίζεται με  
έναν χαρακτήρα.

Ταυτίζεται με οποιοδήποτε string **και  
με το κενό.**

# Παραδείγματα SQL (7)

**Σχέση EMPLOYEE (ΥΠΑΛΛΗΛΟΣ)**

Empl Id (Κωδικός υπαλλήλου)	Name (Όνομα)	Address (Διεύθυνση)	SSN (ΑΦΜ)
25X15	Γιάννης Αθανασίου	Ακαδημίας 1234	111223333
34Y70	Μαρία Γεωργιάδου	Πανεπιστημίου 982	999009999
23Y34	Στέλιος Μπέλλος	Σταδίου 724	111005555
•	•	•	•
•	•	•	•
•	•	•	•

- `insert into EMPLOYEE  
values ('43W12', 'Ελένη Αλεξίου', 'Πατησίων 76', '444661111')`
- `insert into EMPLOYEE (EmplId, Name)  
values ('83X21', 'Γεώργιος Χριστιάς')`
- `delete from EMPLOYEE  
where Name = 'Στέλιος Μπέλλος'`
- `update EMPLOYEE  
set Address = 'Ιπποκράτους 35'  
where Name = 'Μαρία Γεωργιάδου'`



# Συναλλαγές

- **Συναλλαγή (transaction):** Ακολουθία λειτουργιών που πρέπει να συμβούν όλες μαζί και διαβάζουν ή αλλάζουν τη ΒΔ.
  - Παράδειγμα: μεταφορά χρημάτων μεταξύ τραπεζικών λογαριασμών, αλλαγή στην ΒΔ των στοιχείων ενός υπαλλήλου
  - Π.χ.: `read(balance)`, `balance = balance - 10`, `write(balance)`
  - Τα **read** και **write** είναι λειτουργίες που επικοινωνούν με τη ΒΔ
- Μία συναλλαγή
  - είτε είναι επιτυχής (**commits**), εκτελείται και πιθανότατα αλλάζει τη ΒΔ
  - είτε αποτυγχάνει (**aborts**), και πρέπει να επαναφέρει τις αλλαγές που έκανε στη ΒΔ

# Ιδιότητες Συναλλαγών

- **Atomicity** – «Όλα ή τίποτα», μία συναλλαγή είτε θα εκτελεστεί όλη επιτυχώς, είτε θα αφήσει τη ΒΔ χωρίς αλλαγές.
- **Consistency** – Πρέπει να μετατρέψει τη ΒΔ από μία σταθερή κατάσταση σε μία άλλη.
  - Π.χ. στη μεταφορά χρημάτων μεταξύ τραπεζικών λογαριασμών, δεν επιτρέπεται να αφαιρεθούν τα χρήματα από τον ένα λογαριασμό και να μην προστεθούν στον άλλο.
- **Isolation** – Κάθε συναλλαγή εκτελείται ανεξάρτητα από τις άλλες.
  - Οι αλλαγές που κάνουν μη ολοκληρωμένες συναλλαγές δεν πρέπει να είναι εμφανείς στις άλλες συναλλαγές.
- **Durability** – Οι αλλαγές που κάνει στη ΒΔ μία επιτυχώς ολοκληρωμένη συναλλαγή (committed transaction) πρέπει να είναι μόνιμες.

# Συγχρονισμός συναλλαγών

- Οι συναλλαγές μπορεί να εκτελούνται και να έχουν πρόσβαση στη ΒΔ **παράλληλα**.
  - **Εξοικονόμηση χρόνου!**
  - Μπορεί η μία συναλλαγή να **«μπλέκει»** με την άλλη και να δημιουργούνται προβλήματα. Π.χ.
    - Πρόβλημα **χαμένης ενημέρωσης**
    - Πρόβλημα **λανθασμένης σύνοψης**

# Πρόβλημα χαμένης ενημέρωσης

- $bal_x$  αρχικά 100€
- $T_1$  κάνει ανάληψη 10€
- $T_2$  καταθέτει 100€
- Τελικό σωστό ποσό 190€.

**Πρόβλημα:** η αλλαγή της  $T_2$  παρακάπτεται από την  $T_1$ .

**Λύση:** μην επιτρέψεις στην  $T_1$  να διαβάσει πριν ολοκληρωθεί η  $T_2$ .

Time	$T_1$	$T_2$	$bal_x$
$t_1$		begin_transaction	100
$t_2$	begin_transaction	read( $bal_x$ ) ← 100	100
$t_3$	100 → read( $bal_x$ )	$bal_x = bal_x + 100$	100
$t_4$	$bal_x = bal_x - 10$	write( $bal_x$ ) → 200	200
$t_5$	90 ← write( $bal_x$ )	commit	90
$t_6$	commit		90

# Πρόβλημα λανθασμένης σύνοψης

- $T_6$  υπολογίζει το **άθροισμα** των λογαριασμών  $x$ ,  $y$ ,  $z$  που πρέπει να είναι 175€.
- $T_5$  **μεταφέρει** 10€ από το λογαριασμό  $x$  στο  $z$ .

**Πρόβλημα:** η  $T_6$  διαβάζει την παλιά τιμή της  $x$  και την καινούργια της  $z$ .

**Λύση:** μην επιτρέψεις στην  $T_6$  να διαβάσει τίποτα πριν ολοκληρωθεί η  $T_5$ .

Time	$T_5$	$T_6$	$bal_x$	$bal_y$	$bal_z$	sum
$t_1$		begin_transaction	100	50	25	→ 175
$t_2$	begin_transaction	sum = 0	100	50	25	0
$t_3$	read( $bal_x$ )	read( $bal_x$ ) ← old value	100	50	25	0
$t_4$	$bal_x = bal_x - 10$	sum = sum + $bal_x$	100	50	25	100
$t_5$	write( $bal_x$ )	read( $bal_y$ )	90	50	25	100
$t_6$	read( $bal_z$ )	sum = sum + $bal_y$	90	50	25	150
$t_7$	$bal_z = bal_z + 10$		90	50	25	150
$t_8$	write( $bal_z$ )		90	50	35	150
$t_9$	commit	read( $bal_z$ ) ← updated value	90	50	35	150
$t_{10}$		sum = sum + $bal_z$	90	50	35	185
$t_{11}$		commit	90	50	35	185

# Έλεγχος Συγχρονισμού (1)

- Το ΣΔΒΔ πρέπει να αναγκάζει τις συναλλαγές που «μπλέκονται» να εκτελούνται **μια κάθε φορά**
  - Χρησιμοποιούνται τεχνικές **χρονο-προγραμματισμού** και **αμοιβαίου αποκλεισμού**
  - Σε μια ΒΔ **πρέπει να επιτρέπεται** να εκτελούνται **ταυτόχρονα** συναλλαγές **που δεν επηρεάζουν η μια το αποτέλεσμα της άλλης**
- **Κλείδωμα (lock)**: αποτροπή άλλων συναλλαγών από το να προσπελάσουν δεδομένα που χρησιμοποιούνται από μια συναλλαγή
  - **Κοινόχρηστο (shared)** κλείδωμα: χρησιμοποιείται κατά την ανάγνωση δεδομένων (δεν τροποποιούνται τα δεδομένα)
  - **Αποκλειστικό (exclusive)** κλείδωμα: χρησιμοποιείται κατά την τροποποίηση δεδομένων
  - **Επιτρέπεται ταυτόχρονη πρόσβαση σε συναλλαγές που ζητούν κοινόχρηστη πρόσβαση**

## Έλεγχος Συγχρονισμού (2)

- **Αδιέξοδο (deadlock)** αν 2 συναλλαγές απαιτούν η καθεμία αποκλειστική πρόσβαση σε ίδια δεδομένα (μπορεί να μπλοκάρουν η μια την άλλη)
  - Αν η καθεμιά παίρνει αποκλειστική πρόσβαση σε ένα (διαφορετικό) τμήμα δεδομένων και επιμένει να περιμένει την άλλη

Time	T <sub>17</sub>	T <sub>18</sub>
t <sub>1</sub>	begin_transaction	
t <sub>2</sub>	write_lock(bal <sub>x</sub> )	begin_transaction
t <sub>3</sub>	read(bal <sub>x</sub> )	write_lock(bal <sub>y</sub> )
t <sub>4</sub>	bal <sub>x</sub> = bal <sub>x</sub> - 10	read(bal <sub>y</sub> )
t <sub>5</sub>	write(bal <sub>x</sub> )	bal <sub>y</sub> = bal <sub>y</sub> + 100
t <sub>6</sub>	write_lock(bal <sub>y</sub> )	write(bal <sub>y</sub> )
t <sub>7</sub>	WAIT	write_lock(bal <sub>x</sub> )
t <sub>8</sub>	WAIT	WAIT
t <sub>9</sub>	WAIT	WAIT
t <sub>10</sub>	⋮	WAIT
t <sub>11</sub>	⋮	⋮

## Έλεγχος Συγχρονισμού (3)

- **Αποφυγή τέτοιων φαινομένων.** Π.χ. με ανάθεση προτεραιοτήτων σε συναλλαγές:
  - Προτεραιότητα σε παλαιότερες συναλλαγές
  - Αν μια παλαιότερη συναλλαγή απαιτεί πρόσβαση σε δεδομένα που είναι κλειδωμένα από μια νεότερη συναλλαγή, η τελευταία αναγκάζεται να **απελευθερώσει** τα δεδομένα από το κλείδωμα, και η παλαιότερη αποκτά πρόσβαση
  - Οι συναλλαγές καθώς «παλαιώνουν» παίρνουν μεγαλύτερη προτεραιότητα
- **Αναίρεση (roll back):** Ακύρωση μερικώς εκτελεσμένης ή αποτυχημένης συναλλαγής
  - αν υπάρξει βλάβη πριν φτάσουμε στο σημείο δέσμευσης (commit)
  - Αναίρεση ενεργειών που έγιναν στο μεταξύ
- **Συνεχόμενη αναίρεση (cascading rollback):** Διαδοχική αναίρεση λειτουργιών αν η τελευταία είχε χρησιμοποιηθεί από μια άλλη, η οποία με τη σειρά της είχε χρησιμοποιηθεί από άλλη, κ.ο.κ.



# Εξόρυξη Δεδομένων

- **Εξόρυξη δεδομένων (data mining):** Ο τομέας της Επιστήμης Υπολογιστών που ασχολείται με την ανακάλυψη προτύπων (pattern recognition), τάσεων (trends), και κρυμμένων/άγνωστων συμπερασμάτων (inference) από συλλογές δεδομένων
  - Εφαρμογές: Κοινωνικά δίκτυα, συστήματα συστάσεων, παγκόσμιος ιστός, οικονομία, βιοπληροφορική, ασφάλεια υπολογιστών,...
- **Σήμερα: Εξαιρετικά μεγάλος όγκος δεδομένων (Big Data)**
  - Οι αλγόριθμοι πρέπει να είναι υψηλής κλιμάκωσης ώστε να μπορούν να διαχειρίζονται tera-bytes από δεδομένα
- Υψηλών διαστάσεων και πολυπλοκότητας δεδομένα
  - Ροές δεδομένων (Data streams), π.χ. δεδομένα αισθητήρων
  - Δομημένα δεδομένα: γραφήματα, κοινωνικά δίκτυα (social networks)
  - Βάσεις δεδομένων
  - Χωροχρονικά δεδομένα, εικόνες, βίντεο, κείμενο

# Στόχοι Εξόρυξης Δεδομένων (1)

- **Περιγραφή κλάσεων (class description):** αναγνώριση ιδιοτήτων που χαρακτηρίζει μια (γνωστή) ομάδα δεδομένων
  - Π.χ. Αναγνώριση χαρακτηριστικών των ανθρώπων που αγοράζουν iphone
- **Διάκριση κλάσεων ή Κατηγοριοποίηση (class discrimination, classification):** αναγνώριση ιδιοτήτων που ξεχωρίζουν δυο ή περισσότερες (γνωστές) ομάδες δεδομένων
  - Π.χ. Αναγνώριση χαρακτηριστικών που ξεχωρίζουν αυτούς που αγοράζουν καινούρια από αυτούς που αγοράζουν μεταχειρισμένα αυτοκίνητα
- **Ανάλυση συστάδων (cluster analysis):** εύρεση ιδιοτήτων με στόχο την **ανακάλυψη ομαδοποιήσεων** από δεδομένα
  - Π.χ. Άτομα 4-10 ετών και 25-40 ετών είδαν μια κινηματογραφική ταινία
- **Ανάλυση συσχετισμών (association analysis):** εύρεση συσχετισμών ανάμεσα σε ομάδες δεδομένων
  - Π.χ. Πελάτες που αγοράζουν πατατάκια, αγοράζουν και κοκα-κόλα

## Στόχοι Εξόρυξης Δεδομένων (2)

- **Ανάλυση αποκλίσεων (outlier analysis):** Αναγνωρίζει δεδομένα που δεν εμπίπτουν στην κανονικότητα των υπολοίπων
  - Π.χ. για αναγνώριση λαθών σε συλλογές δεδομένων, ανίχνευση υποκλοπής πιστωτικής κάρτας, ανίχνευση τρομοκρατίας, cyber-bullying, κλπ
- **Σειριακή ανάλυση προτύπων (sequential pattern analysis):** Αναγνώριση πρότυπα συμπεριφοράς/τάσεων κατά τη διάρκεια του χρόνου
  - Π.χ. Πρόβλεψη τιμής μετοχών, αναγνώριση τάσεων (trendy topic detection) σε κοινωνικά δίκτυα
- **Πρόβλεψη (prediction):** Αν μία οντότητα διαθέτει τις ιδιότητες που χαρακτηρίζουν μία κλάση, τότε πιθανότατα συμπεριφέρεται όπως τα μέλη της.

# Μηχανική Μάθηση

- Μάθηση βασισμένη σε δεδομένα

- η σημαντικές ιδιότητες/χαρακτηριστικά των δεδομένων (features)
- Δεδομένα στη μορφή: (ιδιότητα<sub>1</sub>, ιδιότητα<sub>2</sub>,... ιδιότητα<sub>n</sub>, label)
- Το label παίρνει τιμές σε ένα διακριτό ή συνεχές σύνολο τιμών

- Επιτηρούμενη μάθηση (Supervised learning)

- Αν το label παίρνει διακριτές τιμές, η επιτηρούμενη μάθηση λέγεται **κατηγοριοποίηση (classification)**
- Αν το label παίρνει συνεχείς τιμές, η επιτηρούμενη μάθηση λέγεται **παλινδρόμηση (regression)**

- Μη επιτηρούμενη μάθηση (Unsupervised learning) ή **συσταδοποίηση (clustering)**

- Δεν υπάρχουν labels στα δεδομένα
- Δεδομένα στη μορφή: (ιδιότητα<sub>1</sub>, ιδιότητα<sub>2</sub>,... ιδιότητα<sub>n</sub>)

# Παραδείγματα Μηχανικής Μάθησης

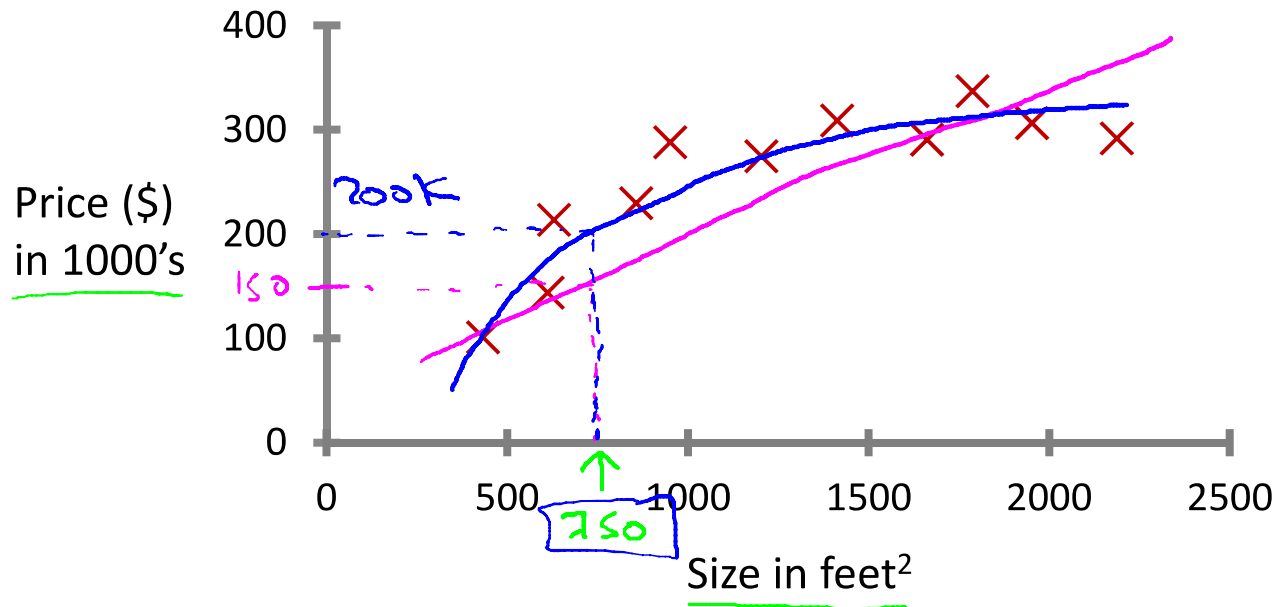
- **Κατηγοριοποίηση**
  - Εκτίμηση αν ένα email είναι spam ή όχι spam
  - Εκτίμηση αν ένας χρήστης θα κάνει click σε μια διαφήμιση ή όχι
  - Πρόβλεψη τι καιρό θα κάνει αύριο {ηλιοφάνεια, βροχή, χιονόνερο, συννεφιά, χιόνι}
  - Εκτίμηση αν μια προτεινόμενη ταινία θα αρέσει ή όχι σε έναν χρήστη
  - Συμπέρασμα αν μια εικόνα δείχνει μια γάτα ή έναν σκύλο
- **Παλινδρόμηση**
  - Πρόβλεψη τιμής μετοχών, πρόβλεψη πωλήσεων
  - Πρόβλεψη τηλεπικοινωνιακής κίνησης
  - Πρόβλεψη πιθανότητας κλικ σε διαφημίσεις
- **Συσταδοποίηση**
  - Ομαδοποίηση χρηστών μιας online πλατφόρμας
  - Ομαδοποίηση πελατών με βάση ορισμένα χαρακτηριστικά (π.χ. μισθός κλπ)
- Στην συσταδοποίηση δεν γνωρίζω την κατηγορία (όπως στην κατηγοριοποίηση) ή ποιο μέγεθος θέλω να προβλέψω (όπως στην παλινδρόμηση)

# Στάδια μηχανικής μάθησης

- 1. Εκπαίδευση (Training)
  - Με βάση ένα σύνολο από δεδομένα, εξάγω ένα μοντέλο Μηχανικής Μάθησης
  - Το μοντέλο Μηχανικής Μάθησης θα με καθοδηγεί να προβλέπω/κατηγοριοποιώ άγνωστα δεδομένα
- 2. Έλεγχος (Testing)
  - Αποτιμώ το μοντέλο σε άγνωστα δεδομένα
    - π.χ. ως προς το ποσοστό σφαλμάτων
- Συνήθως από ένα σύνολο δεδομένων, κρατώ το 80% των δεδομένων για εκπαίδευση και το υπόλοιπο 20% για έλεγχο

# Παράδειγμα: Παλινδρόμηση (regression)

Housing price prediction.



Supervised Learning

"right answers" given

Regression: Predict continuous valued output (price)

Πρόβλημα παλινδρόμησης:

Θέλω να βρω την μορφή της συνάρτησης  $y = f(\mathbf{x})$ ,  $f: \mathbb{R}^n \rightarrow \mathbb{R}$

Μου δίνεται ένα σύνολο από  $N$  δεδομένα  $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)$

# Δεδομένα Εκπαίδευσης (training data) για παλινδρόμηση

Training set of housing prices (Portland, OR)

Size in feet <sup>2</sup> (x)	Price (\$) in 1000's (y)
→ 2104	460
1416	232
→ 1534	315
852	178
...	...

$m = 47$

Notation:

- $m$  = Number of training examples
- $x$ 's = "input" variable / features
- $y$ 's = "output" variable / "target" variable

$(x, y)$  - one training example

$(x^{(i)}, y^{(i)})$  -  $i$ th training example

$$\left\{ \begin{array}{l} x^{(1)} = 2104 \\ x^{(2)} = 1416 \\ y^{(1)} = 460 \end{array} \right.$$



# Γραμμική Παλινδρόμηση (linear regression)-(1)

Training Set	Size in feet <sup>2</sup> (x)	Price (\$) in 1000's (y)
	2104'	460'
	1416'	232'
	1534'	315'
	852'	178'
	...'	...'

*m* = 47

Hypothesis:  $h_{\theta}(x) = \theta_0 + \theta_1 x$

$\theta_i$ 's: Parameters

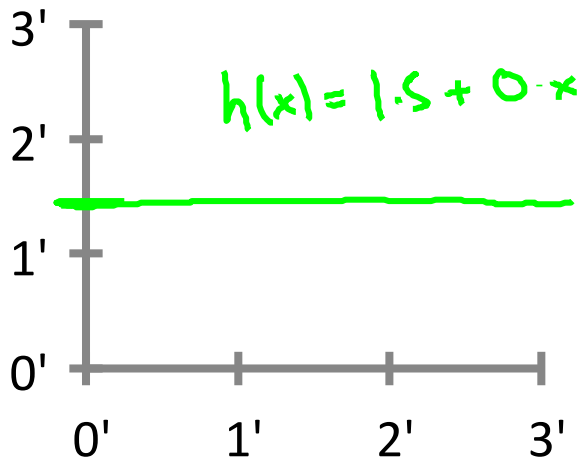
How to choose  $\theta_i$ 's?

**Γραμμική παλινδρόμηση:** αναζητώ την άγνωστη γραμμική συνάρτηση  $h$

Αν βρω το μοντέλο (hypothesis), θα μπορώ να προβλέπω την τιμή του σπιτιού στο μέλλον για οποιοδήποτε άγνωστο μέγεθος (size) σπιτιού

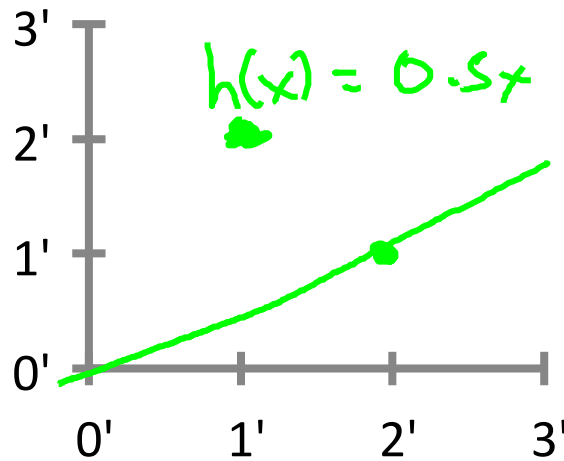
# Γραμμική Παλινδρόμηση (linear regression)-(2)

$$\underline{h_{\theta}(x)} = \theta_0 + \theta_1 x$$



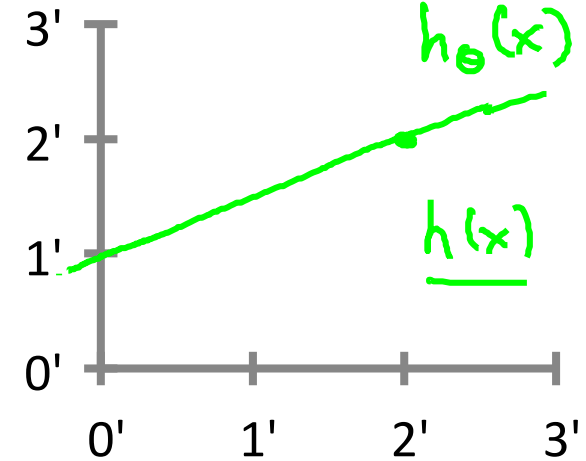
→  $\theta_0 = 1.5$

→  $\theta_1 = 0$



→  $\theta_0 = 0$

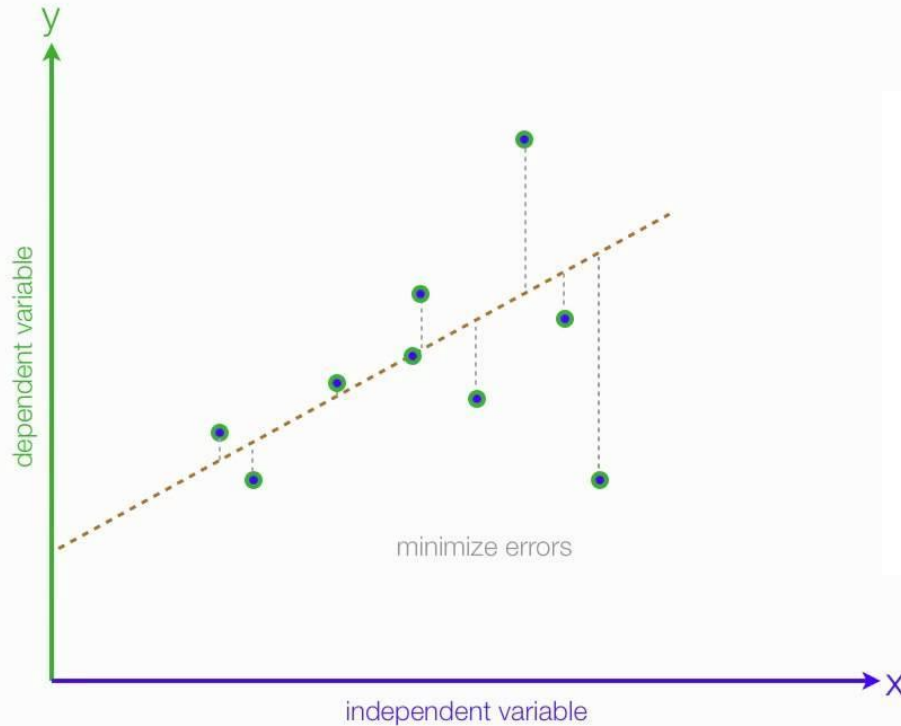
→  $\theta_1 = 0.5$



→  $\theta_0 = 1$

→  $\theta_1 = 0.5$

# Στόχος Γραμμικής Παλινδρόμησης



**Hypothesis:**  $h_{\theta}(x) = \theta_0 + \theta_1 \cdot x$   
How to choose  $\theta_0$  and  $\theta_1$ ?

**Idea:** choose  $\theta_0$  and  $\theta_1$  such that  $h_{\theta}(x)$  is “close” to  $y$  for the training set

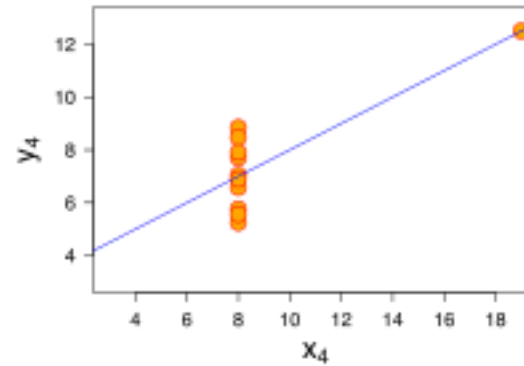
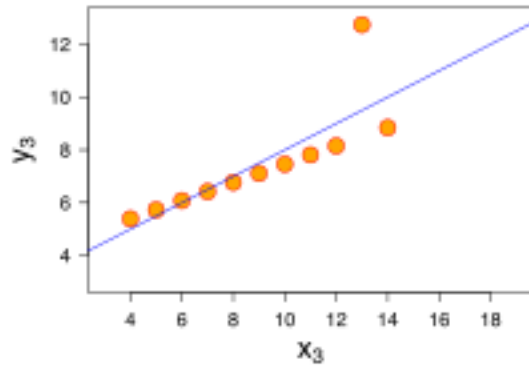
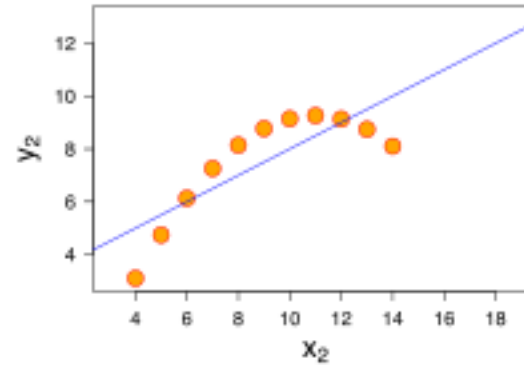
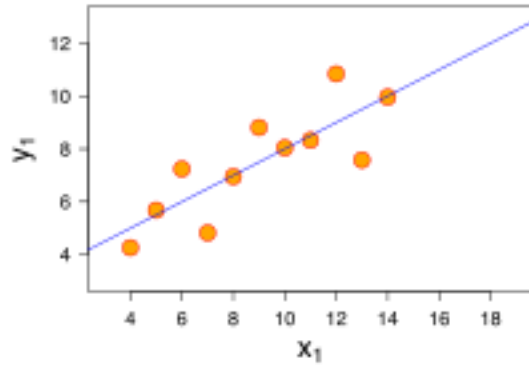
For  $(x^{(i)}, y^{(i)})$ , the **error** is  $|h_{\theta}(x^{(i)}) - y^{(i)}|$

Use of **square error function:**

$$J(\theta_0, \theta_1) = \frac{1}{2m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2$$

Find  $\theta_0$  and  $\theta_1$  that **minimize**  $J(\theta_0, \theta_1)$

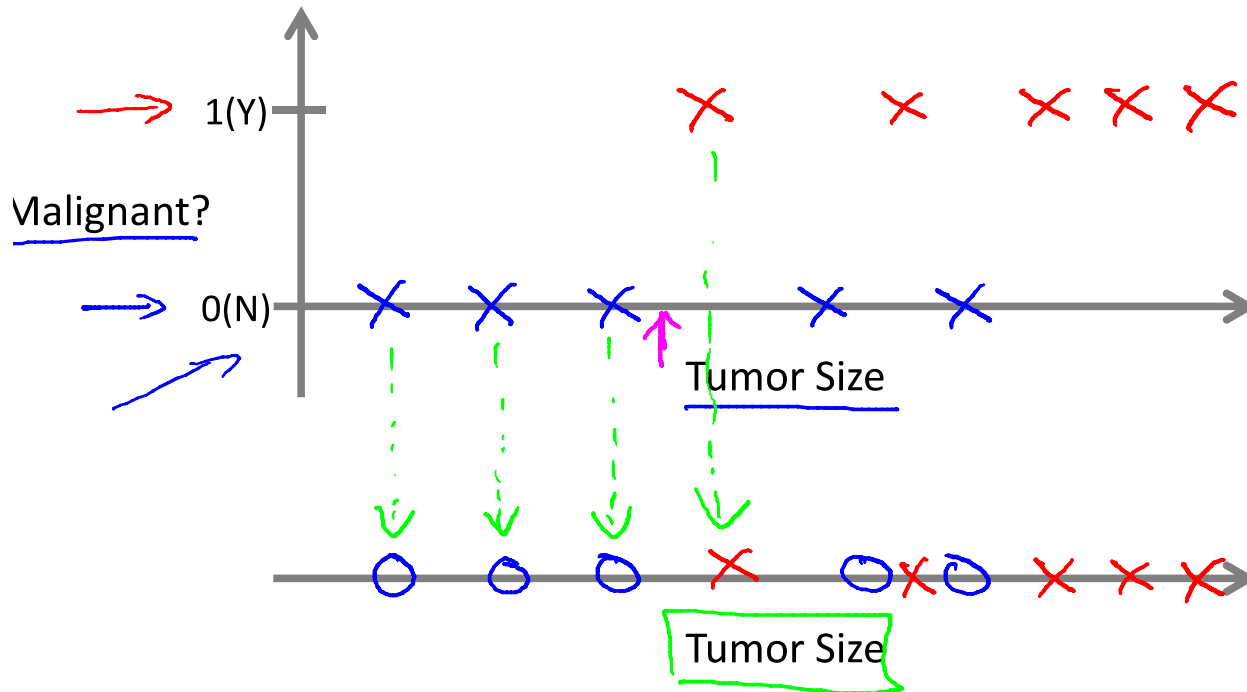
# Γραμμική Παλινδρόμηση (linear regression)-(3)



Η γραμμική παλινδρόμηση δεν είναι πάντα η κατάλληλη για πρόβλεψη.

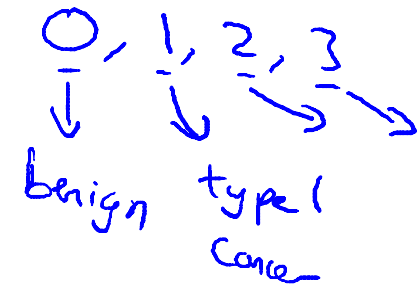
# Παράδειγμα κατηγοριοποίησης (classification)

## Breast cancer (malignant, benign)



### Classification

Discrete valued output (0 or 1)



### Πρόβλημα κατηγοριοποίησης:

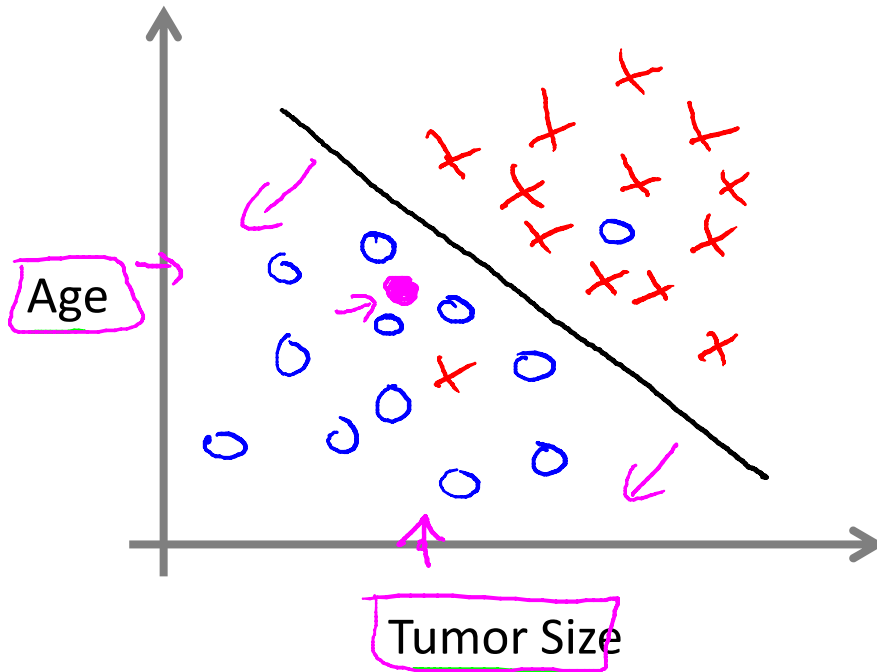
Θέλω να βρω την μορφή της συνάρτησης  $y = f(\mathbf{x})$ ,  $f: \mathbb{R}^n \rightarrow \{0,1\}$

Μου δίνεται ένα σύνολο από  $N$  δεδομένα  $(\mathbf{x}_1, y_1), \dots, (\mathbf{x}_N, y_N)$

Στην κατηγοριοποίηση, έχουμε label σε ένα διακριτό σύνολο (εδώ στο  $\{0,1\}$ )

Στο παράδειγμα αυτό υπάρχει μια ιδιότητα (tumor size) που καθορίζει αν ο όγκος είναι καλοήθης/κακοήθης

# Παράδειγμα κατηγοριοποίησης με 2 ιδιότητες



- Clump Thickness
- Uniformity of Cell Size
- Uniformity of Cell Shape
- ...

# Και άλλα παραδείγματα κατηγοριοποίησης

## Classific (on+

- Email: Spam / Not Spam?
- Online Transactions: Fraudulent (Yes / No)?
- Tumor: Malignant / Benign ?

→  $y \in \{0, 1\}$

0: "Negative Class" (e.g., benign tumor)

1: "Positive Class" (e.g., malignant tumor)

→  $y \in \{0, 1, 2, 3\}$

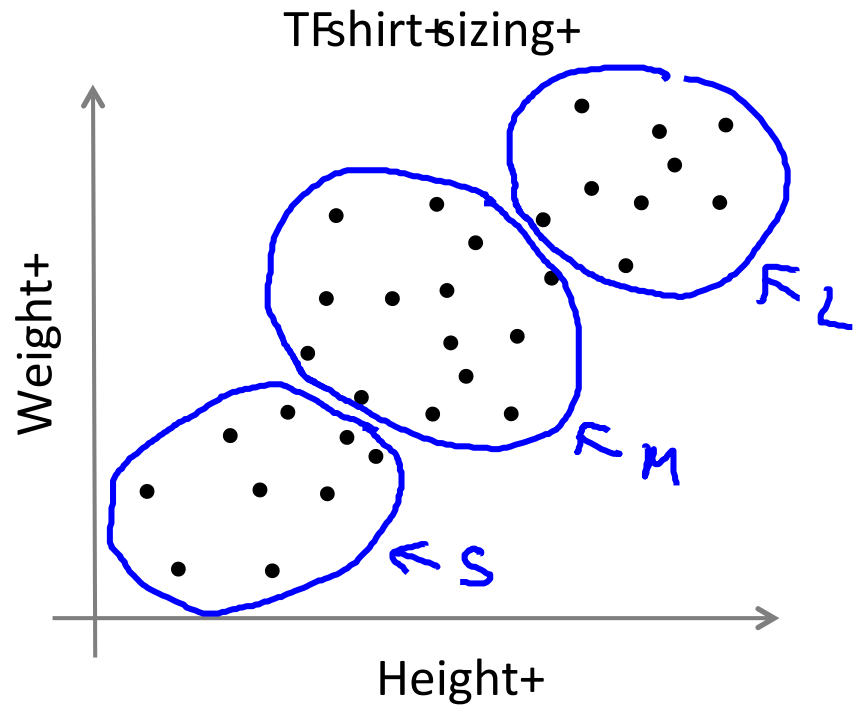
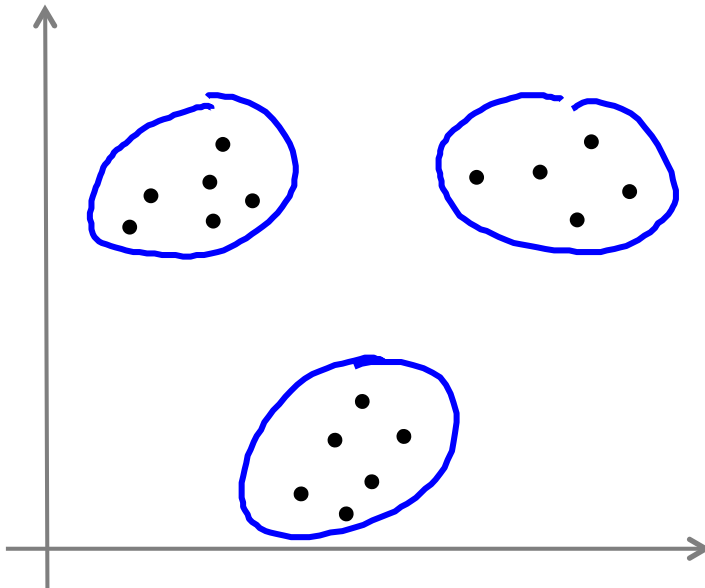
# Συσταδοποίηση

- Στη συσταδοποίηση έχω τα δεδομένα στην μορφή  $\mathbf{x}_1, \dots, \mathbf{x}_N$
- Το κάθε δεδομένο (σημείο)  $\mathbf{x}_i$  είναι ένα διάνυσμα διάστασης  $n$  (όσα και τα χαρακτηριστικά των δεδομένων)
  - Παριστάνεται σαν ένα σημείο σε έναν Ευκλείδιο χώρο  $n$  διαστάσεων
- ΑΛΛΑ **δεν έχω κατηγορίες** (κλάσεις)
- Στόχος είναι **να βρω ομάδες** (clusters) δεδομένων



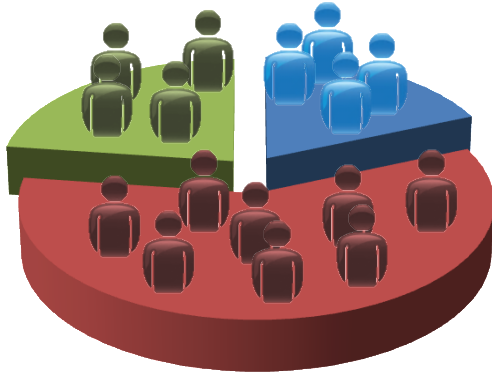
# Αποτέλεσμα Συσταδοποίησης

- Συστάδες

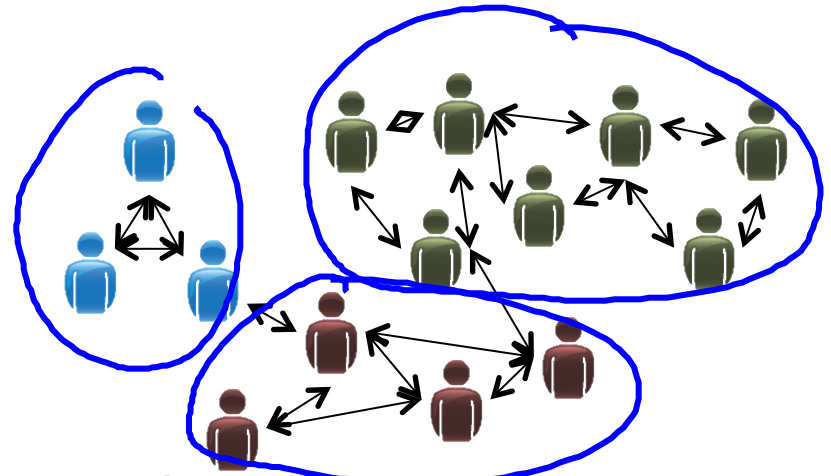


# Εφαρμογές Συσταδοποίησης

## Applications of clustering



→ Market segmentation



→ Social network analysis

- Συστήματα συστάσεων (συσταδοποίηση χρηστών με παρόμοια ενδιαφέροντα)
  - ώστε να προτείνουμε ταινίες ή μουσική σε αυτούς
- Ομαδοποίηση χρηστών ώστε να τους προβληθούν σχετικές παρόμοιες διαφημίσεις
- Ανίχνευση κοινοτήτων (Community Detection) σε κοινωνικά δίκτυα

# Κοινωνικές επιπτώσεις της τεχνολογίας των βάσεων δεδομένων

- Προβλήματα:
  - Συλλογή προσωπικών δεδομένων σε μαζική κλίμακα
    - Συνήθως χωρίς γνώση / χωρίς οικειοθελή συναίνεση των εμπλεκόμενων ατόμων
  - Τα δεδομένα μπορεί να χρησιμοποιηθούν από τρίτους που ο χρήστης που τα δίνει ούτε καν γνωρίζει
    - Π.χ. Δεδομένα που συλλέγονται από Google, Yahoo!, Amazon, κλπ και μέσω cookies
    - Μπορούν να πωληθούν σε τρίτους με σκοπό κυρίως την στοχευμένη διαφήμιση (targeted advertising)
  - Ζητήματα Ιδιωτικότητας (privacy) για τον χρήστη
  - Δικαίωμα στη λήθη (“The right to be forgotten”)
- Κυρώσεις:
  - Οι υπάρχουσες ποινικές κυρώσεις είναι αναποτελεσματικές
  - Η «αρνητική δημοσιότητα» (negative publicity) μπορεί να είναι η πιο αποτελεσματική αποτρεπτική τακτική

# Παραδείγματα (ιδιωτικού) προφίλ χρήστη



Πλοήγηση στο διαδίκτυο



Έξυπνοι μετρητές



Υπηρεσίες με βάση την κινητή θέση



Συστήματα συστάσεων

# Παραδείγματα Κινδύνων Ιδιωτικότητας Δεδομένων

Περίπτωση	Προφίλ χρήστη	Εμπλεκόμενοι	Σκοπός
Πλοήγηση στο web	Συμπεριφορά πλοήγησης (επισκεψιμότητα ιστοσελίδων, συχνότητα, κκλ)	Google, Yahoo!, Facebook, Twitter, Amazon, ...	Στοχευμένη διαφήμιση
Θέση (κινητού)	Θέσεις που πέρασε το κινητό, διάρκεια διαμονής, τροχιά	Πάροχοι κινητής τηλεφωνίας, κατασκευαστές εξοπλισμού για κινητά (π.χ. Apple, Samsung...)	Υπηρεσίες με βάση τη θέση (Location-based services), διαφημίσεις (location-based ads, εντοπισμός στο χάρτη, ειδοποιήσεις, κλπ
Έξυπνοι μετρητές ηλ. Ενέργειας	Προφίλ κατανάλωσης ενέργειας, συσκευές ανοιχτές, κλπ	Πάροχοι ηλ. Ενέργειας	Υπηρεσίες όπως demand response, εκπτώσεις, διαφημίσεις
Συστήματα Συστάσεων	Βαθμολόγηση αντικειμένων (ταινίες, βιβλία, ξενοδοχεία)	Online αγορές (Amazon), πάροχοι περιεχομένου (Spotify, Netflix),...	Συστάσεις αντικειμένων στο χρήστη

# Για όποιον/α ενδιαφέρεται να το ψάξει περισσότερο

- Τι άλλα μοντέλα βάσεων δεδομένων υπάρχουν εκτός από το σχεσιακό;
- Εξηγείστε τον αλγόριθμο **K-means** για επίλυση του προβλήματος της συσταδοποίησης (clustering) προσπαθώντας να δώσετε τα βασικά μαθηματικά πίσω από αυτόν και το πως δουλεύει
- Τι είναι ο κανονισμός **GDPR** που θεσπίστηκε σχετικά πρόσφατα για την ιδιωτικότητα στη διαχείριση δεδομένων και ποιες οι βασικές του απαιτήσεις;

# Τέλος Κεφαλαίου 9

- Μιλήσαμε για **βάσεις δεδομένων**, το **σχεσιακό μοντέλο** και τη γλώσσα **SQL**. Κάναμε και μία μικρή εισαγωγή στη **μηχανική μάθηση**.
- Στο επόμενο κεφάλαιο θα ασχοληθούμε με τη **θεωρία υπολογισμού**.