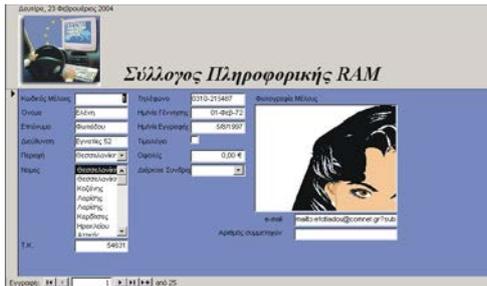


# ΔΙΑΧΕΙΡΙΣΗ ΚΑΙ ΔΙΑΣΥΝΔΕΣΗ ΣΧΕΣΙΑΚΩΝ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ



**Κατάλογος Μελών**

Νομός: Αττικής

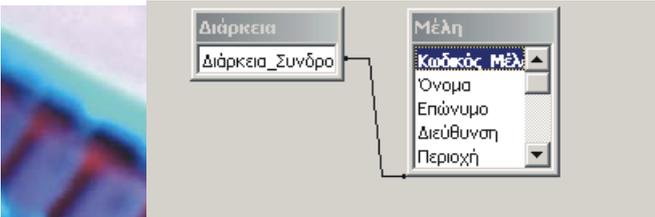
Επώνυμο	Όνομα	Τηλέφωνο
Βαρδάσις	Μιχαήλ	010-6332019
Ζουρβίης	Ιωάννης	010-611 7838
Μιχαλόπου	Στέλλα	010-8123863
Μπενέτας	Γεώργιος	010-5872742
Παπαδόση	Μαρία	010-1698732
Ρουτιάσις	Ιωσήφ	02990-86165
Τρακάλης	Αντώνιος	010-8345749
Χασώτη	Χρυστίνα	010-6618886
Χρυστακό	Ιωάννης	010-5259183

Νομός: Ηρακλείου

Επώνυμο	Όνομα	Τηλέφωνο
Δασκαλάκης	Αντώνιος	0810-540799
Λυράκης	Μιχαήλ	0810-232347
Σάκας	Χρ.ήσπος	0810-321116

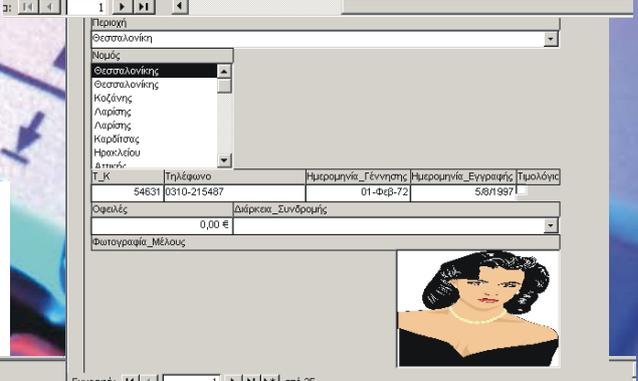
Νομός: Θεσσαλονίκης

Επώνυμο	Όνομα	Τηλέφωνο
...	...	...



**Κατάλογος διευθύνσεων**

Επώνυμο	Όνομα	αποστέλ	Διεύθυνση	Τηλ.οι	Τηλ.ερ	Κινητό	Αριθμ. Δείκτρο τηλέφω
ΑΘΑΝΑΣΙΟΥ	2	ΛΑΥΡΥΝΝΙΟΣ 218	2101098	3457845	4357345	2567457	2101234567
ΑΥΓΕΡΩΝΟΥ	4	ΣΟΛΩΝΟΣ 1	2109876	2875674	45734		
ΓΕΩΡΓΙΑΔΟΥ	6	Α. ΠΑΡΑΣΚΕΥΗΣ	2103245	3657356	45678		
ΝΙΚΟΛΑΪΔΗΣ	1	Σ. ΠΙΝΗΣ 18	2109876	2109876	88876		
ΝΙΚΟΛΑΟΥ							



Προβολή αλλαγών σε:

- Κατάλογος διευθύνσεων
- Όνομα
- Επώνυμο
- Διεύθυνση

Πεδίο:	Κατάλογος διευθύνσεων	Όνομα	Επώνυμο	Διεύθυνση	Τηλέφωνο/Οκίας	Τηλέφωνο/Ερ
Πίνακας:	Κατάλογος διευθύνσεων					
Ταξινόμηση:	<input checked="" type="checkbox"/>					
Εμφάνιση:						
Κριτήρια:						>"2109000000"
ή:						

Επιβλέπων Καθηγητής: κ. ΜΠΑΚΑΣ ΧΡΗΣΤΟΣ

Σπουδάστρια: ΠΑΝΑΓΙΩΤΑ ΜΑΣΤΡΟΓΑΛΙΑ

ΠΡΕΒΕΖΑ 2004

<b>ΕΙΣΑΓΩΓΗ</b>	<b>3</b>
<b>ΤΙ ΕΙΝΑΙ ΜΙΑ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ</b>	<b>6</b>
<b>ΚΕΦΑΛΑΙΟ Ι</b>	<b>7</b>
<b>ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ</b>	<b>7</b>
1.1 ΤΙ ΣΗΜΑΙΝΕΙ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ;	7
1.2 ΓΙΑΤΙ ΧΡΗΣΙΜΟΠΟΙΟΥΜΕ ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ	8
1.3 ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ	10
<b>ΚΕΦΑΛΑΙΟ ΙΙ</b>	<b>14</b>
<b>ΠΡΟΤΥΠΑ ΟΡΓΑΝΩΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ</b>	<b>14</b>
2.1 ΑΦΑΙΡΕΣΗ ΔΕΔΟΜΕΝΩΝ	14
2.2 ΜΟΝΤΕΛΑ ΔΕΔΟΜΕΝΩΝ	14
2.3 ΙΕΡΑΡΧΙΚΟ ΠΡΟΤΥΠΟ (HIERARCHICAL DATA MODEL)	16
2.4 ΔΙΚΤΥΩΤΟ ΠΡΟΤΥΠΟ (NETWORK MODEL)	17
2.5 ΣΧΕΣΙΑΚΟ ΠΡΟΤΥΠΟ (RELATION DATA MODEL)	19
2.6 ΠΛΕΟΝΕΚΤΗΜΑΤΑ ΚΑΙ ΜΕΙΟΝΕΚΤΗΜΑΤΑ ΤΩΝ ΤΡΙΩΝ ΠΡΟΤΥΠΩΝ ΟΡΓΑΝΩΣΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ.	24
2.7 ΣΧΕΣΙΑΚΕΣ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ	25
2.8 ΔΥΝΑΤΟΤΗΤΕΣ ΤΩΝ ΣΧΕΣΙΑΚΩΝ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ	26
<b>ΚΕΦΑΛΑΙΟ ΙΙΙ</b>	<b>27</b>
<b>ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ</b>	<b>27</b>
3.1 ΤΟ ΘΕΜΑ ΤΗΣ ΣΧΕΔΙΑΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ	27
3.2 Η ΔΙΑΔΙΚΑΣΙΑ ΤΗΣ ΣΧΕΔΙΑΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ	29
3.3 ΠΡΟΣΕΓΓΙΣΗ ΠΡΟΣΑΝΑΤΟΛΙΣΜΕΝΗ ΣΤΟ ΑΝΤΙΚΕΙΜΕΝΟ ΣΤΟ ΣΧΕΔΙΑΣΜΟ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ	30
3.4 ΕΞΕΤΑΖΟΝΤΑΣ ΤΙΣ ΣΤΑΤΙΚΕΣ ΚΑΙ ΔΥΝΑΜΙΚΕΣ ΙΔΙΟΤΗΤΕΣ ΑΝΤΙΚΕΙΜΕΝΩΝ	30
3.5 ΠΕΡΙΓΡΑΦΟΝΤΑΣ ΤΑ ΔΕΔΟΜΕΝΑ ΚΑΙ ΤΙΣ ΙΔΙΟΤΗΤΕΣ ΤΟΥΣ	30
3.6 ΛΑΜΒΑΝΟΝΤΑΣ ΥΠΟΨΗ ΤΗ ΣΥΜΠΕΡΙΦΟΡΑ ΑΝΤΙΚΕΙΜΕΝΩΝ ΜΕ ΜΕΘΟΔΟΥΣ	31
3.7 ΣΥΝΔΥΑΖΟΝΤΑΣ ΔΙΑΦΟΡΕΤΙΚΕΣ ΟΝΤΟΤΗΤΕΣ ΣΕ ΕΝΑ ΠΙΝΑΚΑ	33
3.8 ΟΡΟΛΟΓΙΑ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ	33
3.9 ΤΥΠΟΙ ΠΙΝΑΚΩΝ ΚΑΙ ΚΛΕΙΔΙΩΝ ΣΕ ΣΧΕΣΙΑΚΕΣ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ	35
3.9.1 ΔΙΑΓΡΑΜΜΑΤΑ ΜΟΝΤΕΛΩΝ ΔΕΔΟΜΕΝΩΝ	36
3.9.2 ΚΑΝΟΝΙΚΟΠΟΙΟΥΝΤΑΣ ΔΕΔΟΜΕΝΑ ΣΤΟ ΣΧΕΣΙΑΚΟ ΜΟΝΤΕΛΟ	37
3.9.3 ΚΑΝΟΝΕΣ ΚΑΝΟΝΙΚΟΠΟΙΗΣΗΣ	38
3.9.4 ΠΡΩΤΗ ΚΑΝΟΝΙΚΗ ΜΟΡΦΗ	38
3.9.5 ΔΕΥΤΕΡΗ ΚΑΝΟΝΙΚΗ ΜΟΡΦΗ	38
3.9.6 ΤΡΙΤΗ ΚΑΝΟΝΙΚΗ ΜΟΡΦΗ	39
3.9.7 ΤΕΤΑΡΤΗ ΚΑΝΟΝΙΚΗ ΜΟΡΦΗ	39
3.9.8 ΠΕΜΠΤΗ ΚΑΝΟΝΙΚΗ ΜΟΡΦΗ ΚΑΙ ΣΥΝΔΥΑΣΜΕΝΕΣ ΟΝΤΟΤΗΤΕΣ	39
3.9.10 ΤΥΠΟΙ ΣΧΕΣΕΩΝ	39
3.9.10.1 ΣΧΕΣΕΙΣ ΈΝΑ - ΠΡΟΣ - ΈΝΑ	40
3.9.10.2 ΣΧΕΣΕΙΣ ΈΝΑ – ΠΡΟΣ – ΠΟΛΛΑ	41
3.9.10.3 ΣΧΕΣΕΙΣ ΠΟΛΛΑ – ΠΡΟΣ – ΈΝΑ	42
3.9.10.4 ΣΧΕΣΕΙΣ ΠΟΛΛΑ – ΠΡΟΣ – ΠΟΛΛΑ ΚΑΙ Η ΤΕΤΑΡΤΗ ΚΑΝΟΝΙΚΗ ΜΟΡΦΗ.	43

<b>ΚΕΦΑΛΑΙΟ IV</b>	<b>48</b>
<b>Η MICROSOFT ACCESS ΣΑΝ ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ</b>	<b>48</b>
<b>ΣΧΕΣΙΑΚΩΝ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ</b>	<b>48</b>
4.1 ΟΡΙΣΜΟΣ ΔΕΔΟΜΕΝΩΝ & ΑΠΟΘΗΚΕΥΣΗ	49
4.2 ΧΕΙΡΙΣΜΟΣ ΔΕΔΟΜΕΝΩΝ	51
4.3 ΈΛΕΓΧΟΣ ΔΕΔΟΜΕΝΩΝ	52
4.4 ΔΙΑΔΙΚΑΣΙΑ ΑΝΑΠΤΥΞΗΣ ΕΝΟΣ ΣΥΣΤΗΜΑΤΟΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ	53
<b>ΚΕΦΑΛΑΙΟ V</b>	<b>56</b>
Η ΓΛΩΣΣΑ SQL	56
5.1 ΤΥΠΟΙ ΔΕΔΟΜΕΝΩΝ	57
5.2 Η SQL ΣΤΗ MICROSOFT ACCESS	58
<b>ΚΕΦΑΛΑΙΟ VI</b>	<b>60</b>
<b>ΠΡΑΚΤΙΚΟ ΜΕΡΟΣ – ΕΦΑΡΜΟΓΗ</b>	<b>60</b>
ΕΙΣΑΓΩΓΗ	60
6.1 ΣΧΕΔΙΑΣΜΟΣ ΤΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ	60
6.2 ΕΙΣΑΓΩΓΗ ΤΩΝ ΔΕΔΟΜΕΝΩΝ ΣΤΟ ΑΡΧΕΙΟ	64
6.3 ΧΡΗΣΗ ΕΡΩΤΗΜΑΤΩΝ	66
6.4 ΧΡΗΣΗ ΕΚΘΕΣΕΩΝ	68
<b>ΒΙΒΛΙΟΓΡΑΦΙΑ</b>	<b>70</b>

---

---

## Εισαγωγή

---

---

Έχετε ποτέ σκεφτεί πόσα μικρά ή μεγάλα προβλήματα αντιμετωπίζουμε και λύνουμε καθημερινά; Ή πόσες μικρές ή μεγάλες αποφάσεις παίρνουμε σε μια μέρα; Κάποιες φορές οι αποφάσεις είναι εύκολες, ούτε καν τις σκεφτόμαστε. Είναι όμως και κάποιες φορές που πρέπει να σπαζοκεφαλιάσουμε. Ας δούμε λοιπόν μέσα από το παρακάτω παράδειγμα πώς φτάνουμε στη λήψη μιας απόφασης ή στη λύση ενός προβλήματος.

«Πόσα χρήματα κρατάω μαζί μου;» αναρωτιέται η Μαρία καθώς πλησιάζει το αγαπημένο της εμπορικό κέντρο. «Α! ναι 150 euro» Φτάνοντας βλέπει τις τιμές στην βιτρίνα:»ΦΟΥΣΤΑ :40 euro, ΠΑΛΤΟ :80 euro, ΜΠΛΟΥΖΑ :30 euro».«Α, ωραία! Φτάνουν τα χρήματα που έχω!».

Σκέφτεται:

-Φτάνουν τα λεφτά μου για να αγοράσω μια Φούστα και μία μπλούζα ή μόνο για παλτό

Πλησιάζει λοιπόν την πωλήτρια και λέει:

- Την καρό φούστα & την μαύρη μπλούζα που έχετε στην βιτρίνα, θα ήθελα παρακαλώ!

Το πρόβλημα που έχει να λύσει η Μαρία στο παράδειγμά μας είναι το εξής απλό: «Τι μπορώ να ψωνίσω από το εμπορικό με τα λεφτά που έχω στην τσέπη μου;». Ας προσπαθήσουμε τώρα να αναλύσουμε τον τρόπο σκέψης της Μαρίας. Η Μαρία, λοιπόν, ξεκίνησε από το γεγονός ότι διέθετε 150 euro. Αυτό είναι ένα από τα **δεδομένα** που διαθέτει. Όμως από μόνο του δε λέει πολλά, επειδή δεν ξέρει ακόμα τι μπορεί να κάνει μ'αυτό το ποσό. Όταν διάβασε τις ετικέτες με τις τιμές «40 euro», «80 euro», «30 euro», έμαθε κάτι παραπάνω, συγκέντρωσε μερικά ακόμα **δεδομένα**.

---

**Δεδομένα** ονομάζουμε μια ομάδα από γεγονότα, σύμβολα, αριθμούς, λέξεις κ.ά. που περιγράφουν ή αντιπροσωπεύουν αντικείμενα, καταστάσεις, έννοιες, ποσότητες, ιδέες, λειτουργίες.

---

Όλα τα παραπάνω είναι στοιχεία που περιγράφουν την πραγματικότητα, τον κόσμο που μας περιβάλλει, και μπορούμε να τα συλλέξουμε από διάφορες πηγές και με πολλούς τρόπους. Όσο αυτά τα στοιχεία δεν είναι κατάλληλα οργανωμένα, δε σημαίνουν κάτι. Αν όμως τα συσχετίσουμε κατάλληλα, αν τα **επεξεργαστούμε**, αποτελούν την πρώτη ύλη για την παραγωγή χρήσιμων και αξιοποιήσιμων πληροφοριών.

Ας επιστρέψουμε όμως πάλι στο παράδειγμά μας. Τα μέχρι τώρα δεδομένα της Μαρίας είναι τα χρήματα που διαθέτει και οι τιμές που διάβασε στις ετικέτες. Οι τιμές αυτές όμως δε θα σημαίνουν τίποτα, αν πλάι σε κάθε τιμή δεν υπήρχε το αντίστοιχο είδος («φούστα, παλτό»)

Ο **συσχετισμός** μεταξύ τιμών και ειδών είναι αυτός που βοηθάει τη Μαρία να προχωρήσει στο επόμενο στάδιο για την επίλυση του προβλήματός της. Ο συσχετισμός είναι αυτός που μετατρέπει σύμβολα, όπως λέξεις ή αριθμούς, τα οποία από μόνα τους δε μας λένε τίποτα, σε κάτι που έχει νόημα. Παράγει, δηλαδή, μια **πληροφορία** που μπορεί να χρησιμεύσει σε κάτι. Ο συσχετισμός των δεδομένων «40 euro», «30 euro», «80 euro», «Φούστα», «Μπλούζα», «Παλτό» έδωσε στη Μαρία τις χρήσιμες πληροφορίες, «Τιμή Φούστας», «Τιμή Μπλούζας» και «Τιμή Παλτού». Ένας τρόπος λοιπόν για να πάρουμε, να αντλήσουμε, όπως λέμε, μια πληροφορία, είναι να συσχετίσουμε κατάλληλα κάποια δεδομένα

Έχοντας πλέον στη διάθεση της τα παραπάνω δεδομένα και πληροφορίες, εύκολα καταλαβαίνουμε ποια ήταν η επόμενη σκέψη της Μαρίας. Πρόσθεσε τα δεδομένα «Τιμή Φούστας» και «Τιμή Παλτού», σύγκρινε το αποτέλεσμα με τα χρήματα που είχε και πήρε την πληροφορία ότι τα χρήματά της δε φτάνουν.

Στη συνέχεια πρόσθεσε «Τιμή Φούστας» και «Τιμή Μπλούζας» και είδε ότι φτάνουν. Για να πάρει αυτή την τελευταία πληροφορία η Μαρία χρειάστηκε να **επεξεργαστεί** τα δεδομένα της κάνοντας κάποιες πράξεις με αυτά. Μπορούμε λοιπόν να πούμε ότι ένας άλλος τρόπος επεξεργασίας των δεδομένων είναι να εφαρμόζουμε σε αυτά κάποιες πράξεις, ενέργειες ή αποφάσεις.

---

***Επεξεργασία των δεδομένων είναι η διαδικασία με την οποία τροποποιούμε, οργανώνουμε, διαμορφώνουμε τα δεδομένα ή, ακόμα, εφαρμόζουμε με αριθμητικές ή λογικές πράξεις, προκειμένου να παράγουμε κάποια πληροφορία.***

---

Τι σημαίνει όμως, τελικά, πληροφορία; Είδαμε ότι τα πρώτα δεδομένα που πήρε η Μαρία διαβάζοντας τις ετικέτες με τις τιμές και τα είδη συσχετίστηκαν μεταξύ τους παράγοντας την πρώτη πληροφορία. Αυτή χρησιμοποιήθηκε στη συνέχεια ως δεδομένο στο επόμενο στάδιο της επεξεργασίας, ώστε να παραχθεί η επόμενη πληροφορία.

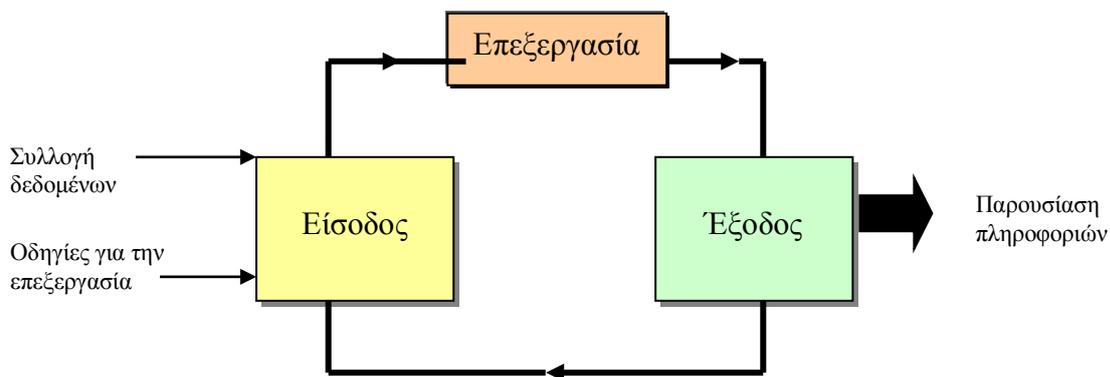
---

**Η πληροφορία, λοιπόν, προκύπτει ως αποτέλεσμα της επεξεργασίας των δεδομένων και προσπαθούμε να την αντλήσουμε από αυτά, με σκοπό να αυξήσουμε τις γνώσεις μας, να πάρουμε κάποια απόφαση ή να λύσουμε κάποιο πρόβλημα.**

---

Αν θελήσουμε να περιγράψουμε συνοπτικά όλες τις ενέργειες της Μαρίας , θα δούμε ότι έκανε τα εξής:

- ✱ Πήρε τα δεδομένα για τις τιμές των διάφορων ειδών (**συλλογή**).
- ✱ Έκανε κάποιες αριθμητικές και λογικές πράξεις με αυτά (**επεξεργασία**).
- ✱ Έφτασε στο συμπέρασμα ότι τα χρήματά του έφταναν και έδωσε την παραγγελία του (**αποτέλεσμα**).



*Σχηματική αναπαράσταση της μετατροπής δεδομένων σε πληροφορία*

Γενικεύοντας το παράδειγμά μας, μπορούμε να πούμε ότι, για να μετατρέψουμε δεδομένα σε πληροφορία, ανεξάρτητα από το αν θα χρησιμοποιήσουμε γι' αυτό το μυαλό μας ή κάποιο βοηθητικό εργαλείο, πρέπει να ακολουθήσουμε τα εξής τρία στάδια:

- ✱ **ΕΙΣΟΔΟΣ:** Συλλογή, επαλήθευση δεδομένων καθώς και οδηγίες για τον τρόπο επεξεργασίας τους.
- ✱ **ΕΠΕΞΕΡΓΑΣΙΑ:** Ταξινόμηση, υπολογισμοί, αποθήκευση δεδομένων και πληροφοριών
- ✱ **ΕΞΟΔΟΣ:** Παρουσίαση αποτελεσμάτων επεξεργασίας.

## **Τι είναι μια ΒΑΣΗ ΔΕΔΟΜΕΝΩΝ**

Μια **βάση δεδομένων (Data Base)** αποτελεί μία οργανωμένη συλλογή ειδικά ταξινομημένων δεδομένων, σχεδιασμένη με τέτοιο τρόπο, ώστε να μπορεί να εξυπηρετήσει αποτελεσματικά πολλές εφαρμογές, μειώνοντας τις άσκοπες επαναλήψεις των δεδομένων.

Αντί να δημιουργούνται ξεχωριστά αρχεία για κάθε εφαρμογή, όλα τα δεδομένα καταχωρούνται με τέτοια μέθοδο, ώστε να παρουσιάζονται στο χρήστη με έναν ενιαίο τρόπο. Κάθε εφαρμογή μπορεί να κάνει χρήση εκείνων των δεδομένων που είναι απαραίτητα για την επεξεργασία και την παραγωγή των αντίστοιχων πληροφοριών.

Τα απλά συστήματα επεξεργασίας αρχείων ικανοποιούν πλείστες απαιτήσεις σε επιχειρήσεις και οργανισμούς και κυρίως σε περιπτώσεις όπου υπάρχει μικρός αριθμός χρηστών και τα δεδομένα μπορούν να κατανεμηθούν εύκολα χωρίς να απαιτείται δημιουργία ολοκληρωμένων Πληροφοριακών Συστημάτων.

Σε άλλες όμως περιπτώσεις, όπως σε ορισμένα συστήματα συναλλαγών, στα Πληροφοριακά Συστήματα Διοίκησης και στα Συστήματα Υποστήριξης Αποφάσεων, τα συστήματα επεξεργασίας αρχείων δεν μπορούν να ικανοποιήσουν τις απαιτήσεις καθώς όλα αυτά τα συστήματα περιλαμβάνουν μεγάλο αριθμό χρηστών, πολλά αρχεία με δεδομένα που σχετίζονται μεταξύ τους και διαφορετικές όψεις των ίδιων δεδομένων. Ακόμη περισσότερο, χρήστες αυτών των συστημάτων απαιτούν γρήγορες απαντήσεις σε μη προσχεδιασμένες ερωτήσεις μια λειτουργία που δεν υποστηρίζεται από τα απλά συστήματα επεξεργασίας.

### ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

#### 1.1 Τι σημαίνει βάση δεδομένων;

Με την απλούστερη έννοια, βάση δεδομένων (database) είναι μια συλλογή από έγγραφες και αρχεία, τα οποία είναι οργανωμένα έτσι ώστε να εξυπηρετούν ένα συγκεκριμένο σκοπό. Στον υπολογιστή σας μπορεί να κρατάτε τα ονόματα και τις διευθύνσεις όλων των φίλων ή πελατών σας. Ίσως να συλλέγετε και όλες τις επιστολές που γράφετε και να τις οργανώνετε με βάση τον παραλήπτη. Μπορεί να έχετε κι ένα άλλο σύνολο αρχείων στα οποία κρατάτε όλα σας τα οικονομικά στοιχεία-πελάτες ή προμηθευτές ή τις καταχωρήσεις των βιβλίων σας και τους ισολογισμούς σας. Τα έγγραφα του επεξεργαστή κειμένου που οργανώνετε με βάση το θέμα τους, είναι άλλο ένα είδος βάσης δεδομένων. Τα αρχεία του λογιστικού φύλλου που οργανώνετε σύμφωνα με τις χρήσεις τους, αποτελούν κι αυτά ένα είδος βάσης δεδομένων.

Αν είστε πολύ οργανωμένοι, πιθανώς να μπορείτε να χειρίζεστε πολλές εκατοντάδες λογιστικών φύλλων, χρησιμοποιώντας φακέλους και υποφακέλους. Όταν το κάνετε αυτό, είστε εσείς οι διαχειριστές της βάσης δεδομένων. Ωστόσο, τι μπορείτε να κάνετε όταν τα προβλήματα που προσπαθείτε να λύσετε μεγαλώσουν πολύ; Πώς μπορείτε να συλλέγετε εύκολα πληροφορίες για όλους σας τους πελάτες και τις παραγγελίες τους όταν τα δεδομένα είναι αποθηκευμένα σε πολλά έγγραφα και αρχεία λογιστικών φύλλων; Πώς μπορείτε να διατηρείτε συνδέσμους μεταξύ των αρχείων όταν εισάγετε νέες πληροφορίες; Πώς εξασφαλίζετε ότι τα δεδομένα εισάγονται σωστά; Τι γίνεται αν πρέπει να μοιραστείτε τις πληροφορίες σας με πολλά άτομα, αλλά δε θέλετε να μπορούν δυο άτομα να προσπαθήσουν να ενημερώσουν τα ίδια δεδομένα την ίδια στιγμή; Επειδή ακριβώς έχετε να αντιμετωπίσετε όλες αυτές τις προκλήσεις, χρειάζεστε ένα σύστημα διαχείρισης βάσεων δεδομένων (database management system, DBMS).

## 1.2 Γιατί χρησιμοποιούμε βάση δεδομένων

Για εκπαιδευτικούς λόγους αλλά και για μία εμβάθυνση σε κάποια θέματα θα αναφέρουμε πλεονεκτήματα της χρήσης βάσεων (αν και ίσως στις μέρες μας δεν έχει πλέον νόημα να συζητάμε γιατί πρέπει να 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 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) δηλαδή:

- ✦ Ο σχεδιασμός των εφαρμογών γίνεται κεντρικά από ειδικούς
- ✦ Υιοθετούνται πρότυπα
- ✦ Αποφεύγονται πλεονασμοί κατά τη φύλαξη στοιχείων
- ✦ Κάθε νέα εφαρμογή αποτελεί αντικείμενο μελέτης και πρέπει να είναι συμβατή με τις παλαιότερες εφαρμογές
- ✦ Οι εφαρμογές κατασκευάζονται με τη χρήση των ίδιων εργαλείων διαχείρισης βάσεων δεδομένων κτλ.

Πρόσθετα μπορούμε να αναφέρουμε ότι:

- ✦ Μειώνεται ο χρόνος υλοποίησης και συντήρησης εφαρμογών
- ✦ Υπάρχει μεγαλύτερη ευελιξία στην περίπτωση αλλαγής απαιτήσεων των χρηστών
- ✦ Υποστηρίζεται η άμεση διαθεσιμότητα ενημερωμένων πληροφοριών
- ✦ Υποστηρίζονται πολλαπλές όψεις των δεδομένων
- ✦ Υποστηρίζεται η αποθήκευση μεταδεδομένων περιγραφής της βάσης
- ✦ Υποστηρίζεται η ανεξαρτησία προγραμμάτων και δεδομένων και η αφαίρεση δεδομένων

- ✱ Υποστηρίζεται η διαμοίραση δεδομένων ανάμεσα σε ταυτόχρονους χρήστες
- ✱ Υποστηρίζεται η άμεση επεξεργασία δοσοληψιών
- ✱ Περιορίζεται η μη εξουσιοδοτημένη προσπέλαση
- ✱ Παρέχονται πολλές διεπαφές χρηστών (user interfaces)
- ✱ Υποστηρίζεται η επιβολή περιορισμών ορθότητας που πρέπει να ισχύουν για τα δεδομένα
- ✱ Υποστηρίζεται η λήψη και χρήση εφεδρικών αντιγράφων (back-up)
- ✱ Υποστηρίζονται διαδικασίες ανάκαμψης από βλάβες (recovery)
- ✱ Υποστηρίζεται η αναπαράσταση στη βάση δεδομένων πολύπλοκων συσχετίσεων μεταξύ των δεδομένων

Οι Elmasri-Navathe αναφέρουν σαν πλεονεκτήματα:

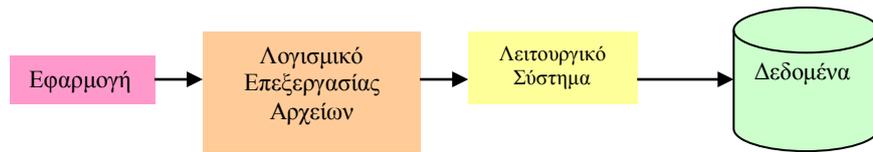
- ✘ Για **επαγωγικά ΣΔΒΔ** τις δυνατότητες ορισμού κανόνων επαγωγής για συμπερασμό νέων πληροφοριών από αποθηκευμένα στη βάση γεγονότα.
- ✘ Για **αντικειμενοστραφή ΣΔΒΔ** τη δυνατότητα μόνιμης αποθήκευσης για αντικείμενα προγραμμάτων και δομές δεδομένων.

Τέλος υπογραμμίζουν ότι η προσέγγιση των ΣΒΔ/ΣΔΒΔ επιτρέπει σε όλο τον οργανισμό να επενδύσει σε πιο ισχυρούς επεξεργαστές, μονάδες αποθήκευσης ή εξοπλισμό επικοινωνιών αντί να αγοράζει κάθε τμήμα ανεξάρτητα το δικό του. Έτσι ελαττώνεται το συνολικό κόστος λειτουργίας και διαχείρισης (**οικονομία κλίματος**)

### **1.3 Συστήματα Διαχείρισης Βάσεων Δεδομένων**

Τα Συστήματα Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΔ), αγγλικός όρος Data Base Management Systems (DBMS), είναι προγράμματα τα οποία ακολουθούν μία πιο πολύπλοκη προσέγγιση στην αποθήκευση και ανάκτηση δεδομένων από αυτή που ακολουθούν τα απλά συστήματα επεξεργασίας αρχείων (File Management System-FMS). Στην πραγματικότητα, τα ΣΔΒΔ αρχίζουν εκεί που σταματούν τα απλά συστήματα επεξεργασίας αρχείων. Τα ΣΔΒΔ εκμεταλλεύονται την ταχύτητα και την ευελιξία που προσφέρει η μέθοδος της άμεσης προσπέλασης, όσον αφορά την αποθήκευση και την ανάκτηση δεδομένων, οργανώνουν τα δεδομένα με πολύ διαφορετικό τρόπο από ότι τα απλά συστήματα.

Ένα από τα μεγαλύτερα πλεονεκτήματα των ΣΔΒΔ είναι ότι το σύστημα βρίσκει και ανακτά τα δεδομένα χωρίς τη μεσολάβηση του χρήστη. Ακόμη και οι προγραμματιστές Η/Υ δεν απαιτείται να γνωρίζουν την τοποθεσία και την μορφή των δεδομένων καθώς μέσω των προγραμμάτων ζητούν να ανεβρεθούν τα δεδομένα και το σύστημα αναλαμβάνει να τα ανακτήσει για λογαριασμό τους. Τα συστήματα επεξεργασίας αρχείων όμως λειτουργούν διαφορετικά και με τρόπο που μοιάζει με την αναζήτηση ενός βιβλίου σε μια βιβλιοθήκη όπου βρίσκουμε κάποιο βιβλίο είτε ψάχνοντας τα ράφια (επεξεργασία μέσω σειριακής αναζήτησης), είτε ψάχνοντας στις καρτέλες της βιβλιοθήκης (επεξεργασία μέσω ευρετηρίου και άμεσης προσπέλασης). Όποιο τρόπο και αν επιλέξουμε από τους δύο θα πρέπει μόνοι μας να δούμε στα ράφια της βιβλιοθήκης για να πάρουμε το βιβλίο. Αντίστοιχα, με ένα ΣΔΒΔ είναι σαν να ζητούμε από τον υπάλληλο της βιβλιοθήκης να βρει το βιβλίο που επιθυμούμε και να μας το φέρει προσδιορίζοντας το τίτλο, το συγγραφέα και το έτος έκδοσης. Η διαδικασία που ακολουθεί ο υπάλληλος και ο τρόπος οργάνωσης των βιβλίων στα ράφια δεν μας ενδιαφέρουν, δηλαδή δεν μας ενδιαφέρει αν ο υπάλληλος χρησιμοποίησε τις καρτέλες της βιβλιοθήκης για να βρει τη θέση του βιβλίου και αν τα βιβλία είναι οργανωμένα στα ράφια βάσει του τίτλου, του συγγραφέα, της ημερομηνίας έκδοσης ή του μεγέθους τους. Ακόμη περισσότερο μπορούμε με την ίδια ευκολία να ζητήσουμε από τον υπάλληλο να μας φέρει όλα τα βιβλία ενός συγκεκριμένου συγγραφέα ή τα άρθρα που δημοσιεύτηκαν πάνω σε ένα συγκεκριμένο θέμα την τελευταία πενταετία χωρίς να μας ενδιαφέρει πως θα γίνει αυτό.



α. Ανάκτηση Δεδομένων σε Συστήματα Διαχείρισης Αρχείων



β. Ανάκτηση Δεδομένων σε Σύστημα Διαχείρισης Βάσεων Δεδομένων

Σχήμα 1.3.1 Συστήματα Επεξεργασίας Αρχείων και Συστήματα Διαχείρισης Βάσεων Δεδομένων.

Αν και τα ΣΔΒΔ παρέχουν ισχύ και ελαστικότητα, πέραν κάθε συμβατικού συστήματος, όλα τα συστήματα που υποστηρίζονται από Η/Υ δεν χρησιμοποιούν την τεχνολογία αυτή κυρίως εξαιτίας του κόστους που τα συνοδεύει. Ένας άλλος λόγος είναι ότι πολλά συστήματα αναπτύχθηκαν πριν την εμφάνιση της τεχνολογίας των Βάσεων Δεδομένων και οι επιχειρήσεις ή οργανισμοί δείχνουν απροθυμία να αλλάξουν τον τρόπο λειτουργίας τους. Ένας ακόμη λόγος είναι η πολυπλοκότητα ανάπτυξης και συντήρησης ενός τέτοιου συστήματος. Η απόφαση ανάπτυξης ενός ΣΔΒΔ πρέπει να είναι απόρροια μίας ανάλυσης κόστους /οφέλους καθώς τόσο τα ΣΔΒΔ όσο και τα απλά συστήματα επεξεργασίας αρχείων είναι κατάλληλα για διαφορετικές περιπτώσεις. **Ο γενικός κανόνας που πρέπει να ακολουθείται στην επιλογή συστήματος είναι να ικανοποιούνται οι απαιτήσεις των χρηστών με το μικρότερο δυνατό κόστος**

Ένα ΣΔΒΔ είναι απλά το Λογισμικό το οποίο υποστηρίζει την συγκέντρωση των δεδομένων, την αποτελεσματική διαχείρισή τους και επιτρέπει τη πρόσβαση των εφαρμογών στις αποθηκευμένες αυτές πληροφορίες.

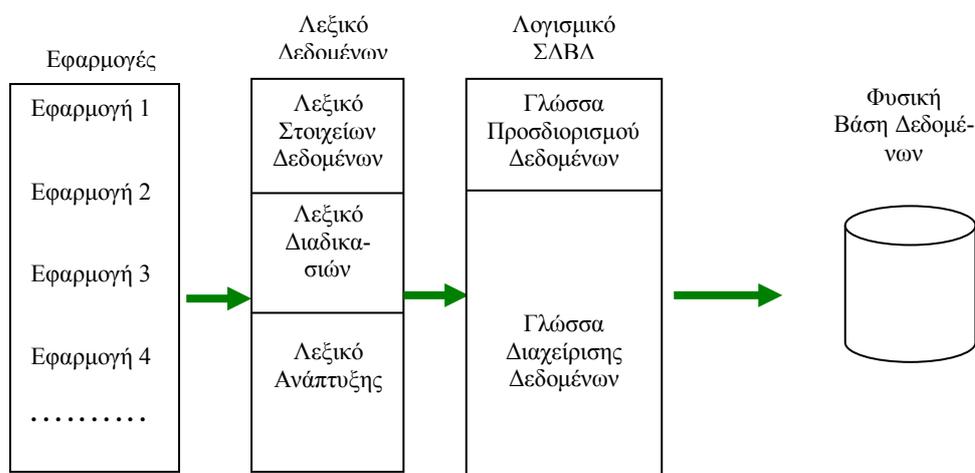
Υπάρχουν τέσσερα συστατικά στον ορισμό της Βάσης Δεδομένων τα οποία πρέπει να τονιστούν ιδιαίτερα. **Πρώτον**, υπάρχει η **φυσική υπόσταση των δεδομένων**, τα οποία τηρούνται σε κάποιο μαγνητικό μέσο, συνήθως σε μαγνητικούς δίσκους. Στη Βάση Δεδομένων υπάρχουν τα δεδομένα που χρησιμοποιούνται από κάθε λειτουργική μονάδα του οργανισμού ή της επιχείρησης όπως π.χ. το Προσωπικό. Εν αντιθέσει με τα συμβατικά συστήματα, τα δεδομένα στα ΣΔΒΔ αποθηκεύονται σε μια τοποθεσία, προσδιορίζονται οριστικά και αμετά-

κλιτα και χρησιμοποιούνται από όλες τις εφαρμογές που έχουν πρόσβαση στην συγκεκριμένη τοποθεσία.

Το δεύτερο συστατικό ενός ΣΔΒΔ είναι το **λογισμικό που χρησιμοποιείται διαμεσολαβητικά** ανάμεσα στις εφαρμογές και στα φυσικά δεδομένα και τα αρχεία όπου αυτά είναι αποθηκευμένα. Όταν μία εφαρμογή ζητήσει κάποια πληροφορία, το ΣΔΒΔ τη βρίσκει και την μεταφέρει στην εφαρμογή χωρίς ο προγραμματιστής ή ο χρήστης να απαιτείται να προσδιορίσει την συγκεκριμένη πληροφορία ή να πει στο σύστημα που βρίσκεται. Όλες οι εντολές αναζήτησης που χρησιμοποιούνται στις συμβατικές γλώσσες προγραμματισμού εδώ είναι περιττές. Τα ΣΔΒΔ προσφέρουν στους προγραμματιστές μία γλώσσα προσδιορισμού δεδομένων η οποία μπορεί να χρησιμοποιηθεί για να παρέχει την σύνδεση μεταξύ εφαρμογών και αρχείων.

Το τρίτο συστατικό είναι ένα **σύνολο εφαρμογών** όπου χρησιμοποιούνται για την ανεύρεση δεδομένων στη Βάση. Οι εφαρμογές αυτές είναι συνήθως γραμμένες σε κάποια συμβατική γλώσσα όπως η COBOL και τα δεδομένα, τα οποία απαιτούνται μέσω μίας εφαρμογής σε COBOL, ανευρίσκονται και μεταφέρονται στο ΣΔΒΔ χωρίς ο χρήστης να έχει προσδιορίσει ούτε πως, ούτε που μπορεί να βρίσκεται η πληροφορία αυτή.

Ένα ΣΔΒΔ συνοδεύεται από ένα πακέτο λογισμικού (σχήμα 1.3.2) το οποίο σε γενικές γραμμές περιλαμβάνει μία γλώσσα προσδιορισμού δεδομένων και μία γλώσσα διαχείρισης αυτών. Η γλώσσα προσδιορισμού των δεδομένων είναι η γλώσσα που χρησιμοποιείται από τους προγραμματιστές για την ανάπτυξη της Βάσης Δεδομένων.



Σχήμα 1.3.2 Τα Συστατικά ενός Συστήματος Διαχείρισης Βάσεων Δεδομένων

Τα περισσότερα ΣΔΒΔ έχουν την δική τους γλώσσα για την ανάπτυξη εφαρμογών, η οποία αποκαλείται γλώσσα διαχείρισης δεδομένων. Αυτή η γλώσσα είναι τέταρτης γενιάς και φιλική προς τον χρήστη παρέχοντας έναν αριθμό εντολών που προσφέρουν τόσο στους χρήστες όσο και στους προγραμματιστές τις απαιτούμενες δυνατότητες για την διαχείριση των δεδομένων. Η πλέον σημαντική από τις γλώσσες αυτές είναι η γλώσσα δομημένων ερωτήσεων (Structured Query Language-SQL).

Η χρησιμοποίηση της γλώσσας διαχείρισης δεδομένων ενός ΣΔΒΔ επιτρέπει την συγγραφή προγραμμάτων από τους τελικούς χρήστες (υπάλληλοι, διευθυντές χωρίς εξειδικευμένη γνώση). Τα προγράμματα αυτά περιέχουν μερικές απλές ερωτήσεις με σκοπό την παραγωγή κάποιας αναφοράς.

Το τέταρτο συστατικό ενός ΣΔΒΔ είναι το **Λεξικό Δεδομένων** το οποίο δεν είναι παρά ένα αυτοματοποιημένο αρχείο όπου τηρούνται οι ορισμοί των δεδομένων και τα χαρακτηριστικά τους όπως χρήση, φυσική αναπαράσταση, ιδιοκτησία (ποιος είναι υπεύθυνος για την συντήρησή τους), δικαιώματα πρόσβασης και ασφάλεια. Σε πολλές περιπτώσεις αυτά τα λεξικά μπορούν να παράγουν αναφορές που καταγράφουν αναλυτικά την χρήση των στοιχείων, τις τυχόν ομαδοποιήσεις των δεδομένων, εφαρμογές που έχουν πρόσβαση σε αυτά κ.λ.π. Τα περισσότερα όμως Λεξικά είναι εντελώς παθητικά δεν παρέχουν δηλαδή πληροφορίες όπως αυτές που προαναφέρθηκαν. Υπάρχουν όμως Λεξικά τα οποία είναι αρκετά προηγμένα και τα οποία φροντίζουν μετά από κάποια μεταβολή στο περιεχόμενό τους να ενημερώνουν όλες τις εφαρμογές που σχετίζονται με αυτή τη μεταβολή. Για παράδειγμα, αν χρειαστεί να αλλάξουμε τους Ταχυδρομικούς Κώδικες από πέντε σε εννιά ψηφία, απλώς εισάγουμε στο Λεξικό την μεταβολή αυτή και μετά αυτό αναλαμβάνει να ενημερώσει όλες τις εφαρμογές οι οποίες κάνουν χρήση Ταχυδρομικών Κωδικών.

## **ΠΡΟΤΥΠΑ ΟΡΓΑΝΩΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ**

### **2.1 Αφαίρεση δεδομένων**

Ένας από τους σημαντικότερους στόχους ενός συστήματος βάσεων δεδομένων είναι η διαμόρφωση μιας αφαιρετικής άποψης των δεδομένων που βλέπει ο τελικός χρήστης. Οι περισσότεροι χρήστες μίας βάσης δεδομένων δεν έχουν το επίπεδο γνώσεων, που τους επιτρέπει να καταλάβουν την πολύπλοκη δομή των δεδομένων, κάτι που είναι σε πολλές περιπτώσεις και ανεπιθύμητο. Για το σκοπό αυτό, το σύστημα σχεδιάζεται έτσι ώστε να κρύβει τη λεπτομέρεια του συστήματος σε διάφορα επίπεδα.

- **Φυσικό επίπεδο:** Το χαμηλότερο επίπεδο αφαίρεσης, όπου περιγράφεται το πώς πραγματικά αποθηκεύονται τα δεδομένα. Εδώ οι δομές δεδομένων χαμηλού επιπέδου περιγράφονται με λεπτομέρεια.
- **Εννοιολογικό επίπεδο:** Το αμέσως υψηλότερο επίπεδο αφαίρεσης περιγράφει τι δεδομένα αποθηκεύονται στη βάση και ποιες είναι οι σχέσεις μεταξύ τους. Περιγράφεται ολόκληρη η βάση δεδομένων με σχετικά απλές δομές. Ο χρήστης αυτού του επιπέδου δεν χρειάζεται να ξέρει το φυσικό επίπεδο. Στο επίπεδο αυτό, δουλεύουν και οι διαχειριστές των βάσεων.
- **Εποπτικό επίπεδο:** Το υψηλότερο επίπεδο αφαίρεσης περιγράφει μόνο ένα τμήμα της βάσης. Ακόμα και στο προηγούμενο επίπεδο, η βάση παραμένει περίπλοκη. Ο τελικός χρήστης δεν χρειάζεται αυτή την πολυπλοκότητα αλλά μόνο ένα μέρος της βάσης. Το σύστημα δίνει διάφορα μέρη της βάσης σε διαφορετικούς χρήστες σε αυτό το επίπεδο.

### **2.2 Μοντέλα δεδομένων**

Η βάση της δομής μιας βάσης δεδομένων είναι η ιδέα του *μοντέλου δεδομένων*, ενός συνόλου εργαλείων για την περιγραφή των δεδομένων, των μεταξύ τους σχέσεων και των τυχόν περιορισμών τους. Τα μοντέλα αυτά χωρίζονται γενικά σε τρεις μεγάλες κατηγορίες: λο-

γικά μοντέλα αντικειμένων (object-based logical models) , λογικά μοντέλα εγγραφών (record-based logical models) και μοντέλα φυσικών δεδομένων (physical data models).

### Λογικά Μοντέλα Αντικειμένων

Αυτά χρησιμοποιούνται για την περιγραφή των δεδομένων στα δύο ανώτερα επίπεδα. Υπάρχουν πολλά διαφορετικά μοντέλα και συνεχώς προτείνονται νέα. Μερικά από τα πιο διαδεδομένα είναι:

- Το μοντέλο οντοτήτων-συσχετίσεων (Entity-Relationship Model)
- Το αντικειμενοστραφές μοντέλο (object-oriented model)
- Το δυαδικό μοντέλο
- Το μοντέλο σημασιολογικών δεδομένων
- Το πληροφοριακό μοντέλο
- Το μοντέλο λειτουργικών δεδομένων

### Λογικά Μοντέλα Εγγραφών

Αυτά τα μοντέλα χρησιμοποιούνται για την περιγραφή των δεδομένων στα δύο υψηλότερα επίπεδα αφαίρεσης. Έχουν σκοπό τον καθορισμό της λογικής δομής της βάσης και την παροχή μιας περιγραφής της υλοποίησης σε υψηλότερο επίπεδο.

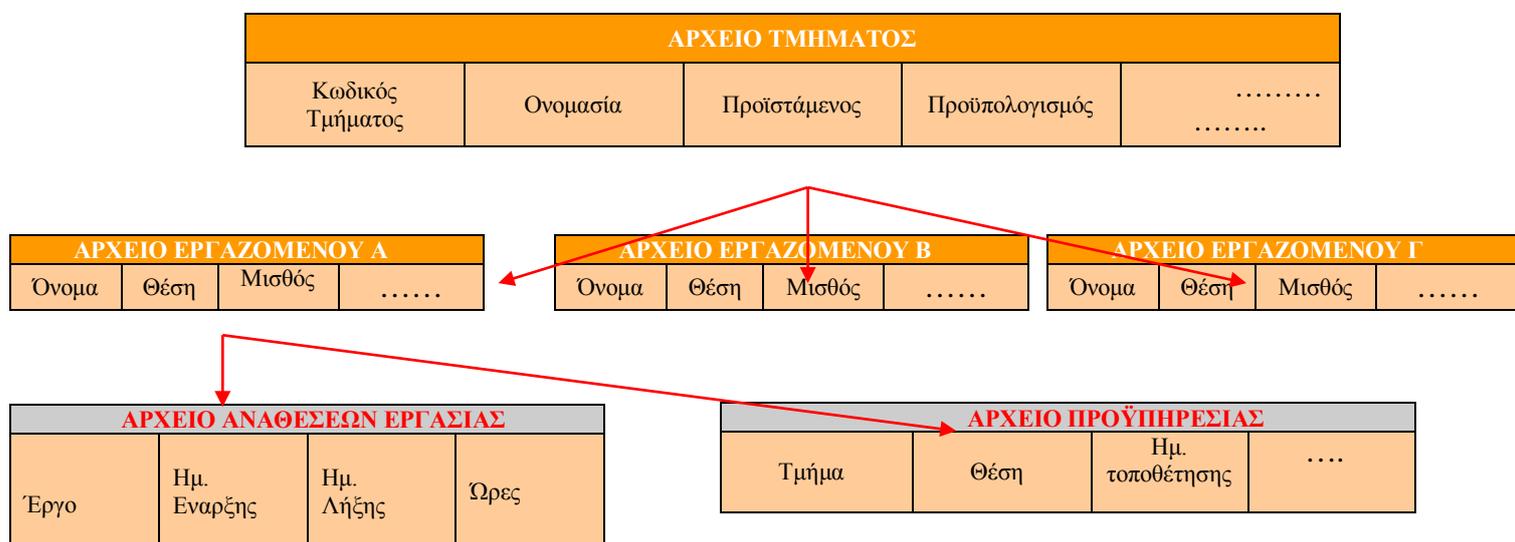
Η ονομασία των μοντέλων αυτών προέρχεται από το γεγονός ότι η βάση αποτελείται από εγγραφές σταθερής μορφής διαφόρων τύπων. Κάθε εγγραφή αποτελείται από συγκεκριμένα πεδία (θέσεις πίνακα) που έχουν συγκεκριμένο μήκος και τύπο.

Τα τρία πιο γνωστά μοντέλα δεδομένων είναι το **σχεσιακό**, το **δικτυακό** και το **ιεραρχικό**. Το σχεσιακό έχει αναδειχθεί σε «δημοφιλέστερο» τα τελευταία χρόνια. Τα άλλα δύο χρησιμοποιούνται σε παλιότερα συστήματα.

## 2.3 Ιεραρχικό Πρότυπο (Hierarchical Data Model)

Το ιεραρχικό πρότυπο αποτελεί το πρότυπο που χρησιμοποιήθηκε στα πρώτα στάδια των εφαρμογών των συστημάτων διαχείρισης βάσεων δεδομένων της ανάπτυξης και παρουσιάζει τα δεδομένα στους χρήστες (λογική όψη) οργανωμένα σε μία ιεραρχική μορφή κόμβων, όπου από έναν κόμβο κάποιου συγκεκριμένου επιπέδου οδηγούμαστε σε πολλούς κόμβους του αμέσως κατωτέρου επιπέδου. Αντίθετα, κάθε κόμβος είναι συνδεδεμένος μόνο με έναν κόμβο ανωτέρου επιπέδου

Ας θεωρήσουμε τη βάση δεδομένων του προσωπικού μιας επιχείρησης, όπως αυτή αναπτύσσεται στο παρακάτω σχήμα 2.1



**Σχήμα 2.1.** Ιεραρχική οργάνωση δεδομένων.

Το αρχείο τμήματος περιλαμβάνει τα δεδομένα του τμήματος. Κάθε τμήμα έχει πολλούς εργαζόμενους. Επομένως, το αρχείο του τμήματος συνδέεται με τα αρχεία κάθε εργαζόμενου στο τμήμα, τα οποία περιέχουν τις βασικές πληροφορίες για κάθε εργαζόμενο. Ο κάθε εργαζόμενος έχει ορισμένες εργασίες που του ανατίθενται και για τις οποίες καταχωρούνται τα στοιχεία που αφορούν την εκτέλεση της καθεμίας. Επίσης, ο κάθε εργαζόμενος έχει μια προϋπηρεσία σε άλλες θέσεις της επιχείρησης, η οποία βρίσκεται σε ξεχωριστό αρχείο.

Με την ιεραρχική δομή είναι εύκολο να εξαγάγουμε πληροφορίες, εφόσον τις αναζητούμε σύμφωνα με την ιεραρχική οργάνωση. Αυτό συμβαίνει γιατί κάθε αρχείο σε ένα επίπεδο

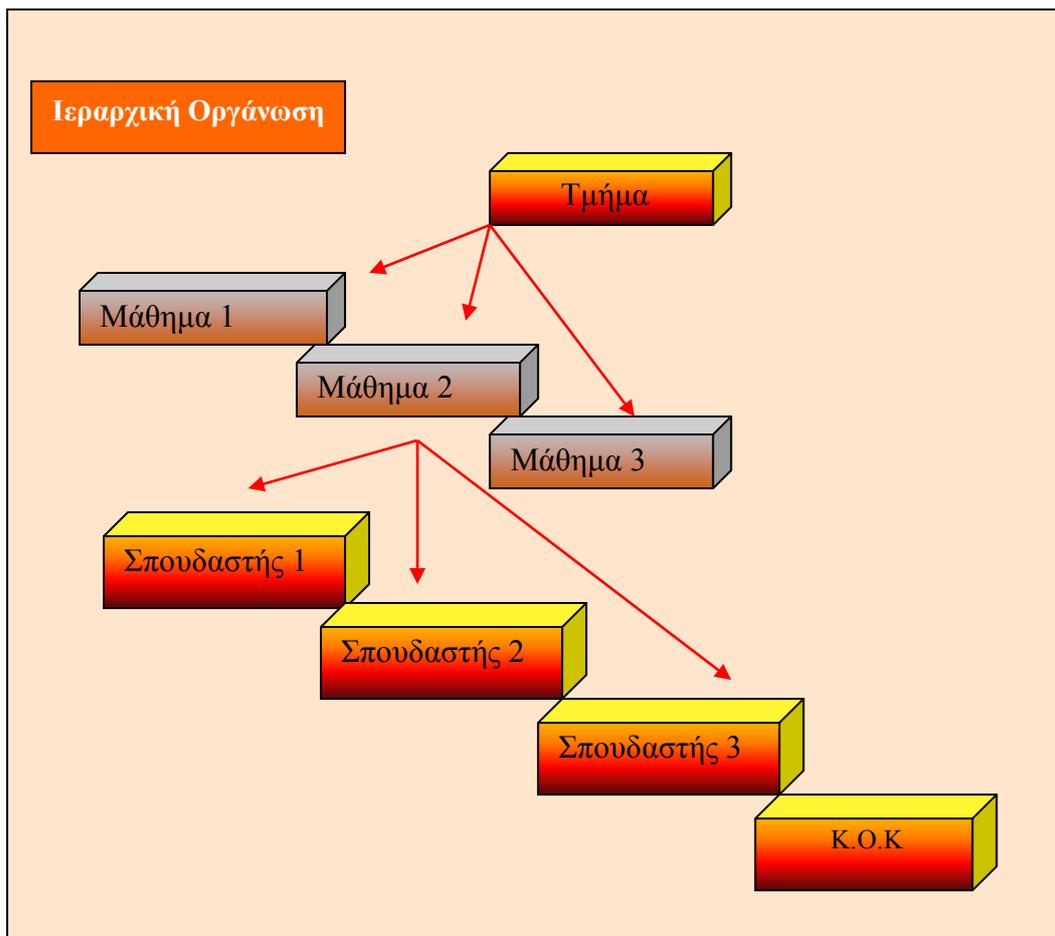
δο είναι συνδεδεμένο, με συνδέσμους-δείκτες (pointers), με τα αρχεία στην αμέσως κατώτερη ιεραρχικά κατηγορία. Μπορούμε για παράδειγμα, να έχουμε την πληροφορία για το άθροισμα της μισθοδοσίας κάθε τμήματος (αθροίζοντας τους μισθούς όλων των εργαζομένων στο τμήμα). Ας υποθέσουμε όμως ότι μία ανάθεση εργασίας μοιράστηκε σε περισσότερα από ένα άτομα, τα οποία μπορεί να βρίσκονται και σε διαφορετικά τμήματα της επιχείρησης. Επειδή δεν υπάρχουν σύνδεσμοι που οδηγούν από τα κατώτερα αρχεία των δεδομένων στα ανώτερα, η εξαγωγή τέτοιου είδους πληροφοριών δε διευκολύνεται σε μία βάση δεδομένων που είναι οργανωμένη σύμφωνα με το ιεραρχικό πρότυπο.

## **2.4 Δικτυωτό πρότυπο (Network Model)**

Τους περιορισμούς που υπάρχουν στο ιεραρχικό πρότυπο των βάσεων δεδομένων έρχεται να λύσει το δικτυωτό πρότυπο, το οποίο αποτελεί μία παραλλαγή του ιεραρχικού προτύπου.

Στο ιεραρχικό μοντέλο, κάθε αρχείο σε ένα επίπεδο ιεραρχίας συνδέεται με περισσότερα αρχεία στο αμέσως κατώτερο επίπεδο (παιδί-child), αλλά μόνο με ένα στο αμέσως ανώτερο (γονέας-parent), έχουμε δηλαδή σχέση σύνδεσης «ένα προς πολλά». Αντίθετα, το δικτυωτό πρότυπο απεικονίζει λογικές σχέσεις των δεδομένων του τύπου «πολλά προς πολλά».

*Ας υποθέσουμε ότι έχουμε τα δεδομένα που αφορούν τα σπουδαστικά αρχεία σε ένα τμήμα ενός πανεπιστημίου. Μια τυπική ιεραρχική οργάνωση των δεδομένων θα ήταν η εξής:*

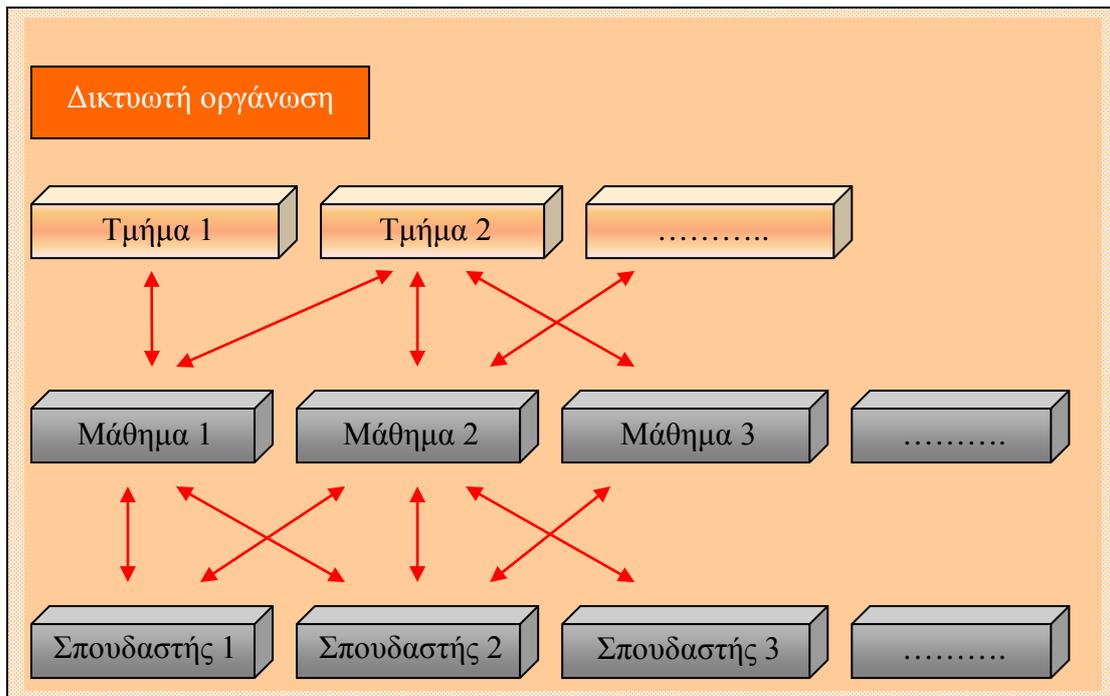


*Κάθε τμήμα έχει ορισμένα μαθήματα και κάθε μάθημα το έχουν δηλώσει για να το παρακολουθούν ορισμένοι σπουδαστές.*

*Ενώ λοιπόν, με την παραπάνω ιεραρχική οργάνωση, είναι εύκολο να αναζητήσουμε ποιοι σπουδαστές παρακολουθούν κάθε μάθημα, είναι αντίθετα, δύσκολο να βρούμε ποια μαθήματα παρακολουθεί κάθε σπουδαστής. Αυτό συμβαίνει γιατί δεν υπάρχει σύνδεση μεταξύ του σπουδαστή και του μαθήματος παρά μόνο στην άλλη κατεύθυνση. Αντίστοιχα προβλήματα θα προέκυπταν αν επιτρεπόταν να παρακολουθήσουν οι σπουδαστές και μαθήματα που ανήκουν σε άλλα τμήματα ή είναι κοινά.*

Στο δικτυωτό πρότυπο, οι συνδέσεις μεταξύ των δεδομένων των αρχείων ακολουθούν το πρότυπο «πολλά προς πολλά».

*Έτσι το προηγούμενο παράδειγμα, σε μια δικτυωτή οργάνωση της βάσης δεδομένων, θα είχε την εξής μορφή:*



*Κάθε μάθημα έχει δείκτες προς τους σπουδαστές που το έχουν δηλώσει, ενώ σε κάθε εγγραφή του σπουδαστή υπάρχουν επίσης δείκτες προς τα μαθήματα που αυτός έχει δηλώσει.*

Με αυτό τον τρόπο οργάνωσης, είναι δυνατή η προσπέλαση των δεδομένων και η αποφυγή επαναλήψεων που θα έπρεπε να είχαμε στην αντίστοιχη ιεραρχική οργάνωση (ο κάθε σπουδαστής θα έπρεπε να επαναλαμβανόταν σε κάθε τμήμα). Το τίμημα αυτής της αποτελεσματικότητας είναι η εκθετική αύξηση των συνδέσμων (pointers) μεταξύ των δεδομένων.

## **2.5 Σχεσιακό πρότυπο (Relation Data Model)**

Το σχεσιακό πρότυπο στις βάσεις δεδομένων αποτελεί την πλέον πρόσφατη εξέλιξη των βάσεων δεδομένων. Το σχεσιακό πρότυπο οργάνωσης ξεπερνά τις δυσκολίες που παρουσιάζουν τα δυο προηγούμενα πρότυπα οργάνωσης.

Στις σχεσιακές βάσεις δεδομένων, τα δεδομένα είναι οργανωμένα με τη μορφή διδιάστατων πινάκων, οι οποίοι πολλές φορές αναφέρονται και με τον όρο «αρχεία»

*Ας υποθέσουμε ότι μια επιχείρηση θέλει να δημιουργήσει μία βάση δεδομένων με τις πωλήσεις της. Οι πωλήσεις της επιχείρησης πραγματοποιούνται μέσω των πωλητών στους πελάτες της. Επειδή δεν υπάρχει αποκλειστική σχέση πελατών και πωλητών, κάθε πωλητής μπορεί να πωλήσει σε οποιοδήποτε πελάτη και αντίστοιχα οι αγορές κάθε πελάτη γίνονται μέσω διαφορετικών πωλητών.*

Τα βασικά στοιχεία των πελατών και των πωλητών της επιχείρησης έχουν οργανωθεί σε πίνακες όπως αυτοί που ακολουθούν :

<b>ΠΙΝΑΚΑΣ:ΠΕΛΑΤΕΣ</b>			
<b>ΚΩΔ .ΠΕΛΑΤΗ</b>	<b>ΟΝΟΜΑ</b>	<b>ΠΟΛΗ</b>	<b>ΠΙΣΤΩΣΗ</b>
2001	ΛΕΚΚΑΣ	ΑΘΗΝΑ	100
2002	ΜΑΝΩΛΑ	ΛΑΡΙΣΑ	200
2003	ΝΑΝΟΥΡΗ	ΛΑΜΙΑ	200
2004	ΝΑΣΤΟΣ	ΔΡΑΜΑ	300
2006	ΚΟΤΣΙΜΠΟΥ	ΑΘΗΝΑ	100
2007	ΜΩΥΣΙΑΔΗ	ΛΑΡΙΣΑ	100
2008	ΜΗΤΡΟΥ	ΛΑΜΙΑ	300

<b>ΠΙΝΑΚΑΣ:ΠΩΛΗΤΕΣ</b>			
<b>ΚΩΔ.ΠΩΛΗΤΗ</b>	<b>ΟΝΟΜΑ</b>	<b>ΠΟΛΗ</b>	<b>ΠΟΣΟΣΤΟ</b>
1001	ΑΚΡΙΤΙΔΗΣ	ΑΘΗΝΑ	0,12
1002	ΒΕΛΩΝΗ	ΛΑΜΙΑ	0,13
1003	ΣΠΑΝΟΣ	ΒΟΛΟΣ	0,1
1004	ΔΗΜΟΥ	ΑΘΗΝΑ	0,11
1007	ΖΑΧΑΡΙΟΥ	ΠΑΤΡΑ	0,15

Οι στήλες κάθε πίνακα ονομάζονται πεδία (fields), ενώ κάθε γραμμή του πίνακα αντιστοιχεί στο σύνολο των δεδομένων (τιμές των πεδίων) ενός πελάτη ή ενός πωλητή, αποτελώντας μία εγγραφή (record) του πίνακα.

Οι πελάτες δίνουν τις παραγγελίες τους μέσω των πωλητών, οι οποίες καταχωρούνται στον πίνακα: «ΠΑΡΑΓΓΕΛΙΕΣ. Κάθε πελάτης οπωσδήποτε έχει περισσότερες από μία παραγγελίες. Κάθε πωλητής έχει περισσότερες από μια παραγγελίες από διαφορετικούς πελάτες. Κάθε παραγγελία, επίσης, περιλαμβάνει περισσότερα από ένα είδη, τα οποία εμπορεύεται η επιχείρηση. Η οργάνωση των δεδομένων των παραγγελιών σε πίνακες είναι όπως παρακάτω :

<b>ΠΙΝΑΚΑΣ:ΠΑΡΑΓΓΕΛΙΕΣ</b>				
<b>ΚΩΔ.ΠΑΡΑΓΓΕΛΙΑΣ</b>	<b>ΚΩΔ.ΠΩΛΗΤΗ</b>	<b>ΚΩΔ.ΠΕΛΑΤΗ</b>	<b>ΗΜΕΡΟΜΗΝΙΑ</b>	<b>ΠΟΣΟ</b>
3001	1007	2008	3/10/1990	185.000
3002	1004	2007	3/10/1990	1.005.000
3003	1001	2001	3/10/1990	1.900.000
3005	1002	2003	3/10/1990	900.000
.....	.....	.....	.....	.....

Στον πίνακα: «ΠΑΡΑΓΓΕΛΙΕΣ» περιλαμβάνεται τόσο το πεδίο: «ΚΩΔ.ΠΕΛΑΤΗ», όσο και το πεδίο : «ΚΩΔ.ΠΩΛΗΤΗ».

Η ύπαρξη των κοινών αυτών πεδίων μεταξύ των διαφορετικών πινάκων επιτρέπει τη συσχέτιση των πινάκων μεταξύ τους. Αν θέλουμε να αναζητήσουμε τις παραγγελίες ενός πελάτη ή ενός πωλητή, αυτό είναι δυνατόν να γίνει με τον ίδιο τρόπο όπως και σε έναν ιεραρχικό τρόπο οργάνωσης δεδομένων, διότι οι πίνακες: «ΠΕΛΑΤΕΣ» και «ΠΩΛΗΤΕΣ» συνδέονται με τον πίνακα: «ΠΑΡΑΓΓΕΛΙΕΣ» με μια σχέση του τύπου «ένα προς πολλά» (ένας πελάτης έχει πολλές παραγγελίες, το ίδιο ισχύει και για κάθε πωλητή).

Ας υποθέσουμε όμως ότι επιθυμούμε μια κατάσταση όλων των πωλητών που εξυπηρέτησαν έναν πελάτη. Η αναζήτηση πληροφοριών αυτού του είδους, η οποία θα απαιτούσε μία οργάνωση δικτυωτής μορφής του τύπου «πολλά προς πολλά» (κάθε πελάτης εξυπηρετείται από πολλούς πωλητές και κάθε πωλητής εξυπηρετεί πολλούς πελάτες), γίνεται στη σχεσιακή οργάνωση του παραδείγματος μέσω του πίνακα: «ΠΑΡΑΓΓΕΛΙΕΣ», ο οποίος συνδέεται με σχέση «ένα προς πολλά» και με τους δύο πίνακες.

Η οργάνωση των σχεσιακών βάσεων δεδομένων οικοδομείται σε δύο βασικές αρχές:

- ✨ Στην ύπαρξη κοινών πεδίων στους διαφόρους πίνακες που αποτελούν τη βάση δεδομένων.

✨ Στις σχέσεις συσχέτισης του τύπου «ένα προς πολλά» μεταξύ των πινάκων.

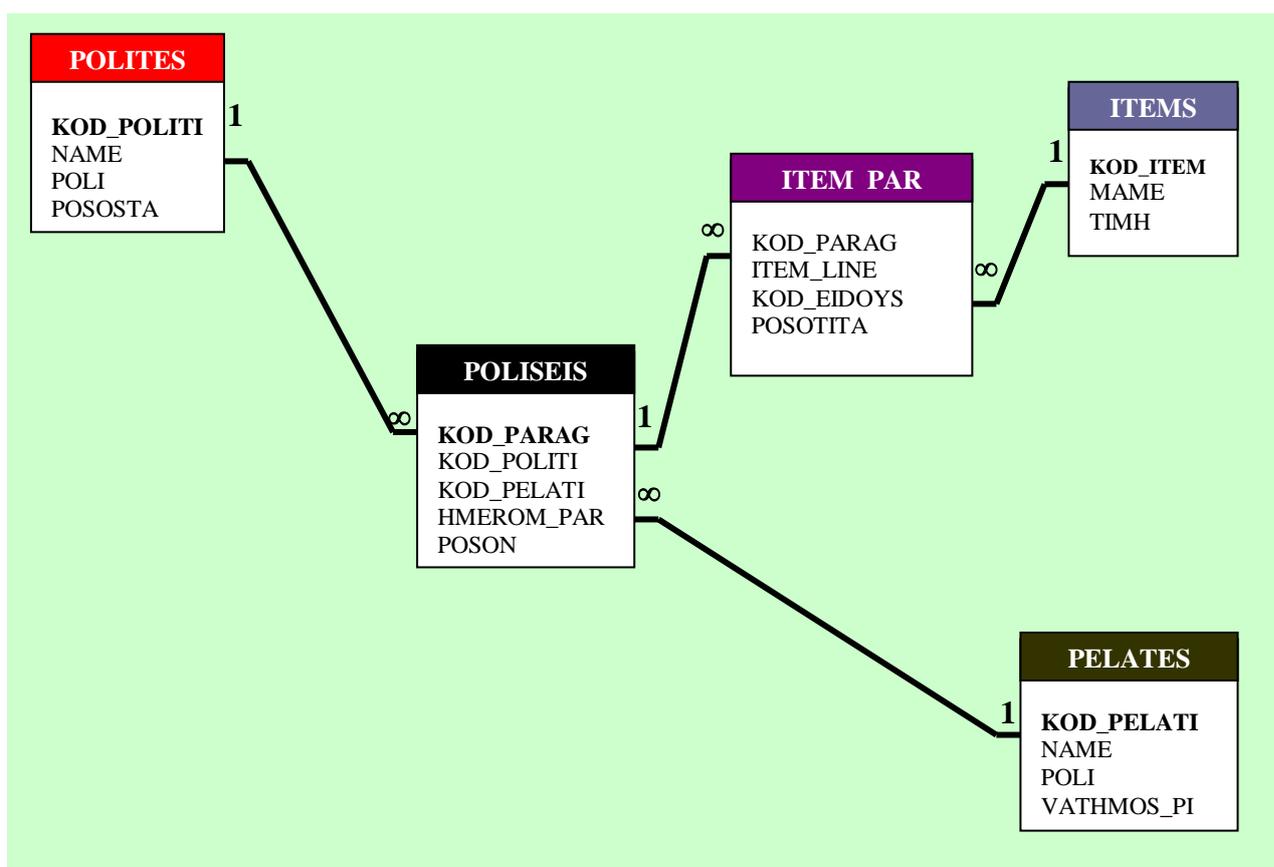
Η οργάνωση αυτή επιτρέπει την αναζήτηση δεδομένων από περισσότερους από έναν πίνακες με αποτελεσματικό τρόπο.

Ολοκληρώνοντας το παράδειγμα της σχεσιακής βάσης δεδομένων που ξεκινήσαμε στο παράδειγμα που αναφερόμαστε, μπορούμε να υποθέσουμε ότι κάθε παραγγελία αναλύεται στα επιμέρους είδη που παραγγέλθηκαν, τα οποία καταχωρούνται στον πίνακα: «ΕΙΔΗ ΠΑΡΑΓΓΕΛΙΑΣ», ενώ ο πίνακας: «ΕΙΔΗ» περιλαμβάνει τα δεδομένα των ειδών που εμπορεύεται η επιχείρηση.

<b>ΠΙΝΑΚΑΣ: ΕΙΔΗ ΠΑΡΑΓΓΕΛΙΑΣ</b>			
<b>ΚΩ- Δ.ΠΑΡΑΓΓΕΛΙΑΣ</b>	<b>ΓΡΑΜΜΗ</b>	<b>ΚΩ- Δ.ΕΙΔΟΥΣ</b>	<b>ΠΟΣΟΤΗΤΑ</b>
3001	1	A101	1
3002	1	A105	3
3002	2	A103	1
3002	3	A106	2
3003	1	A103	2
3003	2	A107	4
3005	1	A104	3
.....	.....	.....	.....

<b>ΠΙΝΑΚΑΣ: ΕΙΔΗ</b>		
<b>ΚΩΔ.ΕΙΔΟΥΣ</b>	<b>ΠΕΡΙΓΡΑΦΗ</b>	<b>ΤΙΜΗ</b>
A101	A105	185.000
A102	ΕΓΧΡ.ΟΘΟΝΗ 14''	85.000
A103	ΕΓΧΡ.ΟΘΟΝΗ 15''	150.000
.....	.....	.....

Παρατηρούμε ότι όλοι οι πίνακες της βάσης δεδομένων έχουν κάποια συσχέτιση μεταξύ τους έτσι, ώστε όλοι μαζί να παρουσιάζουν ένα ενιαίο και λογικό σύνολο δεδομένων. Οι συσχετίσεις μεταξύ των πινάκων της βάσης δεδομένων φαίνονται στο σχήμα 2.2



Σχήμα 2.2 Σχέσεις μεταξύ πινάκων της βάσης δεδομένων του παραδείγματος.

Παρατηρώντας το παραπάνω σχήμα, διαπιστώνουμε τη σύνδεση μεταξύ δύο οποιωνδήποτε πινάκων της βάσης δεδομένων, η οποία βασίζεται στην ύπαρξη κοινών πεδίων σε δύο ή περισσότερους πίνακες. Κάθε συσχέτιση χαρακτηρίζεται από μία σχέση: ένα (όπου υπάρχει ο αριθμός 1) προς πολλά (όπου υπάρχει το σύμβολο του άπειρου  $\infty$ ).

## **2.6 Πλεονεκτήματα και μειονεκτήματα των τριών προτύπων οργάνωσης βάσης δεδομένων.**

Το βασικό πλεονέκτημα του ιεραρχικού και του δικτυωτού προτύπου οργάνωσης μιας βάσης δεδομένων είναι η αποτελεσματικότερη (ταχύτερη) επεξεργασία των δεδομένων. Η αποτελεσματικότητα αυτή έχει ένα τίμημα. Ο τρόπος πρόσβασης των δεδομένων, οι δείκτες σύνδεσης των δεδομένων και οι αντίστοιχοι κατάλογοι πρέπει να οριστούν εκ των προτέρων. Οι μεταβολές σε αυτήν τη δομή δεν είναι εύκολες και απαιτούν την επέμβαση εξειδικευμένων προγραμματιστών. Επομένως, παρέχουν μικρή έως καθόλου ευελιξία στον τελικό χρήστη.

Και τα δύο πρότυπα απαιτούν, σε μεγάλο βαθμό, γνώσεις προγραμματισμού για τη διαχείριση των βάσεων δεδομένων και την αναζήτηση πληροφοριών.

Τα πλεονεκτήματα των σχεσιακών βάσεων δεδομένων είναι η μεγάλη ευελιξία στην κατά περίπτωση αναζήτηση πληροφοριών, οι δυνατότητες σύνθεσης δεδομένων από περισσότερους από έναν πίνακες, ο απλός σχεδιασμός και η ευκολία συντήρησής τους. Είναι συστήματα που ο μέσος χρήστης μπορεί εύκολα να χειριστεί για αναζήτηση πληροφοριών, χωρίς να υπάρχει πάντα η ανάγκη υποστήριξης από εξειδικευμένο προγραμματιστή. Από την άλλη πλευρά, η ευκολία χειρισμού έχει το τίμημά της στην ταχύτητα επεξεργασίας των δεδομένων, καθότι και για μία απλή αναζήτηση απαιτείται η αναζήτηση δεδομένων σε περισσότερους από έναν πίνακες και ο συσχετισμός των δεδομένων προτού να γίνει η εξαγωγή των πληροφοριών που ζητήθηκαν.

---

*Σε ορισμένες περιπτώσεις, η ταχύτητα πρόσβασης στα δεδομένα είναι το καθοριστικό χαρακτηριστικό ενός συστήματος. Φανταστείτε ένα σύστημα επεξεργασίας κράτησης θέσεων σε αεροπορικά δρομολόγια, στο οποίο έχουν πρόσβαση χιλιάδες χρήστες (ταξιδιωτικοί πράκτορες). Η απόκριση του συστή-*

ματος πρέπει να είναι άμεση. Η ευελιξία σε αυτή την περίπτωση δεν είναι το ζητούμενο, καθότι η αναζήτηση πληροφοριών μπορεί να τυποποιηθεί σε μεγάλο βαθμό.

Αντίθετα, σε ένα σύστημα διαχείρισης δεδομένων σε μια επιχείρηση η ευελιξία στην παραγωγή πληροφοριών για την άσκηση της διοίκησης είναι περισσότερο σημαντική από την ταχύτητα, η οποία, ούτως ή άλλως, λόγω του μικρού σχετικά όγκου των δεδομένων, θα είναι ικανοποιητική και με ένα σχεσιακό σύστημα διαχείρισης βάσεων δεδομένων.

---

## **2.7 Σχεσιακές βάσεις δεδομένων**

Σχεδόν όλα τα σύγχρονα συστήματα διαχείρισης βάσεων δεδομένων χειρίζονται και αποθηκεύουν πληροφορίες χρησιμοποιώντας το σχεσιακό (relational) μοντέλο διαχείρισης βάσης δεδομένων. Ο όρος σχεσιακό προκύπτει από το γεγονός ότι κάθε εγγραφή της βάσης δεδομένων περιέχει πληροφορίες συσχετισμένες με ένα μοναδικό θέμα και μόνο με αυτό. Επίσης, τα δεδομένα που αφορούν δυο κατηγορίες πληροφοριών (όπως πελάτες και παραγγελίες) μπορούν να αντιμετωπίζονται σαν μία οντότητα που βασίζεται σε συσχετισμένες τιμές δεδομένων. Για παράδειγμα, θα ήταν περιττή η αποθήκευση του ονόματος και της διεύθυνσης του πελάτη σε κάθε παραγγελία που γίνεται από αυτόν. Έτσι, σε ένα σχεσιακό σύστημα, τα στοιχεία των παραγγελιών περιέχουν κι ένα πεδίο δεδομένων, στο οποίο αποθηκεύονται κάποια στοιχεία όπως ο κωδικός του πελάτη, που χρησιμοποιείται για τη σύνδεση κάθε παραγγελίας με τα στοιχεία του πελάτη.

Ένα σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων, που μερικές φορές ονομάζεται και RDBMS (relational database management system), διαχειρίζεται όλα τα δεδομένα σε πίνακες. Στους πίνακες αποθηκεύονται πληροφορίες για κάποιο θέμα (όπως οι πελάτες ή οι μαθητές) οι στήλες περιέχουν τα διαφορετικά είδη των πληροφοριών γι' αυτό το θέμα (για παράδειγμα, τη διεύθυνση του πελάτη ή του μαθητή) και οι γραμμές περιγράφουν όλες της ιδιότητες μιας απλής περίπτωσης του θέματος (για παράδειγμα, τα στοιχεία ενός συγκεκριμένου πελάτη ή μαθητή). Ακόμα κι όταν ανακτάτε πληροφορίες από έναν ή περισσότερους πίνακες (κάτι που ονομάζεται query, ερώτημα), το αποτέλεσμα είναι πάντα κάτι που μοιάζει με άλλον έναν πίνακα.

Μπορείτε επίσης να ενώνετε τις πληροφορίες πολλών διαφορετικών πινάκων ή ερωτημάτων μέσω των συσχετισμένων τιμών. Για παράδειγμα, μπορείτε να συνδέσετε στοιχεία μαθητών με στοιχεία αιτήσεων σε κολέγια για να βρείτε ποιοι μαθητές έκαναν αίτηση σε

ποια κολέγια, ή να συνδέσετε στοιχεία υπαλλήλων με στοιχεία παραγγελιών για να βρείτε ποιος πωλητής πρέπει να πάρει προμήθεια.

## **2.8 Δυνατότητες των σχεσιακών βάσεων δεδομένων**

Κάθε σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS) σας δίνει τον πλήρη έλεγχο στον τρόπο με τον οποίο θα ορίζετε τα δεδομένα σας, θα δουλεύετε με αυτά, και θα τα μοιράζεστε με άλλους. Το σύστημα σας εφοδιάζει επίσης με ισχυρές δυνατότητες που κάνουν εύκολη την καταγραφή και το χειρισμό μεγάλων ποσοτήτων δεδομένων σε πολλούς πίνακες. Ένα RDBMS έχει τρεις βασικές δυνατότητες: ορισμό, χειρισμό, και έλεγχο δεδομένων. Όλη αυτή η λειτουργικότητα περιέχεται στις ισχυρές δυνατότητες της Microsoft Access.

## **ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ**

### **3.1 Το θέμα της σχεδίασης βάσεων δεδομένων**

Η στρατηγική σχεδίασης βάσεων δεδομένων, σύμφωνα με τον Roger Jennings(μέλος της ομάδας Microsoft Access Insiders), θέλει να επιτύχει τους παρακάτω στόχους :

- ✦ Να εκπληρώσει τις ανάγκες σας ή τις ανάγκες της επιχείρησής σας για πληροφορίες για ένα συνεπή, οικονομικό και έγκαιρο τρόπο.
  
- ✦ Να απαλείψει ή να ελαχιστοποιήσει τα διπλά δεδομένα των περιεχομένων των βάσεων δεδομένων στην επιχείρηση. Σε μια μεγάλη επιχείρηση, η απαλοιφή επαναλαμβανόμενων δεδομένων μπορεί να απαιτεί μια κατανεμημένη βάση δεδομένων. Αυτές οι βάσεις δεδομένων απαιτούν πολλούς διακομιστές για να αποθηκεύουν τις βάσεις. Οι ξεχωριστές βάσεις δεδομένων συνδέονται μεταξύ τους (για να χρησιμοποιήσουμε την ορολογία της Access) μέσα από ένα τοπικό δίκτυο ή ένα ευρύ δίκτυο, ώστε να εμφανίζονται σαν μία βάση δεδομένων στο χρήστη.
  
- ✦ Να παρέχεται γρήγορη πρόσβαση στις πληροφορίες που απαιτούνται από κάθε κατηγορία χρήστη. Η ταχύτητα είναι μία συνάρτηση του συστήματος σχεσιακών βάσεων δεδομένων, της σχεδίασης των εφαρμογών που δημιουργείτε, της ικανότητας του διακομιστή και του πελάτη και των χαρακτηριστικών του δικτύου.
  
- ✦ Να μπορεί να επεκτείνετε τις βάσεις δεδομένων για να προσαρμόζεται στις ανάγκες ενός οργανισμού που επεκτείνεται, όπως να προσθέτετε νέα προϊόντα και διαδικασίες κτλ.
  
- ✦ Να διατηρείτε την ακεραιότητα της βάσης δεδομένων ώστε να περιέχει μόνο έγκυρες πληροφορίες. Οι περισσότερες βάσεις δεδομένων πελάτη / διακομιστή, MSDE και SQL Server, παρέχουν ενσωματωμένα trigger για να διατηρούν την ακεραιότητα των βάσεων δεδομένων και να κάνουν άλλες λειτουργίες. Τα trigger είναι ένα σύνολο από κανόνες

που συμπεριλαμβάνονται στη βάση δεδομένων. Εάν παραβιάσετε έναν κανόνα, το trigger στέλνει ένα μήνυμα λάθους αντί να εκτελέσει την συναλλαγή.

✱ Να εμποδίζετε την πρόσβαση στη βάση δεδομένων σε άτομα που δεν έχουν δικαιώματα πρόσβασης. Η Access παρέχει ένα σύστημα ασφαλείας που απαιτεί από τους χρήστες να βάλουν ένα κωδικό πρόσβασης για να ανοίξουν μία συγκεκριμένη βάση δεδομένων.

✱ Να επιτρέψετε πρόσβαση μόνο σε αυτά τα στοιχεία των πληροφοριών που κάθε χρήστης ή κατηγορία χρηστών χρειάζεται στη δουλειά του. Μπορείτε να επιτρέψετε ή να αρνηθείτε στους χρήστες το δικαίωμα να βλέπουν τα δεδομένα σε κάποιους πίνακες, ανάλογα με το όνομα χρήστη και το κωδικό πρόσβασης.

✱ Να επιτρέψετε μόνο σε συγκεκριμένα άτομα να προσθέτουν ή να επεξεργάζονται πληροφορίες. Οι άδειες στην Access είναι πολλών επιπέδων. Μπορείτε επιλεκτικά να επιτρέψετε σε χρήστες να επεξεργάζονται πίνακες ή να αλλάζουν τη δομή τους όπως επίσης να επεξεργάζονται ή να δημιουργούν τις δικές τους εφαρμογές.

✱ Να διευκολύνετε τη δημιουργία εφαρμογών εισαγωγής δεδομένων, επεξεργασίας, εμφάνισης και αναφορών που εξυπηρετούν αποτελεσματικά τις ανάγκες των χρηστών της βάσης δεδομένων. Η σχεδίαση εφαρμογών RDBMS προσδιορίζει την ευκολία με την οποία δημιουργούνται νέες εφαρμογές ή μπορούν να αλλάξουν οι υπάρχουσες.

Οι πρώτοι δυο στόχοι είναι ανεξάρτητοι από τον διαχειριστή της βάσης δεδομένων που επιλέγετε. Το RDBMS επηρεάζει ή προσδιορίζει τους άλλους στόχους. Η ταχύτητα, ή παραβίαση των δεδομένων, η ασφάλεια και η δημιουργία εφαρμογών περιορίζεται από τις δυνατότητες του ίδιου του RDBMS και του υπολογιστή στον οποίο λειτουργεί. Εάν η βάση δεδομένων είναι κοινόχρηστη σε ένα δίκτυο, θα πρέπει να σκεφτείτε λειτουργίες ασφαλείας του δικτύου και το σύστημα πελάτη/διακομιστή (αν χρησιμοποιείτε κάποιο) στη στρατηγική ασφαλείας.

## 3.2 Η διαδικασία της σχεδίασης βάσεων δεδομένων

Η διαδικασία της σχεδίασης ενός συστήματος βάσης δεδομένων αποτελείται από δέκα βασικά βήματα:

- ✦ Προσδιορίστε τα αντικείμενα (δεδομένα) που θα παρουσιάσει το σύστημα.
- ✦ Ανακαλύψτε σχέσεις μεταξύ των αντικειμένων (όταν έχετε περισσότερα από ένα).
- ✦ Προσδιορίστε τις σημαντικές ιδιότητες και συμπεριφορά των αντικειμένων.
- ✦ Ορίστε πώς σχετίζονται μεταξύ τους οι ιδιότητες των αντικειμένων.
- ✦ Δημιουργήστε ένα βασικό λεξικό δεδομένων για να ορίσετε τους πίνακες που αποτελούν τη βάση της βάσης δεδομένων σας.
- ✦ Ορίστε τις σχέσεις μεταξύ πινάκων, που βασίζονται σε σχέσεις μεταξύ αντικειμένων που περιέχονται σε πίνακες και εισάγετε αυτές τις πληροφορίες στο λεξικό της βάσης δεδομένων.
- ✦ Ορίστε τον τύπο των ενημερώσεων και συναλλαγών που δημιουργούν και αλλάζουν τα δεδομένα στους πίνακες, όπως και οποιεσδήποτε απαιτήσεις ακεραιότητας δεδομένων.
- ✦ Προσδιορίστε πως θα χρησιμοποιείτε ευρετήρια για να επιταχύνετε τις λειτουργίες ερωτημάτων χωρίς να επιβραδύνετε πολύ την προσθήκη δεδομένων σε πίνακες ή να καταναλώνετε επιπλέον χώρο στο δίσκο.
- ✦ Αποφασίστε ποιος μπορεί να έχει πρόσβαση και να αλλάξει τα δεδομένα σε κάθε πίνακα (ασφάλεια δεδομένων) και να αλλάξει τη δομή των πινάκων, αν χρειάζεται για να πιστοποιήσει την ασφάλεια των δεδομένων.
- ✦ Να τεκμηριώσετε τη σχεδίαση της βάσης δεδομένων σαν σύνολο. Συμπληρώστε τα λεξικά δεδομένων για τη βάση δεδομένων σαν σύνολο και για κάθε πίνακα που περιέχει και γράψτε διαδικασίες για τη συντήρηση της βάσης δεδομένων.

### **3.3 Προσέγγιση προσανατολισμένη στο αντικείμενο στο σχεδιασμό βάσεων δεδομένων**

Οι βάσεις δεδομένων περιέχουν πληροφορίες για αντικείμενα που υπάρχουν στον πραγματικό κόσμο. Αυτά τα αντικείμενα μπορεί να είναι άνθρωποι, βιβλία μίας βιβλιοθήκης, έντυπα, τιμολόγια ή παραγγελίες, χάρτες, χρήματα σε λογαριασμούς τραπεζών ή κυκλώματα. Οποιοδήποτε και αν είναι το αντικείμενο, πρέπει να έχει μία φυσική αναπαράσταση, ακόμα και αν είναι μια εικόνα σε μια οθόνη υπολογιστή που δεν τυπώνεται ποτέ.

Τα αντικείμενα έχουν *ιδιότητες* και *συμπεριφορά*. Κατά αρχάς, αυτός ο συνδυασμός μπορεί να εμφανίζεται ότι είναι εφαρμόσιμος μόνο σε βάσεις δεδομένων ατόμων και όχι σε βιβλία ή σε λογαριασμούς τραπεζών. Όμως, όλα τα αντικείμενα βάσεων δεδομένων, εκτός από αυτά στις αρχειοθετημένες βάσεις δεδομένων, έχουν και ιδιότητες και συμπεριφορά.

### **3.4 Εξετάζοντας τις στατικές και δυναμικές ιδιότητες αντικειμένων**

Οι ιδιότητες ενός αντικειμένου προσδιορίζουν τα περιεχόμενα μιας βάσης δεδομένων ή πίνακα που περιέχει αναπαράσταση αντικειμένων του ίδιου τύπου. Τα βιβλία έχουν κωδικούς θεμάτων, που προέρχονται από το Dewey δεκαδικό σύστημα. Τα μοντέρνα βιβλία έχουν έναν κωδικό ISBN και τα περισσότερα έχουν τώρα και έναν αριθμό καταλόγου. Αυτοί οι αριθμοί είναι ιδιότητες ενός βιβλίου, όπως είναι και ο τίτλος, ο συγγραφέας, ο αριθμός των σελίδων. Αυτές οι ιδιότητες είναι στατικές: είναι οι ίδιες είτε το βιβλίο είναι σε βιβλιοθήκη ή τα έχει δανειστεί κάποιος. Οι πληροφορίες πελάτη για έναν λογαριασμό τραπεζής, όπως ο αριθμός λογαριασμού, το όνομα και η διεύθυνση είναι στατικές πληροφορίες, ακόμα και αν οι πελάτες μερικές φορές αλλάζουν διεύθυνση. Οι πληροφορίες κατάστασης δανεισμού των βιβλίων και το υπόλοιπο του λογαριασμού είναι δυναμικές ιδιότητες: αλλάζουν συνεχώς.

### **3.5 Περιγράφοντας τα δεδομένα και τις ιδιότητες τους**

Ένα αντικείμενο, όπως και όλες οι στατικές του ιδιότητες, ονομάζεται *οντότητα δεδομένων*. Κάθε ξεχωριστή οντότητα πρέπει να είναι μοναδική, ώστε να μπορείτε να την ξεχωρίσετε από άλλες. Ένας λογαριασμός τραπεζής πελάτη είναι μία οντότητα, για παράδειγμα, αλλά τα χρήματα στο λογαριασμό του πελάτη δεν προσδιορίζονται μοναδικά. Ένας

πελάτης μπορεί να έχει περισσότερους από έναν λογαριασμό, έτσι ο αριθμός ταυτότητας δεν αρκεί σαν μοναδικός προσδιορισμός. Ένας αριθμός λογαριασμού πρέπει να έχει αντιστοιχισθεί για να είναι σίγουρη η μοναδικότητα κάθε οντότητας πελάτη.

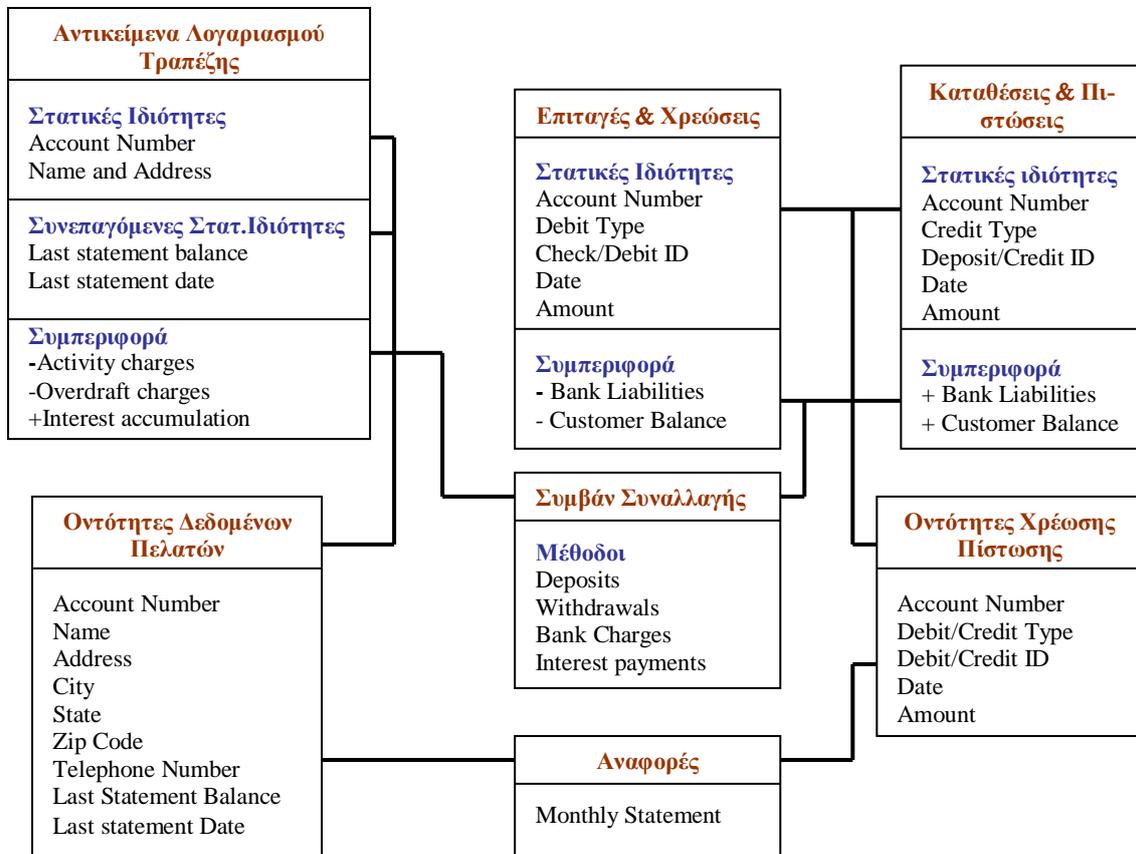
Το έντυπο κατάθεσης και οι επιταγές είναι αντικείμενα που αντιπροσωπεύονται στη βάση δεδομένων σαν άλλες οντότητες που σχετίζονται με την οντότητα πελάτη. Οι αριθμοί επιταγών δεν είναι αρκετά μοναδικοί για να ξεχωρίζουν σαν οντότητες. Πολλοί διαφορετικοί πελάτες μπορεί να έχουν μία επιταγή με αριθμό 1553. ο συνδυασμός του αριθμού του πελάτη και της επιταγής δεν είναι αρκετός σαν μοναδικός προσδιορισμός επειδή διαφορετικές τράπεζες μπορεί να χρησιμοποιούν τον ίδιο αριθμό πελάτη για να προσδιορίζουν διαφορετικά άτομα. Ένας αριθμός τραπεζής, ο αριθμός πελάτη και ο αριθμός επιταγής μαζί, μπορεί να προσδιορίζουν μοναδικά μία οντότητα πίστωσης. Κάθε επιταγή περιέχει αυτές τις πληροφορίες τυπωμένες με μαγνητικό μελάνι. Η ιδιότητα του ποσού κάθε χρέωσης (επιταγής) ή πίστωσης (κατάθεσης) χρησιμοποιείται για να αλλάξει το υπόλοιπο του λογαριασμού με απλή αφαίρεση και πρόσθεση, μία διαδικασία που ονομάζεται συναλλαγή.

Δεν θέλετε να περιμένετε μέχρι το ATM (αυτοματοποιημένο σύστημα της τραπεζής) υπολογίσει το υπόλοιπο σας, υπολογίζοντας κάθε συναλλαγή από τότε που ανοίξατε το λογαριασμό. Συνεπώς, μία στατική ιδιότητα, το τελευταίο υπόλοιπο, μπορεί να συμπεριληφθεί στην οντότητα των δεδομένων του πελάτη και να ενημερώνετε μία φορά τον μήνα. Μόνο το τελευταίο υπόλοιπο και οι σημερινές συναλλαγές χρειάζεται να υπολογιστούν για να υπολογιστεί το τρέχον υπόλοιπο - μία δυναμική ιδιότητα υπολογισμού. Στην Εικόνα 3.1, οι γραμμές συνδέουν στατικές ιδιότητες λογαριασμών τραπεζής με τις οντότητες δεδομένων που προέρχονται από αυτές. Οι ιδιότητες των αντικειμένων που περιλαμβάνονται στις οντότητες δεδομένων, όπως ο αριθμός λογαριασμού και το όνομα πελάτη, ονομάζονται *ιδιότητες*.

### **3.6 Λαμβάνοντας υπόψη τη συμπεριφορά αντικειμένων με μεθόδους**

Η συμπεριφορά σχετιζόμενων αντικειμένων βάσεων δεδομένων προσδιορίζει τα χαρακτηριστικά των συναλλαγών στις οποίες συμμετέχουν τα δεδομένα. Τα βιβλία μίας βιβλιοθήκης μπορούν να ζητηθούν, να δανειστούν, να επιστραφούν και να χαθούν. Η συμπεριφορά των λογαριασμών τραπεζής είναι πολύ εύκολο να περιγραφούν: ένας πελάτης ανοίγει λογαριασμό, έχει συναλλαγές κατάθεσης και τόκους που αυξάνουν το υπόλοιπο του και επιταγές, αναλήψεις και χρεώσεις που περιορίζουν το υπόλοιπο του. Η χρέωση ή πίστωση ενός

λογαριασμού είναι ένα παράδειγμα μίας συναλλαγής. Οι συναλλαγές συμβαίνουν σε απόκριση συμβάντων, όπως όταν κάνετε μία κατάθεση ή ανάληψη.



**Εικόνα 3.1** Σχέσεις μεταξύ αντικειμένων, οντοτήτων, συμβάντων και μεθόδων σε μία βάση δεδομένων τραπεζής.

Στις συμβατικές σχεσιακές βάσεις δεδομένων, μπορείτε να αναπαριστάνετε τα αντικείμενα και τις ιδιότητες σαν οντότητες, αλλά όχι με συμπεριφορά του πραγματικού κόσμου. Η συμπεριφορά ενός OLE αντικειμένου προσδιορίζεται από τις μεθόδους που είναι διαθέσιμες στον OLE διακομιστή που χρησιμοποιείτε για να δημιουργήσει το αντικείμενο ή να προσθέσει το αντικείμενο στο πεδίο OLE Object. Με συμβατικές οντότητες δεδομένων, προσομοιάζετε τη συμπεριφορά των αντικειμένων εφαρμόζοντας τις μεθόδους που χρησιμοποιείτε στις εφαρμογές σας.

### **3.7 Συνδυάζοντας διαφορετικές οντότητες σε ένα πίνακα**

Μπορείτε να συμπεριλάβετε ξεχωριστούς τύπους αντικειμένων σε ένα μόνο πίνακα εφόσον μπορείτε να αντιπροσωπεύσετε τις ιδιότητες και συμπεριφορά τους με τον ίδιο τρόπο και ακόμα να ξεχωρίσετε μεταξύ των διαφορετικών αντικειμένων. Για παράδειγμα, οι επιταγές και οι χρεώσεις εμφανίζονται σαν ένας τύπος αντικειμένου στην Εικόνα 3.1, αν και το ένα είναι ένα έντυπο χαρτί και το άλλο μία ηλεκτρονική χρέωση. Το πεδίο Debit Type μπορεί να δείξει τις διαφορετικές πηγές. Μπορείτε να συνδυάσετε καταθέσεις σε ρευστό και τις μεταφορές από ένα λογαριασμό Ταμειυτηρίου, σαν μία οντότητα ενός πίνακα Credits. Μπορείτε να συνδυάσετε χρεώσεις και πιστώσεις σε έναν πίνακα που μπορείτε να το κάνετε χρησιμοποιώντας διαφορετικούς κωδικούς για χρεώσεις και πιστώσεις. Για να προσδιορίσετε μοναδικά μία χρέωση ή πίστωση, πρέπει να συμπεριλάβετε πεδία για τον αριθμό τραπέζης, αριθμό πελάτη, τύπο χρέωσης / πίστωσης και τον αριθμό συναλλαγής. Αν και ένας αριθμός επιταγής μπορεί να εξυπηρετήσει σαν αριθμός συναλλαγής, το σύστημα πρέπει να δίνει αριθμούς συναλλαγής για άλλους τύπους συναλλαγών, όπως αυτές που γίνονται από τα ATM. Η Access 2000 μπορεί να χρησιμοποιήσει ένα πεδίο Auto Number, που ονομάζεται πεδίο Counter στην Access 2 και νωρίτερα, για να προσθέσει ένα μοναδικό αριθμό συναλλαγής (είτε με αύξηση είτε τυχαία) σε κάθε οντότητα δεδομένων, όπως και στις επιταγές. Ο αριθμός επιταγής γίνεται μία ξεχωριστή ιδιότητα.

### **3.8 Ορολογία βάσεων δεδομένων**

Τα αντικείμενα του πραγματικού κόσμου είναι η βασική πηγή πληροφοριών που αντιπροσωπεύονται σε μία βάση δεδομένων, σαν οντότητα. Η παρακάτω λίστα αρχίζει με μία οντότητα, χωρίζεται σε ξεχωριστά στοιχεία και μετά ορίζει τη θέση της στην ιεραρχία των βάσεων δεδομένων και πινάκων.

◆ **Οντότητα.** Μία μοναδική αναπαράσταση ενός μοναδικού αντικειμένου του πραγματικού κόσμου, που δημιουργείται χρησιμοποιώντας τις τιμές των ιδιοτήτων σε μορφή αναγνώσιμη από υπολογιστή. Για να είστε σίγουροι για τη μοναδικότητα, μία ή περισσότερες ιδιότητες της οντότητας πρέπει να έχουν τιμές διαφορετικές από τις αντίστοιχες τιμές οποιασδήποτε άλλης παρόμοιας οντότητας.

◆ **Ιδιότητα** μία σημαντική ιδιότητα ενός πραγματικού αντικειμένου. Κάθε ιδιότητα έχει μία τιμή που βοηθά στον προσδιορισμό της οντότητας για να την ξεχωρίσει από τα άλλα μέρη. Οι οντότητες περιέχονται σε πεδία πινάκων ή σε στήλες ερωτημάτων.

◆ **Τύπος δεδομένων ιδιότητας.** Οι βασικοί τύποι αποτελούνται από όλους τους αριθμητικούς τύπους δεδομένων (ακέραιους, πραγματικούς κλπ) και συμβολοσειρές (κείμενο ή αλφαριθμητικό), χωρίς ενσωματωμένα κενά ή στίξη. Οι συμβολοσειρές μπορεί να περιέχουν γράμματα, αριθμούς και ειδικούς χαρακτήρες.

Οι τύποι δεδομένων κειμένου με κενά ή άλλη στίξη ονομάζονται σύνθετοι τύποι δεδομένων. Μπορείτε να χωρίσετε τους περισσότερους τύπους δεδομένων σε βασικούς τύπους.

◆ **Σύνολο τιμών ιδιοτήτων.** Το επιτρεπτό εύρος τιμών για μία ιδιότητα. Ο τύπος δεδομένων της ιδιότητας προσδιορίζει το σύνολο τιμών, εκτός και αν η ιδιότητα είναι περιορισμένοι από μία διαδικασία που είναι εξωτερική στα δεδομένα του πίνακα. Σαν παράδειγμα του περιορισμού του πεδίου δεδομένων, το σύνολο δεδομένων της ιδιότητας της ηλικίας ενός εργαζομένου που έχει ακέραιο τύπο δεδομένων, μπορεί να παίρνει τα όρια από μία μέθοδο επικύρωσης δεδομένων, σε έναν ακέραιο μεγαλύτερο από 13 και μικρότερο από 90. Οι κανόνες επικύρωσης της Access, αποθηκεύονται σε πίνακες.

◆ **Τιμή ιδιότητας.** Η μικρότερη μονάδα δεδομένων μίας οντότητας. Οι τιμές είναι περιορισμένες σε αυτές του συνόλου δεδομένων της ιδιότητας.

◆ **Αναγνωριστικό.** Μία ιδιότητα ή συνδυασμός ιδιοτήτων που απαιτούνται για να προσδιορίσετε μία συγκεκριμένη οντότητα, μοναδικά. Τα αναγνωριστικά ονομάζονται πεδία πρωτεύοντος κλειδιού στην Access και χρησιμοποιούνται για να δημιουργήσουν πρωτεύοντα ευρετήρια οντοτήτων.

◆ **Πίνακας.** Η συλλογή όλων των οντοτήτων ενός τύπου δεδομένων, ονομάζεται τάξη οντότητας ή τύπος οντότητας.

◆ **Βάση δεδομένων.** Το σύνολο από πίνακες που αποθηκεύουν σχετιζόμενες οντότητες.

Μεγάλο μέρος της τυπικής ορολογίας χρησιμοποιείτε για να περιγράψει αντικείμενα δεδομένων σε σχεσιακές βάσεις δεδομένων, με μάλλον τεχνική ορολογία αντί για αφηρημένη. Πρέπει να καταλάβετε την έννοια αυτών των όρων όμως, όταν δημιουργείτε μοντέλα δεδομένων που αποτελούν τη βάση σας για τη βάση δεδομένων σας.

### **3.9 Τύποι πινάκων και κλειδιών σε σχεσιακές βάσεις δεδομένων**

Σε σχέση με τις σχεσιακές βάσεις δεδομένων, υπάρχουν συγκεκριμένοι τύποι πινάκων και πεδίων που ενεργοποιούν τις σχέσεις μεταξύ πινάκων. Η κατανόηση αυτών των πινάκων και των κλειδιών είναι απαραίτητη για να καταλάβετε τις σχεσιακές βάσεις δεδομένων και τους κανόνες κανονικοποίησης. Η παρακάτω λίστα ορίζει τα διάφορα σχεσιακά κλειδιά και πίνακες:

◆ **Βασικός πίνακας.** Σε μία σχεσιακή βάση δεδομένων, ένας βασικός πίνακας είναι ο πίνακας που έχει μία ή περισσότερες στήλες με τις ιδιότητες ενός αντικείμενου και περιέχει το πρωτεύον κλειδί που προσδιορίζει μοναδικά το αντικείμενο σαν οντότητα. Οι βασικοί πίνακες καλούνται συχνά πρωτεύοντες πίνακες, εξαιτίας της απαίτησης για ένα πρωτεύον κλειδί.

◆ **Σχεσιακός πίνακας.** Ένας πίνακας που χρησιμοποιείται για να δώσει συνδέσεις μεταξύ άλλων πινάκων και δεν είναι βασικός πίνακας (επειδή δεν έχει ιδιότητες ενός αντικείμενου ή ενός πρωτεύοντος κλειδιού), ονομάζεται σχεσιακός πίνακας. Τα πεδία κλειδιά των σχεσιακών πινάκων πρέπει να έχουν ξένα κλειδιά που να σχετίζονται με το πρωτεύον κλειδί του βασικού πίνακα.

Τεχνικά, ένας πραγματικά σχεσιακός πίνακας αποτελείται πλήρως από ξένα κλειδιά και δεν περιέχει ανεξάρτητα δεδομένα. Ο πίνακας Order Details της βάσης δεδομένων North wind Traders είναι ένα παράδειγμα ενός σχεσιακού πίνακα, που περιέχει τιμές που δεν είναι ξένα κλειδιά (τα πεδία Unit Price και Quantity, για παράδειγμα). Το πεδίο OrderID σχετίζεται με το πεδίο με το ίδιο όνομα στον πίνακα Orders. Παρόμοια, το πεδίο ProductID σχετίζεται με το πεδίο ProductID του πίνακα Products. Αν και ο πίνακας έχει ένα σύνθετο κλειδί, δεν είναι πραγματικά πρωτεύον κλειδί. Σκοπός του είναι να εμποδίσει διπλές εγγραφές για ένα προϊόν, σε μία συγκεκριμένη σειρά.

◆ **Πρωτεύον κλειδί.** Ένα πρωτεύον κλειδί αποτελείται από ένα σύνολο από τιμές που καθορίζουν μοναδικά μία γραμμή ενός βασικού πίνακα, που για την Access είναι ο πρωτεύον πίνακας. Για οποιοδήποτε πρωτεύον κλειδί, μία και μόνο μία γραμμή του πίνακα ταιριάζει με αυτή την τιμή. Μπορείτε να βασίζεστε το πρωτεύον κλειδί σε ένα μόνο πεδίο, εάν η τιμή κάθε κελιού είναι πάντα μοναδική.

◆ **Υποψήφια κλειδιά.** Οποιαδήποτε στήλη ή ομάδα από στήλες που ανταποκρίνονται στις απαιτήσεις για ένα πρωτεύον κλειδί είναι υποψήφια να γίνουν το πρωτεύον κλειδί του πίνακα. Ο αριθμός Name και Social Security είναι υποψήφια κλειδιά για τον για τον προσδιορισμό ενός ατόμου στις Η.Π.Α. Όμως, ο αριθμός Social Security είναι πιο κατάλληλη επιλογή, επειδή δυο άτομα μπορεί να έχουν το ίδιο όνομα αλλά όχι τον ίδιο αριθμό ταυτότητας.

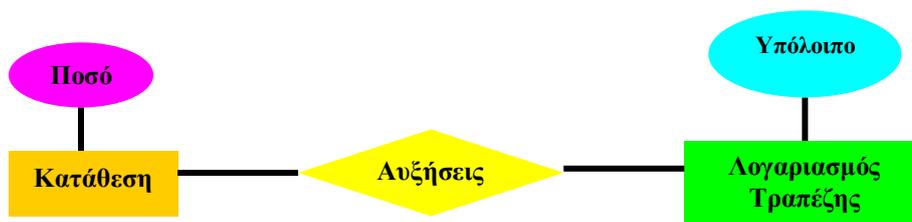
◆ **Σύνθετα κλειδιά.** Εάν χρειάζεστε δεδομένα από περισσότερες από μία στήλες του πίνακα για να ανταποκρίνονται στην απαίτηση της μοναδικότητας ενός πρωτεύοντος κλειδιού, το κλειδί λέγεται ότι είναι σύνθετο κλειδί.

◆ **Ξένα κλειδιά.** Ένα ξένο κλειδί είναι μία στήλη της οποίας οι τιμές αντιστοιχούν σε αυτές που περιέχονται σε ένα πρωτεύον κλειδί, ή το πιο αριστερό στοιχείο ενός σύνθετου κλειδιού, σε έναν άλλο σχετιζόμενο πίνακα. Ένα ξένο κλειδί μπορεί να αποτελείται από μία στήλη ή μία ομάδα από στήλες (ένα σύνθετο ξένο κλειδί). Εάν το μήκος ενός ξένου κλειδιού είναι μικρότερο από το αντίστοιχο πρωτεύον κλειδί, το κλειδί ονομάζεται μερικό ή αποκομμένο ξένο κλειδί.

Η σχεδίαση βάσεων δεδομένων θεμάτων περιλαμβάνουν τη δημιουργία ενός διαγράμματος των σχετικών αντικειμένων και των συσχετίσεων μεταξύ τους και τη δημιουργία μοντέλων για κάθε βάση δεδομένων.

### **3.9.1 Διαγράμματα μοντέλων δεδομένων**

Υπάρχουν πολλές μέθοδοι για τη δημιουργία διαγραμμάτων για την εμφάνιση μοντέλων δεδομένων. Μια από τις πιο χρήσιμες μεθόδους είναι το διάγραμμα οντότητας σχέσης (Entity-Relationships E-R) που αναπτύχθηκε από τον Peter Chen το 1976 (δείτε εικόνα 3.1.2).



Εικόνα 3.2 Ένα διάγραμμα E-R με δύο οντότητες δεδομένων από την εικόνα 3.1

Οι οντότητες δεδομένων περιλαμβάνονται μέσα σε ορθογώνια, οι ιδιότητες των δεδομένων μέσα σε ελλείψεις και οι σχέσεις μεταξύ των οντοτήτων μέσα σε ρόμβους. Οι σχέσεις μεταξύ αντικειμένων, στη θεωρητική βάση, μπορούν να οριστούν από την συμπεριφορά τους. Συνεπώς, τα E-R διαγράμματα περιλαμβάνουν τουλάχιστον ένα ρήμα του οποίου το αντικείμενο είναι στα δεξιά του συμβόλου του ρόμβου, εκτός και αν υποδεικνύεται διαφορετικά. Προσθέτετε τα σύμβολα στο διάγραμμα καθώς αυξάνονται οι λεπτομέρειες του μοντέλου. Ένα από τα πλεονεκτήματα του διαγράμματος E-R είναι ότι μπορείτε να το χρησιμοποιήσετε για να εμφανίσετε τη θεωρητική σχεδίαση πολύ μεγάλων συστημάτων, με πολλές βάσεις δεδομένων, σε σχετικά μικρό χώρο.

### 3.9.2 Κανονικοποιώντας δεδομένα στο σχεσιακό μοντέλο

Τα περισσότερα θέματα του κεφαλαίου αυτού εφαρμόζονται σε οποιοδήποτε είδος βάσης δεδομένων-ιεραρχική σχεσιακή ή ακόμα και τη νέα τάξη των συστημάτων βάσεων δεδομένων αντικειμένων. Όμως, επειδή η Access είναι RDBMS, η ισορροπία του κεφαλαίου είναι αφιερωμένη σε σχεσιακές βάσεις δεδομένων. Η Access χειρίζεται πλήρως το σχεσιακό μοντέλο στην εγγενή δομή της και μπορείτε να συνδέσετε πίνακες από άλλες RDBMS (όπως πίνακες πελατών διακομιστών), με τις βάσεις δεδομένων της Access.

Η θεωρία των σχεσιακών βάσεων δεδομένων έχει ξεκινήσει από ένα κλάδο των μαθηματικών που ονομάζεται *θεωρία συνόλων* και σχετίζεται πολύ με τη σχεδιαστική ανάλυση και μερική στατιστική μεθοδολογία. Το σύνολο των κανόνων και των συμβόλων με τις οποίες ορίζονται οι σχεσιακές βάσεις δεδομένων, ονομάζονται *σχεσιακή άλγεβρα*.

### **3.9.3 Κανόνες κανονικοποίησης**

Η κανονικοποίηση είναι μία τυποποιημένη προσέγγιση, κατά την οποία οι ιδιότητες των δεδομένων είναι ομαδοποιημένες σε πίνακες και οι πίνακες σε βάσεις δεδομένων. Ο σκοπός της κανονικοποίησης περιλαμβάνει τα παρακάτω:

- ❑ Απαλοιφή διπλών πληροφοριών σε πίνακες.
- ❑ Να αντιμετωπίζει μελλοντικές αλλαγές στη δομή των πινάκων.
- ❑ Να ελαχιστοποιεί την επίδραση των δομικών αλλαγών της βάσης δεδομένων σε εφαρμογές του χρήστη που έχουν πρόσβαση στα δεδομένα.

Η κανονικοποίηση γίνεται σε βήματα. Παρακάτω περιγράφονται καθένα από τα πέντε βήματα ολόκληρης της διαδικασίας κανονικοποίησης.

### **3.9.4 Πρώτη κανονική μορφή**

Η *πρώτη κανονική μορφή* απαιτεί οι πίνακες να είναι επίπεδοι και να μην περιέχουν επαναλαμβανόμενες ομάδες. Ένας επίπεδος πίνακας έχει μόνο δύο διαστάσεις – το μήκος (ο αριθμός των εγγραφών ή γραμμών) και το πλάτος (ο αριθμός των πεδίων ή στηλών) – και δεν μπορεί να περιέχει κελιά με περισσότερες από μία τιμές. Για να περιέχει ένα μόνο κελί περισσότερες από μία τιμές, η αναπαράσταση των περιεχομένων του κελιού απαιτεί μία τρίτη διάσταση, το βάθος, για να εμφανισθούν οι πολλαπλές τιμές. Οι επίπεδοι πίνακες και οι επίπεδες βάσεις δεδομένων είναι παρόμοια γιατί έχουν δυο διαστάσεις. Οι επίπεδες βάσεις δεδομένων όμως αποτελούνται μόνο από έναν πίνακα και δεν έχουν περιορισμούς στα περιεχόμενα των κελιών μέσα στον πίνακα.

### **3.9.5 Δεύτερη κανονική μορφή**

Η *δεύτερη κανονική μορφή* απαιτεί, τα δεδομένα σε όλες τις στήλες που δεν είναι κλειδιά, να είναι πλήρως εξαρτώμενα από το πρωτεύον κλειδί και από κάθε στοιχείο (στήλη) του πρωτεύοντος κλειδιού, όταν το πρωτεύον κλειδί είναι σύνθετο. Το πλήρως εξαρτώμενα σημαίνει ότι οι τιμές των δεδομένων σε κάθε στήλη που δεν είναι κλειδί μίας εγγραφής, προσδιορίζεται μοναδικά από την τιμή του πρωτεύοντος κλειδιού. Εάν απαιτείτε ένα σύνθετο

πρωτεύον κλειδί για να ορισθεί η μοναδικότητα μίας εγγραφής, ο ίδιος κανόνας εφαρμόζεται σε κάθε τιμή των πεδίων που αποτελούν το σύνθετο κλειδί της εγγραφής. Ο πίνακας σας πρέπει να είναι στην πρώτη κανονική μορφή πριν τον εξετάσετε για την δεύτερη κανονική μορφή. Η δεύτερη κανονική μορφή αφαιρεί πολλές από τις επαναλήψεις δεδομένων που είναι πιθανό να συμβούν στην πρώτη κανονική μορφή.

### **3.9.6 Τρίτη κανονική μορφή**

Η τρίτη κανονική μορφή απαιτεί όλες οι στήλες χωρίς κλειδιά ενός πίνακα να εξαρτώνται από το πρωτεύον κλειδί του πίνακα και να είναι ανεξάρτητες μεταξύ τους. Οι πίνακες πρέπει να ανταποκρίνονται και στην πρώτη και τη δεύτερη κανονική μορφή για να μπορούν να έρθει στην τρίτη κανονική μορφή.

### **3.9.7 Τέταρτη κανονική μορφή**

Η τέταρτη κανονική μορφή απαιτεί ότι δε θα αποθηκεύονται ανεξάρτητα δεδομένα στον ίδιο πίνακα όταν υπάρχουν σχέσεις πολλά-προς-πολλά μεταξύ αυτών των οντοτήτων.

### **3.9.8 Πέμπτη κανονική μορφή και συνδυασμένες οντότητες**

Η πέμπτη κανονική μορφή απαιτεί να ξανακατασκευάσετε ακριβώς τον αρχικό πίνακα από τους πίνακες στους οποίους αναλύθηκε. Η πέμπτη κανονική μορφή απαιτεί οι πίνακες να ανταποκρίνονται στους κανόνες για την τρίτη κανονική μορφή και, όταν υπάρχουν σχέσεις πολλά-προς-πολλά, με τον κανόνα της τέταρτης κανονικής μορφής.

### **3.9.10 Τύποι σχέσεων**

Το θέμα των σχέσεων μεταξύ οντοτήτων συνήθως προϋπάρχει στις συζητήσεις κανονικοποίησης. Οι σχέσεις εδώ έρχονται δεύτερες, επειδή μπορούμε να δημιουργήσουμε έγκυρες σχέσεις μόνο μεταξύ πινάκων που έχουν δομηθεί τουλάχιστον σε σχέση με τις τρεις πρώτες κανονικές μορφές. Στη συνέχεια περιγράφουμε τους τέσσερις βασικούς τύπους σχέσεων μεταξύ πινάκων και χρησιμοποιούμε διαγράμματα E-R για να δείξουμε γραφικά τις σχέσεις.

### 3.9.10.1 Σχέσεις Ένα - Προς - Ένα

Η απλούστερη σχέση μεταξύ πινάκων είναι μια σχέση ένα-προς-ένα. Σε μία τέτοια σχέση, οι πίνακες έχουν ακριβώς αντιστοιχία μία-προς-μία γραμμή. Καμία γραμμή του πίνακα δεν έχει περισσότερες από μία αντίστοιχες γραμμές στους άλλους πίνακες. Μπορείτε να συνδυάσετε πίνακες ένα-προς-ένα σε ένα μόνο πίνακα, που να αποτελείται από όλες τις στήλες των πινάκων.

Οι σχέσεις ένα-προς-ένα μερικές φορές χρησιμοποιούνται για να χωρίσουν πολύ μεγάλους βασικούς πίνακες σε μικρότερους. Μπορείτε να χωρίσετε έναν πλατύ πίνακα για να περιορίσετε την ώρα που χρειάζεστε για να δείτε τα πεδία που περιέχουν συγκεκριμένα σύνολα δεδομένων. Μερικές φορές, μπορεί να ελέγχετε την πρόσβαση στα μέρη των πινάκων που περιέχουν ευαίσθητα ή εμπιστευτικά δεδομένα. Ένα παράδειγμα είναι ένας πίνακας εργαζομένων: όλοι μπορεί να έχουν πρόσβαση "ανάγνωσης-μόνο" στα ονόματα των εργαζομένων, αλλά μόνο τα μέλη του προσωπικού μπορούν να δουν τις πληροφορίες μισθού (δείτε εικόνα 3.3)

Employee	Position	Last	First	Employee	Salary
00668	Master	Johansson	Lars	00668	6500.00
00721	Master	Karlsson	Bo	00721	6250.00
00885	Chief	Macgregor	Paul	00885	5100.00
00912	Chief	McDermott	John	00912	5000.00
00955	Master	Olafson	Karl	00955	6100.00
01023	Master	Kekkonen	Elna	01023	6050.00
01156	Chief	McDougal	William	01156	4900.00
01203	Chief	Kashihara	Matsuo	01203	4850.00

Εικόνα 3.3 Δύο πίνακες με μία σχέση ένα – προς –ένα.

Η παρακάτω εικόνα 3.4 δείχνει το διάγραμμα E-R για τους πίνακες Employees και Salaries. Ο αριθμός 1 Έχει προστεθεί σε κάθε πλευρά της σχέσης με ένα ρόμβο, που δείχνει μια σχέση ένα-προς-ένα. Η συμμετοχή των οντοτήτων στις σχέσεις μπορεί να είναι απαραίτητη ή προαιρετική. Οι προαιρετικές σχέσεις συμβολίζονται από έναν κύκλο που έχει σχεδια-

σθεί στη γραμμή που συνδέει το προαιρετικό στοιχείο με το ρόμβο. Στην εικόνα, η σχέση Paid και Salaries είναι προαιρετική, επειδή μερικοί εργαζόμενοι μπορεί να πληρωθούν σε ωριαία βάση και είναι συνδεδεμένοι με τον πίνακα Wage (μισθοί). Οι πίνακες με απαιτούμενες σχέσεις ένα-προς-ένα είναι βασικοί πίνακες. Ένας πίνακας με μία προαιρετική σχέση ένα-προς-ένα με ένα βασικό πίνακα είναι ένας σχετιζόμενος πίνακας. Πολλαπλοί πίνακες, με σχέσεις ένα-προς-ένα, στις οποίες οι αντίστοιχες εγγραφές στους άλλους πίνακες είναι προαιρετικές, μπορούν να περιορίσουν το χώρο που καταλαμβάνουν στο δίσκο.



Εικόνα 3.4 Ένα διάγραμμα E-R για μία προαιρετική σχέση ένα – προς – ένα

### 3.9.10.2 Σχέσεις Ένα – Προς – Πολλά

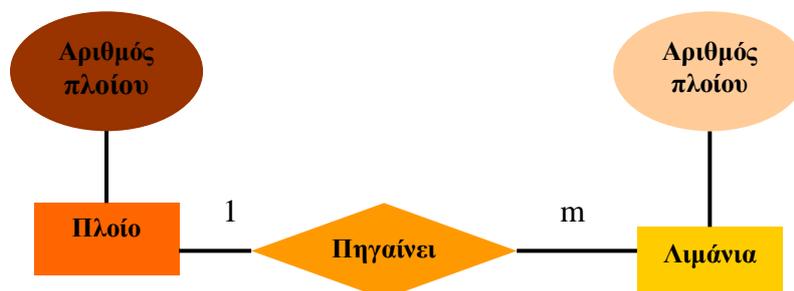
Οι σχέσεις ένα – προς – πολλά συνδέουν μία γραμμή ενός πίνακα με πολλές γραμμές ενός άλλου πίνακα, μέσω μίας σχέσης μεταξύ του πρωτεύοντος κλειδιού του βασικού πίνακα και του αντίστοιχου ξένου κλειδιού στο σχετιζόμενο πίνακα. Αν και το ξένο κλειδί του πίνακα που περιέχει τις σχέσεις "πολλά" μπορεί να είναι στοιχείο ενός σύνθετου πρωτεύοντος κλειδιού στο δικό του πίνακα, είναι ξένο κλειδί για το σκοπό της σχέσης. Οι σχέσεις ένα-προς-πολλά είναι ο πιο κοινός τύπος σχέσης

Η σχέση ένα-προς-πολλά, που φαίνεται στην παρακάτω εικόνα 3.5, συνδέει όλες τις εγγραφές στον πίνακα Ports με μία εγγραφή στον πίνακα Vessels. Η σχέση Ένα-προς-πολλά σας επιτρέπει να εμφανίζετε όλες τις εγγραφές στον πίνακα Ports για τα προγραμματισμένα λιμάνια στο Japan Bear.

Vessel parts	Vessel Name	Vessel	Voyage	Port	Type	Arrives	De-
528	Japan Bear	528	9203W	HNL	S	6/6/1998	6/8/1996
		528	9204W	HNL	S	8/27/1998	8/29/1998
		528	9203W	OSA	S	7/15/1998	7/18/1998
		528	9204W	OSA	S	9/30/1998	10/2/1998
		528	9203W	SFO	E		5/31/1998
		528	9204W	SFO	E		8/20/1998

Εικόνα 3.5 Μία σχέση ένα-προς-πολλά μεταξύ των πινάκων Vessel και Ports

Το διάγραμμα E-R στην εικόνα 3.6 εκφράζει αυτή τη σχέση. Ο βαθμός των σχέσεων της οντότητας Vessel μεταξύ των δυο πινάκων υποδεικνύεται από το "1" και το "m" δίπλα στις οντότητες. Το παράθυρο Relationships της Access χρησιμοποιεί το σύμβολο του απείρου (μ) για να εμφανίσει την πλευρά "πολλά" της σχέσης.



Εικόνα 3.6 Το E-R διάγραμμα για τη σχέση ένα-προς-πολλά της εικόνας 3.5

### 3.9.10.3 Σχέσεις Πολλά – Προς – Ένα

Οι σχέσεις πολλά-προς-ένα είναι το αντίθετο του ένα-προς-πολλά. Με τη σχέση πολλά-προς-ένα εμφανίζετε το όνομα του πλοίου για οποιαδήποτε εγγραφή στον πίνακα Ports. Εάν ο ρόλος των οντοτήτων που συμμετέχουν είναι απλώς αντίστροφος της σχέσης πολλά-προς-ένα, η σχέση λέμε ότι είναι αντανakλαστική, δηλαδή, η σχέση πολλά-προς-ένα είναι η αντανakλαση της αντίστοιχης σχέσης Ένα-προς-πολλά (δείτε εικόνα 3.7). Όλες οι σχέσεις πολλά-προς-ένα της Access είναι αντανakλαστικές. Μπορείτε να καθορίσετε μόνο μία σχέση

ένα-προς-ένα ή ένα-προς-πολλά μεταξύ του πρωτεύοντος πίνακα και του σχετιζόμενου πίνακα, χρησιμοποιώντας τα δύο κουμπιά επιλογής στο παράθυρο διαλόγου Relationships της Access.

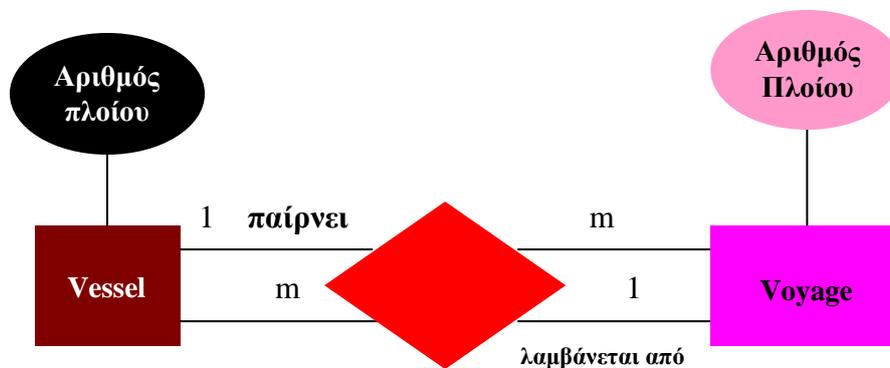
Vessel	Voyage	Port	Type	Arrives	Departs
528	9203W	HNL	S	6/6/1998	6/8/1998
528	9204W	HNL	S	8/27/1998	8/29/1998
528	9203W	OSA	S	7/15/1998	7/18/1998
528	9204W	OSA	S	9/30/1998	10/2/1998
528	9203W	SFO	E		5/31/1998
528	9204W	SFO	E		8/20/1998

Vessel	Vessel Name
528	Japan Bear

Εικόνα 3.7 Οι πίνακες Ports και Vessels σε μία ανακλαστική σχέση πολλά-προς-ένα.

Εάν επιλέξετε μία εγγραφή από την πλευρά πολλά της σχέσης, μπορείτε να εμφανίσετε την εγγραφή που αντιστοιχεί στο ξένο κλειδί στην πλευρά 1. Τα διαγράμματα E-R για αντανακλαστικές σχέσεις, σχεδιάζονται συχνά όπως το διάγραμμα της εικόνας 3.8. Οι αντανακλαστικές σχέσεις υποδεικνύονται από την κατάλληλη μορφή του ρήματος που μπαίνει έξω από το ρόμβο που ορίζει τη σχέση.



Εικόνα 3.8 Το E-R διάγραμμα για μία αντανακλαστική σχέση πολλά-προς-ένα.

### 3.9.10.4 Σχέσεις Πολλά – Προς – Πολλά και η τέταρτη κανονική μορφή.

Ο Πίνακας Crews στην παρακάτω εικόνα 3.9 δημιουργεί μία σχέση Πολλά-Προς-Πολλά μεταξύ του πίνακα Vessels (βασισμένου στην οντότητα Vessel) και του πίνακα Employees (βασισμένου στον αριθμό εργαζομένων στα πεδία Master, Chief και 1stMate).

Vessel	Voyage	Port	To	Master	Chief	1stMate
528	9203W	SFO	HNL	01023	01156	01367
528	9203W	HNL	HNL	01023	01156	01367
528	9203W	HNL	OSA	01023	01156	01367
528	9203W	OSA	OSA	01023	01156	01367
528	9203W	OSA	INC	01023	01156	01367

*Εικόνα 3.9* Η πρώτη έκδοση του πίνακα Crews

Η παραπάνω έκδοση του πίνακα Crews δημιουργεί μία σχέση πολλά-προς-πολλά μεταξύ των πινάκων Vessels και Employees.

Οι εργαζόμενοι που είναι πλήρωμα στο πλοίο είναι ανεξάρτητοι μεταξύ τους, οποιοσδήποτε έχει τα προσόντα μπορεί θεωρητικά να πάρει μία θέση σε οποιοδήποτε ταξίδι.

Η παρακάτω Εικόνα 3.10 δείχνει τον ξαναδομημένο πίνακα Crews που είναι απαραίτητος για να αντιστοιχίσετε τους εργαζόμενους με τα ταξίδια. Ο πίνακας έχει μία εγγραφή για κάθε εργαζόμενο, για κάθε ταξίδι.

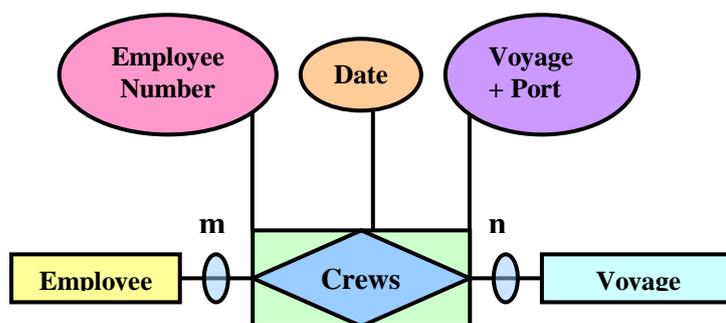
<b>Employee</b>	<b>Vessel</b>	<b>Voyage</b>	<b>Port</b>	<b>To</b>
01023	528	9203W	SFO	HNL
01156	528	9203W	SFO	HNL
01367	528	9203W	SFO	HNL
01023	528	9203W	HNL	HNL
01156	528	9203W	HNL	HNL
01367	528	9203W	HNL	HNL
01023	528	9203W	HNL	OSA
01156	528	9203W	HNL	OSA
01367	528	9203W	HNL	OSA
01023	528	9203W	OSA	OSA
01156	528	9203W	OSA	OSA
01367	528	9203W	OSA	OSA
01023	528	9203W	OSA	INC
01156	528	9203W	OSA	INC
01367	528	9203W	OSA	INC

*Εικόνα 3.10 Ο πίνακας Crews αναδομημένος σε τέταρτη κανονική μορφή*

Μπορούμε να προσθέσουμε νέες οντότητες σε αυτό τον πίνακα, εφόσον οι οντότητες είναι πλήρως εξαρτώμενες από το ξένο κλειδί. Ένα παράδειγμα μίας εξαρτώμενης οντότητας είναι τα δεδομένα μισθοδοσίας που μπορεί να περιλαμβάνουν ιδιότητες, όπως τις ώρες εργασίας, τις ώρες υπερωριών και τα έξοδα που έχουν γίνει από κάθε εργαζόμενο σε κάθε ταξίδι.

Τέτοιες οντότητες ονομάζονται *ασθενείς* ή *σχετικές οντότητες*, επειδή βασίζονται σε άλλους βασικούς πίνακες. Ο πίνακας Crews δε θεωρείται πλέον αυστηρά ένας σχεσιακός πίνακας όταν προσθέτετε σχετικές οντότητες, επειδή δεν αποτελείται πλήρως από πεδία που αποτελούν ξένα κλειδιά.

Το E-R διάγραμμα για το σχεσιακό πίνακα πολλά-προς-πολλά, που συσχετίζει τους εργαζομένους και το μέρος του ταξιδιού που θα ταξιδέψουν, φαίνεται στην παρακάτω εικόνα 3.11. Το Date που είναι μέσα σε κύκλο και συνδέεται με τη σχέση Assigned Crew, εκφράζει την τάξη της σχέσης: ένας εργαζόμενος μπορεί να αντιστοιχηθεί μόνο σε ένα ταξίδι μίας ημερομηνίας. Η τάξη της σχέσης συνεπώς, βασίζεται στις ημερομηνίες αναχώρησης και άφιξης. Η αυτόματη ενεργοποίηση της συνθήκης ότι οι εργαζόμενοι δε χρειάζεται πλέον να είναι σε περισσότερα από ένα μέρη ταυτόχρονα, μπορεί να επιτευχθεί δημιουργώντας ένα ευρετήριο χωρίς διπλές εγγραφές, που να αποτελείται από όλα τα πεδία του πίνακα Crews. Οι σχετικές οντότητες φαίνονται στα διαγράμματα E-R σαν σχέση ρόμβου μέσα σε ένα ορθογώνιο. Εάν προσθέσετε δεδομένα μισθοδοσίας στον πίνακα Crews, δημιουργείται μία σχετική οντότητα. Η αντιστοίχιση ενός εργαζομένου σε ένα ταξίδι είναι προαιρετική, όπως φαίνεται από τις κυκλικές γραμμές. Οι εργαζόμενοι μπορεί να έχουν άδεια στη στεριά, μπορεί να μην είναι διαθέσιμοι, ή μπορεί να έχουν καθήκοντα στην ξηρά.



Εικόνα 3.11. Ένα E-R διάγραμμα για μία σχέση πολλά-προς-πολλά με μία σχετική οντότητα.

Χρησιμοποιώντας γραφικά σχήματα και E-R διαγράμματα όταν σχεδιάζουμε μία Access βάση δεδομένων, μας βοηθούν να καταλάβουμε ότι η βάση δεδομένων ανταποκρίνεται στον αρχικό μας στόχο. Το σχήμα, μας είναι επίσης χρήσιμο για να εξηγήσουμε τη δομή της βάσης δεδομένων στους χρήστες μας. Τα E-R διαγράμματα μπορούν να αποκαλύψουν λάθη σχεδίασης, όπως την αποτυχία να κανονικοποιούνται πίνακες, τουλάχιστον σε τέταρτη κανο-

νική μορφή. Έτσι, η σχεδίαση και τα διαγράμματα είναι ένας τρόπος ώστε να παρατηρούμε την επιτυχία σχεδίασης της βάσης δεδομένων.

## **Η MICROSOFT ACCESS ΣΑΝ ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΣΧΕΣΙΑΚΩΝ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ**

Η Microsoft Access είναι ένα πλήρες λειτουργικό σύστημα διαχείρισης σχεσιακών βάσεων δεδομένων (RDBMS). Μας εφοδιάζει με όλες τις δυνατότητες ορισμού, χειρισμού, και ελέγχου των δεδομένων που χρειαζόμαστε για τη διαχείριση μεγάλων ποσοτήτων πληροφοριών.

### **Κύριες λειτουργίες μιας βάσης δεδομένων**

☀ **Ορισμός δεδομένων (Data definition)** – Μπορούμε να ορίζουμε τι δεδομένα θα αποθηκεύονται στη βάση δεδομένων μας, τον τύπο τους (για παράδειγμα, Αριθμοί ή χαρακτήρες), και τον τρόπο με τον οποίο θα συσχετίζονται μεταξύ τους. Σε μερικές περιπτώσεις, μπορούμε επίσης να ορίζουμε τον τρόπο μορφοποίησης των δεδομένων και τον τρόπο ελέγχου της εγκυρότητάς τους.

☀ **Χειρισμός δεδομένων (Data manipulation)** – Μπορούμε να επεξεργαζόμαστε τα δεδομένα με πολλούς τρόπους. Μπορούμε να επιλέγουμε ποια πεδία θέλουμε, να φιλτράρουμε και να ταξινομούμε τα δεδομένα. Μπορούμε επίσης να ενώνουμε δεδομένα με άλλες συσχετισμένες πληροφορίες και να συνοψίζουμε τα δεδομένα (να παίρνουμε σύνολα και άλλα στατιστικά στοιχεία). Επίσης, μπορούμε να επιλέξουμε μια ομάδα πληροφοριών και να ζητήσουμε από το RDBMS να τις ενημερώσει, να τις διαγράψει, να τις αντιγράψει σε άλλο πίνακα, ή να δημιουργήσει με αυτές έναν άλλο πίνακα.

☀ **Έλεγχος δεδομένων (Data control)** – Μπορούμε να ορίζουμε σε ποιον θα επιτρέπεται η ανάγνωση, η ενημέρωση, ή η εισαγωγή των δεδομένων. Σε πολλές περιπτώσεις, μπορούμε επίσης να ορίζουμε τον τρόπο με τον οποίο θα μοιράζονται και θα ενημερώνονται τα δεδομένα από πολλούς διαφορετικούς χρήστες ταυτόχρονα.

## 4.1 Ορισμός Δεδομένων & Αποθήκευση

Γενικά, όταν δουλεύουμε με κάποιο έγγραφο ή λογιστικό φύλλο, έχουμε πλήρη ελευθερία στον ορισμό των περιεχομένων του εγγράφου ή του κάθε κελιού. Μέσα σε μία σελίδα του εγγράφου, μπορούμε να περιλαμβάνουμε παραγράφους κειμένου, πίνακες, διαγράμματα, ή πολλές διαφορετικές στήλες δεδομένων που θα εμφανίζονται με διάφορες γραμματοσειρές. Μέσα σε μια στήλη ενός λογιστικού φύλλου μπορούμε να έχουμε κείμενο στην κορυφή για να ορίσουμε τις κεφαλίδες των στηλών για την εκτύπωση ή την εμφάνιση, και μέσα στη στήλη μπορούμε να έχουμε διάφορες μορφές αριθμών, ανάλογα με το σκοπό της γραμμής. Χρειαζόμαστε αυτήν την ευελιξία επειδή το έγγραφο του επεξεργαστή κειμένου πρέπει να μπορεί να αποδώσει αυτό που θέλουμε μέσα σε μία τυπωμένη σελίδα, και το λογιστικό μας φύλλο πρέπει να αποθηκεύει τα δεδομένα που αναλύουμε, καθώς και να φροντίζει για τον υπολογισμό και την παρουσίαση του επιθυμητού αποτελέσματος.

Η ευελιξία αυτή είναι πολύ καλή για τη λύση των σχετικά μικρών, και καλά ορισμένων, προβλημάτων μιας επιχείρησης. Ωστόσο, ένα λογιστικό φύλλο γίνεται δύσκολο στη διαχείρισή του όταν περιέχει περισσότερες από μερικές εκατοντάδες γραμμές στοιχείων, και τα έγγραφα γίνονται δύσχρηστα όταν έχουν περισσότερες από μερικές δεκάδες σελίδες. Καθώς η ποσότητα των δεδομένων μεγαλώνει, μπορεί επίσης να ανακαλύψουμε ότι ξεπεράσαμε το όριο αποθήκευσης δεδομένων του λογιστικού μας φύλλου ή του επεξεργαστή κειμένου, ή ακόμα και του υπολογιστικού μας συστήματος. Αν σχεδιάζουμε ένα έγγραφο ή λογιστικό φύλλο που θα χρησιμοποιηθεί από άλλους, είναι δύσκολο (αν όχι αδύνατο) να ελέγξουμε πως θα χρησιμοποιήσουν τα δεδομένα ή πως θα καταχωρίσουν καινούργια. Για παράδειγμα, σε ένα λογιστικό φύλλο ο χρήστης μπορεί εύκολα να εισαγάγει κατά λάθος χαρακτήρες σε ένα κελί που προσδιορίζεται για ημερομηνία ή χρηματική τιμή.

Κάποια προγράμματα λογιστικών φύλλων μας επιτρέπουν να ορίσουμε μία περιοχή "βάσης δεδομένων" μέσα στο φύλλο μας, για να μας βοηθήσουν στη διαχείριση των στοιχείων που χρειαζόμαστε, έτσι ώστε να πάρουμε το αποτέλεσμα που θέλουμε. Ωστόσο εξακολουθούν να ισχύουν οι βασικοί περιορισμοί αποθήκευσης του λογιστικού μας φύλλου, και εξακολουθούμε να μην έχουμε αρκετό έλεγχο σε αυτά που εισαγάγουμε στις γραμμές και τις στήλες της περιοχής της " βάσης δεδομένων". Επίσης, αν πρέπει να χειριστούμε κάτι περισσότερο από αριθμούς και χαρακτήρες, μπορεί να ανακαλύψουμε ότι το λογιστικό μας φύλλο δεν "καταλαβαίνει" κάποια αντικείμενα όπως είναι οι εικόνες και οι ήχοι.

Ένα RDBMS (Σύστημα Διαχείρισης Σχεσιακής Βάσης Δεδομένων), μας επιτρέπει να ορίζουμε το είδος των δεδομένων μας και τον τρόπο με τον οποίο θέλετε να αποθηκεύονται. Επίσης, συνήθως μπορούμε να ορίζουμε κανόνες, τους οποίους θα χρησιμοποιεί το RDBMS για να εξασφαλίζεται η ακεραιότητα των δεδομένων. Στην απλούστερη μορφή του, ένας *κανόνας εγκυρότητας* (validation rule) μπορεί να απαγορεύει την καταχώριση αλφαβητικών χαρακτήρων σε ένα πεδίο που θα έπρεπε να περιέχει αριθμούς. Άλλοι κανόνες μπορεί να ορίζουν τις επιτρεπτές τιμές ή τις περιοχές τιμών που μπορούν να πάρουν τα δεδομένα μας. Σε πιο εξελιγμένα συστήματα, μπορούμε να ορίζουμε τις σχέσεις μεταξύ κάποιων συλλογών δεδομένων (που συνήθως ονομάζονται πίνακες ή αρχεία) και να ζητάμε από το RDBMS να εξασφαλίζει ότι τα δεδομένα μας παραμένουν σωστά. Για παράδειγμα, θα μπορούσαμε να ζητήσουμε από το σύστημα να ελέγχει αυτόματα αν κάθε παραγγελία αντιστοιχεί σε κάποιον έγκυρο πελάτη.

Με την Access, έχουμε πλήρη ελευθερία στον ορισμό των δεδομένων μας (σαν κείμενο, αριθμούς, ημερομηνίες, ώρες, χρηματικές τιμές, εικόνες, ήχους, έγγραφα, ή λογιστικά φύλλα) σε ότι αφορά τον ορισμό του τρόπου αποθήκευσής τους (μήκος αλφαριθμητικών, ακρίβεια αριθμών, μορφή ημερομηνιών/ωρών) και τον ορισμό της μορφής τους όταν τα εμφανίζουμε ή τα τυπώνουμε. Μπορούμε να ορίζουμε απλούς ή σύνθετους κανόνες εγκυρότητας για να είμαστε βέβαιοι ότι στη βάση δεδομένων μας υπάρχουν μόνο ορθές τιμές. Μπορούμε επίσης να ζητάμε από την Access να ελέγχει τη βάση δεδομένων μας για έγκυρα είδη σχέσεων μεταξύ των αρχείων ή των πινάκων.

Επειδή η Access είναι η τελευταία λέξη της τεχνολογίας στις εφαρμογές για Microsoft windows, μπορούμε να χρησιμοποιήσουμε όλες τις λειτουργίες της *Δυναμικής Ανταλλαγής Δεδομένων*, ΔΑΔ (Dynamic Data Exchange, DDE) και της *Σύνδεσης και Ενσωμάτωσης Αντικειμένων*, ΣΕΑ (Object Linking and Embedding, OLE). Η ΔΑΔ μας επιτρέπει να εκτελούμε λειτουργίες και να ανταλλάσσουμε δεδομένα μεταξύ της Access και άλλων εφαρμογών για Windows που υποστηρίζουν τη ΔΑΔ. Μπορούμε επίσης να κάνουμε συνδέσεις ΔΑΔ με άλλες εφαρμογές χρησιμοποιώντας μακροεντολές ή τη Microsoft Visual Basic for Applications (VBA). Η ΣΕΑ είναι μια εξελιγμένη δυνατότητα των Windows, που μας επιτρέπει να συνδέουμε ή να ενσωματώνουμε αντικείμενα στη βάση δεδομένων της Access. Τα αντικείμενα μπορεί να είναι εικόνες, γραφήματα, λογιστικά φύλλα, ή έγγραφα από άλλες εφαρμογές για Windows, που και αυτές υποστηρίζουν τη ΣΕΑ.

Η Access μπορεί επίσης να καταλαβαίνει και να χρησιμοποιεί μία μεγάλη ποικιλία άλλων μορφών δεδομένων, όπως είναι οι δομές αρχείων άλλων συστημάτων διαχείρισης βάσεων δεδομένων. Μπορούμε να εισάγουμε και να εξάγουμε δεδομένα από αρχεία επεξεργαστών κειμένου ή λογιστικά φύλλα, καθώς και να προσπελάζουμε απευθείας και να ενημερώνουμε αρχεία του Paradox, των dBASE III και dBASE IV, Του Btrieve, της FoxPro, και άλλων εφαρμογών. Μπορούμε επίσης να εισάγουμε δεδομένα από αυτά τα αρχεία σε έναν πίνακα της Access. Επιπλέον, η Access μπορεί να επεξεργαστεί αρχεία των δημοφιλέστερων βάσεων δεδομένων που υποστηρίζουν το πρότυπο της *Ανοιχτής Συνδεσιμότητας Βάσεων Δεδομένων*, ΑΣΒΔ (Open DataBase Connectivity, ODBC), Όπως των Microsoft SQL Server, Oracle, DB2, και Rdb.

## **4.2 Χειρισμός Δεδομένων**

Η επεξεργασία των δεδομένων σε ένα πρόγραμμα επεξεργασίας κειμένου ή σε κάποιο λογιστικό φύλλο, διαφέρει από αυτήν μιας βάσης δεδομένων. Σε ένα έγγραφο επεξεργαστή κειμένου, μπορούμε να έχουμε στοιχεία με μορφή πίνακα και να εφαρμόζουμε σε αυτά ένα περιορισμένο σύνολο λειτουργιών. Επίσης, μπορούμε να αναζητάμε αλφαριθμητικά στο πρωτότυπο έγγραφο και, με τη βοήθεια της Τεχνικής Σύνδεσης και Ενσωμάτωσης Αντικειμένων, να συμπεριλαμβανετε σε αυτό πίνακες, διαγράμματα, ή εικόνες από άλλες εφαρμογές. Στο λογιστικό φύλλο, μερικά κελιά περιέχουν συναρτήσεις που καθορίζουν το επιθυμητό αποτέλεσμα, ενώ εμείς εισάγουμε τα δεδομένα σε άλλα κελιά, τα οποία παρέχουν τις πληροφορίες προέλευσης για τις συναρτήσεις. Τα δεδομένα ενός λογιστικού φύλλου εξυπηρετούν ένα συγκεκριμένο σκοπό, και είναι δύσκολο να χρησιμοποιήσουμε τα ίδια δεδομένα για να αντιμετωπίσουμε ένα διαφορετικό πρόβλημα. Για να το πετύχουμε αυτό, μπορούμε να συνδεθούμε με τα δεδομένα ενός άλλου λογιστικού φύλλου ή να χρησιμοποιήσουμε τις περιορισμένες δυνατότητες αναζήτησης για να αντιγράψουμε ένα επιλεγμένο υποσύνολο των δεδομένων σε ένα άλλο λογιστικό φύλλο, το οποίο και θα χρησιμοποιήσουμε στη συνέχεια για την επίλυση του νέου προβλήματος.

Ένα RDBMS μας παρέχει πολλούς τρόπους εργασίας με τα δεδομένα μας. Μπορούμε, για παράδειγμα, να ψάξουμε σε ένα μεμονωμένο πίνακα για πληροφορίες, ή να εκτελέσουμε μια πολύπλοκη έρευνα μέσα σε πολλούς συσχετισμένους πίνακες ή αρχεία. Μπορούμε να ενημερώνουμε ένα μεμονωμένο πεδίο ή πολλές εγγραφές με μία διαταγή, και να γράφουμε

προγράμματα που θα χρησιμοποιούν τις λειτουργίες του RDBMS για την ανάγνωση και την ενημέρωση των δεδομένων μας. Πολλά συστήματα μας εφοδιάζουν με λειτουργίες καταχώρισης δεδομένων και γεννήτριας αναφορών.

Η Access χρησιμοποιεί την ισχυρή γλώσσα βάσεων δεδομένων SQL (Structured Query Language, Δομημένη Γλώσσα Ερωτημάτων) για την επεξεργασία των δεδομένων που έχουμε αποθηκεύσει στους πίνακες μας. Με την SQL, μπορούμε να ορίζουμε το σύνολο των πληροφοριών που χρειαζόμαστε για την επίλυση ενός συγκεκριμένου προβλήματος, όπως είναι τα δεδομένα από πολλούς διαφορετικούς πίνακες. Όμως, η Access απλοποιεί τις εργασίες χειρισμού των δεδομένων. Για να τη χρησιμοποιήσουμε δε χρειάζεται καν να γνωρίζουμε SQL. Χρησιμοποιεί τους ορισμούς των τύπων σχέσεων που της έχουμε ορίσει για να συνδέσει αυτόματα τους πίνακες που πρέπει. Έτσι, μπορούμε να επικεντρωθούμε στον τρόπο με τον οποίο θα επιλύσουμε τα προβλήματα που αφορούν τις πληροφορίες χωρίς να ανησυχούμε για το πώς θα κατασκευάσουμε τα πολύπλοκα συστήματα σύνδεσης όλων των δομών των στοιχείων της βάσης δεδομένων μας. Η Access έχει επίσης μια εξαιρετικά απλή, αλλά και πανίσχυρη λειτουργία ορισμού ερωτημάτων με τη βοήθεια γραφικών που μπορούμε να χρησιμοποιούμε για να καθορίζουμε τα δεδομένα που χρειαζόμαστε για την επίλυση ενός προβλήματος χρησιμοποιώντας τις τεχνικές "κατάδειξης και πατήματος" ( Point and click ) και "μεταφοράς και ενσωμάτωσης" ( drag and drop), και μερικά πατήματα πλήκτρων, μπορούμε να κατασκευάσουμε ένα πολύπλοκο ερώτημα μέσα σε λίγα μόνο δευτερόλεπτα.

### **4.3 Έλεγχος Δεδομένων**

Τα λογιστικά φύλλα και τα έγγραφα είναι θαυμάσια για την επίλυση προβλημάτων ενός χρήστη, αλλά δύσκολα όταν τα δεδομένα πρέπει να μοιράζονται σε περισσότερα από ένα άτομα. Τα λογιστικά φύλλα είναι επίσης χρήσιμα για να παρέχουν κάποιο πρότυπο απλής εισαγωγής στοιχείων, αλλά δεν κάνουν καλά τη δουλειά τους όταν απαιτούνται περίπλοκοι έλεγχοι εγκυρότητας αυτών των στοιχείων. Για παράδειγμα, ένα λογιστικό φύλλο δουλεύει καλά σαν πρότυπο για την τιμολόγηση μιας μικρής επιχείρησης ενός ιδιοκτήτη, αλλά, όταν η δουλειά επεκταθεί και οι παραγγελίες καταχωρίζονται από περισσότερους από έναν πωλητές, τότε χρειαζόμαστε βάση δεδομένων. Παρόμοια, ένα λογιστικό φύλλο μπορεί να δουλεύει καλά για να βοηθά τους υπαλλήλους στην παραγωγή αναφορών δαπανών σε μεγάλες επιχειρήσεις, αλλά είναι πιθανό να πρέπει να συγκεντρωθούν τα δεδομένα και να τοποθε-

τηθούν σε μία βάση δεδομένων για να τα επεξεργάζεται το κεντρικό λογιστήριο της εταιρείας.

Όταν πρέπει να μοιραζόμαστε τα δεδομένα μας με άλλους, τα πραγματικά συστήματα διαχείρισης σχεσιακών βάσεων δεδομένων μας επιτρέπουν να τα προστατεύουμε, έτσι ώστε μόνο εξουσιοδοτημένοι χρήστες να μπορούν να τα διαβάζουν ή να τα ενημερώνουν. Ένα RDBMS που έχει σχεδιαστεί για να επιτρέπει το μερισμό δεδομένων, παρέχει επίσης δυνατότητες που εξασφαλίζουν πως δε θα προσπαθήσουν δύο άτομα, την ίδια στιγμή, να αλλάξουν τα ίδια δεδομένα. Τα καλύτερα συστήματα, μας επιτρέπουν επίσης να ομαδοποιούμε τις μεταβολές (μια σειρά μεταβολών ονομάζεται πολλές φορές κίνηση, transaction), έτσι ώστε ή να ισχύουν όλες μαζί, ή να μην ισχύει καμία από αυτές. Για παράδειγμα, καθώς εισάγουμε μια νέα παραγγελία για κάποιον πελάτη, είναι πιθανό να θέλουμε να γνωρίζουμε ότι καταγράφηκαν όλα τα είδη, ή, αν διαπιστώσουμε κάποιο λάθος, να θέλουμε να μην αποθηκευτεί καμία από τις μεταβολές. Θα θέλαμε επίσης να είμαστε βέβαιοι ότι κανείς άλλος δε θα μπορεί να δει κάποιο μέρος της παραγγελίας μέχρι να την εισαγάγουμε ολόκληρη.

Η Access σχεδιάστηκε για να χρησιμοποιείται είτε σαν απλό RDBMS σε ένα σταθμό εργασίας, είτε σε κατάσταση μερισμού πελάτη-διακομιστή ( client-server ) σε δίκτυο. Επειδή μπορούμε να μοιραζόμαστε τα στοιχεία μας με άλλους χρήστες, η Access έχει εξαιρετικές δυνατότητες προστασίας και ακεραιότητας των στοιχείων. Μπορούμε να ορίζουμε ποιοι χρήστες ή ομάδες χρηστών θα έχουν πρόσβαση στα αντικείμενα (πίνακες, φόρμες, ερωτήματα) της βάσης δεδομένων μας. Η Access παρέχει αυτόματα μηχανισμούς κλειδώματος, ώστε να μην μπορούν δύο άτομα να ενημερώσουν το ίδιο αντικείμενο την ίδια στιγμή. Μπορεί επίσης να κατανοήσει και να χρησιμοποιήσει τους μηχανισμούς κλειδώματος άλλων δομών βάσεων δεδομένων (όπως του paradox, της dBASE , και των βάσεων δεδομένων SQL), τις οποίες έχουμε συνδέσει με τη βάση δεδομένων μας.

#### **4.4 Διαδικασία Ανάπτυξης Ενός Συστήματος Βάσης Δεδομένων**

Η πολυπλοκότητα των σημερινών εφαρμογών βάσεων δεδομένων έχει οδηγήσει στην ανάπτυξη μεθοδολογιών και εργαλείων (CASE tools) για όλα τα στάδια της ανάπτυξης ενός συστήματος βάσεων δεδομένων. Σύμφωνα με τις περισσότερες μεθοδολογίες ανάπτυξης, τα στάδια της κατασκευής μίας βάσης δεδομένων είναι τα ακόλουθα:

- ✿ Καθορισμός του συστήματος βάσης δεδομένων (σκοπός του συστήματος, χρήστες του συστήματος, απαιτήσεις των χρηστών από το σύστημα).
- ✿ Σχεδίαση του συστήματος (Ιδεατή, λογική και φυσική σχεδίαση της βάσης δεδομένων).
- ✿ Υλοποίηση της εφαρμογής (περιλαμβάνει την υλοποίηση της σχεδίασης της βάσης δεδομένων με τη γλώσσα ορισμού και χειρισμού δεδομένων του επιλεγμένου συστήματος διαχείρισης βάσεων δεδομένων).
- ✿ Φόρτωση των δεδομένων (στο στάδιο αυτό τα πραγματικά δεδομένα φορτώνονται στις υλοποιημένες δομές δεδομένων, είτε άμεσα είτε με τη μετατροπή τους από υπάρχοντα αρχεία).
- ✿ Έλεγχος και αξιολόγηση του συστήματος βάσεων δεδομένων.
- ✿ Λειτουργία του συστήματος.
- ✿ Παρακολούθηση της λειτουργίας και συντήρηση του συστήματος.

Ιδιαίτερη σημασία αποδίδεται στη φάση της σχεδίασης ενός συστήματος βάσης δεδομένων. Ο στόχος της σχεδίασης είναι διττός: αποσκοπεί τόσο στο να ικανοποιήσει τις ανάγκες των χρηστών σε πληροφορίες μίας εφαρμογής, αποτυπώνοντας τις πληροφορίες σε κατάλληλες δομές δεδομένων, όσο και στο να εξυπηρετήσει τις λειτουργικές απαιτήσεις και τις ανάγκες επίδοσης μίας εφαρμογής (χρόνος απόκρισης, χρόνος επεξεργασίας, απαιτήσεις σε μνήμη). Τα στάδια της σχεδίασης, είναι τα ακόλουθα:

- ✿ Συγκέντρωση και ανάλυση των απαιτήσεων των χρηστών.
- ✿ Σχεδίαση του ιδεατού σχήματος (conceptual modeling).
- ✿ Επιλογή του συστήματος διαχείρισης βάσεων δεδομένων που θα χρησιμοποιηθεί.
- ✿ Σχεδίαση του λογικού σχήματος (logical modelling).
- ✿ Σχεδίαση του φυσικού σχήματος (physical modelling).

Στόχος του ιδεατού σχήματος είναι η σαφής περιγραφή του περιεχομένου της βάσης δεδομένων, ανεξάρτητα από τον τρόπο υλοποίησής της. Το ιδεατό σχήμα χρησιμοποιείται ως κοινή πλατφόρμα επικοινωνίας μεταξύ των μελλοντικών χρηστών, των σχεδιαστών, των αναλυτών και των προγραμματιστών εφαρμογών της βάσης δεδομένων. Για το λόγο αυτό, τα μοντέλα που χρησιμοποιούνται για την ιδεατή σχεδίαση είναι εκφραστικά, απλά, χρησιμοποιούν λίγα δομικά στοιχεία και αναπαριστούν τα δεδομένα και τις συσχετίσεις τους με διαγράμμα-

τα, ώστε να είναι κατανοητά. Το ποιο συχνά χρησιμοποιούμενο ιδεατό μοντέλο σήμερα είναι το **μοντέλο οντοτήτων-συσχετίσεων** (γνωστό ως E-R model).

Στο προσεχές μέλλον εκτιμάται ότι η παρουσία των συστημάτων βάσεων δεδομένων θα γίνει απαραίτητη, ακόμα και σε μη παραδοσιακούς τομείς μηχανογράφησης, όπως είναι τα βιομηχανικά περιβάλλοντα (Computer Integrated Manufacturing), τα γεωγραφικά συστήματα πληροφοριών, καθώς και σε πολιτιστικές, τουριστικές, ιατρικές και άλλες εφαρμογές, όπου απαιτείται και χειρισμός πολυμέσων (εικόνας, ήχου, video κ.λπ.). Σε όλες τις παραπάνω περιοχές, ο όγκος της πληροφορίας είναι σημαντικός και αξιοποιείται αποδοτικά μέσω αυστηρών κανόνων οργάνωσης της. Για παράδειγμα, σε ιατρικά περιβάλλοντα, η καταγραφή όχι μόνο των σταθερών στοιχείων ασθενών αλλά και των ακτινογραφιών, ιατρικών εξετάσεων κ.λπ. και η δυνατότητα αναζήτησης ατόμων με βάση τέτοια χαρακτηριστικά (και όχι απλώς το όνομα, ή άλλα σταθερά στοιχεία) ανοίγει νέους ορίζοντες για την αξιοποίηση των περιοχών αυτών.

### Η γλώσσα SQL

Για να μπορέσουμε να εξαγάγουμε πληροφορίες από μια βάση χρειαζόμαστε τη βοήθεια μιας γλώσσας που επιτρέπει την υποβολή ερωτήσεων σε ένα ΣΔΒΔ. Η πιο διαδεδομένη γλώσσα διαχείρισης σχεσιακών βάσεων δεδομένων είναι η SQL (Structured Query Language). Είναι μια πλήρης γλώσσα βάσεων δεδομένων. Παρέχει δυνατότητες για:

- τον ορισμό, τη διαγραφή και τη μεταβολή πινάκων και κλειδιών,
- τη σύνταξη ερωτήσεων (*queries*),
- την εισαγωγή, διαγραφή και μεταβολή στοιχείων,
- τον ορισμό όψεων (*views*) πάνω στα δεδομένα,
- τον ορισμό δικαιωμάτων πρόσβασης,
- τον έλεγχο της ακεραιότητας των στοιχείων,
- τον έλεγχο συναλλαγών (*transaction*)

Από τα παραπάνω γίνεται σαφές ότι είναι ταυτόχρονα και μια Γλώσσα Ορισμού Δεδομένων αλλά και Γλώσσα Χειρισμού Δεδομένων.

Η SQL είναι ορισμένη ως διεθνές πρότυπο.

Επιτρέπει να γίνονται εύκολα πολύπλοκες αναζητήσεις μέσα στη βάση δεδομένων. Είναι φτιαγμένη έτσι ώστε μια εντολή SQL να είναι ανεξάρτητη από το λογισμικό με το οποίο δημιουργήθηκε και τηρείται μια βάση δεδομένων. Η SQL υποστηρίζεται σήμερα από τα περισσότερα σύγχρονα προγράμματα ΣΔΒΔ.

Η βασική δομή μιας έκφρασης αναζήτησης σε γλώσσα SQL αποτελείται από τρία τμήματα, τα:

**Select:** Περιγράφει τα χαρακτηριστικά που αναζητούμε και περιμένουμε σαν αποτέλεσμα από αυτήν την εντολή.

**From:** Είναι ένας κατάλογος από πίνακες ή ερωτήματα που χρησιμοποιείται για την αναζήτηση.

**Where:** Ορίζει τιμές των χαρακτηριστικών των εγγραφών που αναζητούνται. Μπορεί να περιέχει και σχέσεις που να συνδυάζουν τιμές διαφόρων χαρακτηριστικών

Παράδειγμα : Αν θέλουμε να αναζητήσουμε όλους τους μαθητές που μένουν στην Αθήνα, θα δώσουμε την εντολή:

**Select** όνομα **from** Μαθητολόγιο **Where** πόλη διαμονής = Αθήνα

Αν θέλουμε όλους τους μαθητές και τους καθηγητές που επίσης διανέμουν στην Αθήνα (με την προϋπόθεση πως υπάρχει ένας αντίστοιχος πίνακας «ΚΑΘΗΓΗΤΕΣ», θα δώσουμε την εντολή:

**Select** όνομα **from** Μαθητολόγιο, Καθηγητές **Where** πόλη διαμονής = Αθήνα

Αν θέλουμε να εντοπίσουμε τους μαθητές και καθηγητές που μένουν στην Αθήνα ή στο Βόλο, θα δώσουμε την εντολή:

**Select** όνομα **from** Μαθητολόγιο, Καθηγητές **Where** πόλη διαμονής = Αθήνα **or** πόλη διαμονής = Βόλος

## 5.1 Τύποι δεδομένων

Τα δεδομένα κάθε στήλης ενός πίνακα πρέπει να έχουν ένα συγκεκριμένο τύπο. Οι βασικοί τύποι που υποστηρίζονται από την SQL είναι οι παρακάτω:

BIT

Ναι ή Όχι

CURRENCY

Τιμή που παριστάνει με ακρίβεια αριθμούς από 922.337.203.685.477,5808 έως 922.337.203.685.477,5807

DATETIME

Χρόνος

SINGLE

Αριθμός κινητής υποδιαστολή μονής ακρίβειας

DOUBLE

Αριθμός κινητής υποδιαστολή διπλής ακρίβειας

SHORT

Ακέραιος 2 byte (-32768 έως 32767)

LONG

Ακέραιος 4 byte (-2.147.483.648 έως 2.147.483.647)

TEXT

Κείμενο μέχρι 255 χαρακτήρες

LONGTEXT

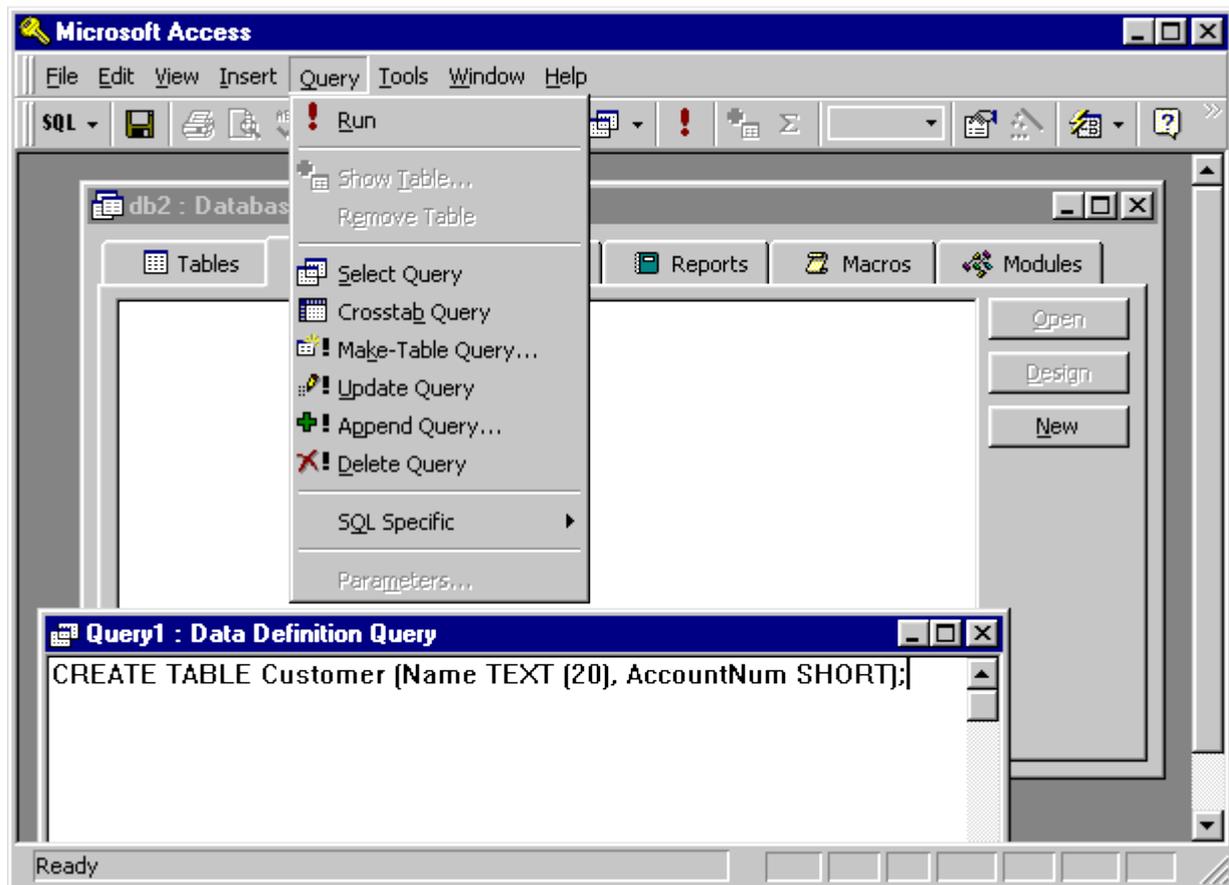
Κείμενο μέχρι 1.2GB

## **5.2 Η SQL στη Microsoft Access**

Για να δώσουμε εντολές SQL στη Microsoft Access ακολουθούμε την παρακάτω διαδικασία:

- Δημιουργούμε μια κενή βάση δεδομένων
- View - Database objects - Queries
- New - Design view
- Show table - close
- View - SQL
- Γράφουμε την εντολή SQL που θέλουμε
- Query - Run

Τα αποτελέσματα της εντολής μπορούμε να τα δούμε στην περιοχή Tables.



## **ΠΡΑΚΤΙΚΟ ΜΕΡΟΣ – ΕΦΑΡΜΟΓΗ**

### **ΕΙΣΑΓΩΓΗ**

Η εφαρμογή αυτή αφορά στη δημιουργία μιας Βάσης Δεδομένων για τον έλεγχο της επίδοσης μαθητών ελληνικού εκπαιδευτικού ιδρύματος.

Για το σχεδιασμό μιας Βάσης Δεδομένων για τη εφαρμογή αυτή πρέπει να ορισθούν οι πίνακες της Βάσης. Η δημιουργία κάθε πίνακα περιλαμβάνει τον ορισμό και το χαρακτηρισμό των πεδίων και τον ορισμό του πρωτεύοντος κλειδιού όπου αυτό είναι αναγκαίο. Διακρίνουμε τους παρακάτω πίνακες.

- **ΜΑΘΗΤΟΛΟΓΙΟ.** Στον πίνακα αυτό καταχωρούνται τα στοιχεία του κάθε μαθητή.
- **ΒΑΘΜΟΛΟΓΙΕΣ.** Στον πίνακα αυτών καταχωρούνται οι βαθμολογίες των μαθητών και στα 3 τρίμηνα

### **6.1 ΣΧΕΔΙΑΣΜΟΣ ΤΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ**

Για τη δημιουργία της Βάσης πρέπει πρώτα να επιλεγούν τα απαραίτητα πεδία Όταν φορτωθεί πρόγραμμα της Βάσης Δεδομένων και ζητηθεί η δημιουργία δομής των πινάκων θα πρέπει να δοθούν διάφορες πληροφορίες για τα παρακάτω.

•	Όνόματα	πεδίων
•	Τύπο	πεδίων
•	Εύρος	πεδίων
•	Πεδία	κλειδιά

Τα ονόματα των τύπων των δεδομένων που θα χρησιμοποιηθούν είναι τα **Ονόματα πεδίων**. Κάθε πεδίο δεδομένων πρέπει να έχει ένα ξεχωριστό όνομα. Οι πιο συνηθισμένοι **τύποι** είναι:

- Πεδία **χαρακτήρων**, τα οποία περιέχουν δεδομένα όπως ονόματα, διευθύνσεις, αριθμούς τηλεφώνων κλπ.
- **Αριθμητικά** πεδία, τα οποία μπορεί να χρησιμοποιηθούν σε υπολογισμούς, όπως βαθμοί, σειρά ή άλλα κλπ. Στα αριθμητικά δεδομένα συνήθως δηλώνεται και ο αριθμός των δεκαδικών ψηφίων που περιέχουν.
- Πεδία **ημερομηνίας**, τα οποία περιλαμβάνουν συνήθως οκτώ χαρακτήρες, π.χ. 12/01/98.
- **Λογικά** πεδία τα οποία δέχονται ένα μόνο χαρακτήρα για να δηλωθεί εάν αληθεύει (Ν ή Σ) ή όχι (Ο ή Λ) μία συνθήκη.

Το **Εύρος των Πεδίων** δηλώνει το μέγιστο αριθμό χαρακτήρων ή ψηφίων που θα περιέχει το πεδίο συμπεριλαμβανομένης και της υποδιαστολής.

Τα **πρωτεύοντα κλειδιά** προσδιορίζουν μονοσήμαντα μία εγγραφή, π.χ. ο κωδικό του βιβλίου

Η δομή του πινάκων της Βάσης ορίζεται στους παρακάτω πίνακες:

**Πίνακας ΜΑΘΗΤΟΛΟΓΙΟ** με πρωτεύον κλειδί τον **Κωδικό Μαθητή**

A/A	Όνομα πεδίου	Τύπος πεδίου	Εύρος πεδίου
1	Κωδικός Μαθητή	κείμενο	8
2	Όνομα	κείμενο	30
3	Επώνυμο	κείμενο	30
4	Τάξη	κείμενο	15
5	Τμήμα	κείμενο	2

**Πίνακας ΒΑΘΜΟΛΟΓΙΕΣ** με πρωτεύον κλειδί τον **Κωδικό Μαθητή**

A/A	Όνομα πεδίου	Τύπος πεδίου	Εύρος πεδίου
1	Κωδικός Μαθητή	κείμενο	8
2	Βαθμός Α τριμήνου	αριθμός	3
3	Βαθμός Β τριμήνου	αριθμός	3
4	Βαθμός Γ τριμήνου	αριθμός	3

Microsoft Access - [ΒΑΘΜΟΛΟΓΙΕΣ : Πίνακας]

Αρχείο Επεξεργασία Προβολή Εισαγωγή Εργαλεία Παράθυρο Βοήθεια Πληκτρολογήστε ερώτηση

Όνομα πεδίου	Τύπος δεδομένων	Περιγραφή
ΚΩΔΙΚΟΣ ΜΑΘΗΤΗ	Κείμενο	
ΒΑΘΜΟΣ Α ΤΡΙΜΗΝΟΥ	Αριθμός	
ΒΑΘΜΟΣ Β ΤΡΙΜΗΝΟΥ	Αριθμός	
ΒΑΘΜΟΣ Γ ΤΡΙΜΗΝΟΥ	Αριθμός	

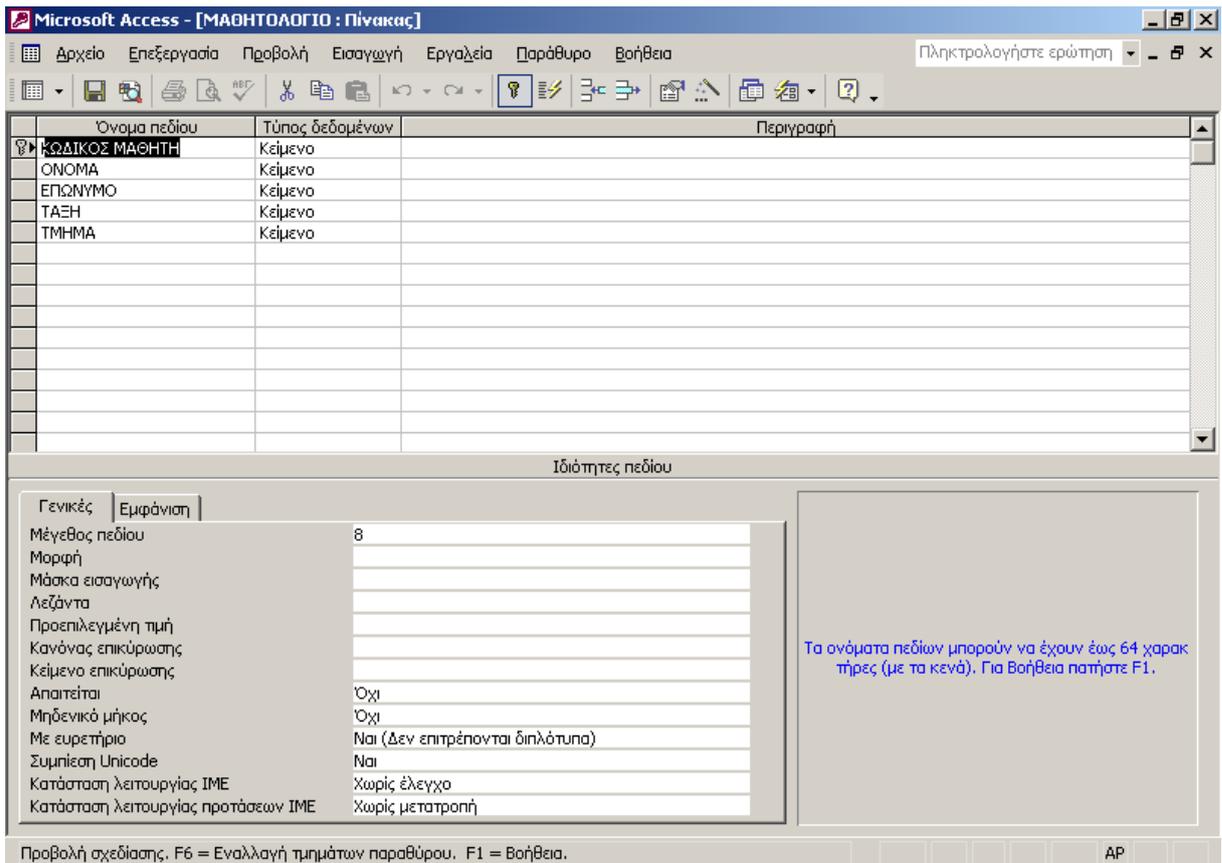
Ιδιότητες πεδίου

Γενικές Εμφάνιση

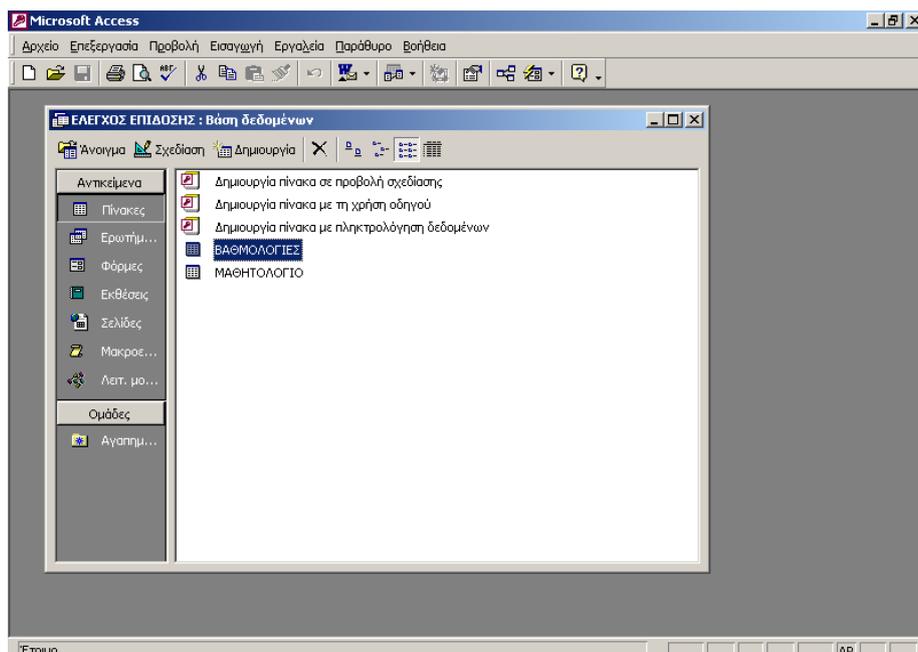
Μέγεθος πεδίου	8
Μορφή	
Μάσκα εισαγωγής	
Λεζάντα	
Προεπιλεγμένη τιμή	
Κανόνας επικύρωσης	
Κείμενο επικύρωσης	
Απαιτείται	Όχι
Μηδενικό μήκος	Όχι
Με ευρετήριο	Ναι (Δεν επιτρέπονται διπλάτιστα)
Συμπίεση Unicode	Ναι
Κατάσταση λειτουργίας IME	Χωρίς έλεγχο
Κατάσταση λειτουργίας προτάσεων IME	Χωρίς μετατροπή

Τα ονόματα πεδίων μπορούν να έχουν έως 64 χαρακτήρες (με τα κενά). Για βοήθεια πατήστε F1.

Προβολή σχεδίασης. F6 = Εναλλαγή τμημάτων παραθύρου. F1 = Βοήθεια. AP

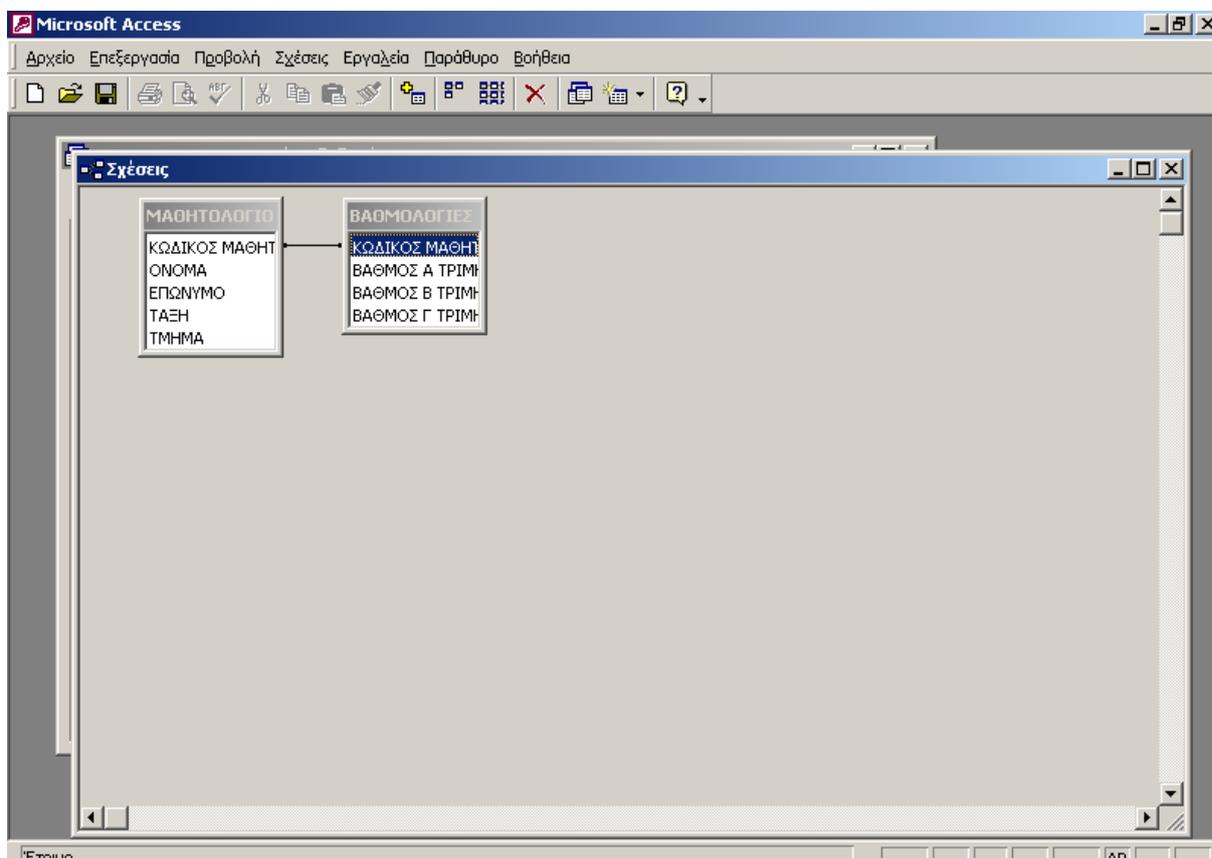


Σε περιβάλλον ACCESS μετά τη δημιουργία των πινάκων η βασική οθόνη της ACCESS θα έχει την παρακάτω μορφή:



Εικόνα 6-1. Η Βάση Δεδομένων της Βιβλιοθήκης σε περιβάλλον ACCESS.

Στη συνέχεια θα πρέπει να οριστούν οι **συσχετίσεις ή σχέσεις (relationships)** μεταξύ των πινάκων. Η διασύνδεση των πινάκων με τις σχέσεις σε περιβάλλον της ACCESS φαίνεται στην παρακάτω εικόνα:



Εικόνα 6-2. Η διασύνδεση των πινάκων της Βάσης Δεδομένων της Βιβλιοθήκη σε περιβάλλον ACCESS.

## 6.2 Εισαγωγή των δεδομένων στο αρχείο

Κατά την πληκτρολόγηση των δεδομένων η οθόνη παρουσιάζει τα πεδία σε κάποια κατάλληλη μορφή για την εισαγωγή των δεδομένων.

Microsoft Access - [ΜΑΘΗΤΟΛΟΓΙΟ : Πίνακας]

Αρχείο Επεξεργασία Προβολή Εισαγωγή Μορφή Εγγραφές Εργαλεία Παράθυρο Βοήθεια

ΚΩΔΙΚΟΣ ΜΑΘΗΤΗ	ΟΝΟΜΑ	ΕΠΩΝΥΜΟ	ΤΑΞΗ	ΤΜΗΜΑ
+ A-A-1	ΑΓΓΕΛΙΝΑ	ΑΛΕΞΙΟΥ	Β' ΛΥΚΕΙΟΥ	B1
+ A-A-2	ΑΝΑΣΤΑΣΙΟΣ-ΝΕΚΤΑΡΙΟΣ	ΑΝΤΩΝΙΟΥ	Γ' ΛΥΚΕΙΟΥ	Γ3
+ A-Γ-3	ΑΘΗΝΑ-ΧΡΙΣΤΙΝΑ	ΓΕΩΡΓΙΟΥ	Α' ΛΥΚΕΙΟΥ	A1
+ A-Γ-4	ΑΛΕΞΑΝΔΡΟΣ	ΠΑΝΝΑΚΗΣ	Γ' ΛΥΚΕΙΟΥ	Γ1
+ A-Γ-5	ΑΙΚΙΣΩΝ	ΓΕΝΗΣ	Γ' ΛΥΚΕΙΟΥ	Γ3
+ A-Γ-6	ΑΝΑΣΤΑΣΙΑ	ΓΚΕΚΑ	Α' ΛΥΚΕΙΟΥ	A1
+ A-Γ-7	ΑΝΔΡΟΝΙΚΗ	ΓΕΩΡΓΑΚΗ	Γ' ΛΥΚΕΙΟΥ	Γ1
+ A-Z-10	ΑΝΔΡΕΑΣ	ΖΑΧΑΡΑΚΗΣ	Α' ΛΥΚΕΙΟΥ	A1
+ A-Z-8	ΑΛΚΗΣΤΗ-ΘΕΟΔΩΡΑ	ΖΑΧΑΡΟΠΟΥΡΓΑ	Β' ΛΥΚΕΙΟΥ	B1
+ A-Z-9	ΑΛΚΗΣΤΗ-ΝΑΤΑΛΙΑ	ΖΗΣΗ	Γ' ΛΥΚΕΙΟΥ	Γ1
+ A-I-11	ΑΡΙΣΤΕΙΔΗΣ	ΙΑΣΩΝΙΔΗΣ	Β' ΛΥΚΕΙΟΥ	B3
+ A-K-12	ΑΓΑΘΗ	ΚΟΛΛΙΑ	Γ' ΛΥΚΕΙΟΥ	Γ3
+ A-K-13	ΑΙΚΑΤΕΡΙΝΗ	ΚΑΤΣΑΝΟΥ	Α' ΛΥΚΕΙΟΥ	A1
+ A-K-14	ΑΙΚΑΤΕΡΙΝΗ	ΚΟΝΤΟΜΙΧΗ	Γ' ΛΥΚΕΙΟΥ	Γ1
+ A-K-15	ΑΛΕΞΑΝΔΡΟΣ	ΚΩΣΤΑΣ	Γ' ΛΥΚΕΙΟΥ	Γ1
+ A-K-16	ΑΛΕΞΗΣ	ΚΑΒΒΑΔΑΣ	Β' ΛΥΚΕΙΟΥ	B3
+ A-K-17	ΑΝΑΣΤΑΣΙΑ	ΚΩΝΣΤΑΝΤΙΝΗ	Α' ΛΥΚΕΙΟΥ	A2
+ A-K-18	ΑΝΔΡΕΑΣ	ΚΟΝΤΟΔΗΜΑΣ	Γ' ΛΥΚΕΙΟΥ	Γ3
+ A-Λ-19	ΑΛΙΚΗ-ΗΛΙΑΝΑ	ΛΑΖΑΡΗ	Β' ΛΥΚΕΙΟΥ	B3
+ A-Λ-20	ΑΝΤΥ	ΛΙΓΚΟΥ	Α' ΛΥΚΕΙΟΥ	A4
+ A-M-21	ΑΘΗΝΑ	ΜΑΥΡΟΚΕΦΑΛΟΥ	Γ' ΛΥΚΕΙΟΥ	Γ1
+ A-M-22	ΑΛΕΞΑΝΔΡΟΣ	ΜΑΝΙΚΗΣ	Α' ΛΥΚΕΙΟΥ	A2
+ A-M-23	ΑΛΕΞΑΝΔΡΟΣ	ΜΗΤΣΙΟΣ	Γ' ΛΥΚΕΙΟΥ	Γ2
+ A-M-24	ΑΛΕΞΑΝΔΡΟΣ	ΜΠΑΙΛΗΣ	Β' ΛΥΚΕΙΟΥ	B1
+ A-M-25	ΑΡΙΣΤΕΙΑΗΣ	ΜΑΝΩΣ	Β' ΛΥΚΕΙΟΥ	B1

Εγγραφή: 1 από 256

Προβολή φύλλου δεδομένων

Εικόνα 6-3. Τα περιεχόμενα του πίνακα ΜΑΘΗΤΟΛΟΓΙΟ

Microsoft Access - [ΒΑΘΜΟΛΟΓΙΕΣ : Πίνακας]

Αρχείο Επεξεργασία Προβολή Εισαγωγή Μορφή Εγγραφές Εργαλεία Παράθυρο Βοήθεια

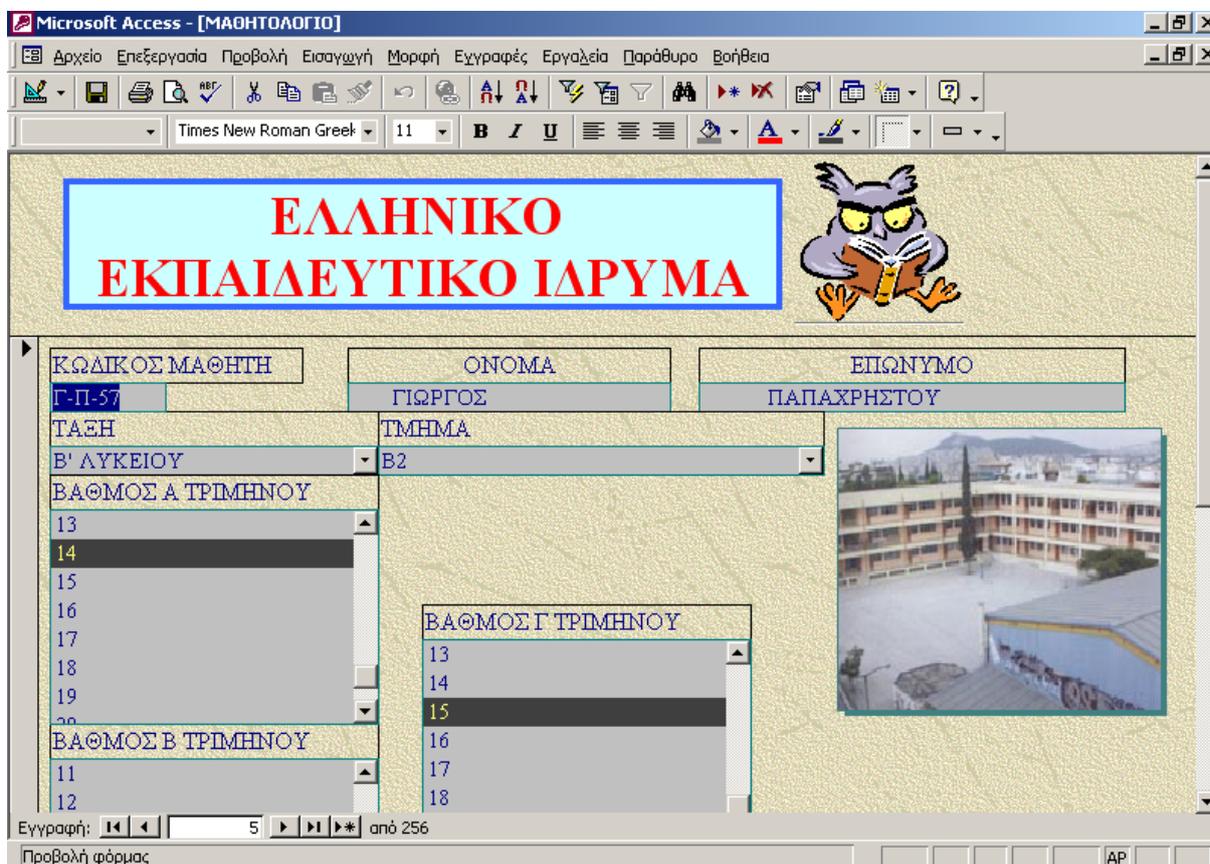
ΚΩΔΙΚΟΣ ΜΑΘΗΤΗ	ΒΑΘΜΟΣ Α ΤΡΙΜΗΝΟΥ	ΒΑΘΜΟΣ Β ΤΡΙΜΗΝΟΥ	ΒΑΘΜΟΣ Γ ΤΡΙΜΗΝΟΥ
+ A-A-1	7	18	17
A-A-2	15	16	16
A-Γ-3	18	8	19
A-Γ-4	10	12	12
A-Γ-5	17	18	18
A-Γ-6	11	14	13
A-Γ-7	17	18	19
A-Z-10	15	15	16
A-Z-8	19	19	20
A-Z-9	19	19	20
A-I-11	13	15	16
A-K-12	14	16	16
A-K-13	18	17	17
A-K-14	18	17	17
A-K-15	15	15	17
A-K-16	11	13	14
A-K-17	16	16	17
A-K-18	17	17	18
A-Λ-19	19	16	17
A-Λ-20	12	15	16
A-M-21	18	19	19
A-M-22	14	15	16
A-M-23	12	13	14
A-M-24	15	16	16
A-M-25	13	14	15

Εγγραφή: 1 από 256

Προβολή φύλλου δεδομένων

Εικόνα 6-4. Τα περιεχόμενα του πίνακα ΒΑΘΜΟΛΟΓΙΕΣ

Ο χρήστης όμως μπορεί να δημιουργήσει δικές του μορφές (φόρμες) εισαγωγής των εγγραφών. Μία μορφή εισαγωγής δεδομένων φαίνεται στο σχήμα 6-5. Για να εισαχθούν τα δεδομένα πληκτρολογούνται οι αντίστοιχες τιμές στις θέσεις που αντιστοιχούν στα ονόματα δεδομένων όπως φαίνεται παρακάτω.



Σχήμα 6-5

### 6.3 Χρήση Ερωτημάτων

Ο χρήστης με την χρήση ερωτημάτων μπορεί να επιλέξει και να εμφανίσει συγκεκριμένες ομάδες εγγραφών που ανταποκρίνονται στα κριτήρια που ορίζει ή να συνδυάσει πληροφορίες από διαφορετικούς πίνακες, για να έχει μια ομοιογενή προβολή σχετιζόμενων δεδομένων. Στο σχήμα 6-6, αναπαρίστανται οι αριστούχοι μαθητές, ενώ στο σχήμα 6-7, οι απορριπτέοι.

Microsoft Access - [ΑΡΙΣΤΟΙ ΜΑΘΗΤΕΣ : Ερώτημα επιλογής]

Αρχείο Επεξεργασία Προβολή Εισαγωγή Μορφή Εγγραφές Εργαλεία Παράθυρο Βοήθεια

ΚΩΔΙΚΟΣ ΜΑΘΗΤΗ	ΟΝΟΜΑ	ΕΠΩΝΥΜΟ	ΤΑΞΗ	ΤΜΗΜΑ	ΒΑΘΜΟΣ Α	ΒΑΘΜΟΣ Β	ΒΑΘΜΟΣ Γ	ΜΕΣΟΣ ΟΡΟΣ
Μ-Π-176	ΜΑΡΙΑ	ΠΑΝΑΓΙΩΤΟΥ	Γ' ΛΥΚΕΙΟΥ	Γ4	19	19	19	19
A-Z-8	ΑΛΚΗΣΙΤΗ-ΘΕΟΔΩΡΑ	ΖΑΧΑΡΟΠΟΥΡΓΑ	Β' ΛΥΚΕΙΟΥ	Β1	19	19	20	19,3333333333
A-M-21	ΑΘΗΝΑ	ΜΑΥΡΟΚΕΦΑΛΟΥ	Γ' ΛΥΚΕΙΟΥ	Γ1	18	19	19	18,6666666667
E-O-95	ΕΡΑΝΤΑ-ΒΑΣΙΛΙΚΗ	ΟΜΕΡΙ	Α' ΛΥΚΕΙΟΥ	Α3	18	19	19	18,6666666667
M-T-181	ΜΑΡΙΑΝΝΑ	ΤΣΑΤΣΟΥ	Γ' ΛΥΚΕΙΟΥ	Γ2	19	19	19	19
K-M-144	ΚΩΝΣΤΑΝΤΙΝΟΣ	ΜΠΑΛΗΣ	Γ' ΛΥΚΕΙΟΥ	Γ2	19	19	19	19
X-Γ-245	ΧΡΗΣΤΟΣ	ΓΙΝΗ	Β' ΛΥΚΕΙΟΥ	Β1	18	19	19	18,6666666667
A-Z-9	ΑΛΚΗΣΙΤΗ-ΝΑΤΑΛΙΑ	ΖΗΣΗ	Γ' ΛΥΚΕΙΟΥ	Γ1	19	19	20	19,3333333333
Λ-Ρ-159	ΛΑΜΠΡΟΣ	ΡΑΠΤΗΣ	Α' ΛΥΚΕΙΟΥ	Α3	18	19	19	18,6666666667
X-K-248	ΧΑΡΙΛΑΚΗΣ	ΚΩΤΣΙΑΣ	Γ' ΛΥΚΕΙΟΥ	Γ3	19	19	20	19,3333333333
E-M-92	ΕΥΡΥΔΙΚΗ	ΜΠΟΛΑΝΟΥ	Α' ΛΥΚΕΙΟΥ	Α4	19	19	20	19,3333333333
N-N-190	ΝΙΚΟΛΑΟΣ	ΝΤΑΛΛΑΣ	Β' ΛΥΚΕΙΟΥ	Β4	18	19	20	19
*								

Εγγραφή: 1 από 12  
Προβολή φύλλου δεδομένων

Σχήμα 6-6

Microsoft Access - [ΑΠΟΡΡΙΠΤΑΙΟΙ ΜΑΘΗΤΕΣ : Ερώτημα επιλογής]

Αρχείο Επεξεργασία Προβολή Εισαγωγή Μορφή Εγγραφές Εργαλεία Παράθυρο Βοήθεια

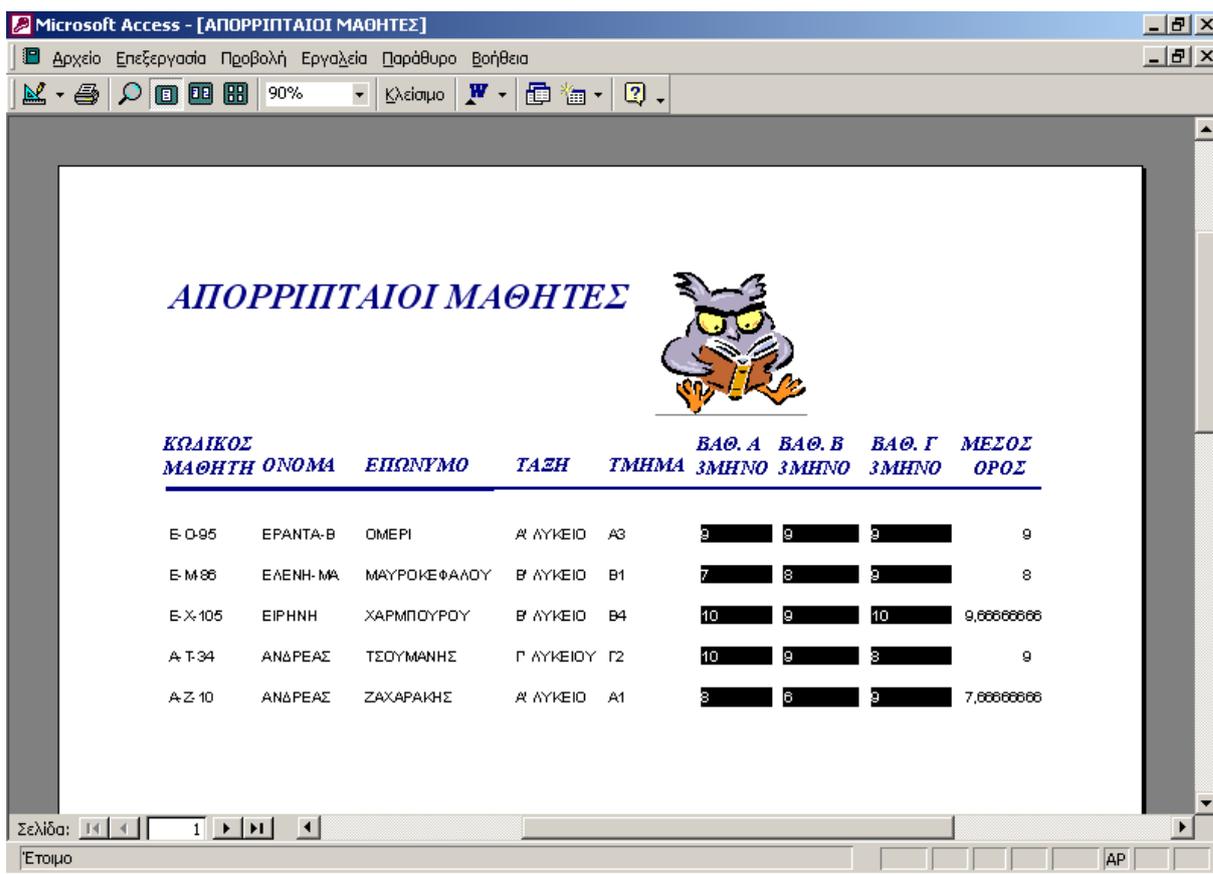
ΚΩΔΙΚΟΣ ΜΑΘΗΤΗ	ΟΝΟΜΑ	ΕΠΩΝΥΜΟ	ΤΑΞΗ	ΤΜΗΜΑ	ΒΑΘΜΟΣ Α	ΒΑΘΜΟΣ Β	ΒΑΘΜΟΣ Γ	ΜΕΣΟΣ ΟΡΟΣ
E-O-95	ΕΡΑΝΤΑ-ΒΑΣΙΛΙΚΗ	ΟΜΕΡΙ	Α' ΛΥΚΕΙΟΥ	Α3	9	9	9	9
E-M-86	ΕΛΕΝΗ-ΜΑΡΙΑ	ΜΑΥΡΟΚΕΦΑΛΟΥ	Β' ΛΥΚΕΙΟΥ	Β1	7	8	9	8
E-X-105	ΕΙΡΗΝΗ	ΧΑΡΜΠΟΥΡΟΥ	Β' ΛΥΚΕΙΟΥ	Β4	10	9	10	9,6666666667
A-T-34	ΑΝΔΡΕΑΣ	ΤΣΟΥΜΑΝΗΣ	Γ' ΛΥΚΕΙΟΥ	Γ2	10	9	8	9
A-Z-10	ΑΝΔΡΕΑΣ	ΖΑΧΑΡΑΚΗΣ	Α' ΛΥΚΕΙΟΥ	Α1	8	6	9	7,6666666667
*								

Εγγραφή: 3 από 5  
Προβολή φύλλου δεδομένων

σχήμα 6-7

## 6.4 Χρήση Εκθέσεων

Τέλος, συχνά ο χρήστης χρειάζεται να επιλέξει, να οργανώσει και να τυπώσει συγκεκριμένα στοιχεία της βάσης δεδομένων του. Η Access, του παρέχει τις Εκθέσεις, οι οποίες είναι εργαλεία που έχουν ως βασικό σκοπό τους την αποτελεσματική παρουσίαση των δεδομένων σε έντυπη μορφή. Στο σχήμα 6-8, αναπαρίσταται μια Έκθεση για τους απορριπτέους μαθητές, και στο σχήμα 6-9, μια Έκθεση για τους αριστούχους.



The screenshot shows a Microsoft Access window titled "Microsoft Access - [ΑΠΟΡΡΙΠΤΑΙΟΙ ΜΑΘΗΤΕΣ]". The window contains a report with the following content:

**ΑΠΟΡΡΙΠΤΑΙΟΙ ΜΑΘΗΤΕΣ**



ΚΩΔΙΚΟΣ ΜΑΘΗΤΗ	ΟΝΟΜΑ	ΕΠΩΝΥΜΟ	ΤΑΞΗ	ΤΜΗΜΑ	ΒΑΘ. Α ΒΑΘ. Β ΒΑΘ. Γ			ΜΕΣΟΣ ΟΡΟΣ
					3ΜΗΝΟ	3ΜΗΝΟ	3ΜΗΝΟ	
E-095	ΕΡΑΝΤΑ-Β	ΟΜΕΡΙ	Α' ΛΥΚΕΙΟ	Α3	9	9	9	9
E-M86	ΕΛΕΝΗ-ΜΑ	ΜΑΥΡΟΚΕΦΑΛΟΥ	Β' ΛΥΚΕΙΟ	Β1	7	8	9	8
E-X-105	ΕΙΡΗΝΗ	ΧΑΡΜΠΟΥΡΟΥ	Β' ΛΥΚΕΙΟ	Β4	10	9	10	9,66666666
A-T-34	ΑΝΔΡΕΑΣ	ΤΣΟΥΜΑΝΗΣ	Γ' ΛΥΚΕΙΟΥ	Γ2	10	9	8	9
A-Z 10	ΑΝΔΡΕΑΣ	ΖΑΧΑΡΑΚΗΣ	Α' ΛΥΚΕΙΟ	Α1	8	6	9	7,66666666

At the bottom of the window, there is a navigation bar with "Σελίδα: 1" and a status bar with "Έτοιμο".

Σχήμα 6-8

Microsoft Access - [ΑΡΙΣΤΟΙ ΜΑΘΗΤΕΣ]

Αρχείο Επεξεργασία Προβολή Εργαλεία Παράθυρο Βοήθεια

90% Κλείσιμο

Ζουμ

## ΑΡΙΣΤΟΙ ΜΑΘΗΤΕΣ



ΚΩΔΙΚΟΣ ΜΑΘΗΤΗ	ΟΝΟΜΑ	ΕΠΙΘΥΜΟ	ΤΑΞΗ	ΤΜΗΜΑ	ΒΑΘΜΟΣ Α ΤΡΙΜΗΝΟΥ	ΒΑΘΜΟΣ Β ΤΡΙΜΗΝΟΥ	ΒΑΘΜΟΣ Γ ΤΡΙΜΗΝΟΥ	ΜΕΣΟΣ ΟΡΟΣ
M-P-178	ΜΑΡΙΑ	ΠΑΝΑΓΙΩΤ	Γ ΛΥΚΕΙΟ	Γ4	19	19	19	19
A-Z-8	ΑΛΙΚΗΣΤΗ	ΖΑΧΑΡΟΠ	Β ΛΥΚΕΙΟ	Β1	19	19	20	19,333333
A-M-21	ΑΘΗΝΑ	ΜΑΥΡΟΚΕ	Γ ΛΥΚΕΙΟ	Γ1	18	19	19	18,666667
M-T-181	ΜΑΡΙΑΝΝΑ	ΤΣΑΤΣΟΥ	Γ ΛΥΚΕΙΟ	Γ2	19	19	19	19
K-M-144	ΚΩΝΣΤΑΝΤ	ΜΠΑΙΑΗΣ	Γ ΛΥΚΕΙΟ	Γ2	19	19	19	19
X-G-246	ΧΡΗΣΤΟΣ	ΓΙΝΗ	Β ΛΥΚΕΙΟ	Β1	18	19	19	18,666667
A-Z-9	ΑΛΙΚΗΣΤΗ	ΖΗΣΗ	Γ ΛΥΚΕΙΟ	Γ1	19	19	20	19,333333
A-P-159	ΛΑΜΠΡΟΣ	ΡΑΠΤΗΣ	Α ΛΥΚΕΙΟ	Α3	18	19	19	18,666667

Σελίδα: 1

Έτοιμο

Σχήμα 6-9

## Βιβλιογραφία

---

Τίτλοι ξένων συγγραμμάτων που χρησιμοποιήθηκαν, κατά κύριο λόγο, στην εργασία μου είναι:

- International Organization for Standardization, Geneva, Switzerland. *Information technology --- Database languages --- SQL*, 1992. ISO/IEC 9075:1992.
- Peter Pin-Shan Chen. The entity-relationship model --- toward a unified view of data. *ACM Transactions on Database Systems*, 1(1):9-36, March 1976.
- Ron Soukup & Kalen Delaney, *Microsoft Press inside Microsoft SQL server*, 1999

Για την εργαστηριακή άσκηση και τα παραδείγματα που συνοδεύουν την εργασία μου, δεν χρησιμοποιήθηκε βιβλιογραφία

Συμβουλευτήκα, επίσης, άρθρα του περιοδικού RAM που σε κάποια σημεία της εργασίας επιτρέπουν να παραθέσω κάποιες κλασικές απόψεις για το ρόλο των βάσεων δεδομένων και των εφαρμογών τους. Επίσης, χρήσιμη στάθηκε και η τοποθεσία στο Internet, [www.in.gr](http://www.in.gr), από την οποία αναζήτησα πληροφορίες.

Τέλος, η ελληνική βιβλιογραφία που χρησιμοποίησα είναι η εξής:

- Αντωνιάδης Νίκος, *Βάσεις Δεδομένων*, ΤΕΙ ΗΠΕΙΡΟΥ, Τμήμα Τηλεπληροφορικής & Διοίκησης, 2002
- Γιαννακόπουλος Διον., Παπουτσή Ιωάν., *Πληροφοριακά Συστήματα Διοίκησης*, ΕΛΛΗΝ – Γ. Παρίκος & ΣΙΑ ΕΕ, 1996
- Δεσπότης Γιώργος, *Αναλυτικό Λεξικό Πληροφορικής & Ηλ. Υπολογιστών*, Data PIM ΕΠΕ, 1992
- Ομάδα QUE, *MICROSOFT SPECIAL EDITION – ΠΛΗΡΗΣ ΟΔΗΓΟΣ ΤΗΣ ACCESS*, Γκιούρδας, 2000
- ΟΕΔΒ, *Πληροφορική Γυμνασίου*, 2003
- Σκουρλάς Χρήστος, *Σχισιακές Βάσεις Δεδομένων*, Νέων Τεχνολογιών, 2000
- Παντελής Υψηλάντης, *Πληροφοριακά Συστήματα Διοίκησης*, Πατάκης, 2001
- Χρήστος Γουλτίδης, *Βάσεις Δεδομένων*, *Access 2002-ECDL*, Κλειδάριθμος, 2003
- Χρήστος Γουλτίδης, *Βάσεις Δεδομένων*, *Access 2002-ECDL ADVANCE*, Κλειδάριθμος, 2003
- Μ. Λεόντιος – Α. Γαβανά, *Βάσεις Δεδομένων*, *Access 2002-ECDL*, Β. Γκιούρδας Εκδοτική, 2003