

**ΤΕΧΝΟΛΟΓΙΚΟ ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ
ΜΕΣΟΛΟΓΓΙΟΥ**

ΣΧΟΛΗ ΔΙΟΙΚΗΣΗΣ ΚΑΙ ΟΙΚΟΝΟΜΙΑΣ

ΤΜΗΜΑ ΛΟΓΙΣΤΙΚΗΣ



ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

**ΑΝΑΛΥΣΗ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ ΠΟΥ
ΝΑ ΚΑΝΕΙ ΥΠΗΡΕΣΙΕΣ
ΤΑΥΤΟΠΟΙΗΣΗΣ ΠΡΟΣΩΠΩΝ ΚΑΙ
ΕΦΑΡΜΟΓΕΣ ΚΡΥΠΤΟΓΡΑΦΗΣΗΣ RSA**



**Άνδρεα Κωνσταντίνου ΑΜ (15397)
Γεώργιος Μαρίνης ΑΜ (14637)
Κωνσταντίνος Γιαννακόπουλος ΑΜ (14472)**

ΕΙΣΗΓΗΤΗΣ : ΚΟΜΠΟΘΡΕΚΑΣ ΑΡΙΣΤΟΤΕΛΗΣ

ΜΕΣΟΛΟΓΓΙ 2015



ΠΕΡΙΕΧΟΜΕΝΑ

ΕΙΣΑΓΩΓΗ	5
ΚΕΦΑΛΑΙΟ 1	6
ΔΕΔΟΜΕΝΑ ΚΑΙ ΠΛΗΡΟΦΟΡΙΕΣ	6
1.1 Ορισμός δεδομένων και πληροφορίας	6
1.2 Η Οργάνωση Αρχείων.....	7
1.3 Προβλήματα της Οργάνωσης Αρχείων.....	10
ΚΕΦΑΛΑΙΟ 2	12
ΟΙ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΤΑ ΣΔΒΔ (DBMS).....	12
2.1 Ορισμός βάσης δεδομένων.....	12
2.2 Σύστημα Διαχείρισης Βάσεων Δεδομένων	12
2.3 Διαφορά Βάσης Δεδομένων με Σύστημα Διαχείρισης Βάσεων Δεδομένων	14
2.4 Μέρη που έχει το Σύστημα Βάσης Δεδομένων (ΣΒΔ) ή DBS (Data Base System).....	15
2.5 Τι παρέχει ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων	16
2.6 Τι παρέχει η Βάση Δεδομένων.....	18
ΚΕΦΑΛΑΙΟ 3	20
Η ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΩΝ ΣΔΒΔ	20
3.1 Εισαγωγή.....	20
3.2 Οι Οντότητες (Entities)	21
3.3 Οι Ιδιότητες (Attributes)	21
3.4 Τα Στιγμιότυπα (Snapshots).....	22
3.5 Το Πρωτεύον Κλειδί (Primary Key)	22
3.6 Οι Συσχετίσεις (Relationships)	23
3.7 Το Ιεραρχικό Μοντέλο Βάσεων Δεδομένων.....	23
3.8 Το Δικτυωτό Μοντέλο Βάσεων Δεδομένων	24
3.9 Το Σχεσιακό Μοντέλο Βάσεων Δεδομένων	24
3.10 Τα Σχεσιακά ΣΔΒΔ (RDBMS)	26
3.11 Το Μοντέλο Οντοτήτων–Συσχετίσεων.....	27
3.11.1 Οι Οντότητες.....	28
3.11.2 Οι Ιδιότητες (Χαρακτηριστικά) των Οντοτήτων	29
3.11.3 Τα Κλειδιά	29

3.11.4	Οι Συσχετίσεις Μεταξύ Οντοτήτων.....	31
3.11.5	Οι Διμελείς Συσχετίσεις.....	32
3.11.6	Το Διάγραμμα Οντοτήτων Συσχετίσεων	33
ΚΕΦΑΛΑΙΟ 4	35
ΛΟΓΙΚΟΣ ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ	35
4.1	Εισαγωγή.....	35
4.2	Το αντικείμενο του λογικού σχεδιασμού	35
4.3	Ο λογικός σχεδιασμός.....	35
4.4	Κανονικοποίηση λογικού σχεδιασμού βάσης δεδομένων	36
ΚΕΦΑΛΑΙΟ 5	39
Η ΓΛΩΣΣΑ SQL	39
5.1	Εισαγωγή.....	39
5.2	Αναζήτηση/ανάκτηση Δεδομένων στην SQL.....	40
5.3	Διάταξη (Ταξινόμηση) Αποτελεσμάτων	41
5.4	Σύγκριση συμβολοσειρών.....	42
5.5	Η τιμή του NULL στην SQL.....	42
5.6	Ερωτήματα σύνδεσης πινάκων	43
5.7	Εσωτερική Άλγεβρα.....	43
ΚΕΦΑΛΑΙΟ 6	44
ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ - ACCESS	44
6.1	Εισαγωγή.....	44
6.2	Εκκίνηση της Access.....	45
6.3	Δημιουργία πίνακα- τύποι δεδομένων	46
6.4	Γενικές ιδιότητες δεδομένων.....	47
6.4.1	Ευρετήρια.....	47
6.4.2	Μορφοποίηση Δεδομένων	48
6.4.3	Επικύρωση Δεδομένων.....	48
6.4.4	Μάσκα Εισαγωγής.....	49
6.5	Η ιδιότητα Πεδίου Εμφάνιση.....	49
6.6	Πρωτεύον κλειδί.....	49
6.7	Δηλώσεις Συσχετίσεων	50
ΚΕΦΑΛΑΙΟ 7	53
ΠΡΑΚΤΙΚΗ ΕΦΑΡΜΟΓΗ ΔΗΜΙΟΥΡΓΙΑΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ	53

7.1	Εισαγωγή.....	53
7.2	Σκοπός – στόχος πρακτικής εφαρμογής.....	53
7.3	Σχεδιασμός βάσης δεδομένων.....	53
7.4	Παρουσίαση βάσης δεδομένων.....	54
7.4.1.	Δημιουργία πινάκων	54
7.4.2.	Αναφορές	60
7.4.3.	Φόρμες	63
7.4.4.	Σχέσεις των πινάκων.....	70
7.4.5.	Πρωτεύον κλειδί	71
7.5.	Κρυπτογράφηση και αποκρυπτογράφηση Βάσης Δεδομένων.....	73
7.6.	Ασφάλεια σε επίπεδο χρηστών	77
	ΣΥΜΠΕΡΑΣΜΑΤΑ	82
	ΒΙΒΛΙΟΓΡΑΦΙΑ	84

ΕΙΣΑΓΩΓΗ

Η ραγδαία ανάπτυξη των τελευταίων χρόνων στον επιστημονικό χώρο και στο χώρο της πληροφορικής και των επικοινωνιών έχει κάνει την πληροφορία ένα από τα πολυτιμότερα αγαθά. Είναι επιθυμητή απ' όλους τους εργαζόμενους αλλά και τους εκπαιδευόμενους. Βοηθά στην βελτίωση της απόδοσης, της ανταγωνιστικότητας αλλά και παραγωγικότητάς τους στην εργασία.

Οι βάσεις δεδομένων χρησιμοποιούνται ώστε να αποθηκεύουμε, να επεξεργαζόμαστε αλλά και να εκμεταλλευόμαστε αποδοτικά αυτόν τον τεράστιο όγκο των πληροφοριών που αυξάνονται διαρκώς.

Εξαιτίας του ότι η χρήση των ηλεκτρονικών υπολογιστών και κατά συνέπεια η ηλεκτρονική καταχώρηση και επεξεργασία δεδομένων αυξήθηκε από τη δεκαετία του '70 και μετά στις μεγάλες επιχειρήσεις είχαμε πάρα πολλές εφαρμογές να επεξεργάζονται δεδομένα σε πάρα πολλά αρχεία ταυτόχρονα. Γι' αυτό προτάθηκε η συνένωση όλων των αρχείων μιας εφαρμογής. Εκτός, όμως, από τη συνένωση των αρχείων, απαραίτητη ήταν και η σωστή οργάνωσή τους. Δημιουργήθηκαν έτσι οι Τράπεζες Πληροφοριών ή Βάσεις Δεδομένων (Data Bases).

ΚΕΦΑΛΑΙΟ 1

ΔΕΔΟΜΕΝΑ ΚΑΙ ΠΛΗΡΟΦΟΡΙΕΣ

1.1 Ορισμός δεδομένων και πληροφορίας

Όταν χρησιμοποιούμε τον όρο **πληροφορία** αναφερόμαστε συνήθως σε ειδήσεις, γεγονότα και έννοιες που αποκτάμε από την καθημερινή επικοινωνία και τα θεωρούμε ως επίκτητη γνώση. Ενώ τα δεδομένα μπορούν να είναι μη επεξεργασμένα και μη ταξινομημένα καταλλήλως σύνολα πληροφοριών. Ένας αυστηρός ορισμός για το τι είναι δεδομένα και τι είναι πληροφορία, σύμφωνα με την επιτροπή ANSI των ΗΠΑ, είναι ο εξής :

- **Δεδομένα (data)** είναι μια παράσταση, όπως γράμματα, αριθμοί, σύμβολα κ.ά. στα οποία μπορούμε να δώσουμε κάποια σημασία (έννοια).
- **Πληροφορία (information)** είναι η σημασία που δίνουμε σ' ένα σύνολο από δεδομένα, τα οποία μπορούμε να επεξεργαστούμε βάσει προκαθορισμένων κανόνων και να βγάλουμε έτσι κάποια χρήσιμα συμπεράσματα. Με τις πληροφορίες περιορίζεται η αβεβαιότητα που έχουμε για διάφορα πράγματα και βοηθιόμαστε έτσι στο να λάβουμε σωστές αποφάσεις.

Τα δεδομένα μπορούν να χαρακτηρισθούν ως τρόποι αναπαράστασης εννοιών και γεγονότων που έχουμε τη δυνατότητα να τροποποιήσουμε και να επεξεργαστούμε. Οι παροντικές κοινωνικές συνθήκες απαιτούν συλλογή και αποθήκευση ενός τεράστιου όγκου δεδομένων. Ωστόσο αποτρεπτικό παράγοντα αποτελεί το πρόβλημα της σωστής οργάνωσης και ταξινόμησης των δεδομένων. Τα δεδομένα θα πρέπει να οργανωθούν με έτσι ώστε να μπορούμε να τα εντοπίζουμε και να τα χρησιμοποιούμε εύκολα και γρήγορα και τη στιγμή που τα χρειαζόμαστε.

Ένα κλασικό παράδειγμα μη σωστής οργάνωσης δεδομένων θα ήταν για παράδειγμα ο τηλεφωνικός κατάλογος της πόλης της Θεσσαλονίκης. Στην προκειμένη περίπτωση οι συνδρομητές δεν θα ήταν καταχωρημένοι αλφαβητικά σύμφωνα με το επώνυμο και

το όνομά τους, αλλά εντελώς τυχαία. Ένας τέτοιος τηλεφωνικός κατάλογος θα περιείχε μια τεράστια ποσότητα δεδομένων αλλά θα ήταν ουσιαστικά άχρηστος.

Ένα άλλο παράδειγμα μη σωστής οργάνωσης δεδομένων θα ήταν μια πολύ μεγάλη βιβλιοθήκη με χιλιάδες τόμους βιβλίων στην οποία δεν θα τηρούταν κανένα σύστημα οργάνωσης και ταξινόμησης των βιβλίων. Τόσο οι υπάλληλοι της βιβλιοθήκης όσο και οι επισκέπτες δεν θα μπορούσαν να αξιοποιήσουν την πληθώρα των πληροφοριών που περιέχονται στα βιβλία. Η μόνιμη αποθήκευση των δεδομένων δεν είναι χρήσιμη αν δεν μπορούμε να χειριζόμαστε τις πληροφορίες που παρέχει ευέλικτα και αποδοτικά.¹

Χαρακτηριστικά παραδείγματα δεδομένων που απαιτούν σωστή και αποδοτική οργάνωση είναι τα εξής :

- Τα στοιχεία υπαλλήλων, πελατών, προμηθευτών και παραγγελιών μιας εμπορικής επιχείρησης.
- Τα στοιχεία υλικών μιας αποθήκης.
- Τα στοιχεία ταινιών, πελατών και δανεισμών μιας βιντεολέσχης.
- Τα στοιχεία υπαλλήλων, γιατρών, ασθενών αλλά και υλικών ενός νοσοκομείου.
- Τα στοιχεία βιβλίων, χρηστών (δανειστών) και δανεισμών μιας βιβλιοθήκης.

1.2 Η Οργάνωση Αρχείων

Ο πιο διαδεδομένος τρόπος οργάνωσης δεδομένων είναι σε αρχεία εγγραφών με τη χρήση ηλεκτρονικών υπολογιστών. Για να κατανοήσουμε καλύτερα ορισμένες έννοιες, θα εξετάσουμε την περίπτωση ενός αρχείου πελατών και παραγγελιών μιας εμπορικής επιχείρησης. Για να οργανώσουμε σωστά το αρχείο μας, θα πρέπει να δημιουργήσουμε καρτέλες για τους πελάτες, αλλά και για τις παραγγελίες τους αργότερα, που θα πρέπει να περιέχουν τα εξής στοιχεία ανά πελάτη:

¹ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

- Κωδικός
- Επώνυμο
- Όνομα
- Διεύθυνση
- ΤΚ
- Πόλη
- Τηλέφωνο
- ΑΦΜ
- ΔΟΥ

Η αντιστοίχιση του παλιού τρόπου οργάνωσης με τις καρτέλες σε σχέση με τον σύγχρονο ηλεκτρονικό τρόπο οργάνωσης, έχει ως εξής :

- Συρτάρι – Αρχείο Δεδομένων
- Καρτέλα πελάτη – Εγγραφή του αρχείου δεδομένων
- Στοιχείο της καρτέλας – Πεδίο της εγγραφής

Ένα **αρχείο (file)** θα μπορούμε να το χαρακτηρίσουμε σαν ένα σύνολο που αποτελείται από οργανωμένα στοιχεία παρεμφερούς περιεχομένου. Τα στοιχεία ενός αρχείου μπορούμε να τα οργανώσουμε σε λογικές ενότητες και το σύνολο των στοιχείων που περιέχει μια λογική ενότητα καλείται **εγγραφή (record)**.²

Το κάθε στοιχείο της εγγραφής καλείται **πεδίο (field)**. Το πεδίο αποτελεί και τη μικρότερη δυνατή υποδιαίρεση των στοιχείων ενός αρχείου. Ένα πεδίο χαρακτηρίζεται από τον μέγιστο αριθμό των χαρακτήρων (bytes) που απαιτούνται για την καταχώρησή του στη μνήμη του υπολογιστή και που αποκαλείται **μήκος του πεδίου (field length)**.

Σε μια οργάνωση αρχείου όπως είναι οι πελάτες μιας εμπορικής επιχείρησης που είδαμε νωρίτερα, τα αντίστοιχα πεδία όλων των εγγραφών καταλαμβάνουν έναν προκαθορισμένο από τον χρήστη αριθμό σε bytes. Για παράδειγμα, αν ορίσαμε ότι το πεδίο Επώνυμο θα έχει μήκος 15 χαρακτήρες, τότε το πεδίο της εγγραφής του πελάτη

² (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

με επώνυμο Παπαδόπουλος, αλλά και το πεδίο της εγγραφής του πελάτη με επώνυμο Βεζ θα καταλαμβάνουν από 15 bytes στη μνήμη του υπολογιστή, ενώ αν ένας πελάτης ονομάζεται Παπαχριστοδουλόπουλος, τότε θα γίνει αποκοπή του επωνύμου του και θα καταχωρηθούν στη μνήμη του υπολογιστή μόνο τα 15 πρώτα γράμματα, δηλ. τα Παπαχριστοδουλό.

Ένα πεδίο χαρακτηρίζεται ακόμη και από το είδος των δεδομένων που περιέχει, όπως :

- **Αλφαριθμητικό (alphanumeric)**, μπορεί να περιέχει γράμματα, ψηφία ή και ειδικούς χαρακτήρες.
- **Αριθμητικό (numeric)**, μπορεί να περιέχει μόνο αριθμούς.
- **Αλφαβητικό (alphabetic)**, μπορεί να περιέχει μόνο γράμματα (αλφαβητικούς χαρακτήρες).
- **Ημερομηνίας (date)**, μπορεί να περιέχει μόνο ημερομηνίες.
- **Διαδικό (binary)**, μπορεί να περιέχει ειδικού τύπου δεδομένα, όπως εικόνες, ήχους κ.ά.
- **Λογικό (logical)**, μπορεί να περιέχει μόνο μία από δύο τιμές, οι οποίες αντιστοιχούν σε δύο διακριτές καταστάσεις και μπορούν να χαρακτηρισθούν σαν 0 και 1 ή σαν αληθές (true) και ψευδές (false).
- **Σημειώσεων (memo)**, μπορεί να περιέχει κείμενο με μεταβλητό μήκος, το οποίο μπορεί να είναι και αρκετά μεγάλο και συνήθως αποθηκεύεται σαν ξεχωριστό αρχείο από το κύριο αρχείο.³

Όσον αφορά τις εγγραφές, χρήσιμοι είναι οι εξής ορισμοί: **Μήκος εγγραφής (record length)** χαρακτηρίζεται το άθροισμα που προκύπτει από τα μήκη των πεδίων που την αποτελούν.

- **Δομή εγγραφής (record layout) ή γραμμογράφηση** καλείται ο τρόπος οργάνωσης των πεδίων μιας εγγραφής.
- **Διάβασμα (read)** από αρχείο είναι η μεταφορά των δεδομένων του αρχείου, από το μέσο αποθήκευσης (σκληρό δίσκο ή δισκέτα) στην

³ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

κεντρική μνήμη του υπολογιστή για επεξεργασία. Αυτό γίνεται συνήθως ανά μία εγγραφή.

- **Γράψιμο (write)** σε αρχείο σημαίνει μεταφορά των δεδομένων του αρχείου, από την κεντρική μνήμη του υπολογιστή στο μέσο αποθήκευσης (σκληρό δίσκο ή δισκέτα). Αυτό γίνεται συνήθως ανά μία εγγραφή.⁴

1.3 Προβλήματα της Οργάνωσης Αρχείων

Στα πρώτα στάδια της οργάνωσης αρχείων, ήταν πολύ συνηθισμένη πρακτική το να δημιουργούνται ξεχωριστές εφαρμογές (προγράμματα) και ξεχωριστά αρχεία, (για παράδειγμα η δημιουργία ενός αρχείου πελατών και ενός άλλου ανεξάρτητου αρχείου για τις παραγγελίες των πελατών). Τα προβλήματα που προέκυψαν από την πρακτική αυτή είναι τα εξής :

- **Πλεονασμός των δεδομένων (data redundancy).** Υπάρχει η περίπτωση να συναντήσουμε τα ίδια δεδομένα σε αρχεία διαφορετικών εφαρμογών. Για παράδειγμα, αν έχουμε ένα αρχείο πελατών και ένα αρχείο παραγγελιών αυτών των πελατών, είναι σχεδόν σίγουρο ότι θα υπάρχουν κάποια στοιχεία των πελατών που θα υπάρχουν και στα δύο αρχεία.
- **Ασυνέπεια των δεδομένων (data inconsistency).** Αυτό μπορεί να συμβεί όταν υπάρχουν τα ίδια στοιχεία πελατών (πλεονασμός) και στο αρχείο πελατών και στο αρχείο παραγγελιών. Σε περίπτωση που χρειασθεί να γίνει κάποια αλλαγή στη διεύθυνση ή στα τηλέφωνα κάποιου πελάτη, είναι πολύ πιθανό η διόρθωση να γίνει μόνο στο ένα αρχείο και όχι και σε όλα.
- **Αδυναμία μερισμού δεδομένων (data sharing).** Μερисμός δεδομένων σημαίνει δυνατότητα για κοινή χρήση των στοιχείων κάποιων αρχείων. Ο μερισμός δεδομένων θα ήταν χρήσιμος αν για παράδειγμα με την παραγγελία ενός πελάτη μπορούμε να δούμε ταυτόχρονα στο αρχείο πελατών το υπόλοιπο του πελάτη και μετά στο αρχείο της αποθήκης αν είναι διαθέσιμα τα προϊόντα που παρήγγειλε ο συγκεκριμένος πελάτης. Η αδυναμία μερισμού δεδομένων δημιουργεί

⁴ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

καθυστέρηση στη λήψη αποφάσεων και στην εξυπηρέτηση των χρηστών.

- **Αδυναμία προτυποποίησης.** Σχετίζεται με την ανομοιομορφία και με την διαφορετική αναπαράσταση και οργάνωση των δεδομένων στα αρχεία των εφαρμογών. Έτσι δημιουργούνται προβλήματα προσαρμογής των χρηστών καθώς και προβλήματα στην ανταλλαγή δεδομένων μεταξύ διαφορετικών συστημάτων.⁵

⁵ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

ΚΕΦΑΛΑΙΟ 2

ΟΙ ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ ΚΑΙ ΤΑ ΣΔΒΔ (DBMS)

2.1 Ορισμός βάσης δεδομένων

Ένα βασικό χαρακτηριστικό των δεδομένων, είναι ότι είναι κατάλληλα για να αποθηκευτούν σε ηλεκτρονικό υπολογιστή. Αυτό οδηγεί στο επόμενο βήμα που είναι η δημιουργία και χρήση Βάσεων Δεδομένων (databases).

Η σχετικότητα των δεδομένων είναι σημαντική ώστε να οριστεί μια Βάση Δεδομένων. Δεδομένα που απλά έχουν αποθηκευτεί σε έναν ηλεκτρονικό υπολογιστή μη συσχετιζόμενα μεταξύ τους δεν αποτελούν μια Βάση Δεδομένων. Μια Βάση Δεδομένων πρέπει να αντικατοπτρίζει ένα περιβάλλον του πραγματικού κόσμου.

Τα δεδομένα της πρέπει να έχουν λογική συνέχεια και νόημα. Επίσης οι λειτουργίες που παρέχει η Βάση Δεδομένων είναι σημαντικές για τον ορισμό της. Συλλογές δεδομένων που δεν χρήζουν δυνατότητας ενός αυτόματου, κοινού και κεντρικού τρόπου χειρισμού των δεδομένων αυτών, δεν αποτελούν Βάση Δεδομένων.

Η Βάση Δεδομένων έχει ένα σκοπό. Υλοποιείται για να απεικονίσει στον υπολογιστή ένα πρόβλημα και να διευκολύνει την επίλυσή του. Αυτό συνήθως σημαίνει ότι τα δεδομένα που χειρίζεται είναι δυναμικά, δηλαδή τροποποιούνται διαρκώς. Η Βάση Δεδομένων περιέχει τα δεδομένα τα οποία περιγράφουν τη δεδομένη χρονική στιγμή το «πρόβλημα», μπορεί όμως να περιέχει και δεδομένα από την ιστορία του προβλήματος.⁶

2.2 Σύστημα Διαχείρισης Βάσεων Δεδομένων

Η εξέλιξη των Βάσεων Δεδομένων και οι ανάγκες για δημιουργία όλο και περισσότερων Βάσεων Δεδομένων, οδήγησαν στη δημιουργία των

⁶ Γιωτόπουλος Σ, Σημειώσεις Μαθήματος Βάσεις Δεδομένων, ΑΤΕΙ ΠΑΤΡΑΣ, Δεκέμβριος 2010

Συστημάτων Διαχείρισης Βάσεων Δεδομένων (Database Management Systems ή DBMS). Το Σύστημα Διαχείρισης Βάσεων Δεδομένων είναι ένα εργαλείο που διευκολύνει την εργασία με Βάσεις Δεδομένων. Με τη χρήση του Συστήματος Διαχείρισης Βάσεων Δεδομένων οι χρήστες έχουν τη δυνατότητα να κατασκευάσουν και να χρησιμοποιήσουν Βάσεις Δεδομένων. Ακολουθεί ο ορισμός του Συστήματος Διαχείρισης Βάσεων Δεδομένων.

Ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων κατά κανόνα «φιλοξενεί» πολλές Βάσεις Δεδομένων που έχουν κατασκευαστεί από διαφορετικούς χρήστες. Οι δυνατότητες που παρέχει ένα Σύστημα χρήστες συνοψίζονται στις παρακάτω:

- Ορισμός της Βάσης Δεδομένων
- Κατασκευή της Βάσης Δεδομένων
- Διαγραφή της Βάσης Δεδομένων
- Χρήση της Βάσης Δεδομένων

Στον ορισμό της Βάσης Δεδομένων ο χρήστης μπορεί να καθορίσει το μοντέλο της Βάσης Δεδομένων, να ορίσει τους τύπους δεδομένων που θα χρησιμοποιήσει και να ελέγξει τη Βάση Δεδομένων χωρίς να προχωρήσει σε υλοποίησή της. Στην κατασκευή της Βάσης Δεδομένων οι τύποι των δεδομένων και τα δεδομένα αποθηκεύονται στο υλικό (hardware) του ηλεκτρονικού υπολογιστή μέσω διαδικασιών που ελέγχονται από το Σύστημα Διαχείρισης Βάσεων Δεδομένων και δεν απασχολούν τον χρήστη. Στη διαγραφή της Βάσης Δεδομένων ο χρήστης διαγράφει και απομακρύνει μια Βάση Δεδομένων ή μεμονομένα αυτής δεδομένα από το υλικό. Τέλος, κατά τη χρήση της Βάσης Δεδομένων ο χρήστης, ή χειρίζεται τα δεδομένα (εισάγει νέα δεδομένα, τροποποιεί δεδομένα, ή διαγράφει δεδομένα), ή υποβάλει ερωτήσεις στη Βάση Δεδομένων για εξαγωγή πληροφοριών.⁷

Οι χρήστες των εφαρμογών παίρνουν τα στοιχεία που θέλουν από τη βάση δεδομένων χωρίς δύνανται να γνωρίζουν με ποιο τρόπο είναι οργανωμένα τα δεδομένα σ' αυτήν. Το ΣΔΒΔ παίζει τον ρόλο του μεσολαβητή ανάμεσα στον χρήστη και τη βάση

⁷ Γιωτόπουλος Σ, Σημειώσεις Μαθήματος Βάσεις Δεδομένων, ΑΤΕΙ ΠΑΤΡΑΣ, Δεκέμβριος 2010

δεδομένων. Μόνο μέσω του ΣΔΒΔ μπορεί ο χρήστης να αντλήσει πληροφορίες από τη βάση δεδομένων. Ένα ΣΔΒΔ μπορεί να είναι εγκατεστημένο σ' έναν μόνο υπολογιστή ή και σ' ένα δίκτυο υπολογιστών και μπορεί να χρησιμοποιείται από έναν ή από πολλούς χρήστες.

2.3 Διαφορά Βάσης Δεδομένων με Σύστημα Διαχείρισης Βάσεων Δεδομένων

Ως εδώ πρέπει να είναι ξεκάθαρη σε εσάς η διαφορά μεταξύ ενός Συστήματος Διαχείρισης Βάσεων Δεδομένων και μίας Βάσης Δεδομένων. Το Σύστημα Διαχείρισης Βάσεων Δεδομένων είναι το λογισμικό το οποίο διευκολύνει τους χρήστες να υλοποιήσουν Βάσεις Δεδομένων. Αντίθετα η Βάση Δεδομένων κατασκευάζεται με τη βοήθεια του Συστήματος Διαχείρισης Βάσεων Δεδομένων που χειρίζεται τα δεδομένα που αποθηκεύονται στο υλικό του υπολογιστή. Όλη η διαδικασία χειρισμού και αποθήκευσης δεδομένων διευκολύνεται από το Σύστημα Διαχείρισης Βάσεων Δεδομένων, που αναλαμβάνει τη τροποποίηση των εντολών του χρήστη σε εντολές προς τον υπολογιστή και λειτουργίες που έχουν σχέση με το χειρισμό των δεδομένων στο υλικό του υπολογιστή.

Ο χρήστης μιας Βάσης Δεδομένων επιτελεί λειτουργίες που δύνανται να κατηγοριοποιηθούν στις παρακάτω: εισαγωγή δεδομένων, διαγραφή δεδομένων, αλλαγή δεδομένων και ανάκληση δεδομένων. Οι παραπάνω λειτουργίες προϋποθέτουν επικοινωνία με το υλικό του υπολογιστή (π.χ. το μαγνητικό δίσκο του υπολογιστή), η οποία πραγματοποιείται μέσω του λογισμικού του Συστήματος Διαχείρισης Βάσεων Δεδομένων. Η επικοινωνία αυτή δεν είναι συνήθως ορατή από τον χρήστη της Βάσης Δεδομένων. Ο τελευταίος απλά βλέπει τα αποτελέσματα από τις λειτουργίες που ζήτησε να επιτελούνται. Αυτό είναι και το σημείο που κάνει αρκετούς χρήστες να συγχέουν τις έννοιες Βάση Δεδομένων και Σύστημα Διαχείρισης Βάσεων Δεδομένων.⁸

⁸ Γιωτόπουλος Σ, Σημειώσεις Μαθήματος Βάσεις Δεδομένων, ΑΤΕΙ ΠΑΤΡΑΣ, Δεκέμβριος 2010

2.4 Μέρη που έχει το Σύστημα Βάσης Δεδομένων (ΣΒΔ) ή DBS (Data Base System)

Ένα Σύστημα Βάσης Δεδομένων (ΣΒΔ) ή DBS (Data Base System) περιλαμβάνει το υλικό, το λογισμικό, τη βάση δεδομένων και τους χρήστες. Πρόκειται για ένα σύστημα στο οποίο μπορούμε να αποθηκεύσουμε και να αξιοποιήσουμε δεδομένα με τη βοήθεια ηλεκτρονικού υπολογιστή. Αναλυτικά :

- Το **υλικό (hardware)** αποτελείται από τους ηλεκτρονικούς υπολογιστές, τα περιφερειακά, τους σκληρούς δίσκους, τις μαγνητικές ταινίες κ.ά., όπου είναι αποθηκευμένα τα αρχεία της βάσης δεδομένων αλλά και τα προγράμματα που χρησιμοποιούνται για την επεξεργασία τους.
- Το **λογισμικό (software)** είναι τα προγράμματα που χρησιμοποιούνται για την επεξεργασία των δεδομένων (στοιχείων) της βάσης δεδομένων.
- Η **βάση δεδομένων (data base)** αποτελείται από το σύνολο των αρχείων όπου είναι αποθηκευμένα τα δεδομένα του συστήματος, που μπορεί να βρίσκονται αποθηκευμένα σ' έναν φυσικό υπολογιστή αλλά και σε περισσότερους. Παρ'όλα αυτά ο χρήστης θεωρεί πως ίσως βρίσκονται συγκεντρωμένα στον ίδιο υπολογιστή. Τα δεδομένα των αρχείων αυτών είναι **ενοποιημένα (data integration)**, δηλ. δεν υπάρχει πλεονασμός (άσκοπη επανάληψη) δεδομένων και **μερισμένα (data sharing)**, δηλ. υπάρχει δυνατότητα προσπέλασης των δεδομένων από πολλούς χρήστες την ίδια χρονική στιγμή. Ο κάθε χρήστης έχει δικαιώματα διαφορετικά, καθώς ανάλογα με τον σκοπό για τον οποίο συνδέεται βλέπει και διαφορετικό κομμάτι της βάσης δεδομένων.
- Οι **χρήστες (users)** μιας βάσης δεδομένων χωρίζονται στις εξής κατηγορίες :
 - **Τελικοί χρήστες (end users)**. Χρησιμοποιούν κάποια εφαρμογή για να αντλούν στοιχεία από μια βάση δεδομένων. Έχουν τις λιγότερες δυνατότητες επέμβασης στα στοιχεία της βάσης δεδομένων. Τέλος, χρησιμοποιούν ειδικούς κωδικούς πρόσβασης και το σύστημα τους επιτρέπει την απαιτούμενη

και ανάλογη πρόσβαση σε συγκεκριμένο κομμάτι της βάσης δεδομένων.

- **Προγραμματιστές εφαρμογών (application programmers).** Αναπτύσσουν τις εφαρμογές του ΣΒΔ σε κάποια από τις γνωστές γλώσσες προγραμματισμού.
- **Διαχειριστής δεδομένων (data administrator – DA).** Έχει διοικητική αρμοδιότητα και ευθύνη για την οργάνωση της βάσης δεδομένων και την απόδοση δικαιωμάτων πρόσβασης στους χρήστες.
- **Διαχειριστής βάσης δεδομένων (database administrator – DBA).** Λαμβάνει οδηγίες από τον διαχειριστή δεδομένων και είναι αυτός που διαθέτει γνώσεις τεχνικές και αρμοδιότητες για τη ορθή και αποδοτική λειτουργία του ΣΔΒΔ.⁹

2.5 Τι παρέχει ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων

Μία Βάση Δεδομένων μπορεί να υλοποιηθεί και αυτόνομα χωρίς να χρησιμοποιηθεί ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων. Όμως κατά κανόνα οι παροχές ενός Συστήματος Διαχείρισης Βάσεων Δεδομένων είναι αυτές που τελικά ωθούν τους χρήστες να το χρησιμοποιήσουν. Στην ενότητα αυτή θα περιγράψουμε τι παρέχει ένα Σύστημα Διαχείρισης Βάσεων Δεδομένων.

Ευκολία στη Σχεδίαση και στην Υλοποίηση

Όπως ήδη γνωρίζετε ένας χρήστης μίας Βάσης Δεδομένων επιθυμεί πρώτα να σχεδιάσει τη Βάση Δεδομένων και να καθορίσει τους τύπους δεδομένων που θα χρησιμοποιεί. Έπειτα θα δημιουργήσει τη Βάση Δεδομένων και στη συνέχεια θα χρησιμοποιεί τη Βάση Δεδομένων ή θα επιτρέπει και σε άλλους χρήστες να τη χρησιμοποιούν ταυτόχρονα. Όλα τα παραπάνω παρέχονται από το Σύστημα Διαχείρισης Βάσεων Δεδομένων. Επίσης το σύστημα παρέχει στον χρήστη ένα ενιαίο τρόπο για να ορίζει τις Βάσεις Δεδομένων του και μια φιλοσοφία που

⁹ Γιωτόπουλος Σ, Σημειώσεις Μαθήματος Βάσεις Δεδομένων, ΑΤΕΙ ΠΑΤΡΑΣ, Δεκέμβριος 2010

καθοδηγεί τις ενέργειές του.¹⁰

Επίπεδα Χρηστών και Έλεγχος Πρόσβασης

Επειδή μία Βάση Δεδομένων μπορεί να χρησιμοποιείται από πολλούς χρήστες με διαφορετικές ανάγκες και γνώσεις, το Σύστημα Διαχείρισης Βάσεων Δεδομένων δύναται να ορίζει διαφορετικές κατηγορίες χρηστών. Για κάθε κατηγορία χρηστών ορίζονται όρια εξουσιοδότησης που διαθέτονται σε κάθε επίπεδο. Κάποια δεδομένα είναι πιθανόν να είναι εμπιστευτικά έτσι δεν επιτρέπεται να έχουν πρόσβαση σε αυτά όλοι οι χρήστες. Επίσης, κάθε κατηγορία χρηστών πρέπει να έχει το δικαίωμα να επιτελεί διαφορετικές λειτουργίες από τις άλλες κατηγορίες. Κάποιοι χρήστες για παράδειγμα θα μπορούν μόνο να διαβάζουν δεδομένα. Αντίθετα άλλοι θα μπορούν να εισάγουν ή επεξεργάζονται και να τροποποιούν τα δεδομένα.

Προστασία από Βλάβες Υλικού

Το Σύστημα Διαχείρισης Βάσεων Δεδομένων οφείλει να τηρεί αντίγραφα ασφαλείας καθώς και δυνατότητες ανάληψης της Βάσης Δεδομένων μετά από βλάβες. Έτσι εξασφαλίζεται καλύτερη προστασία των δεδομένων. Η ανάληψη (ή ανάνηψη) των δεδομένων είναι η διαδικασία κατά την οποία το Σύστημα Διαχείρισης Βάσεων Δεδομένων είτε αυτόματα (χωρίς τη συμβολή του διαχειριστή) είτε χειροκίνητα (με τη συμβολή του διαχειριστή) ανακτά τα δεδομένα που χάθηκαν σε περίπτωση βλάβης.

Δυνατότητα Ελέγχου Πλεονασμών

Το Σύστημα Διαχείρισης Βάσεων Δεδομένων υλοποιεί και ελέγχους για πλεονασμό δεδομένων. Αναφορικά με το Σύστημα Διαχείρισης Βάσεων Δεδομένων οι πλεονασμοί ελέγχονται αυτόματα από το σύστημα και η συμβατότητα των δεδομένων προστατεύεται από το ίδιο το Σύστημα Διαχείρισης

¹⁰ Γιωτόπουλος Σ, Σημειώσεις Μαθήματος Βάσεις Δεδομένων, ΑΤΕΙ ΠΑΤΡΑΣ, Δεκέμβριος 2010

Βάσεων Δεδομένων διευκολύνοντας την εργασία του χρήστη.¹¹

2.6 Τι παρέχει η Βάση Δεδομένων

Ταυτόχρονη Προσπέλαση

Βασικό όφελος από την χρήση Βάσης Δεδομένων είναι η δυνατότητα παροχής σε πολλούς χρήστες ταυτόχρονης προσπέλασης των δεδομένων, αλλά και η αυτοματοποίηση των ελέγχων που συνεπάγεται αυτή η προσπέλαση. Για παράδειγμα, το Σύστημα Διαχείρισης της Βάσης Δεδομένων φροντίζει ώστε να αποκλείει περιπτώσεις ταυτόχρονης μετροπής ενός δεδομένου από δύο ή περισσότερους χρήστες.

Επίσης, το Σύστημα Διαχείρισης της Βάσης Δεδομένων φροντίζει να ελέγχει εξουσιοδοτημένες ή όχη προσπελάσεις δεδομένων, παρέχοντας σε κάθε χρήστη μόνο αυτά για τα οποία έχει εξουσιοδότηση, και επιτρέποντάς του μόνο τις πράξεις (π.χ. διαγραφή, ή αλλαγή) για τις οποίες έχει αρμοδιότητα.

Ταχύτατη εξαγωγή απαντήσεων

Ένα μεγάλο πρόβλημα της οργάνωσης ενός μοντέλου του πραγματικού κόσμου χωρίς τη χρήση Βάσης Δεδομένων, είναι το γεγονός ότι δεν μπορεί να αποφευχθεί η εργασία από τον άνθρωπο για να αντληθεί η απαιτούμενη πληροφορία, προκειμένου να εξαχθούν απαντήσεις σε απλές ερωτήσεις. Ένα σύστημα αρχείων οργανώνει την πληροφορία σειριακά σε αρχεία, χωρίς τη δυνατότητα αναζήτησης παρά μόνο σειριακά (δηλαδή ανάγνωση ενός τμήματος του αρχείου στη μνήμη, έλεγχος και μετά ανάγνωση του επόμενου τμήματος, κτλ).

Σε ένα παραδοσιακό σύστημα αρχείων σε ηλεκτρονικό υπολογιστή η απάντηση στην ερώτηση «Δώσε μας την πόλη που είχε την υψηλότερη θερμοκρασία για κάθε ημέρα από το 1990 μέχρι σήμερα», θα σηματοδοτούσε άνοιγμα εκατοντάδων αρχείων και ανθρώπινη εργασία για την καταγραφή και σύγκριση δεδομένων. Ακριβώς επειδή τα δεδομένα συνήθως συλλέγονται για την απόκτηση

¹¹ Γιωτόπουλος Σ, Σημειώσεις Μαθήματος Βάσεις Δεδομένων, ΑΤΕΙ ΠΑΤΡΑΣ, Δεκέμβριος 2010

πληροφορίας με την μορφή απαντήσεων σε τέτοιου είδους ερωτήσεις, η χρήση παραδοσιακών μοντέλων οργάνωσης δεν μας εξυπηρετεί, αφού το να παρέχονται τέτοιου είδους πληροφορίες είναι πρακτικά αδύνατο. Ο λόγος είναι ότι ο τρόπος με τον οποίο έχουν οργανωθεί τα δεδομένα δεν υποστηρίζει την αυτόματη εξαγωγή τέτοιους είδους απαντήσεων σε ερωτήσεις.

Αντίθετα, με τη χρήση Βάσης Δεδομένων τέτοιου είδους πληροφορίες μπορούν να είναι διαθέσιμες στο χρήστη πολύ γρήγορα. Ο τρόπος οργάνωσης των δεδομένων δύναται οι Βάσεις Δεδομένων να μπορούν απαντούν αυτόματα και ταχύτατα τόσο σε πολύπλοκες, όσο και σε απλές ερωτήσεις, οι οποίες όμως θα ήταν δύσκολο να απαντηθούν εάν είχαν οργανωθεί σύμφωνα μ'ένα παραδοσιακό σύστημα (είτε με χειρόγραφα, είτε με μη οργανωμένα σε Βάση Δεδομένων αρχεία στο ηλεκτρονικό υπολογιστή).

Ευελιξία

Πολλές φορές το μοντέλο του πραγματικού κόσμου αλλάζει, γεγονός που κατά κανόνα απαιτεί αλλαγές οι οποίες τις περισσότερες φορές είναι πολύ δύσκολο να πραγματοποιηθούν σε ένα παραδοσιακό μοντέλο. Από την άλλη, ο τρόπος οργάνωσης των δεδομένων σε μία Βάση Δεδομένων επιδέχεται αλλαγές, και δύναται εύκολα και ταχύτατα να συγκλίνει με το μοντέλο του πραγματικού κόσμου.

Αυτό σημαίνει ότι οι χρήστες της Βάσης Δεδομένων μπορούν να προσαρμόσουν την μορφή των δεδομένων ευκολότερα ώστε οι προσαρμογές αυτές να εφαρμοστούν σε όλα τα δεδομένα, σε αντιπαράθεση με το παραδοσιακό μοντέλο.

Υψηλή Ποιότητα Δεδομένων

Η Βάση Δεδομένων χρησιμοποιεί τις δυνατότητες που παρέχει το Σύστημα Διαχείρισης Βάσεων Δεδομένων και ελέγχει για πλεονασμούς και παραβιάσεις της ορθότητας των δεδομένων. Τα δεδομένα προστατεύονται μ' αυτόν τον τρόπο από ένα μηχανισμό που δεν θα υπήρχε σε ένα παραδοσιακό σύστημα οργάνωσης. Έτσι διασφαλίζεται η ποιότητα των δεδομένων, και ο χρήστης έχει μεγαλύτερη ασφάλεια ότι τα δεδομένα αντλεί αναφορικά με τις ερωτήσεις του είναι ορθά.

ΚΕΦΑΛΑΙΟ 3

Η ΑΡΧΙΤΕΚΤΟΝΙΚΗ ΤΩΝ ΣΔΒΔ

3.1 Εισαγωγή

Όπως προαναφέρθηκε, ένα ΣΔΒΔ (Σύστημα Διαχείρισης Βάσης Δεδομένων) είναι υπεύθυνο για τη διαχείριση των δεδομένων των αρχείων της βάσης, δηλ. την προσθήκη, διαγραφή, τροποποίηση εγγραφών, την αναζήτηση μέσα στις εγγραφές κ.ά.). Το ΣΔΒΔ δέχεται αιτήσεις από τους χρήστες των εφαρμογών και επικοινωνεί με τα αρχεία της βάσης δεδομένων για να τις διεκπεραιώσει.

Αυτή η κοινή διεπαφή (interface) των εφαρμογών με τα αρχεία αποκαλείται λογική διεπαφή. Οι εφαρμογές που κατασκευάζουμε δεν απασχολούνται με τον τρόπο που είναι αποθηκευμένα τα δεδομένα, πόσο χώρο καταλαμβάνουν και αυτή η ιδιότητα είναι γνωστή ως ανεξαρτησία δεδομένων.

Αυτό σημαίνει πρακτικά ότι οποιαδήποτε αλλαγή σηματοδοτείται στον τρόπο οργάνωσης των αρχείων της βάσης δεδομένων δεν συνεπάγεται και αλλαγή στις εφαρμογές· καθώς αυτό ήταν ένα πρόβλημα που ταλαιπωρούσε πολύ τους προγραμματιστές παλαιότερων εποχών. Επιπλέον, η προσθήκη, η κατάργηση ή και η αλλαγή κάποιων εφαρμογών δεν θα έχει καμία συνέπεια στον τρόπο οργάνωσης των αρχείων της βάσης δεδομένων. Στα ΣΔΒΔ έχει επικρατήσει η λεγόμενη αρχιτεκτονική των τριών επιπέδων (βαθμίδων), και τα τρία επίπεδα είναι τα εξής :

- **Εσωτερικό επίπεδο (internal level)**, σχετίζεται με την αποθήκευση των αρχείων στον σκληρό δίσκο, δηλ. την πραγματική ή φυσική κατάστασή τους.
- **Εξωτερικό επίπεδο (external level)**, έχει να κάνει με τους χρήστες είτε αυτοί είναι απλοί χειριστές, είτε προγραμματιστές ή και οι διαχειριστές της βάσης.
- **Εννοιολογικό επίπεδο (conceptual level)**, είναι ένα ενδιάμεσο επίπεδο που διασυνδέει τα δύο άλλα επίπεδα και πραγματεύεται τη λογική σχεδίαση των αρχείων της βάσης δεδομένων.

3.2 Οι Οντότητες (Entities)

Με τον όρο **οντότητα (entity)** εννοούμε ένα αντικείμενο, ένα πρόσωπο, μια κατάσταση και γενικά ο,τιδήποτε μπορεί να εννοιοποιηθεί σαν ανεξάρτητη ύπαρξη (αυτόνομη μονάδα του φυσικού κόσμου). Για παράδειγμα, σε μια βάση δεδομένων μιας εμπορικής εταιρείας, οντότητες μπορεί να είναι οι εργαζόμενοι, οι πελάτες, οι προμηθευτές, οι παραγγελίες, τα είδη της αποθήκης (προϊόντα) κ.ά.

Το **Μοντέλο Οντοτήτων Συσχετίσεων (Entity Relationship Model, ER Model)** είναι μια διαγραμματική αναπαράσταση της δομής μιας βάσης δεδομένων και χρησιμοποιείται κατά τη φάση του λογικού σχεδιασμού της τελευταίας. Αυτό σημαίνει πως, δεν ασχολείται με τον τρόπο που αποθηκεύονται τα δεδομένα της βάσης, αλλά με την ταυτοποίηση των δεδομένων και με τον τρόπο συσχέτισής τους.¹²

Θα αναφερθούμε σε ένα παράδειγμα μιας εταιρείας, που περιέχει δεδομένα τα οποία έχουν να κάνουν τους υπαλλήλους της (employees), τα τμήματά της (departments) και τα έργα (projects) που έχουν αναλάβει αυτά τα τμήματα. Ένα τμήμα της εταιρείας μπορεί να εποπτεύει ένα ή περισσότερα έργα (projects) και ένας υπάλληλος ανήκει σ' ένα μόνο τμήμα της εταιρείας αλλά μπορεί να απασχολείται ταυτόχρονα σε πολλά έργα, τα οποία δεν περιορίζονται να παρακολουθούνται από το ίδιο τμήμα.

3.3 Οι Ιδιότητες (Attributes)

Με τον όρο **ιδιότητα ή χαρακτηριστικό ή και πεδίο (attribute)** μιας οντότητας, αναφερόμαστε σ' ένα από τα συστατικά της στοιχεία που την περιγράφουν και την διαφοροποιούν από τα άλλα στοιχεία της ίδιας οντότητας. Για παράδειγμα, η οντότητα ΠΕΛΑΤΗΣ μπορεί να έχει ως ιδιότητες (χαρακτηριστικά) τον κωδικό, το επώνυμο, το όνομα, τη διεύθυνση, το τηλέφωνο, το ΑΦΜ κ.ά., με τη βοήθεια των οποίων μπορούμε να ξεχωρίσουμε τους πελάτες μεταξύ τους.

¹² (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

Επίσης, η οντότητα ΠΑΡΑΓΓΕΛΙΑ μπορεί να έχει ως ιδιότητες (χαρακτηριστικά) τον κωδικό, τον αριθμό παραστατικού, την ημερομηνία, τον κωδικό πελάτη, το προϊόν κ.ά., με τη βοήθεια των οποίων μπορούμε να ξεχωρίσουμε τις παραγγελίες μεταξύ τους. Στο παράδειγμα της εταιρείας, μπορούμε να ορίσουμε έναν τύπο οντότητας για τους υπαλλήλους της εταιρείας (EMPLOYEE), έναν τύπο οντότητας για τα τμήματα που έχει η εταιρεία (DEPARTMENT) και έναν τύπο οντότητας για τα έργα που έχει αναλάβει η εταιρεία (PROJECT). Καθένας από τους προαναφερθέντες τύπους οντοτήτων περιγράφεται από ένα όνομα και από το σύνολο των πεδίων που περιέχει. Οι πληροφορίες αυτές αποτελούν το **σχήμα (schema)** της οντότητας.¹³

3.4 Τα Στιγμιότυπα (Snapshots)

Το κάθε διαφορετικό (αυτόνομο) στοιχείο μιας οντότητας αποκαλείται **στιγμιότυπο (snapshot)** ή και **εμφάνιση της** οντότητας. Για παράδειγμα, στην οντότητα ΠΕΛΑΤΗΣ, άλλο στιγμιότυπο είναι ο πελάτης με επώνυμο Παπαδόπουλος και άλλο στιγμιότυπο είναι ο πελάτης με επώνυμο Σουμπάσης.

3.5 Το Πρωτεύον Κλειδί (Primary Key)

Πρωτεύον κλειδί ή **πεδίο κλειδί** (primary key) μιας οντότητας καλείται η ιδιότητα εκείνη (ή ο συνδυασμός ιδιοτήτων) που έχει μοναδική τιμή για όλα τα στιγμιότυπα (εμφανίσεις) της οντότητας. Για παράδειγμα, στην οντότητα ΠΕΛΑΤΗΣ πρωτεύον κλειδί είναι ο κωδικός πελάτη, στην οντότητα ΠΑΡΑΓΓΕΛΙΑ πρωτεύον κλειδί μπορεί να είναι ο κωδικός παραγγελίας ή ο αριθμός παραστατικού κοκ.

Υπάρχουν περιπτώσεις που το πεδίο κλειδί ενός τύπου οντότητας μπορεί να μην είναι απλό αλλά σύνθετο. Να αποτελείται δηλαδή από πολλά απλά πεδία. Σ' αυτήν την περίπτωση η συνθήκη της μοναδικότητας για την τιμή του κλειδιού δεν εφαρμόζεται σε κάθε πεδίο του σύνθετου κλειδιού αλλά στο σύνολο του συνδυασμού αυτών των πεδίων.¹⁴

¹³ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

¹⁴ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

3.6 Οι Συσχετίσεις (Relationships)

Με τον όρο **συσχέτιση (relationship)** αναφερόμαστε στον τρόπο σύνδεσης (επικοινωνίας) δύο ξεχωριστών οντοτήτων με στόχο να έχουμε τη δυνατότητα να παίρνουμε στοιχεία (πληροφορίες) από τον συνδυασμό τους.

Για παράδειγμα, η οντότητα ΓΙΑΤΡΟΣ συσχετίζεται με την οντότητα ΑΣΘΕΝΗΣ αλλά και με την οντότητα ΚΛΙΝΙΚΗ στη βάση δεδομένων ενός νοσοκομείου. Μπορούμε να δεχθούμε ότι ένας γιατρός παρακολουθεί (συσχετίζεται με) πολλούς ασθενείς, αλλά ένας ασθενής παρακολουθείται από (συσχετίζεται με) έναν μόνο γιατρό και επίσης ένας γιατρός συσχετίζεται με (ανήκει σε) μία μόνο κλινική, αλλά μια κλινική συσχετίζεται με (απασχολεί) πολλούς γιατρούς.

Στο παράδειγμα της εταιρείας, η οντότητα EMPLOYEE συσχετίζεται με την οντότητα DEPARTMENT και η οντότητα DEPARTMENT συσχετίζεται με την οντότητα PROJECTS. Ένας υπάλληλος ανήκει σ' ένα μόνο τμήμα και ένα τμήμα μπορεί να έχει πολλούς υπαλλήλους. Επίσης, ένα τμήμα εποπτεύει πολλά έργα αλλά ένα έργο εποπτεύεται από ένα μόνο τμήμα.

3.7 Το Ιεραρχικό Μοντέλο Βάσεων Δεδομένων

Υπάρχουν τρία βασικά μοντέλα που έχουν επικρατήσει στις βάσεις δεδομένων, το ιεραρχικό, το δικτυωτό και το σχεσιακό. Το ιεραρχικό μοντέλο (hierarchical) έχει μια ιεραρχική δομή που θυμίζει δένδρο. Οι οντότητες μοιάζουν με κλαδιά δένδρων και τοποθετούνται σε επίπεδα ιεραρχίας. Τα κλαδιά παριστάνουν τις συσχετίσεις ανάμεσα στις οντότητες.

Από μια οντότητα που βρίσκεται σ' ένα ανώτερο επίπεδο εκκινούν πολλά κλαδιά. Καθένα από αυτά καταλήγει σε μια οντότητα που βρίσκεται σ' ένα χαμηλότερο υποεπίπεδο. Αλλά, σε κάθε οντότητα που βρίσκεται σ' ένα χαμηλότερο υποεπίπεδο αντιστοιχεί μία αυστηρά μόνο οντότητα που βρίσκεται σ' ένα ανώτερο επίπεδο. Το μοντέλο αυτό ήταν το πρώτο που εμφανίσθηκε αλλά σήμερα θεωρείται δύσχρηστο και παλαιωμένο.

3.8 Το Δικτυωτό Μοντέλο Βάσεων Δεδομένων

Και στο δικτυωτό (network) μοντέλο, τα στοιχεία τοποθετούνται σ' ένα επίπεδο ιεραρχίας, αλλά κάθε στοιχείο μπορεί να συσχετισθεί με πολλά στοιχεία είτε σ' ένα κατώτερο ή σ' ένα ανώτερο επίπεδο.¹⁵

3.9 Το Σχεσιακό Μοντέλο Βάσεων Δεδομένων

Το σχεσιακό (relational) μοντέλο έχει επικρατήσει σήμερα στην αναπαράσταση των δεδομένων καθώς διαθέτει σημαντικά πλεονεκτήματα ως προς τα άλλα δύο. Οι βάσεις δεδομένων που σχεδιάζονται σύμφωνα μ' αυτό αποκαλούνται σχεσιακές (relational databases). Με τις σχεσιακές βάσεις δεδομένων διαθέτουμε έναν σαφή, απλό και εύκολα αντιληπτό τρόπο να μπορέσουμε να αναπαραστήσουμε και να διαχειριστούμε τα δεδομένα μας. Μοναδικό τους μειονέκτημα είναι η ταχύτητα υπολογισμών και ο χώρος αποθήκευσης, αλλά μόνο όταν έχουμε να κάνουμε με πολύ μεγάλες βάσεις δεδομένων.

Στο μοντέλο αυτό οι βάσεις δεδομένων περιγράφονται με αυστηρές μαθηματικές έννοιες και ο χρήστης βλέπει τις οντότητες και τις συσχετίσεις με τη μορφή πινάκων (tables) και σχέσεων (relations) αντίστοιχα.

Ένας **πίνακας (table)** αποτελείται από γραμμές (rows) και στήλες (columns). Εκεί τοποθετούμε τα στοιχεία σε οριζόντια και κάθετη μορφή. Η κάθε στήλη του πίνακα χαρακτηρίζει κάποια ιδιότητα της οντότητας και αποκαλείται **χαρακτηριστικό (attribute) ή πεδίο (field)**. Η κάθε γραμμή του πίνακα περιέχει όλες τις πληροφορίες (στήλες) αναφορικά μ' ένα στοιχείο της οντότητας και αποκαλείται **πλειάδα (tuple) ή εγγραφή (record)**.¹⁶

¹⁵ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

¹⁶ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

Κάθε πεδίο του πίνακα μπορεί να πάρει ορισμένες μόνο τιμές. Αυτές καθορίζονται από τον τύπο δεδομένων της ιδιότητας, όπως π.χ. ονόματα ή αριθμοί, ή και από αυτό που εκφράζει, όπως το ότι δεν μπορούμε να έχουμε αρνητικό βάρος ή αρνητικό ΑΦΜ, για παράδειγμα. Το σύνολο των αποδεκτών τιμών μιας οντότητας αποκαλείται **πεδίο ορισμού (domain)**.

Για να μπορέσουμε να αντιληφθούμε τις σχεσιακές βάσεις δεδομένων, ένα πολύ χαρακτηριστικό παράδειγμα αποτελεί ένας πίνακας πελατών και ένας πίνακας παραγγελιών μιας εμπορικής εταιρείας.

Τα πεδία που μπορούν να οριστούν στους πίνακες αυτούς είναι τα παρακάτω:

ΠΙΝΑΚΑΣ (ΟΝΤΟΤΗΤΑ) ΠΕΛΑΤΕΣ

(Κωδικός Πελάτη, Επώνυμο, Όνομα, Διεύθυνση, ΤΚ, Πόλη, ΑΦΜ, Υπόλοιπο)

ΠΙΝΑΚΑΣ (ΟΝΤΟΤΗΤΑ) ΠΑΡΑΓΓΕΛΙΕΣ

(Κωδικός Πελάτη, Κωδικός Παραγγελίας, Ημερομηνία, Είδος, Ποσότητα, Τιμή Μονάδας)

Διακρίνουμε ότι οι δύο πίνακες έχουν ένα κοινό πεδίο (στήλη), τον Κωδικό Πελάτη, κάτι το οποίο είναι απαραίτητο στις σχεσιακές βάσεις δεδομένων, ώστε να είμαστε σε θέση να κάνουμε τη δουλειά μας και να συνδυάσουμε πληροφορίες και από τους δύο πίνακες.

Όπως είναι ευνόητο, η βασικότερη εργασία που πρέπει να πραγματοποιήσουμε κατά τον σχεδιασμό μιας σχεσιακής βάσης δεδομένων, είναι να ορίσουμε τους πίνακες που θα χρησιμοποιήσουμε καθώς και τα πεδία που θα περιέχει ο καθένας απ' αυτούς. Η διαδικασία αυτή αποκαλείται κατασκευή του **σχήματος (schema)** μιας βάσης δεδομένων.

Οι κανόνες που πρέπει να ακολουθούμε πιστά κατά τον σχεδιασμό μιας σχεσιακής βάσης δεδομένων είναι οι εξής :

- Η κάθε οντότητα πρέπει να παριστάνεται ως ένας ξεχωριστός πίνακας.
- Η κάθε στήλη του πίνακα αντιστοιχεί σε μια ιδιότητα της οντότητας.

- Η κάθε γραμμή του πίνακα αντιστοιχεί σε μια εμφάνιση της οντότητας.
- Η κάθε γραμμή πρέπει να είναι μοναδική. Αποκλείεται να υπάρχουν δύο ή και περισσότερες γραμμές που να περιέχουν τα ίδια ακριβώς στοιχεία.
- Η σειρά εμφάνισης των γραμμών είναι άνευ σημασίας.
- Η κάθε στήλη έχει μια δική της μοναδική ονομασία.
- Οι τιμές που ανήκουν στην ίδια στήλη πρέπει να είναι του ίδιου τύπου, δηλ. ή όλες αριθμοί ή όλες αλφαριθμητικές κοκ.
- Η στήλη που αποτελεί το πρωτεύον κλειδί (primary key) μιας οντότητας, δεν πρέπει να είναι ποτέ κενή (null).
- Αποκλείεται να υπάρχουν δύο ή και περισσότερες γραμμές που να περιέχουν την ίδια τιμή στο πρωτεύον κλειδί.
- Το πρωτεύον κλειδί μιας οντότητας αποκαλείται ξένο κλειδί (foreign key) σε μια άλλη οντότητα, με την οποία υπάρχει συσχετισμός.
- Μπορεί να υπάρχουν πολλές γραμμές που να έχουν την ίδια τιμή στο ξένο κλειδί.¹⁷

3.10 Τα Σχεσιακά ΣΔΒΔ (RDBMS)

Τα Σχεσιακά Συστήματα Διαχείρισης Βάσεων Δεδομένων (ΣΣΔΒΔ) ή RBMS (Relational DataBase Management Systems) εξελίχθηκαν βασιζόμενα στο σχεσιακό μοντέλο και έχουν επικρατήσει πλήρως στον χώρο. Κατά τον σχεδιασμό και τη δημιουργία μιας σχεσιακής βάσης δεδομένων, οι πίνακες αποτελούν το μοναδικό δομικό και απαραίτητο στοιχείο ώστε να μπορέσουν να απεικονισθούν οι πληροφορίες που περιέχονται στη βάση δεδομένων.

Για να μπορέσουμε να προσθέσουμε, διαγράψουμε ή τροποποιήσουμε τα στοιχεία που περιέχονται σε μια βάση δεδομένων, χρησιμοποιούμε ειδικές γλώσσες

¹⁷ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

προγραμματισμού που αποκαλούνται **γλώσσες ερωταπαντήσεων (query languages)**. Η ποι διαδεδομένη γλώσσα, η οποία σήμερα αποτελεί ένα διεθνές πρότυπο για την επικοινωνία των χρηστών με τα Σχεσιακά ΣΔΒΔ είναι η **SQL (Structured Query Language) ή Δομημένη Γλώσσα Ερωτημάτων**. Μπορεί να λειτουργήσει ανεξάρτητα αλλά και σε συνεργασία μ' άλλες γλώσσες προγραμματισμού.

Μια άλλη εύχρηστη γλώσσα προγραμματισμού ώστε μπορούμε να υποβάλουμε ερωτήματα σε σχεσιακές βάσεις δεδομένων και να λαμβάνουμε απαντήσεις είναι η **QBE (Query By Example)**. Αυτή χρησιμοποιεί φόρμες για τη γραφική απεικόνιση των ερωτημάτων μας.

Σήμερα, υπάρχουν εξελιγμένα εργαλεία διαχείρισης σε γραφικό και φιλικό προς τον χρήστη περιβάλλον για να κάνουμε τα εξής :

- Δημιουργία πινάκων
- Δημιουργία φορμών
- Δημιουργία ερωτημάτων
- Δημιουργία εκθέσεων (αναφορών)

Τα Σχεσιακά ΣΔΒΔ τα διακρίνουμε στα **μεγάλα**, που αφορούν κυρίως μεγάλους οργανισμούς και επιχειρήσεις, κατέχουν τεράστιο όγκο δεδομένων και πολλούς χρήστες ταυτόχρονα. Τέτοια συστήματα είναι τα Oracle, Ingres, Informix, SQL Server κ.ά.. Από την άλλη έχουμε τα **μικρά**, τα οποία απευθύνονται κυρίως απλούς χρήστες, όπως είναι η Microsoft Access, η Paradox, η FoxPro κ.ά.

3.11 Το Μοντέλο Οντοτήτων-Συσχετίσεων

Το μοντέλο που έχει επικρατήσει σήμερα για να παραστήσει τις έννοιες ή τη δομή μιας βάσης δεδομένων είναι το **Μοντέλο Οντοτήτων-Συσχετίσεων (ΟΣ)**. Οι βασικές (θεμελιώδεις) έννοιες του μοντέλου αυτού είναι οι εξής :

- Οντότητες
- Ιδιότητες ή Χαρακτηριστικά
- Συσχετίσεις

Για να αναπαραστήσουμε ένα Μοντέλο Οντοτήτων – Συσχετίσεων χρησιμοποιούμε ειδικά διαγράμματα, όπου τα ορθογώνια αντιπροσωπεύουν τις οντότητες, οι ρόμβοι τις συσχετίσεις και οι ελλείψεις τις ιδιότητες. Με ευθείες γραμμές συνδέουμε τις οντότητες που με κάποιο τρόπο έχουν σχέση μεταξύ τους. Όλα τα παραπάνω αποτελούν τη λογική δομή μιας βάσης δεδομένων, μια εργασία που είναι αναγκαίο να διεκπεραιωθεί πριν από την καταχώριση και την τροποποίηση των στοιχείων (πληροφοριών) της βάσης δεδομένων.

Το μοντέλο οντοτήτων–συσχετίσεων αποτελεί μια γενική περιγραφή των γενικών στοιχείων που αποτελούν μια βάση δεδομένων και ενσαρκώνουν την αντίληψη που έχουμε για τα δεδομένα (εννοιολογικό), χωρίς να εμπλέκεται σε λεπτομέρειες υλοποίησης.

3.11.1 Οι Οντότητες

Με τον όρο **οντότητα (entity)** αναγγέλουμε κάθε αντικείμενο, έννοια, πρόσωπο ή κατάσταση που έχει μια αυθυπόστατη ύπαρξη. Είναι κάτι που διακρίνεται και μπορούμε να συγκεντρώσουμε πληροφορίες (στοιχεία) γι' αυτό. Η οντότητα είναι αντίστοιχη με την έννοια της εγγραφής των αρχείων και των πινάκων, αλλά και με την έννοια του αντικειμένου στις σύγχρονες αντικειμενοστραφείς γλώσσες προγραμματισμού.

Παραδείγματα οντοτήτων είναι τα εξής : ΠΕΛΑΤΗΣ, ΠΑΡΑΓΓΕΛΙΑ, ΠΡΟΜΗΘΕΥΤΗΣ, ΑΠΟΘΗΚΗ, ΜΑΘΗΤΗΣ, ΚΑΘΗΓΗΤΗΣ, ΑΘΛΗΤΗΣ, ΑΓΩΝΙΣΜΑ, ΧΩΡΑ, ΠΟΛΕΙΣ κ.ά.

Μια βάση δεδομένων δύναται να περιέχει πολλές διαφορετικές οντότητες. Αυτές αναπαριστώνται με ορθογώνια παραλληλόγραμμα και συσχετίζονται μεταξύ τους σε δυάδες.¹⁸

¹⁸ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

3.11.2 Οι Ιδιότητες (Χαρακτηριστικά) των Οντοτήτων

Με τον όρο **ιδιότητες (properties)** ή **χαρακτηριστικά (attributes)** αναφερόμαστε στα συστατικά (δομικά) στοιχεία τα οποία διακρίνουν (αποτελούν) μια οντότητα. Η ιδιότητα είναι παρεμφερής με την έννοια των αρχείων και των πινάκων του πεδίου, αλλά και με την έννοια της μεταβλητής στις γλώσσες προγραμματισμού.

Για παράδειγμα, η οντότητα ΓΙΑΤΡΟΣ μπορεί να αποτελείται από τις ιδιότητες (χαρακτηριστικά) ΑριθμόςΜητρώου, Επώνυμο, Όνομα, Πατρώνυμο, Ειδικότητα, Βαθμός, ΈτοςΓέννησης, Διεύθυνση, ΑΦΜ, Τηλέφωνο, Κινητό κ.ά., ενώ η οντότητα ΑΘΛΗΤΗΣ μπορεί να αποτελείται από τις ιδιότητες (χαρακτηριστικά) Κωδικός Αθλητή, Επώνυμο, Όνομα, Πατρώνυμο, Αγώνισμα, Επίδοση, Σύλλογος, Έτος Γέννησης, Διεύθυνση, Τηλέφωνο, Κινητό κ.ά.

Απ' όλες τις ιδιότητες μιας οντότητας, υπάρχει μία μόνο ιδιότητα, και σπανιότερα ένας συνδυασμός δύο ή και περισσότερων ιδιοτήτων, η τιμή της οποίας είναι μοναδική και αξιολογεί την κάθε εμφάνιση (στιγμιότυπο) της οντότητας. Αυτή αποκαλείται **πρωτεύον κλειδί (primary key)**. Το πρωτεύον κλειδί εμφανίζεται στα διαγράμματα με υπογράμμιση ή με έντονη γραφή ή έχει ως πρόθεμα τον χαρακτήρα #.

Στο διάγραμμα οντοτήτων–συσχετίσεων οι ιδιότητες παριστάνονται με σχήματα ελλειπτικής μορφής, που ενώνονται με ευθείες γραμμές με την οντότητα στην οποία ανήκουν.¹⁹

3.11.3 Τα Κλειδιά

Όπως είδαμε και νωρίτερα, με τον όρο **κλειδί (key)** ή πιο σωστά **πρωτεύον κλειδί (primary key)** αναφερόμαστε σε μια ιδιότητα (πεδίο), ή σπανιότερα σ' ένα σύνολο ιδιοτήτων (πεδίων), η τιμή της οποίας είναι μοναδική σ' ολόκληρη την οντότητα (πίνακας). Στην πράξη, το πρωτεύον κλειδί έχει διαφορετική τιμή για κάθε εμφάνιση της οντότητας ή για κάθε γραμμή (εγγραφή) του πίνακα και ποτέ δεν μπορεί να έχει

¹⁹ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

μηδενική (κενή) τιμή (null). Προσοχή, άλλο πράγμα είναι ο αριθμός 0 και διαφορετικό να είναι κενή η τιμή (null), δηλ. η μη ύπαρξη τιμής.

Όταν συδυάζονται δύο ή περισσότερες ιδιότητες (πεδίων) για τη δημιουργία ενός πρωτεύοντος κλειδιού, το αποτέλεσμα που δημιουργείται από αυτόν τον συνδυασμό αποκαλείται **σύνθετο κλειδί**. Ένα παράδειγμα σύνθετου κλειδιού θα μπορούσε να είναι ο συνδυασμός των ιδιοτήτων Επώνυμο, Όνομα και Πατρώνυμο, εφόσον φυσικά είμαστε απολύτως βέβαιοι ότι δεν υπάρχουν δύο ή και περισσότερα άτομα με κοινές τιμές στις παραπάνω ιδιότητες.²⁰

Ξένο κλειδί αποκαλείται μια ιδιότητα (πεδίο) που είναι πρωτεύον κλειδί σε μια οντότητα (πίνακας) αλλά που υπάρχει και σε μια άλλη οντότητα (πίνακας) σαν απλή ιδιότητα. Τα ξένα κλειδιά είναι απαραίτητα για να μπορέσουμε να κάνουμε τις συσχετίσεις (συνδέσεις, επικοινωνίες) ανάμεσα στις οντότητες (πίνακες).

Για παράδειγμα, στην οντότητα ΣΥΛΛΟΓΟΣ, το πεδίο ΚωδικόςΣυλλόγου είναι πρωτεύον κλειδί, ενώ στην οντότητα ΑΘΛΗΤΗΣ, το πεδίο ΚωδικόςΣυλλόγου είναι ξένο κλειδί και είναι απαραίτητο για να μπορέσουμε να υλοποιήσουμε τη συσχέτιση ΑΝΗΚΕΙ, δηλ. να αντλήσουμε την πληροφορία ποιοι αθλητές ανήκουν σε ποιους συλλόγους. Προφανώς, στην οντότητα ΣΥΛΛΟΓΟΣ, το πεδίο Κωδικός Συλλόγου θα έχει μοναδικές τιμές, ενώ στην οντότητα ΑΘΛΗΤΗΣ, το πεδίο Κωδικός Συλλόγου θα έχει επαναλαμβανόμενες τιμές και αυτό γιατί πολλοί αθλητές θα ανήκουν στον ίδιο σύλλογο, αλλά ένας αθλητής ανήκει υποχρεωτικά σ' έναν και μόνο έναν σύλλογο.

Σ' ένα άλλο παράδειγμα, στην οντότητα ΓΙΑΤΡΟΣ, το πεδίο ΚωδικόςΓιατρού είναι πρωτεύον κλειδί, ενώ στην οντότητα ΑΣΘΕΝΗΣ, το πεδίο ΚωδικόςΓιατρού είναι ξένο κλειδί και είναι απαραίτητο για να μπορέσουμε να υλοποιήσουμε τη συσχέτιση ΠΑΡΑΚΟΛΟΥΘΕΙΤΑΙ, δηλ. να αντλήσουμε την πληροφορία ποιοι ασθενείς παρακολουθούνται από ποιους γιατρούς. Προφανώς, στην οντότητα ΓΙΑΤΡΟΣ, το πεδίο ΚωδικόςΓιατρού θα έχει μοναδικές τιμές, ενώ στην οντότητα ΑΣΘΕΝΗΣ, το πεδίο ΚωδικόςΓιατρού θα έχει επαναλαμβανόμενες τιμές και αυτό γιατί πολλοί

²⁰ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

ασθενείς θα παρακολουθούνται από τον ίδιο γιατρό, αλλά ένας ασθενής παρακολουθείται μόνο από έναν γιατρό.

Αυτό αποτελεί βέβαια μια παραδοχή που κάνουμε για να μπορέσουμε να πραγματοποιήσουμε μια συσχέτιση σαν την παραπάνω σε μια βάση δεδομένων ενός Νοσοκομείου. Ωστόσο μπορεί να θεωρήσει κάποιος ότι ένας ασθενής μπορεί να παρακολουθείται από πολλούς γιατρούς ταυτόχρονα, οπότε θα πρέπει να μεταβάλλουμε και τον τρόπο συσχέτισης των παραπάνω οντοτήτων.²¹

3.11.4 Οι Συσχετίσεις Μεταξύ Οντοτήτων

Ο σωστός σχεδιασμός και προσδιορισμός των οντοτήτων και των ιδιοτήτων τους είναι στοιχειώδη βήματα για τη σωστή σχεδίαση και υλοποίηση μιας βάσης δεδομένων. Μια συσχέτιση συνδέει δύο ή και περισσότερες οντότητες μεταξύ τους και απεικονίζεται στο διάγραμμα οντοτήτων–συσχετίσεων μ' έναν ρόμβο.

Οι συσχετίσεις είναι αναγκαίες για να μπορέσουμε να αντλήσουμε πληροφορίες που αφορούν δύο ή και περισσότερες οντότητες, όπως για παράδειγμα ποιοι πελάτες έκαναν παραγγελίες κάποια συγκεκριμένη χρονική περίοδο (συσχέτιση ΠΑΡΑΓΓΕΛΝΕΙ) ή ποιοι αθλητές ανήκουν σε ποιους συλλόγους (συσχέτιση ΑΝΗΚΕΙ) ή ποιοι αθλητές έλαβαν μέρος σε αγωνίσματα μια συγκεκριμένη χρονιά (συσχέτιση ΣΥΜΜΕΤΕΧΕΙ) κ.ο.κ.

Όταν οι οντότητες που συμμετέχουν σε μια συσχέτιση είναι δύο, η συσχέτιση χαρακτηρίζεται ως *διμελής* ή *δυναδική*. Ο βαθμός μιας συσχέτισης ισούται με το πλήθος των οντοτήτων που συμμετέχουν σ' αυτήν. Μια συσχέτιση μπορεί και η ίδια να έχει ιδιότητες που να περιγράφουν ορισμένα χαρακτηριστικά της, όπως για παράδειγμα η συσχέτιση ΠΑΡΑΓΓΕΛΙΑ ανάμεσα στις οντότητες ΠΕΛΑΤΗΣ και ΠΡΟΪΟΝ μπορεί να περιέχει τις ιδιότητες (πεδία) Κωδικός Πελάτη, Κωδικός Προϊόντος, Κωδικός Παραγγελίας, Ημερομηνία Παραγγελίας, Ποσότητα κ.ά.

²¹ (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

Εδώ το σωστό είναι να κατασκευάσουμε μια ακόμα οντότητα, την οντότητα ΠΑΡΑΓΓΕΛΙΑ, η οποία και θα περιέχει όλες τις παραπάνω ιδιότητες, και να μετονομάσουμε την προηγούμενη συσχέτιση από ΠΑΡΑΓΓΕΛΙΑ σε ΣΥΝΑΛΛΑΓΗ, που δεν θα περιέχει τώρα ιδιότητες. Έτσι, η παραπάνω συσχέτιση θα μετατραπεί από διμελής σε τριμελής.

Κατά τον σχεδιασμό μιας βάσης δεδομένων, θα πρέπει να εκχωρούμε ιδιότητες μόνο στις οντότητες και να έχουμε τις συσχετίσεις απλά και μόνο για να κατανοούμε τις λογικές συνδέσεις ανάμεσα στις οντότητες.

3.11.5 Οι Διμελείς Συσχετίσεις

Οι διμελείς συσχετίσεις μεταξύ οντοτήτων είναι αυτές που θα μας απασχολήσουν ιδιαίτερα και υπάρχουν τρία βασικά είδη συνδέσεων σ' αυτές, τα εξής :

- *Ένα-προς-ένα (1:1)*, στα οποία μια εμφάνιση της μιας οντότητας συνδέεται με μία αυστηρά εμφάνιση της άλλης οντότητας. Για παράδειγμα, η οντότητα ΣΥΛΛΟΓΟΣ έχει έναν μόνο προπονητή, ενώ η οντότητα ΠΡΟΠΟΝΗΤΗΣ συνδέεται μ' έναν και μόνο έναν σύλλογο. Σ' ένα άλλο παράδειγμα, η οντότητα ΝΟΜΟΣ έχει μία μόνο πόλη σαν πρωτεύουσα, ενώ η οντότητα ΠΡΩΤΕΥΟΥΣΑ αντιστοιχεί σ' έναν και μόνο έναν νομό. Στην περίπτωση των διμελών συσχετίσεων του τύπου ένα-προς-ένα, μπορούμε ενώνοντας τα στοιχεία και των δύο ιδιοτήτων να κατασκευάσουμε μια μοναδική οντότητα (πίνακα).
- *Ένα-προς-πολλά (1:M)*, όπου μια εμφάνιση της μιας οντότητας συνδέεται με πολλές εμφανίσεις της άλλης οντότητας αλλά κάθε εμφάνιση της δεύτερης οντότητας συνδέεται με μία αυστηρά εμφάνιση της πρώτης οντότητας. Για παράδειγμα, ένας ΠΕΛΑΤΗΣ κάνει πολλές παραγγελίες, αλλά μια ΠΑΡΑΓΓΕΛΙΑ αντιστοιχεί σ' έναν και μόνο έναν πελάτη. Σ' ένα άλλο παράδειγμα, ένας ΣΥΛΛΟΓΟΣ έχει πολλούς αθλητές, αλλά ένας ΑΘΛΗΤΗΣ ανήκει σ' έναν και μόνο έναν σύλλογο. Οι διμελείς συσχετίσεις του τύπου ένα-προς-ένα είναι συναντώνται συχνότερα και είναι πιο εύχρηστες.

- *Πολλά-προς-πολλά (M:N)*, όπου σε μια εμφάνιση της μιας οντότητας αντιστοιχούν πολλές εμφανίσεις της άλλης οντότητας και σε κάθε εμφάνιση της δεύτερης οντότητας αντιστοιχούν πολλές εμφανίσεις της πρώτης οντότητας. Για παράδειγμα, ένας ΑΘΛΗΤΗΣ συμμετέχει σε πολλούς αγώνες αλλά και σ' έναν ΑΓΩΝΑ λαμβάνουν μέρος πολλοί αθλητές. Σ' ένα άλλο παράδειγμα, ένας ΚΑΘΗΓΗΤΗΣ διδάσκει σε πολλούς μαθητές αλλά και ένας ΜΑΘΗΤΗΣ διδάσκεται από πολλούς καθηγητές. Για να μπορέσουμε να διαχειριστούμε μια διμελή σχέση του τύπου πολλά-προς-πολλά, θα πρέπει να κατασκευάσουμε έναν τρίτο πίνακα που θα αποτελείται από δύο μόνο ιδιότητες (πεδία), δηλ. τα πεδία κλειδιά των δύο οντοτήτων, οπότε ο συνδυασμός τους θα είναι και το πεδίο κλειδί (σύνθετο κλειδί) του νέου πίνακα.

3.11.6 Το Διάγραμμα Οντοτήτων Συσχετίσεων

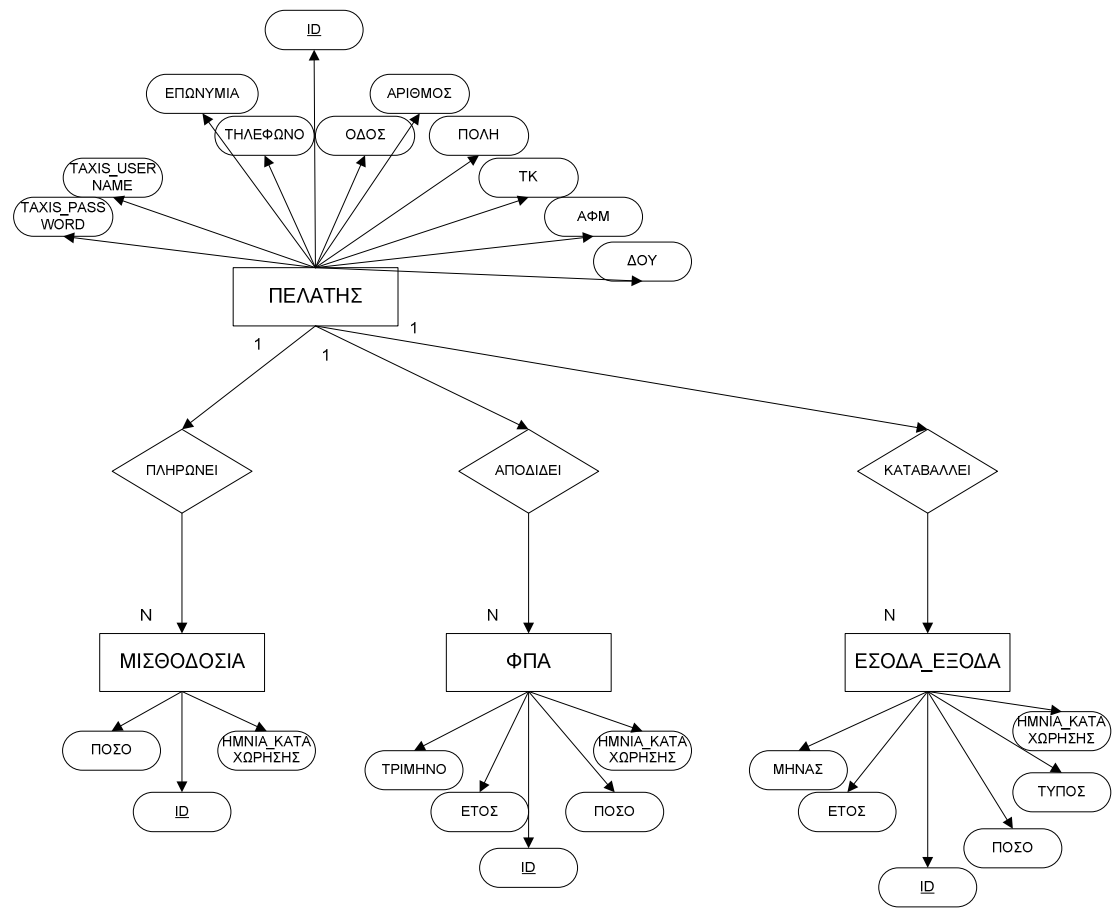
Για να μπορέσουμε να διαμορφώσουμε το διάγραμμα οντοτήτων συσχετίσεων, θα πρέπει να ακολουθήσουμε τα εξής βήματα :

- Να ορίσουμε τις οντότητες (πίνακες) που θα ανήκουν στη βάση δεδομένων που θέλουμε να κατασκευάσουμε.
- Να ορίσουμε τις ιδιότητες (πεδία) και τα πρωτεύοντα κλειδιά της κάθε οντότητας (πίνακα).
- Να ορίσουμε τις συσχετίσεις ανάμεσα στις οντότητες.

Δημιουργούμε το διάγραμμα οντοτήτων συσχετίσεων, όπου θα αναπαραστήσουμε τις οντότητες, τις ιδιότητές τους και τις συσχετίσεις τους.²²

²² (<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

Εικόνα 3.1. Διάγραμμα οντοτήτων – συσχετίσεων



ΚΕΦΑΛΑΙΟ 4

ΛΟΓΙΚΟΣ ΣΧΕΔΙΑΣΜΟΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

4.1 Εισαγωγή

Στο κεφάλαιο αυτό θα παρουσιαστεί μια μεθοδολογία, που υπάγεται στο σχεσιακό μοντέλο. Η μεθοδολογία αυτή οδηγεί στη δημιουργία των κατάλληλων σχέσεων (πινάκων) με βάση το αντίστοιχο εννοιολογικό μοντέλο. Η καταλληλότητα των πινάκων που θα δημιουργηθούν είναι ένα κρίσιμο ζήτημα το οποίο θα κρίνει αφενός τη χρηστικότητα και αφετέρου την απόδοση της βάσης των δεδομένων. Η θεωρία της κανονικοποίησης είναι ένα επιπλέον εργαλείο που επιτρέπει στο σχεδιαστή να ελέγξει το θέμα αυτό και να διορθώσει τυχόν ατέλειες στη δομή των πινάκων.

Το κεφάλαιο αυτό περιέχει ενότητες που αφορούν:

1. το αντικείμενο του λογικού σχεδιασμού, ως ένα αναπόσπαστο κομμάτι της συνολικής διαδικασίας που οδηγεί στην υλοποίηση μιας σχεσιακής βάσης δεδομένων.
2. τη μεθοδολογία δημιουργίας των τελικών πινάκων της ΒΔ χρησιμοποιώντας ως είσοδο τα διαγράμματα οντοτήτων συσχετίσεων, που έχουν δημιουργηθεί κατά τον εννοιολογικό σχεδιασμό.

4.2 Το αντικείμενο του λογικού σχεδιασμού

Στα πλαίσια του εννοιολογικού σχεδιασμού των βάσεων δεδομένων μελετήσαμε διεξοδικά το μοντέλο των οντοτήτων συσχετίσεων. Υπενθυμίζεται πως το αποτέλεσμα του σχεδιασμού με βάση το μοντέλο αυτό είναι η δημιουργία του Διαγράμματος Οντοτήτων Συσχετίσεων (ΔΟΣ). Το ΔΟΣ περιγράφει σε εννοιολογικό επίπεδο, τις πληροφορίες που περιέχει μια βάση δεδομένων.

4.3 Ο λογικός σχεδιασμός.

Ο λογικός σχεδιασμός έχει ως σκοπό, λαμβάνοντας υπόψη τα αποτελέσματα του εννοιολογικού σχεδιασμού, να δημιουργήσει τις κατάλληλες δομές πληροφορίας σε επίπεδο λογικού μοντέλου αναπαράστασης δεδομένων. Για να επιτευχθεί αυτό θα

πρέπει να είναι γνωστό το συγκεκριμένο λογικό μοντέλο των δεδομένων που θα υποστηρίζει το Σύστημα Διαχείρισης Βάσεων Δεδομένων που θα χρησιμοποιηθεί.

4.4 Κανονικοποίηση λογικού σχεδιασμού βάσης δεδομένων

Η διαδικασία της **κανονικοποίησης (normalization process)** προτάθηκε από τον **Codd** το **1970**, και η βασική της λειτουργία, είναι να **δέχεται ως είσοδο το σχεσιακό σχήμα μιας βάσης δεδομένων και να εφαρμόζει πάνω του μια σειρά από ελέγχους προκειμένου να διαπιστώσει εάν ανήκει ή όχι σε κάποια κανονική μορφή**. Ο **Codd** εισήγαγε τρεις κανονικές μορφές, **την πρώτη, δεύτερη και τρίτη κανονική μορφή** οι οποίες συσχετίζονται με τις συναρτησιακές εξαρτήσεις που υφίστανται ανάμεσα στα πεδία των πινάκων της βάσης, ενώ αργότερα προτάθηκαν άλλες δύο κανονικές μορφές – **η τέταρτη και η πέμπτη** – οι οποίες στηρίζονται σε άλλου είδους εξαρτήσεις (**multivalued dependencies και join dependencies**).

Σε μία πρώτη περιγραφή, η **κανονικοποίηση εφαρμόζεται πάνω σε ένα σχεσιακό σχήμα, και προσπαθεί να το μετασχηματίσει σε μια νέα μορφή, η οποία να είναι απαλλαγμένη από τις διάφορες ανωμαλίες εισαγωγής, διαγραφής και τροποποίησης εγγραφών που παρουσιάσαμε σε προηγούμενες σελίδες**. Πιο συγκεκριμένα, αυτή η μεθοδολογία **αναλύει τη δομή των πινάκων του σχεσιακού σχήματος με βάση τα πρωτεύοντα κλειδιά τους και τις συναρτησιακές εξαρτήσεις που υφίστανται ανάμεσα στα πεδία τους**, και στη συνέχεια **ελέγχει αυτές τις δομές προ-κειμένου να διαπιστώσει εάν ικανοποιούν κάποιες συνθήκες**. Εάν οι εν λόγω συν-θήκες δεν ικανοποιούνται, λαμβάνει χώρα **διάσπαση των πινάκων της βάσης σε μικρότερους πίνακες**, κάθε ένας εκ των οποίων, ικανοποιεί πλέον αυτές τις συνθήκες.

Μια κανονική μορφή ορίζεται ως η «κατάσταση» στην οποία βρίσκεται μια σχέση όσον αφορά τους τύπους των εξαρτήσεων που υφίστανται ανάμεσα στα πεδία της. Επειδή γενικά αυτές οι εξαρτήσεις οδηγούν σε προβληματική συμπεριφορά του σχήματος της βάσης – όπως είναι για παράδειγμα η εμφάνιση επαναλαμβανόμενων πεδίων – θα πρέπει να απομακρυνθούν από τους πίνακες. Για το λόγο αυτό η **κανονικοποίηση πραγματοποιείται σε μια ακολουθία βημάτων, με το κάθε βήμα να περιλαμβάνει την απομάκρυνση από τη δομή των πινάκων κάποιου τύπου συναρτησιακής εξάρτησης, διαδικασία, η οποία έχει ως**

αποτέλεσμα, τη μετάβαση του σχεσιακού σχήματος, από μια κανονική μορφή σε μια άλλη. Πιο συγκεκριμένα, οι διαδικασίες που θα πρέπει να λάβουν χώρα για τη μετάβαση των σχεσιακών σχημάτων σε κάθε μια από τις πέντε κανονικές μορφές που έχουν ορισθεί, είναι οι ακόλουθες :

Πρώτη Κανονική Μορφή (1st Normal Form, 1NF): Για να φέρουμε ένα πίνακα σε πρώτη κανονική μορφή, **αν πρέπει να απομακρύνουμε τις επαναλαμβανόμενες ομάδες πεδίων**, έτσι ώστε η τομή μιας γραμμής και μιας στήλης του πίνακα, να αντιστοιχεί πάντα σε μια απλή τιμή.

Δεύτερη Κανονική Μορφή (2nd Normal Form, 2NF): Για να φέρουμε ένα πίνακα σε δεύτερη κανονική μορφή, θα πρέπει πρώτα να τον φέρουμε σε πρώτη κανονική μορφή, και στη συνέχεια **να απομακρύνουμε όλες τις μερικές συναρτησιακές εξαρτήσεις (partial dependencies)** που υφίστανται ανάμεσα στα πεδία του.

Τρίτη Κανονική Μορφή (3rd Normal Form, 3NF): Για να φέρουμε ένα πίνακα σε τρίτη κανονική μορφή, θα πρέπει πρώτα να τον φέρουμε σε δεύτερη κανονική μορφή, και στη συνέχεια **να απομακρύνουμε όλες τις μεταβατικές συναρτησιακές εξαρτήσεις (transitive dependencies)** που υφίστανται ανάμεσα στα πεδία του.

Τέταρτη Κανονική Μορφή (4th Normal Form, 4NF): Για να φέρουμε ένα πίνακα σε τέταρτη κανονική μορφή, θα πρέπει πρώτα να τον φέρουμε σε τρίτη κανονική μορφή και στη συνέχεια **να απομακρύνουμε όλες τις συναρτησιακές εξαρτήσεις που περιλαμβάνουν πεδία πολλαπλών τιμών (multivalued dependencies)**.

Πέμπτη Κανονική Μορφή (5th Normal Form, 5NF) : Για να φέρουμε ένα πίνακα σε πέμπτη κανονική μορφή θα πρέπει πρώτα να τον φέρουμε σε τέταρτη κανονική μορφή, και στη συνέχεια **να απομακρύνουμε όλες τις υπόλοιπες εξαρτήσεις που ενδεχομένως έχουν παραμείνει στη δομή του**, μετά την εφαρμογή των παραπάνω διαδικασιών.

Εκτός από τις πέντε κανονικές μορφές που περιγράψαμε παραπάνω, μια επι-πλέον κανονική μορφή που χρησιμοποιείται σε αρκετές περιπτώσεις, είναι η κανονική

μορφή των **Boyce-Codd (Boyce Codd Normal Form, BCNF)**. Η **BCNF**, μπορεί να θεωρηθεί ως **μια πιο αυστηρά διατυπωμένη 3NF**. Αυτό σημαίνει πως ένας πίνακας που βρίσκεται σε **BCNF** βρίσκεται αυτόματα και σε **3NF** – το αντίστροφο όμως δεν ισχύει. Η **BCNF** θα μελετηθεί αναλυτικά στις επόμενες σελίδες. Στο σχήμα που ακολουθεί αναπαρίστανται με γραφικό τρόπο, τα βήματα που θα πρέπει να ακολουθήσουμε προκειμένου να μετασχηματίσουμε μια μη κανονικοποιημένη σχέση από την πρώτη έως και την πέμπτη κανονική μορφή.

ΚΕΦΑΛΑΙΟ 5

Η ΓΛΩΣΣΑ SQL

5.1 Εισαγωγή

Η SQL (Structured Query Language) είναι μια υπολογιστική γλώσσα που έχει σχεδιαστεί για την ανάκτηση και διαχείριση δεδομένων στις σχεσιακές βάσεις.

Κατά πολλούς, η χρήση της SQL είναι ο σημαντικότερος λόγος επιτυχίας των σχεσιακών βάσεων δεδομένων. Αυτό οφείλεται στην απλότητά της σε σχέση με τις κλασσικές γλώσσες προγραμματισμού αλλά και διότι η συγκεκριμένη γλώσσα αποτελεί ένα πρότυπο διεθνών οργανισμών που προσφέρει σημαντική συμβατότητα μεταξύ των διαφορετικών Συστημάτων Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΔ).

Η συμβατότητα αυτή είναι εξαιρετικά χρήσιμοι για τους χρήστες των ΒΔ για διάφορους λόγους. Για παράδειγμα σκεφτείτε την περίπτωση ενός χρήστη ο οποίος δεν μένει ευχαριστημένος από το ΣΔΒΔ που χρησιμοποιεί. Η μεταφορά σε ένα άλλο ΣΔΒΔ είναι μια σχετικά απλή διαδικασία αν χρησιμοποιείται η τυποποιημένη SQL. Επίσης η συμβατότητα που προσφέρει η SQL επιτρέπει σε ένα πρόγραμμα εφαρμογής να χρησιμοποιεί και να καλεί συγχρόνως δύο διαφορετικά ΣΔΒΔ χωρίς να διαφοροποιείται η γλώσσα των βάσεων δεδομένων που χρησιμοποιείται. Βέβαια στην πράξη τα διάφορα εμπορικά ΣΔΒΔ προσφέρουν την SQL με διαφοροποιήσεις. Αλλά και αυτήν την περίπτωση αν ο χρήστης επιμένει να χρησιμοποιεί το υποσύνολο της SQL που είναι μέρος του προτύπου αλλά και τα ΣΔΒΔ υλοποιούν το συγκεκριμένο πρότυπο, τότε τα παραπάνω πλεονεκτήματα παραμένουν.

Η ιστορία της SQL αρχίζει από τις αρχές της δεκαετίας του 1970. Κατά τη διάρκεια της δεκαετίας αυτής, δημιουργήθηκε από ερευνητική ομάδα της IBM η πρόγονος της SQL, που ονομάστηκε SEQUEL (Structured English Query Language). Η SEQUEL αναπτύχθηκε για την διαχείριση των δεδομένων μίας πειραματικής σχεσιακής βάσης δεδομένων γνωστής ως SYSTEM R που δημιουργήθηκε κατά τα πρότυπα που έθεσε ο πρωτοπόρος στον τομέα Edgar Codd. Το ακρώνυμο SEQUEL αργότερα άλλαξε σε SQL διότι το πρώτο ήδη χρησιμοποιούταν από άλλον οργανισμό. Σε εμπορικό

επίπεδο, από τις πρώτες εταιρίες που χρησιμοποίησαν τη νέα γλώσσα ήταν η Ingers και στα τέλη της δεκαετίας του 1970 η Relational Software που αργότερα ονομάστηκε σε Oracle. Σήμερα, η SQL χρησιμοποιείται από όλα σχεδόν τα εμπορικά ΣΔΒΔ που υποστηρίζουν τις σχεσιακές βάσεις δεδομένων.

Όπως αναφέρθηκε προηγουμένως, η σημαντική δύναμη της SQL οφείλεται εν μέρει στην τυποποίησή της. Η γλώσσα αναγνωρίστηκε ως πρότυπο το 1986 από τον οργανισμό ANSI και το 1987 από τον οργανισμό ISO. Τότε δημιουργήθηκε το πρώτο πρότυπο γνωστό ως SQL – 86 ή απλώς SQL. Η επόμενη σημαντική αναθεώρηση της γλώσσας ως πρότυπο, έγινε το 1992 κατά την οποία δημιουργήθηκε μία σημαντικά εκτεταμένη έκδοση της γλώσσας που έγινε γνωστή ως SQL – 92 η SQL2. Το 1999 δημιουργήθηκε η SQL3, το 2003 και το 2006 δημιουργήθηκε αντίστοιχα η SQL2003 και η SQL2006 με νέες δυνατότητες που σχετίζονται με τη συνεργασία με την XML και το 2008 δημιουργήθηκε η πλέον πρόσφατη έκδοση κατά την οποία δόθηκε έμφαση στις παραθυρικές λειτουργίες της γλώσσας.

Η SQL είναι μία πλήρης γλώσσα βάσεων δεδομένων και διαθέτει εντολές για τον ορισμό δεδομένων όπως για παράδειγμα τη δημιουργία πινάκων, για ερωτήματα ανάκτησης δεδομένων για ενημερώσεις, λόγου χάρη, τροποποίησης δεδομένων. Επομένως είναι ταυτόχρονα Γλώσσα Ορισμού Δεδομένων αλλά και Γλώσσα Χειρισμού Δεδομένων. Επιπλέον, με την SQL μπορούμε να ορίσουμε όψεις, να προσδιορίσουμε κανόνες ασφαλείας και δικαιοδοσίες να προσδιορίσουμε τους κανόνες ακεραιότητας και να χειριστούμε τις δοσοληψίες της ΒΔ . Τέλος, η SQL έχει τη δυνατότητα συνεργασίας με άλλες γλώσσες προγραμματισμού όπως είναι η Pascal και η C, ώστε μέσα από αυτές τις γλώσσες να καλούνται ρουτίνες γραμμένες σε SQL.

5.2 Αναζήτηση/ανάκτηση Δεδομένων στην SQL

Η αναζήτηση/ανάκτηση δεδομένων στην SQL γίνεται με τα ερωτήματα της μορφής:

SELECT (λίστα πεδίων)

FROM (λίστα πινάκων)

WHERE (συνθήκη)

Όπου

(λίστα πεδίων) είναι τα ονόματα των πεδίων που θέλουμε να ανακτηθούν,

(λίστα πινάκων) είναι το όνομα του πίνακα ή των πινάκων από τον οποίο θα γίνει η ανάκτηση της πληροφορίας και

(συνθήκη) είναι μια λογική έκφραση (κριτήριο) που μπορεί να χρησιμοποιεί τους τελεστές >, <, >=, <=, <>, =, AND, OR, NOT και προσδιορίζει τις εγγραφές που θα επιλέγουν.

Σημείωση: όταν παρατίθενται περισσότερα από ένα ονόματα πεδίων ή πινάκων τότε αυτά χωρίζονται με κόμμα (,).

Εν γένει, κατά την εκτέλεση ενός ερωτήματος ανάκτησης δεδομένων, πρώτα δημιουργούνται οι πίνακες του αποτελέσματος, όπως αυτοί περιγράφονται στην εντολή FROM, στην συνέχεια επιλέγονται οι κατάλληλες εγγραφές του πίνακα με βάση την εντολή WHERE και τέλος εκτελούνται οι προβολές στ πεδία που προσδιορίζονται με την εντολή SELECT.

Η χρήση της εντολής WHERE είναι προαιρετική και επομένως μπορεί να παραλειφθεί. Στην εντολή SELECT αντί των ονομάτων πεδίων, μπορούμε να χρησιμοποιήσουμε τον χαρακτήρα «*». Τότε, το αποτέλεσμα θα είναι η προβολή όλων των πεδίων του πίνακα αποτελέσματος.

5.3 Διάταξη (Ταξινόμηση) Αποτελεσμάτων

Η ταξινόμηση των αποτελεσμάτων στην SQL γίνεται με την εντολή ORDER BY. Η εντολή ORDER BY ακολουθεί τη WHERE και χρησιμοποιεί τις παραμέτρους “ASC” (για ταξινόμηση αύξουσας σειράς) και “DESC” (για ταξινόμηση φθίνουσας σειράς). Η σύνταξη του ερωτήματος με ταξινόμηση είναι:

SELECT (λίστα πεδίων)

FROM (λίστα πινάκων)

WHERE (συνθήκη)

ORDER BY πεδίο1 ASC ή DESC, πεδίο2 ASC ή DESC, ...

Η διάταξη μπορεί να είναι ως προς ένα ή περισσότερα πεδία του ίδιου ή διαφορετικών πινάκων. Όταν δεν προσδιορίζεται ο τρόπος ταξινόμησης (ASC ή DESC), τότε εννοείται αύξουσα ταξινόμηση.

5.4 Σύγκριση συμβολοσειρών

Η σύγκριση συμβολοσειρών στην SQL υποστηρίζεται από δύο ειδικούς χαρακτήρες: τον % , ο οποίος σημαίνει «οποιαδήποτε συμβολοσειρά, οποιουδήποτε μήκους» και τον -, ο οποίος σημαίνει «ένας οποιοσδήποτε χαρακτήρας». Συντάσσονται με τον τελεστή LIKE και είναι ιδιαίτεροι χρήσιμοι για την ανάκτηση δεδομένων με βάση τη σύγκριση συμβολοσειρών.

Για παράδειγμα η έκφραση

LIKE '%κόζ'

Σημαίνει «οποιαδήποτε σειρά χαρακτήρων που τελειώνει σε κοζ», ενώ η εντολή

LIKE '-ικός'

Σημαίνει «οποιαδήποτε σειρά τεσσάρων χαρακτήρων που αρχίζει από έναν οποιοδήποτε χαρακτήρα και τελειώνει σε -ικός».

5.5 Η τιμή του NULL στην SQL

Η SQL επιτρέπει το κενό δηλαδή το NULL ως τιμή των πεδίων. Υπενθυμίζεται πως η τιμή του NULL δεν πρέπει να συγχέεται με την μηδενική τιμή για τα αριθμητικά πεδία ή με τον χαρακτήρα «κενό» που χρησιμοποιείται στις συμβολοσειρές. Αντίθετα, η τιμή NULL σημαίνει απουσία τιμής σε ένα πεδίο.

Με την έκφραση IS NULL ή IS NOT NULL μπορούμε να διερευνήσουμε αν η τιμή ενός πεδίου είναι κενή. Για παράδειγμα, το παρακάτω ερώτημα δίνει όλους τους τίτλους των συγγραμμάτων για τα οποία είναι καταγεγραμμένος ο αριθμός σελίδων:

SELECT Τίτλος

FROM Σύγγραμμα

WHERE Αρ Σελίδων IS NOT NULL;

5.6 Ερωτήματα σύνδεσης πινάκων

Η πράξη της σύνδεσης πινάκων αναφέρθηκε και εξηγήθηκε λεπτομερώς στη Σχεσιακή Άλγεβρα. Στην SQL η πράξη της σύνδεσης υλοποιείται με την εντολή JOIN.

5.7 Εσωτερική Άλγεβρα

Η πράξη της εσωτερικής σύνδεσης υλοποιείται με την εντολή INNER JOIN (ή απλώς JOIN). Στο αποτέλεσμα της εσωτερικής σύνδεσης λαμβάνονται οι εγγραφές των συνδεόμενων πινάκων, που ικανοποιούν το κριτήριο της δοσμένης συνθήκης ως προς το πεδίο σύνδεσης. Για παράδειγμα, ας θεωρήσουμε το πρόβλημα της εύρεσης της ονομασίας και του έτους ίδρυσης των τμημάτων και του επώνυμου των καθηγητών που είναι οι αντίστοιχοι προϊστάμενοι τμημάτων. Οι πληροφορίες αυτές μπορούν να βρεθούν συνδέοντας τους πίνακες Καθηγητής και Τμήμα με πεδίο σύνδεσης τον κωδικό καθηγητή και με συνθήκη την ισότητα. Υπενθυμίζεται πως ο κωδικός του προϊστάμενου καθηγητή στον πίνακα τμήμα αναφέρεται με το όνομα πεδίου Προϊστάμενος. Στο αποτέλεσμα πρέπει να συνδυαστούν οι εγγραφές των δύο πινάκων, που έχουν ίδια τιμή στα πεδία Κωδικός K και Προϊστάμενος.

Το ερώτημα σε SQL είναι:

SELECT Ονομασία, ΈτοςΊδρυσης, Επώνυμο

FROM Τμήμα INNER JOIN Καθηγητής ON

Τμήμα Προϊστάμενος= Καθηγητής, ΚωδικόςK

ΚΕΦΑΛΑΙΟ 6

ΣΥΣΤΗΜΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ - ACCESS

6.1 Εισαγωγή

Το παρόν κείμενο αφορά το Σύστημα Διαχείρισης Βάσεων Δεδομένων (ΣΔΒΔ) της Access. Το συγκεκριμένο ΣΔΒΔ είναι παγκοσμίως από τα πλέον δημοφιλή. Κυρίως απευθύνεται στη δημιουργία εφαρμογών διαχείρισης δεδομένων για μικρές επιχειρήσεις και οργανισμούς, ενώ παράλληλα προσφέρει τα απαραίτητα εργαλεία για την σύνδεση και συνεργασία (π.χ. μέσω της τεχνολογίας, ActiveX) με τα υπόλοιπα προϊόντα της Microsoft μεταξύ των οποίων με το Word, το Excel, το PowerPoint και το SQL Server. Τέλος η Access μπορεί να χρησιμοποιηθεί ως ένας άμεσος και δυναμικός τρόπος για την προσπέλαση πληροφοριών σε ενδοδίκτυα (intranets) ή στον παγκόσμιο ιστό (World Wide Web)

Οι πίνακες (σχέσεις) συνιστούν τη βασική δομή οργάνωσης και διαχείρισης δεδομένων για την Access. Η εξοικείωση με την δημιουργία και τον χειρισμό τους αποτελούν βασικές γνώσεις που πρέπει να αποκτηθούν από την αρχή της μελέτης του συγκεκριμένου ΣΔΒΔ. Ένας πίνακας αποτελείται από γραμμές που αναφέρονται ως εγγραφές ή πλειάδες. Κάθε εγγραφή προσδιορίζει μια συλλογή από ατομικές τιμές (στήλες) που ονομάζονται πεδία ή γνωρίσματα. Κάθε πεδίο διαθέτει το δικό του όνομα. Επομένως οι πίνακες (tables) και τα πεδία (fields) αποτελούν τα βασικά στοιχεία οργάνωσης των πληροφοριών Access.

Παρακάτω θα προσφερθούν όλες οι απαραίτητες πληροφορίες σχετικά με την δημιουργία και διαχείριση των πινάκων και αναφερθούν θέματα όπως: ο τρόπος κλήσης και εισαγωγής στο σύστημα και οι υπάρχοντες τύποι δεδομένων, οι γενικές ιδιότητες που μπορούν να προσδιοριστούν για τα δεδομένα, η δυνατότητα των ευρετηρίων, η δυνατότητα μορφοποίησης των διάφορων τύπων δεδομένων και οι κανόνες επικυρώσεις της Access, οι ιδιότητες εμφάνισης των δεδομένων, ο τρόπος προσδιορισμού του πρωτεύοντος κλειδιού, η δημιουργία των πινάκων της ΒΔ «Εκπαιδευτικό Ίδρυμα» στην Access και ο τρόπος προσδιορισμού των συσχετίσεων και των κανόνων αναφορικής ακεραιότητας μεταξύ των πινάκων.

6.2 Εκκίνηση της Access

Για την εκκίνηση Access , μπορεί να ενεργοποιηθεί το αντίστοιχο εικονίδιο από την επιφάνεια εργασίας (desktop) του λειτουργικού συστήματος. Εναλλακτικά μπορεί να επιλεγεί ?έναρξη (Start) -> Προγράμματα (programs) ->Microsoft Access.

Το αρχικό μενού δίνει την δυνατότητα επεξεργασίας μιας υπάρχουσας βάσης που είναι αποθηκευμένη στον υπολογιστή (επιλογή: Κενή βάση δεδομένων) καθώς τέλος και του ανοίγματος οδηγού σελίδων και έργων βάσης δεδομένων της Access.Όταν χρειάζεται να δημιουργηθεί μια ΒΔ από την αρχή, αρκεί να ενεργοποιηθεί η επιλογή κενή βάση δεδομένων της Access .

Στο σημείο αυτό θα πρέπει να επιλεγεί εκ των προτέρων ο φάκελος αλλά και η ονομασία της καινούργιας βάσης δεδομένων που θα δημιουργηθεί. Όταν γίνει η συμπλήρωση των αντίστοιχων πεδίων, ενεργοποιείται η επιλογή Δημιουργία.

Έστω ότι ο χρήστης έδωσε ως όνομα β΄σης δεδομένων ΕΚΠΑΙΔΕΥΤΙΚΟ ΙΔΡΥΜΑ. Επομένως, μετά την ολοκλήρωση της δημιουργίας, εμφανίζεται το βασικό παράθυρο της Access.

Το παράθυρο του σχήματος αποτελείται από δύο μέρη. Όταν ο χρήστης επιλέξει κάτι από το αριστερό μέρος, τα αντίστοιχα περιεχόμενα θα εμφανίζονται στο δεξιό τμήμα του παραθύρου. Η Access διαθέτει επτά κύρια χαρακτηριστικά τα οποία ονομάζονται αντικείμενα:

Πίνακες (tables): χρησιμοποιούνται για την αποθήκευση της πληροφορίας και αποτελούνται από σειρές (εγγραφές ή πλειάδες) και στήλες (πεδία ή γνωρίσματα). Μια βάση δεδομένων στην Access αποτελείται από τουλάχιστον έναν πίνακα, όλα τα άλλα αντικείμενα είναι προαιρετικά.

Ερωτήματα (queries): διαχειρίζονται (φιλτράρουν) τα δεδομένα προκειμένου να ανακτηθούν οι επιθυμητές πληροφορίες.

Φόρμες (forms): δημιουργούν οπτικές φόρμες (οθόνες) για την αλληλεπίδραση με τη ΒΔ (πχ την εισαγωγή και επεξεργασία των δεδομένων), αναφορικά με πίνακες αλλά και ερωτήματα.

Εκθέσεις (reports): εκτυπώνουν τους υπάρχοντες πίνακες και τα αποτελέσματα των ερωτημάτων σε μορφοποιημένη διάταξη.

Σελίδες (pages): δημιουργούν HTML σελίδες από τη βάση δεδομένων, με ένα ιδιαίτερα εύκολο και φιλικό προς το χρήστη τρόπο.

Μακροεντολές (macros): δημιουργούν αυτοματοποιημένες ενέργειες. Για παράδειγμα, μια μακροεντολή μπορεί αυτόματα να ανοίγει και να εκτυπώνει μια έκθεση.

Λειτουργικές μονάδες (modules): υποστηρίζουν σύνθετες μακροεντολές χαμηλότερου επιπέδου. Μια λειτουργική μονάδα αυτοματοποιεί ενέργειες χρησιμοποιώντας ως γλώσσα προγραμματισμού την Visual Basic (VB)

6.3 Δημιουργία πίνακα- τύποι δεδομένων

Για τη δημιουργία πίνακα ενεργοποιείται η επιλογή Δημιουργία πίνακα σε προβολή σχεδίασης. Εφόσον είμαστε σε θέση να προσδιορίσουμε το όνομα και τον τύπο των πεδίων του πίνακα καθώς και τη μορφή των δεδομένων που πρόκειται να εισάγουμε είμαστε σε θέση να δημιουργήσουμε πίνακες της βάσης δεδομένων. Στην επιλογή όνομα πεδίου εισάγετε το όνομα του κάθε πεδίου της αρεσκείας του χρήστη ανάλογα φυσικά με τις ανάγκες της εφαρμογής που θέλει να δημιουργήσει και στην επιλογή Τύπος δεδομένων καθορίζεται ο αντίστοιχος τύπος δεδομένων του πεδίου.

Η Access υποστηρίζει τους ακόλουθους τύπους δεδομένων:

Κείμενο : συμβολοσειρές , δηλαδή σύνολα χαρακτήρων. Το μέγιστο πλήθος χαρακτήρων σε μια συμβολοσειρά μπορεί να είναι έως 255.

Υπόμνημα : συμβολοσειρές πολύ μεγάλου μεγέθους. Ο μέγιστος αριθμός χαρακτήρων σε ένα υπόμνημα μπορεί να είναι 64.000 περίπου δηλαδή 18 σελίδες κειμένου .

Αριθμός: αριθμητικά δεδομένα που μπορούν να χρησιμοποιηθούν σε μαθηματικούς υπολογισμούς .

Ημερομηνία / ώρα : δεδομένα ημερομηνίας ή και ώρας.

Νομισματική μονάδα : νομισματικές τιμές.

Αυτόματη αρίθμηση: αριθμητική τιμή η οποία αυξάνει αυτόματα με την προσθήκη κάθε καινούργιας εγγραφής .

Ναι / όχι : λογικές τιμές όπως με / όχι, αλήθεια / ψέματα και λοιπά .

Αντικείμενο OLE : χρησιμοποιείται για την εισαγωγή αντικειμένων στη ΒΔ από άλλο πρόγραμμα, όπως για παράδειγμα η εισαγωγή μίας φωτογραφίας.

Υπέρ- σύνδεση : κείμενο που χρησιμοποιείται ως σύνδεσμος σε ένα αρχείο στον υπολογιστή μας ή σε μία σελίδα στο διαδίκτυο.

Ιδιαίτερη αναφορά γίνεται στον οδηγό αναζήτησης ο οποίος επιτρέπει, η τιμή σε ένα συγκεκριμένο πεδίο να προέρχεται είτε από τιμές πεδίου σε κάποιον άλλο πίνακα, είτε από μία λίστα τιμών που δίνεται αρχικά κατά τη δημιουργία του συγκεκριμένου πίνακα, η τέλος από κάποιο ερώτημα.

6.4 Γενικές ιδιότητες δεδομένων

Από τη στιγμή που έχει επιλεγεί ο τύπος δεδομένων ενός παιδιού, στη συνέχεια θα πρέπει να καθοριστούν οι ιδιότητες τους . Αυτές διακρίνονται σε γενικές και σε ιδιότητες που αναφέρονται στην εμφάνιση.

Για παράδειγμα, γενικές ιδιότητες ενός αριθμητικού πεδίου είναι ο μέγιστος αριθμός των ψηφίων που μπορεί να έχει ή το πλήθος των δεκαδικών ψηφίων που διαθέτει. Ανάλογα με τον τύπο των δεδομένων που έχει επιλεχθεί διακρίνουμε αντίστοιχες ιδιότητες.

6.4.1 Ευρετήρια

Ιδιαίτερη αναφορά γίνεται στην δυνατότητα της Access για την υποστήριξη των αρτηριών, μέσα από τις γενικές ιδιότητες ενός πεδίου. Ο χρήστης μπορεί να δημιουργήσει ευρετήρια για ένα ή περισσότερα πεδία, τα οποία χρησιμοποιούνται συχνά για λόγους ανάκτησης. Ωστόσο, δεν πρέπει να γίνεται κατάχρηση δημιουργίας ευρετηρίων, γιατί υπάρχει ο κίνδυνος κακής απόδοσης της βάσης δεδομένων.

Σημειώνεται πως δεν είναι δυνατή η δημιουργία ευρετηρίων σε πεδία με τύπο δεδομένων υπέρ – σύνδεσμο, υπόμνημα και αντικείμενο OLE. Τα πεδία που έχουν επιλεγεί ως πρωτεύοντα κλειδιά εισάγονται αυτόματα σε ευρετήριο. Εάν πραγματοποιηθεί η δημιουργία ευρετηρίου για ένα πεδίο τότε δεν επιτρέπεται η εισαγωγή διπλότυπων τιμών για το συγκεκριμένο πεδίο.

6.4.2 Μορφοποίηση Δεδομένων

Για την μορφοποίηση των δεδομένων, ο χρήστης μπορεί να υιοθετήσει τις επιλογές της Access ή να δημιουργήσει δικές του. Για τη μορφοποίηση ημερομηνία και ώρα, χρησιμοποιούνται χαρακτηριστικά όπως για παράδειγμα ο διαχωριστής της ώρας, δηλαδή : , Ο διαχωριστής της ημερομηνίας δηλαδή / , - κλπ, η ημέρα σε ένα ή δύο ψηφία d κλπ. Για την μορφοποίηση τύπου αριθμού και νομισμάτων, είναι δυνατόν να χρησιμοποιηθούν σύμβολα όπως τα: # 0 . , € %

Αντίθετα, για την μορφοποίηση πεδίων τύπου κειμένου η Access προσφέρει σχετικά λιγότερες προεπιλογές. Ο χρήστης έχει τη δυνατότητα να εμφανίζει όλους τους χαρακτήρες ως πεζούς ή κεφαλαίους, να προσδιορίζει τους απαραίτητους προς εμφάνιση χαρακτήρες, να γεμίζει το διαθέσιμο χώρο μίας συμβολοσειράς με συγκεκριμένο χαρακτήρα ή να εμφανίζει μία συμβολοσειρά με κάποιο συγκεκριμένο χρωματισμό.

6.4.3 Επικύρωση Δεδομένων

Η χρήση των κανόνων επικύρωσης είναι ιδιαίτερα σημαντική, καθώς δίνει την ευκαιρία για ελαχιστοποίηση των λαθών κατά τη διάρκεια της εισαγωγής των δεδομένων. Ουσιαστικά γίνεται ένας έλεγχος εάν τα δεδομένα εισαγωγής ικανοποιούν το κριτήριο που τέθηκε . Σε περίπτωση που δεν ικανοποιείται ο κανόνας επικύρωσης, τότε εμφανίζεται ένα μήνυμα λάθους. Οι κανόνες επικύρωσης χρησιμοποιούνται ευκολότερα και αποτελεσματικότερα σε περίπτωση δεδομένων τυπου αριθμού νομισμάτων και ημερομηνίας / ώρας, αλλά δεν αποκλείεται η χρήση τους και με τους υπόλοιπους τύπους δεδομένων.

Στις ιδιότητες ενός πεδίου, υπάρχουν δύο πλαίσια που σχετίζονται με την επικύρωση δεδομένων:

οι κανόνες επικύρωσης, που καθορίζουν τα χαρακτηριστικά για την εισαγωγή των δεδομένων στο πεδίο και το κείμενο επικύρωσης, το οποίο εμφανίζεται όταν παραβιάζεται ο αντίστοιχος κανόνας επικύρωσης.

Στους κανόνες επικύρωσης μπορεί να γίνει χρήση εν γένει των συμβόλων που χρησιμοποιούνται στα ερωτήματα.

6.4.4 Μάσκα Εισαγωγής

Η μάσκα εισαγωγής περιγράφει το πλήθος και τον τύπο των δεδομένων, που θα εισαχθούν σε ένα πεδίο. Γενικά, η δημιουργία μάσκας εισαγωγής θεωρείται ιδιαίτερα βοηθητική για την εισαγωγή των δεδομένων. Ο χρήστης μπορεί να δημιουργήσει μάσκα εισαγωγής δεδομένων με δύο τρόπους :

A) Επιλέγοντας το πλήκτρο ... της δημιουργίας μάσκας στο πεδίο μας εισαγωγής, εμφανίζεται το παράθυρο του οδηγού μάσκας εισαγωγής . Με τον τρόπο αυτό η εισαγωγή μάσκας γίνεται εύκολα και γρήγορα. Ο οδηγός μάσκας εισαγωγής, δίνει τη δυνατότητα να δημιουργηθούν μάσκες για πεδία που έχουν σχέση με ημερομηνία, ώρα, τηλεφωνικούς αριθμούς, Αριθμό Φορολογικού Μητρώου ή αστυνομικής ταυτότητας και ταχυδρομικού κώδικα.

B) Με την εισαγωγή χαρακτήρων στο πεδίο μάσκας εισαγωγής κατευθείαν από το χρήστη .

6.5 Η ιδιότητα Πεδίου Εμφάνιση

Από τη στιγμή που έχουν καθοριστεί γενικές ιδιότητες ενός πεδίου, υπάρχει η δυνατότητα να επιλέγουν και ορισμένες άλλες γιορτές που απεικονίζονται στην καρτέλα εμφάνιση. Η καρτέλα αυτή χρησιμοποιείται στην περίπτωση που τιμές του πεδίου προκύπτουν είτε από κάποιο άλλο πίνακα ή ερώτημα της Access, είτε από κάποια συγκεκριμένη προκαθορισμένη λίστα τιμών.

6.6 Πρωτεύον κλειδί

Από τη στιγμή που έχουν καθοριστεί όλα τα πεδία του πίνακα καθώς και ιδιότητες αυτών, πρέπει να γίνει και η επιλογή του πρωτεύοντος κλειδιού. Ως πρωτεύον κλειδί

θα πρέπει να επιλεγεί το πεδίο εκείνο του πίνακα που καθορίζει μοναδικά κάθε εγγραφή. Ένα τέτοιο παράδειγμα πρωτεύοντος κλειδιού μπορεί να είναι ο αριθμός αστυνομικής ταυτότητας, ο Αριθμός Φορολογικού Μητρώου ή ένας μοναδικός κωδικός.

Το κλειδί καθορίζεται πριν την αποθήκευση του πίνακα. Θα πρέπει να προσδιοριστεί το πεδίο που θα είναι το πρωτεύον κλειδί και στη συνέχεια να επιλεγεί πρωτεύον κλειδί.

Η απενεργοποίηση ενός πεδίου, από πρωτεύον κλειδί, γίνεται με αντίστοιχο τρόπο. Εάν δεν δηλωθεί κάποιο πεδίο ως πρωτεύον κλειδί, τότε η Access θα εισάγει αυτόματα ένα πεδίο αυτόματης αρίθμησης ως πρωτεύον κλειδί στον πίνακα .

Υπάρχει η περίπτωση, το κλειδί σε ένα πίνακα να είναι σύνθετο, δηλαδή να αποτελείται από συνδυασμό δύο ή περισσότερων πεδίων. Τότε, μπορεί να δηλωθεί το πρώτο από τα πεδία ως πρωτεύον κλειδί και στη συνέχεια, με πατημένο το πλήκτρο CTRL , να γίνει η ίδια διαδικασία με το δεύτερο πεδίο, με το τρίτο και ούτω καθεξής, μέχρι να καθοριστεί πλήρως το σύνθετο πρωτεύον κλειδί.

Στην περίπτωση των πεδίων που είναι ξένα κλειδιά, θα πρέπει ληφθεί μέριμνα ώστε αυτά να έχουν ακριβώς τις ίδιες ιδιότητες πεδίου με τα αντίστοιχα πρωτεύοντα κλειδιά, από τα οποία δημιουργήθηκαν.

6.7 Δηλώσεις Συσχετίσεων

Αφού υλοποιηθούν και οι υπόλοιποι πίνακες, θα πρέπει να δηλωθούν οι συσχετίσεις. Από την γραμμή μενού ενεργοποιούνται οι επιλογές εργαλεία → σχέσεις ή από την γραμμή εργαλείων επιλέγεται το πλήκτρο των σχέσεων.

Ως αποτέλεσμα, εμφανίζονται όλοι οι πίνακες οι οποίοι είναι αποθηκευμένοι στην βάση δεδομένων. Στη συνέχεια θα πρέπει να επιλεγούν όλοι πίνακες και να δηλωθούν οι συσχετίσεις που υπάρχουν μεταξύ τους .

Για τη δήλωση των συσχετίσεων θα πρέπει να συνδεθούν τα αντίστοιχα πρωτεύοντα κλειδιά, καθώς επίσης και τα ξένα κλειδιά με τα αντίστοιχα πρωτεύοντα κλειδιά. Για να πραγματοποιηθεί μία τέτοια σύνδεση θα πρέπει να γίνει διπλό κλικ στην γκρίζα

επιφάνεια. Ο χρήστης στη συνέχεια επιλέγει το πλήκτρο Δημιουργία νέου... . Γενικά, ως αριστερός πίνακας θεωρείται ο πίνακας από την πλευρά του ένα, ενώ ως δεξιός πίνακας αυτός που βρίσκεται από την πλευρά του πολλά (N) . Υπενθυμίζεται πως ο πίνακας Παρακολουθεί προέρχεται από την συσχέτιση M:N μεταξύ των πινάκων Σπουδαστής και Μάθημα.

Η συγκεκριμένη συσχέτιση – αλλά και οποιαδήποτε συσχέτιση M: N στην Access – αναλύεται ως εξής: μεταξύ των πινάκων Σπουδαστής και Παρακολουθεί υπάρχει συσχέτιση ένα προς πολλά για το πεδίο AM Σπουδαστής το οποίο προέρχεται από το πεδίο AM του πίνακα Σπουδαστής. Μεταξύ των πινάκων Μάθημα και Παρακολουθεί υπάρχει συσχέτιση ένα προς πολλά για το πεδίο Κωδ Μαθήματος το οποίο προέρχεται από το πεδίο Κωδικός M του πίνακα Μάθημα.

Επομένως η δήλωση της συσχέτισης M:N μεταξύ των πινάκων Σπουδαστή και Μάθημα θα πρέπει να δηλωθούν δύο συσχετίσεις 1:N και συγκριμένα η συσχέτιση μεταξύ Σπουδαστής – Παρακολουθεί και Μάθημα-Παρακολουθεί.

Για να δημιουργηθεί η συσχέτιση θα πρέπει να ενεργοποιηθεί η επιλογή δημιουργία η οποία τώρα φαίνεται με έντονο χρώμα και όχι αχνό όπως προηγουμένως. Εμφανίζονται τρεις επιλογές, συγκεκριμένα:

ενεργοποίηση ακεραιότητας αναφορών: όταν τίθεται θέμα διαγραφής, εξασφαλίζει την μη διαγραφή των συσχετιζόμενων δεδομένων από τον κυρίως πίνακα,

διαδοχική ενημέρωση των σχετικών εγγράφων: όταν γίνεται εισαγωγή των δεδομένων η Access αυτόματα ενημερώνει και τα δεδομένα που ταιριάζουν στους συσχετιζόμενους πίνακες,

Διαδοχική διαγραφή των σχετικών εγγράφων: όταν σβήνεται μια εγγραφή από τον κυρίως πίνακα, η Access διαγραφεί και τις αντίστοιχες εγγραφές στους συσχετιζόμενους πίνακες.

Όταν δημιουργείται συσχέτιση μεταξύ δύο πινάκων, σε αρκετές περιπτώσεις είναι σημαντικό να είναι ενεργοποιημένη η επιλογή της ακεραιότητας των αναφορών. Αυτό εμποδίζει τη δημιουργία μη συνεπών δεδομένων εξαιτίας λαθών ή κακής διαχείρισης. Για να μπορεί να ενεργοποιηθεί η ακεραιότητα θα πρέπει τα

συσχετιζόμενα πεδία να έχουν την ίδια ακριβώς μορφή, δηλαδή οι ιδιότητες του τύπου των δεδομένων να είναι ακριβώς οι ίδιες και στους δύο πίνακες.

Ως εναλλακτικό τρόπο δημιουργίας μίας συσχέτισης, μπορεί να επιλεγεί το πεδίο από την πλευρά του ένα και στη συνέχεια με τη μέθοδο του drag n drop , αυτό να συνδυαστεί με το πεδίο από την πλευρά των πολλών. Έπειτα μπορεί να ολοκληρωθεί η δήλωση των υπολοίπων συσχετίσεων μεταξύ των πινάκων.

Για την επεξεργασία μιας υπάρχουσας συσχέτισης, θα πρέπει πρώτα να ενεργοποιηθεί η συσχέτιση (αριστερό κλικ) και στη συνέχεια με δεξί κλικ λαμβάνεται το σχήμα 10.20.

Με την ενεργοποίηση της επιλογής διαγραφή, διαγράφεται η συγκεκριμένη σχέση, ενώ με την απενεργοποίηση της επιλογής Επεξεργασία σχέσης..., εμφανίζεται το σχήμα 10.15.

Κλείνοντας το παράθυρο των σχέσεων πραγματοποιείται η συνολική αποθήκευση αυτών, οπότε και γίνεται η επιστροφή στην αρχική εικόνα δημιουργίας πινάκων. Η δημιουργία και η διαχείριση των πινάκων έχει ολοκληρωθεί.

ΚΕΦΑΛΑΙΟ 7

ΠΡΑΚΤΙΚΗ ΕΦΑΡΜΟΓΗ ΔΗΜΙΟΥΡΓΙΑΣ ΒΑΣΗΣ ΔΕΔΟΜΕΝΩΝ

7.1 Εισαγωγή

Στην προσπάθεια μας για την καλύτερη κατανόηση της εργασίας μας, και σε συνεργασία με τον καθηγητή μας, δημιουργήσαμε μια βάση δεδομένων για ένα λογιστικό γραφείο όπου θα παρακολουθούμε όλα τα έσοδα και τα έξοδα του καθώς επίσης θα μας δίνετε η δυνατότητα να έχουμε αναλυτικές αναφορές ανά πελάτη του γραφείου για τα έσοδα – έξοδα τους και το ποσό του ΦΠΑ ανά περίοδο.

Στην συνέχεια παρουσιάζουμε την βάση δεδομένων όπως την έχουμε δημιουργήσει.

7.2 Σκοπός – στόχος πρακτικής εφαρμογής

Ο βασικός στόχος που δημιουργήσαμε την βάση δεδομένων ήταν για να μπορέσουμε να έχουμε μια ποιο καλή γνώση της συγκεκριμένης εφαρμογής, καθώς επίσης να καταλάβουμε πολύ ποιο καλά αυτά που αναφέρουμε στο θεωρητικό μέρος της εργασίας μας.

Επίσης με βάση την εφαρμογή αυτή να μας δοθεί η δυνατότητα να μπορούμε να κάνουμε χρήση της εφαρμογής στην μελλοντική μας επαγγελματική πορεία.

7.3 Σχεδιασμός βάσης δεδομένων

Ο βασικός μας σχεδιασμός ήταν να δημιουργήσουμε μια βάση όπου θα υπάρχουν βασικά οι παρακάτω πίνακες :

- Πελάτες
- Μισθοδοσία
- Φπα
- Έσοδα – έξοδα

Με βάση αυτή την παραδοχή δημιουργήσαμε και συγκεκριμένες αναφορές και φόρμες όπου μας δίνουν την δυνατότητα να έχουμε αναλυτική εικόνα ανά πελάτη του λογιστικού γραφείου.

Επιλέξαμε να δημιουργήσουμε 30 εικονικούς πελάτες όπου σε μερικούς από αυτούς έχουμε καταχωρήσει εικονικές συναλλαγές έτσι ώστε να μπορέσουμε να έχουμε αποτελέσματα στις αναφορές και στις φόρμες.

7.4 Παρουσίαση βάσης δεδομένων

7.4.1. Δημιουργία πινάκων

Παρακάτω παρουσιάζουμε τους πίνακες που έχουμε δημιουργήσει :

Εικόνα εσόδων – εξόδων (Εικόνα 7.1)

Έχουμε δημιουργήσει τον πίνακα όπου υπάρχουν οι στήλες

- Id
- Τύπος
- Πελάτης id
- Μήνας
- Ημερομηνία καταχώρησης
- Πόσο

Εικόνα 7.1 Εικόνα εσόδων – εξόδων

ID	ΤΥΠΟΣ	ΠΕΛΑΤΗΣ_ID	ΕΤΟΣ	ΜΗΝΑΣ	ΗΜΕΡΙΑ_ΚΑΤ	ΠΟΣΟ	Προσθήκη νέου πεδίου
1	ΕΣΟΔΑ		2015	01	20/5/2015	100,00	
2	ΕΣΟΔΑ		2015	02	20/5/2015	150,00	
3	ΕΣΟΔΑ		2015	01	20/5/2015	90,00	
4	ΕΣΟΔΑ		2015	01	20/5/2015	50,00	
5	ΕΣΟΔΑ		2015	03	15/6/2015	120,00	
*	(Νέο)				27/7/2015	0,00	

Επίσης στον πίνακα 7.2 φαίνεται ο τρόπος που έχουμε κάνει την σχεδίαση του πίνακα

Εικόνα 7.2 Εικόνα εσόδων – εξόδων

Όνομα πεδίου	Τύπος δεδομένων	Περιγραφή
Αυτόματη Αρίθμηση	Αυτόματη Αρίθμηση	
ΤΥΠΟΣ	Κείμενο	
ΠΕΛΑΤΗΣ_ID	Αριθμός	
ΕΤΟΣ	Κείμενο	
ΜΗΝΑΣ	Κείμενο	
ΗΜΕΡΙΑ_ΚΑΤΑΧΩΡΗΣΗΣ	Ημερομηνία/Ωρα	
ΠΟΣΟ	Αριθμός	

Γενικές	Εμφάνιση
Μέγεθος πεδίου	Μεγάλος ακέραιος
Νέες τιμές	Βημιακά
Μορφή	
Λεξιάντα	
Με ευρετήριο	Ναι (Δεν επιτρέπονται διπλότυπα)
Εξυπνες ετικέτες	
Στάθιση κειμένου	Γενική

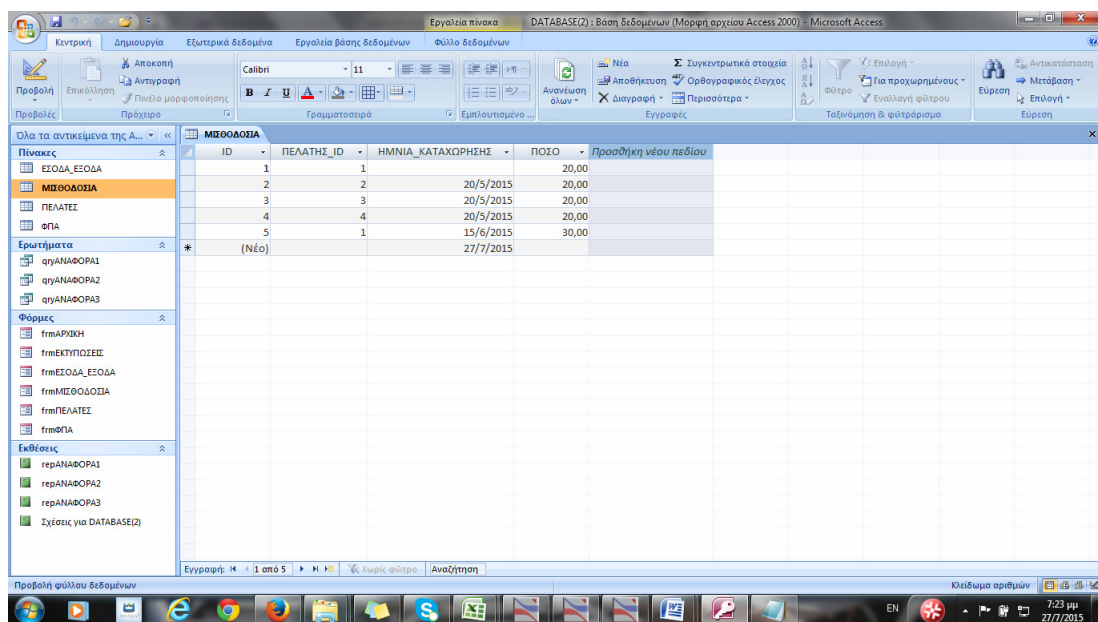
Τα ονόματα πεδίων μπορούν να έχουν έως 64 χαρακτήρες (με τα κενά). Για βοήθεια πατήστε F1.

Εικόνα μισθοδοσίας (Εικόνα 7.3)

Έχουμε δημιουργήσει τον πίνακα όπου υπάρχουν οι στήλες

- Id
- Πελάτης id
- Ημερομηνία καταχώρησης
- Πόσο

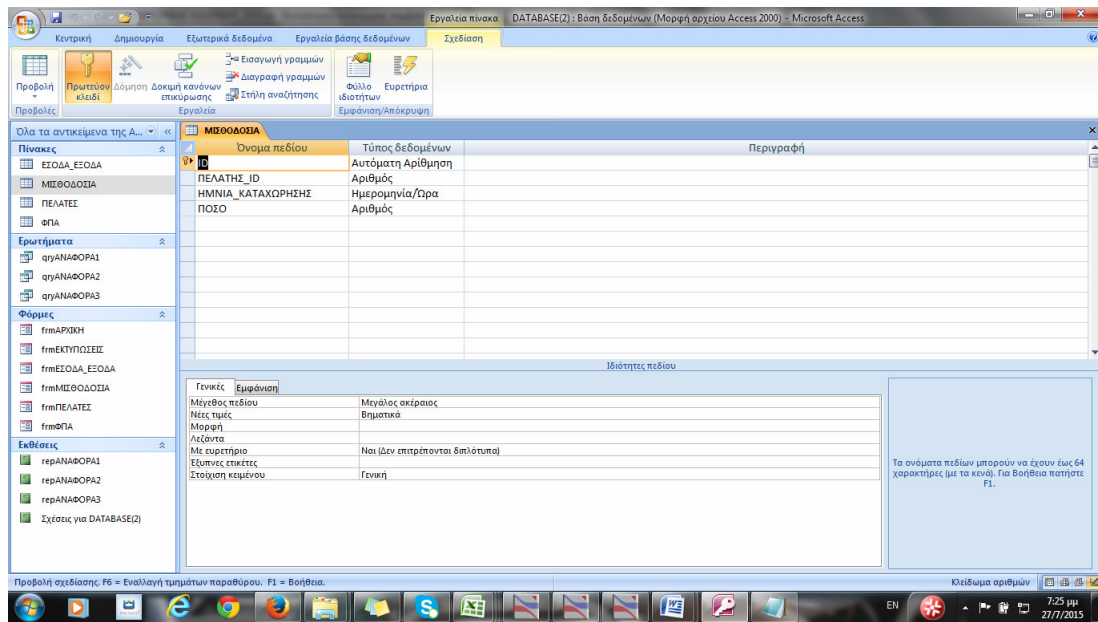
Εικόνα 7.3 Μισθοδοσία



ID	ΠΕΛΑΤΗΣ_ID	ΗΜΕΡΙΑ ΚΑΤΑΧΩΡΗΣΗΣ	ΠΟΣΟ	Προσθήκη νέου πεδίου
1	1		20,00	
2	2	20/5/2015	20,00	
3	3	20/5/2015	20,00	
4	4	20/5/2015	20,00	
5	1	15/6/2015	30,00	
(Νέο)		27/7/2015		

Επίσης στον πίνακα 7.4 φαίνεται ο τρόπος που έχουμε κάνει την σχεδίαση του πίνακα

Εικόνα 7.4 Μισθοδοσία



Εικόνα μισθοδοσίας (Εικόνα 7.5)

Έχουμε δημιουργήσει τον πίνακα όπου υπάρχουν οι στήλες

- Id
- Επωνυμία
- Οδός
- Αριθμός
- Πόλη
- ΤΚ
- Τηλέφωνο
- ΑΦΜ
- Δ.Ο.Υ.
- Taxis user
- Taxis pass

Εικόνα 7.5 Πελάτες

ID	ΕΠΩΝΥΜΙΑ	ΟΔΟΣ	ΑΡΙΘΜΟΣ	ΠΟΛΗ	ΤΚ	ΤΗΛΕΦΩΝΟ	ΑΦΜ	ΔΟΥ	TAXIS_USER	TAXIS_PASS	Προσθήκη
1	ΠΕΛΑΤΗΣ 1	ΕΡΜΟΥ	1	ΠΑΤΡΑ	26331	2610622423	999999999	Γ	1		
2	ΠΕΛΑΤΗΣ 2	ΚΟΡΙΝΘΟΥ	2	ΠΑΤΡΑ	26331	2610623578	888888888	Α	2		
3	ΠΕΛΑΤΗΣ 3	ΜΑΙΩΝΟΣ	3	ΠΑΤΡΑ	26332	2610425632	111111111	Γ	3		
4	ΠΕΛΑΤΗΣ 4	ΑΡΑΤΟΥ	4	ΠΑΤΡΑ	26333	2610752145	445563456	Γ	4		
5	ΠΕΛΑΤΗΣ 5	ΖΑΙΜΗ	5	ΠΑΤΡΑ	26334	2610622478	099765543	Γ	5		
6	ΠΕΛΑΤΗΣ 6	ΚΟΡΑΗ	6	ΠΑΤΡΑ	26500	2610622458	085745995	Γ	6		
7	ΠΕΛΑΤΗΣ 7	ΚΟΡΙΝΘΟΥ	7	ΠΑΤΡΑ	26500	2610478526	045789564	Γ	7		
8	ΠΕΛΑΤΗΣ 8	ΑΠΑΣ ΣΟΦΙΑΣ	8	ΠΑΤΡΑ	26501	2610526859	052697898	Α	8		
9	ΠΕΛΑΤΗΣ 9	ΑΠΩΝ ΘΕΟΔΩ	9	ΠΑΤΡΑ	26337	2610533696	054521782	Α	9		
10	ΠΕΛΑΤΗΣ 10	ΚΑΡΑΒΑΓΓΕΛΗ	10	ΠΑΤΡΑ	26335	2610541526	054786954	Α	10		
11	ΠΕΛΑΤΗΣ 11	ΑΡΑΤΟΥ	11	ΠΑΤΡΑ	26330	2610522369	047848996	Α	11		
12	ΠΕΛΑΤΗΣ 12	ΖΑΙΜΗ	12	ΠΑΤΡΑ	26330	2610521458	041528954	Α	12		
13	ΠΕΛΑΤΗΣ 13	ΜΕΣΟΛΟΓΙΟΥ	13	ΠΑΤΡΑ	26336	2610422369	054147893	Β	13		
14	ΠΕΛΑΤΗΣ 14	ΡΟΥΦΟΥ	14	ΠΑΤΡΑ	26331	2610522748	000585974	Β	14		
15	ΠΕΛΑΤΗΣ 15	ΑΘΗΝΩΝ	15	ΠΑΤΡΑ	26334	2610521478	057949971	Β	15		
16	ΠΕΛΑΤΗΣ 16	ΑΙΓΙΟΥ	16	ΠΑΤΡΑ	26334	2610654789	023457894	Β	16		
17	ΠΕΛΑΤΗΣ 17	ΕΡΜΟΥ	17	ΠΑΤΡΑ	26334	2610622478	011596447	Β	17		
18	ΠΕΛΑΤΗΣ 18	ΑΠΟΥ ΝΙΚΟΛΑ	18	ΠΑΤΡΑ	26331	2610632478	202694489	Β	18		
19	ΠΕΛΑΤΗΣ 19	ΡΗΓΑ ΦΕΡΑΙΟΥ	19	ΠΑΤΡΑ	26330	2610643023	114783989	Β	19		
20	ΠΕΛΑΤΗΣ 20	Κ.ΑΧΑΙΑ	20	Κ.ΑΧΑΙΑ	26229	2613023145	114458749	Κ ΑΧΑΙΑΣ	20		
21	ΠΕΛΑΤΗΣ 21	ΚΑΝΑΡΗ	21	ΠΑΤΡΑ	26228	2610275178	325698744	Α	21		
22	ΠΕΛΑΤΗΣ 22	ΚΑΝΑΚΑΡΗ	22	ΠΑΤΡΑ	26228	2610275137	124785996	Α	22		
23	ΠΕΛΑΤΗΣ 23	ΕΥΒΟΙΑΣ	23	ΠΑΤΡΑ	26225	2610275896	457854146	Α	23		
24	ΠΕΛΑΤΗΣ 24	ΑΚΡΟΤΗΡΙΟΥ	24	ΠΑΤΡΑ	26330	2610622152	675463434	Α	24		
25	ΠΕΛΑΤΗΣ 25	ΚΑΛΑΒΡΥΤΩΝ	25	ΠΑΤΡΑ	26334	2610632589	324567865	Γ	25		

Επίσης στον πίνακα 7.6 φαίνεται ο τρόπος που έχουμε κάνει την σχεδίαση του πίνακα

Εικόνα 7.6 Πελάτες

Όνομα πεδίου	Τύπος δεδομένων	Περιγραφή
ΕΠΩΝΥΜΙΑ	Αυτόματη Αρίθμηση	
ΟΔΟΣ	Κείμενο	
ΑΡΙΘΜΟΣ	Κείμενο	
ΠΟΛΗ	Κείμενο	
ΤΚ	Κείμενο	
ΤΗΛΕΦΩΝΟ	Κείμενο	
ΑΦΜ	Κείμενο	
ΔΟΥ	Κείμενο	
TAXIS_USERNAME	Κείμενο	
TAXIS_PASSWORD	Κείμενο	

Γενικές	Εμφάνιση
Μέγεθος πεδίου	Μεγάλος αέρας
Νέες τιμές	Βηματικά
Μορφή	
Λέξεις	
Με ευρετήριο	Ναι (Δεν επηρεάζονται διπλότυπα)
Εξυπνές ετικέτες	
Σταθμής κειμένου	Γενική

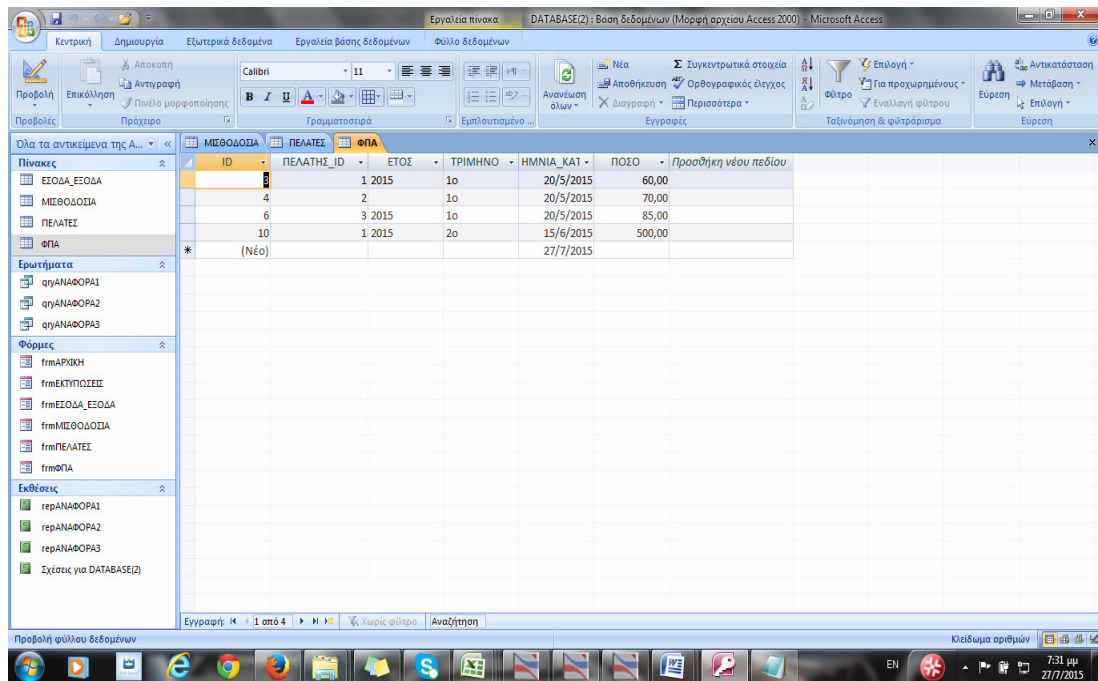
Τα ονόματα πεδίων μπορούν να έχουν έως 64 χαρακτήρες (με τα κενά). Για βοήθεια πατήστε F1.

Εικόνα Φ.Π.Α. (Εικόνα 7.6)

Έχουμε δημιουργήσει τον πίνακα όπου υπάρχουν οι στήλες

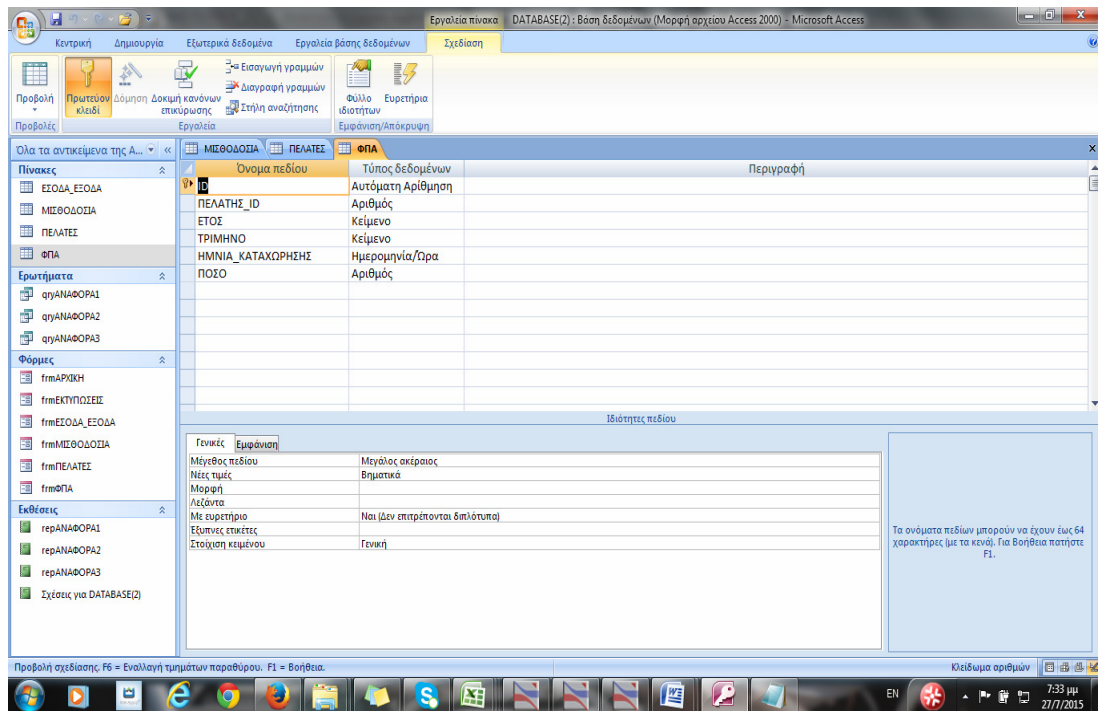
- Id
- Πελάτης id
- Έτος
- Τρίμηνο
- Ημερομηνία καταχώρησης
- Πόσο

Εικόνα 7.7 Φ.Π.Α.



ID	ΠΕΛΑΤΗΣ_ID	ΕΤΟΣ	ΤΡΙΜΗΝΟ	ΗΜΝΙΑ ΚΑΤ	ΠΟΣΟ	Προσθήκη νέου πεδίου
1		2015	1ο	20/5/2015	60,00	
4		2	1ο	20/5/2015	70,00	
6		3	2015	1ο	20/5/2015	85,00
10		1	2015	2ο	15/6/2015	500,00
*	(Νέο)			27/7/2015		

Επίσης στον πίνακα 7.8 φαίνεται ο τρόπος που έχουμε κάνει την σχεδίαση του πίνακα



7.4.2. Αναφορές

Παρακάτω παρουσιάζουμε τις αναφορές που έχουμε δημιουργήσει που μας δείχνουν στοιχεία σχετικά με τα έσοδα και έξοδα που έχουν οι πελάτες ανά περίοδο, καθώς επίσης και τα χρήματα που έχει εισπράξει το λογιστικό γραφείο ανά πελάτη.

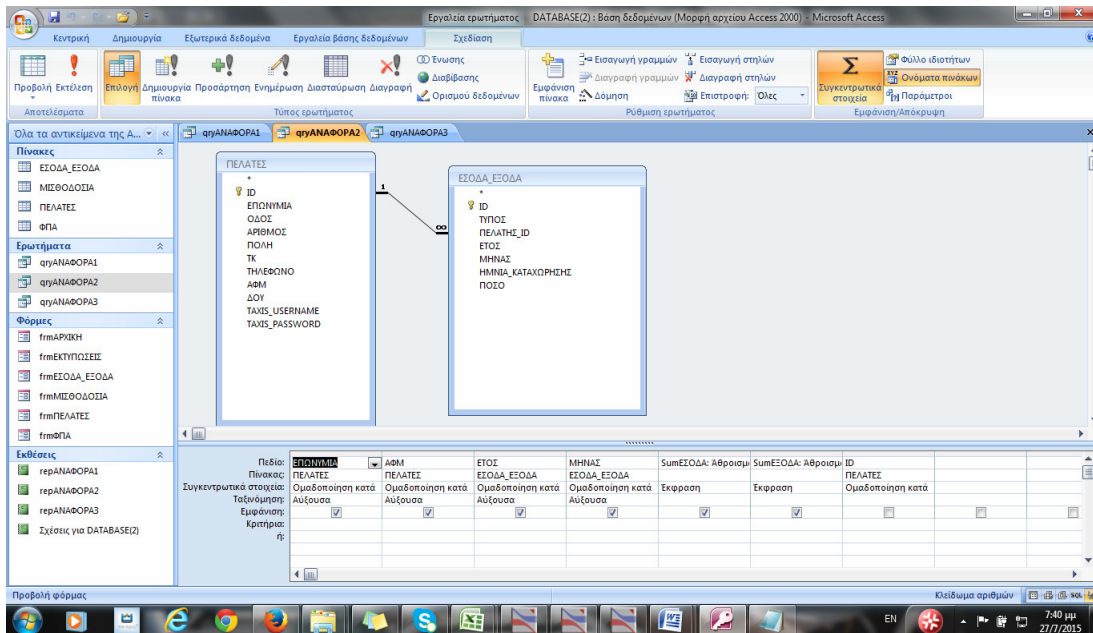
Αναφορά εσόδων – εξόδων (Εικόνα 7.9)

Εικόνα 7.9 Αναφορά εσόδων - εξόδων

ΕΠΙΘΥΜΙΑ	ΑΦΜ	ΕΤΟΣ	ΜΗΝΑΣ	SumΕΣΟΔΑ	SumΕΞΟΔΑ
ΠΕΛΑΤΗΣ	999999999	2015	01	100	50
ΠΕΛΑΤΗΣ 1	999999999	2015	02	150	0
ΠΕΛΑΤΗΣ 1	999999999	2015	03	120	0
ΠΕΛΑΤΗΣ 2	888888888	2015	01	90	0

Επίσης στον πίνακα 7.10 φαίνεται ο τρόπος που έχουμε κάνει την σχεδίαση της αναφοράς

Εικόνα 7.10 Αναφορά εσόδων - εξόδων



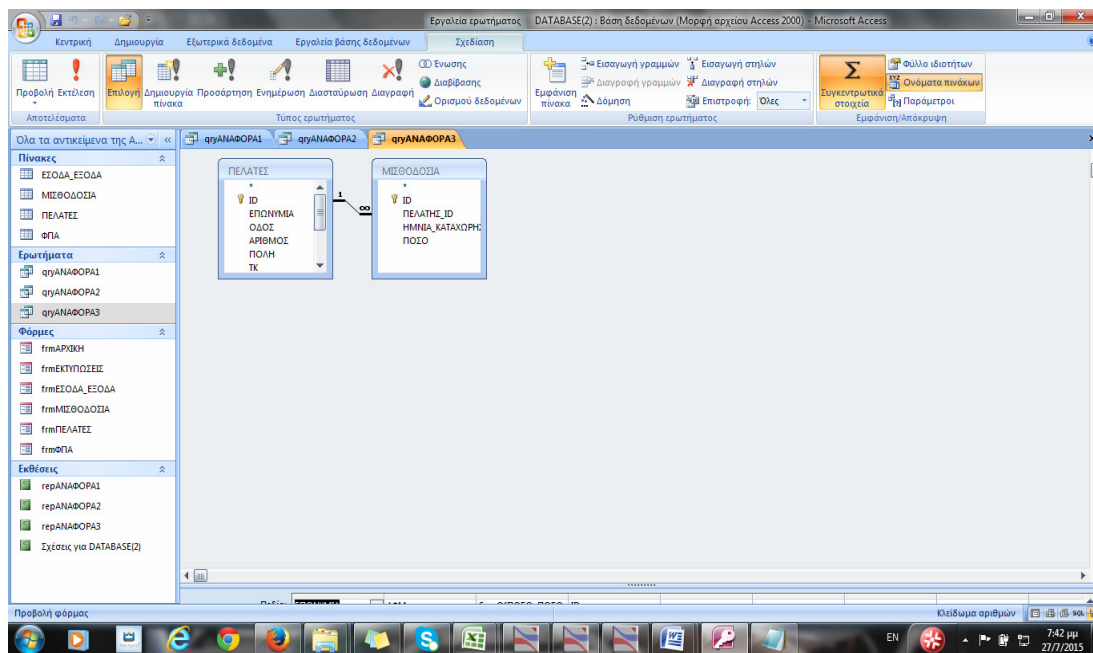
Αναφορά εσόδων ανά πελάτη (Εικόνα 7.11)

Εικόνα 7.11 Αναφορά εσόδων ανά πελάτη

ΕΠΩΝΥΜΙΑ	ΑΦΜ	SumOfΠΟΣΟ
ΠΕΛΑΤΗΣ 1	999999999	50
ΠΕΛΑΤΗΣ 2	888888888	20
ΠΕΛΑΤΗΣ 3	111111111	20
ΠΕΛΑΤΗΣ 4	445563456	20

Επίσης στον πίνακα 7.12 φαίνεται ο τρόπος που έχουμε κάνει την σχεδίαση της αναφοράς

Εικόνα 7.12 Αναφορά εσόδων ανά πελάτη



7.4.3. Φόρμες

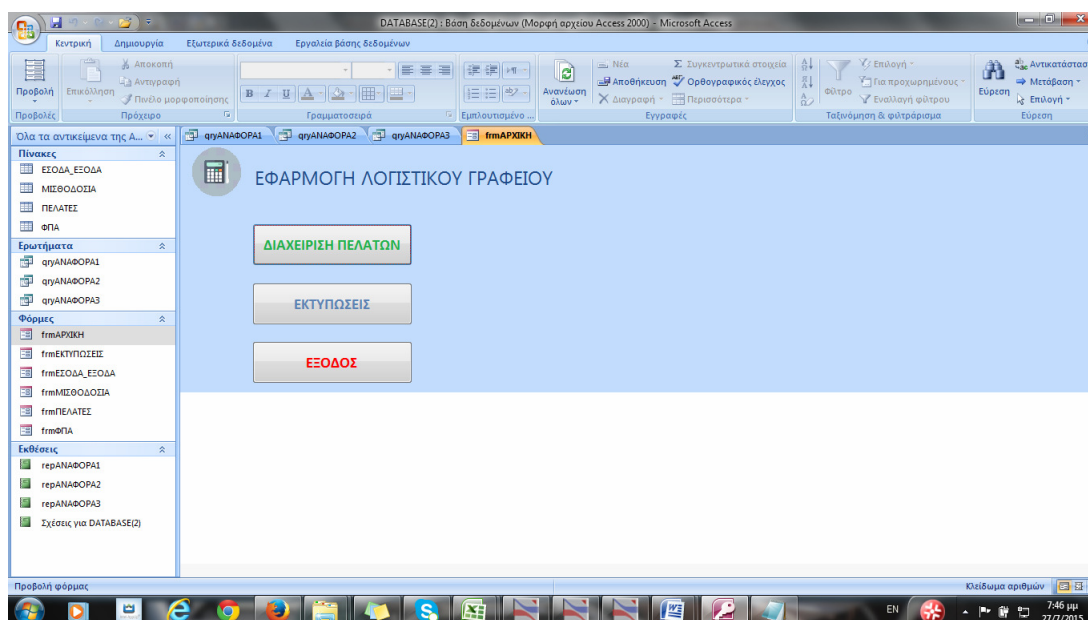
Με τις φόρμες που δημιουργήσαμε θέλαμε να δώσουμε την δυνατότητα στον χρήστη να έχει πληροφορίες για κάθε πελάτη σχετικά με :

- Τα έσοδα – έξοδα
- Το ποσό πληρωμής του Φ.Π.Α ανά τρίμηνο
- Το ποσό της εκκαθαριστικής δήλωσης του Φ.Π.Α.
- Το ποσό της μισθοδοσίας
- Την διαχείριση των πελατών

Αρχική φόρμα

Με την επιλογή αυτή μας δίνεται η δυνατότητα να επιλέξουμε αν θέλουμε να διαχειρισθούμε κάποιον από τουε πελάτες μας, να κάνουμε εκτυπώσεις και τέλος να βγούμε από την συγκεκριμένη φόρμα.(Εικόνα 7.13)

Εικόνα 7.13 Αρχική φόρμα

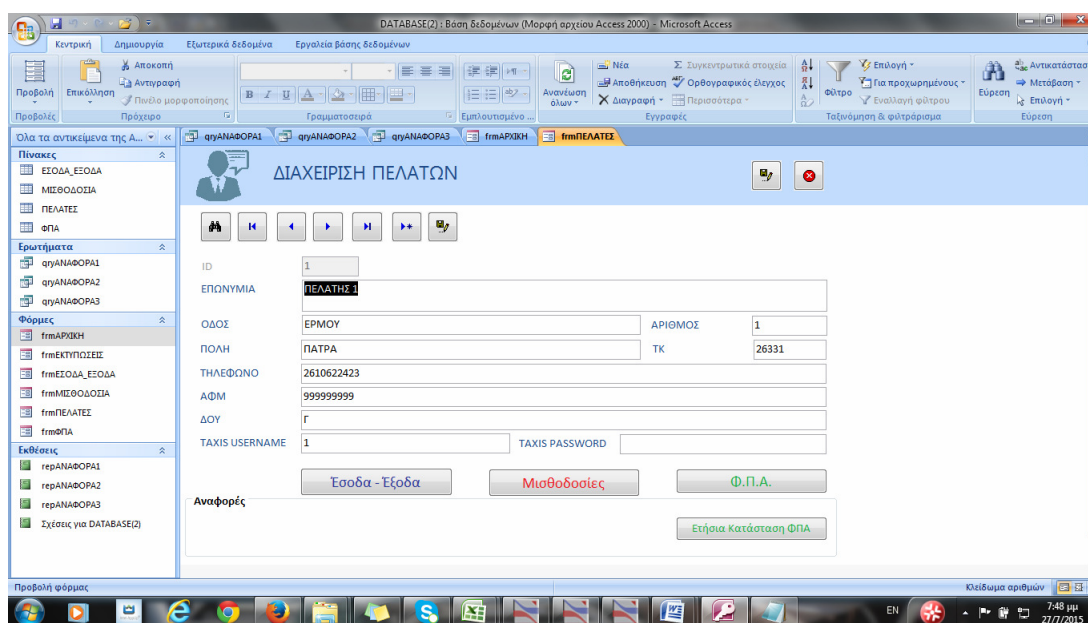


Διαχείριση πελατών

Με την επιλογή διαχείριση πελατών μας εμφανίζεται ο Εικόνα 7.14 όπου μπορούμε να επιλέξουμε μεταξύ των παρακάτω και να έχουμε έτσι μια πολύ γρήγορη εικόνα των πελατών μας

- Έσοδα – έξοδα
- Μισθοδοσία
- Φ.Π.Α.
- Εκκαθαριστική δήλωση Φ.Π.Α.

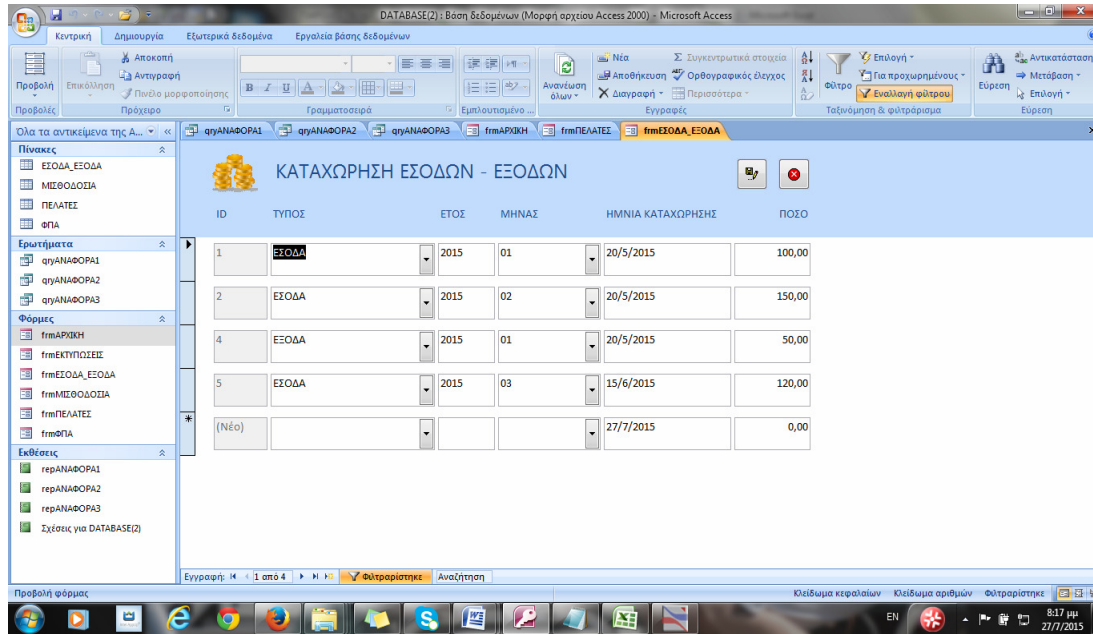
Εικόνα 7.14 Διαχείριση πελατών



Έσοδα – έξοδα

Με την επιλογή αυτή μας δίνεται η εικόνα των εσόδων και εξόδων που έχουν καταχωρηθεί μέχρι και εκείνη την στιγμή στην βάση μας (Εικόνα 7.15 έσοδα – έξοδα)

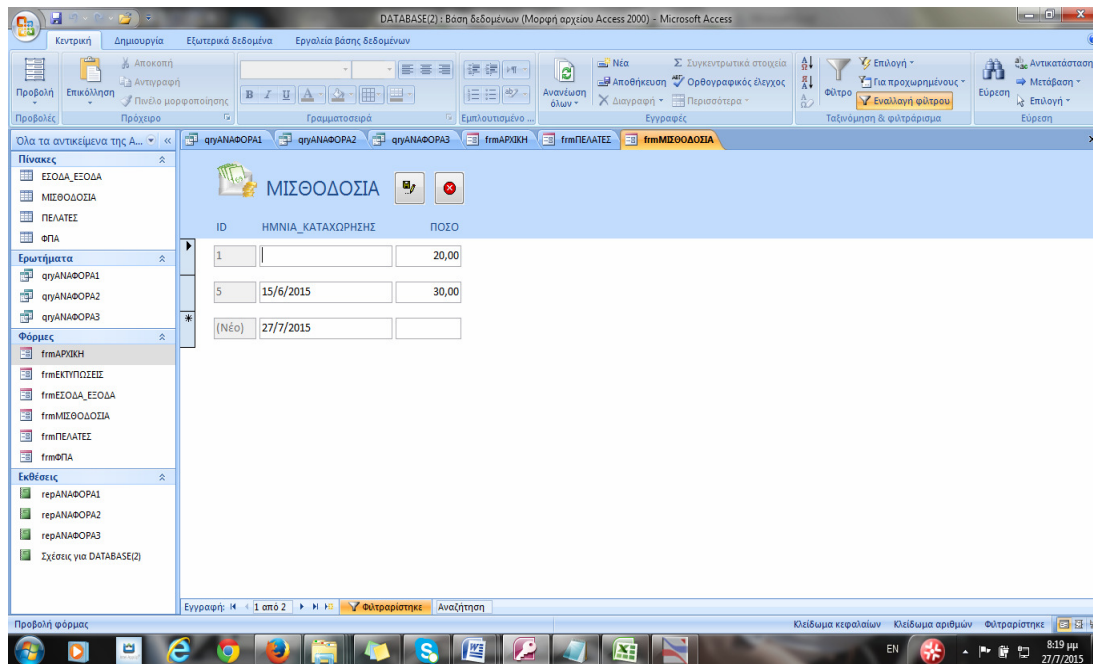
Εικόνα 7.15 έσοδα – έξοδα



Μισθοδοσία

Καταχωρούμε τα ποσά της μισθοδοσίας ανά πελάτη (Εικόνα 7.16)

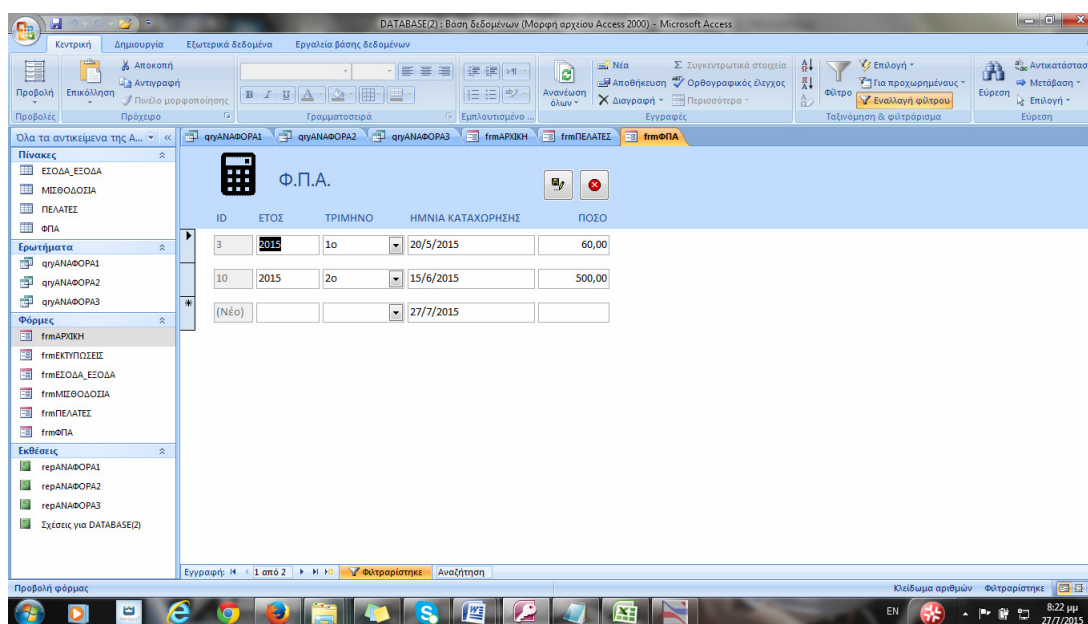
Εικόνα 7.16 Μισθοδοσία



Φ.Π.Α.

Μπορούμε να καταχωρήσουμε τα ποσά των περιοδικών δηλώσεων του Φ.Π.Α. ανά περίοδο (Εικόνα 7.17)

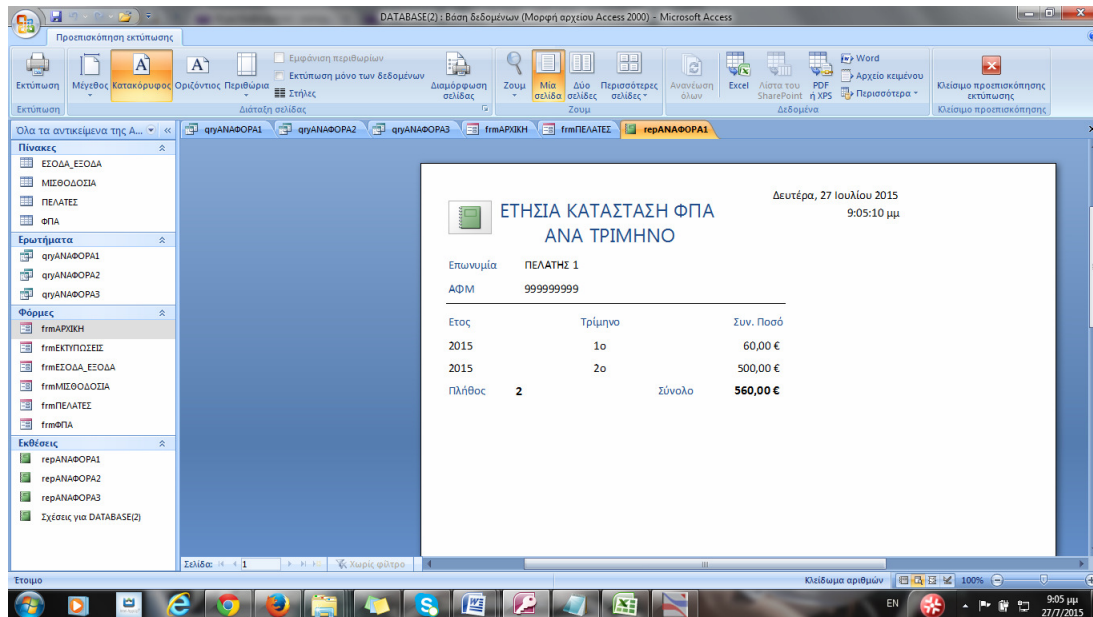
Εικόνα 7.17 Φ.Π.Α.



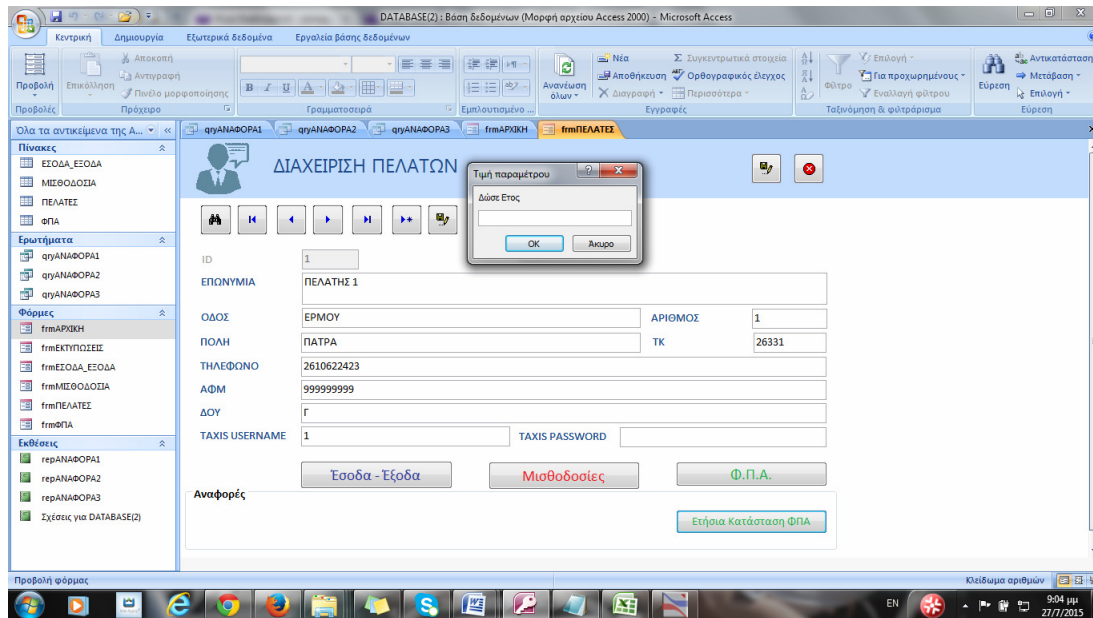
Εκκαθαριστική δήλωση Φ.Π.Α.

Με την επιλογή αυτή μπορούμε να δούμε την εκκαθάριση του Φ.Π.Α. για κάθε πελάτη που έχουμε επιλέξει από την διαχείριση των πελατών (Εικόνα 7.18) , αφού όμως πρώτα έχουμε ορίσει το έτος που θέλουμε να έχουμε το αποτέλεσμα.(Εικόνα 7.19).

Εικόνα 7.18 Εκκαθαριστική δήλωση Φ.Π.Α



Εικόνα 7.19 Εκκαθαριστική δήλωση Φ.Π.Α

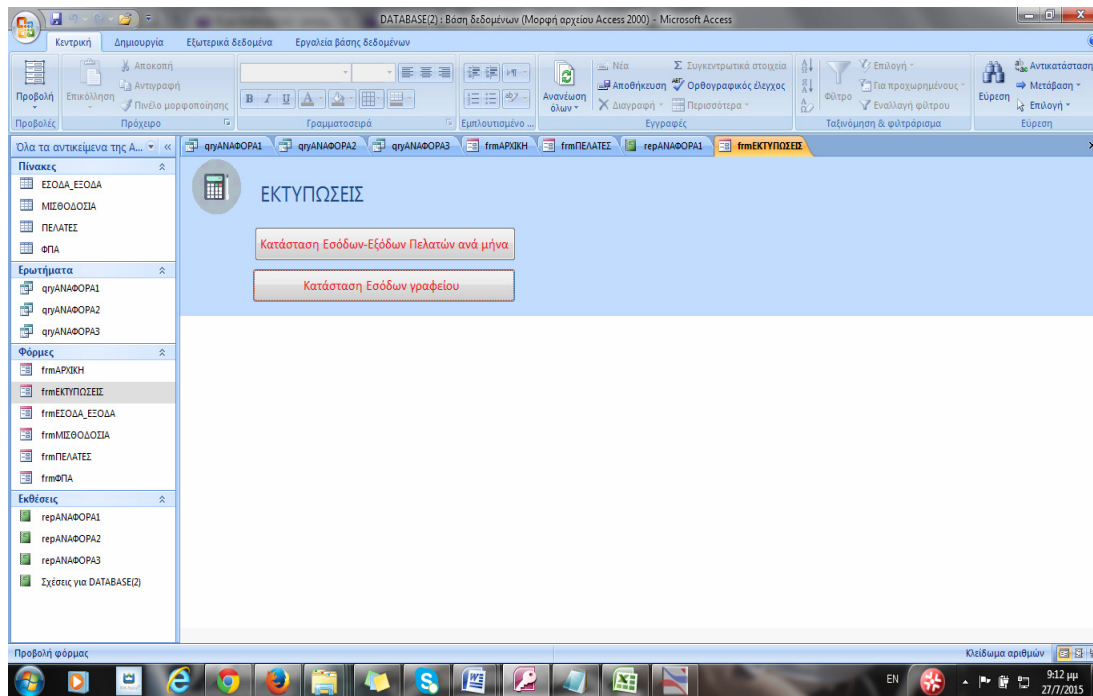


Φόρμα εκτυπώσεων

Η φόρμα αυτή έχει να κάνει με δυο βασικά στοιχεία που είναι : (Εικόνα 7.20)

- Η κατάσταση εσόδων – εξόδων πελατών ανά μήνα (Εικόνα 7.21)
- Η κατάσταση εσόδων γραφείου (Εικόνα 7.22)

Εικόνα 7.20 Φόρμα εκτυπώσεων



Εικόνα 7.21 Κατάσταση εσόδων – εξόδων ανά τρίμηνο

ΚΑΤΑΣΤΑΣΗ ΕΣΟΔΩΝ-ΕΞΟΔΩΝ
ΑΝΑ ΤΡΙΜΗΝΟ

Δευτέρα, 27 Ιουλίου 2015
9:14:36 μμ

ΕΠΩΝΥΜΙΑ	ΑΦΜ	ΕΤΟΣ	ΜΗΝΑΣ	ΕΣΟΔΑ	ΕΞΟΔΑ
PEΛΑΤΗΣ 1					
	999999999				
		2015	01	100,00 €	50,00 €
		2015	02	150,00 €	0,00 €
		2015	03	120,00 €	0,00 €
				75,00%	370,00 €
					50,00 €
PEΛΑΤΗΣ 2					
	888888888				
		2015	01	90,00 €	0,00 €
				25,00%	90,00 €
					0,00 €
Πλήθος			Σύνολο	460,00 €	50,00 €

Εικόνα 7.22 Κατάσταση εσόδων γραφείου

ΚΑΤΑΣΤΑΣΗ ΕΣΟΔΩΝ
ΓΡΑΦΕΙΟΥ

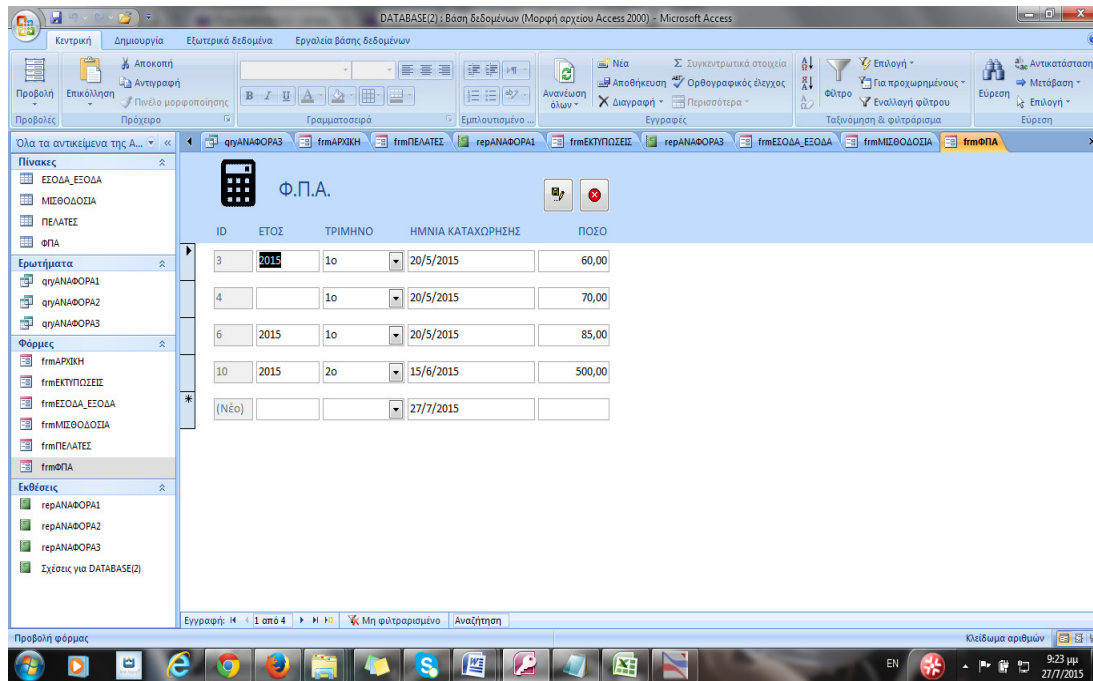
Δευτέρα, 27 Ιουλίου 2015
9:16:00 μμ

Επωνυμία	ΑΦΜ	Ποσό
PEΛΑΤΗΣ 1	999999999	80,00 €
PEΛΑΤΗΣ 2	888888888	20,00 €
PEΛΑΤΗΣ 3	111111111	20,00 €
PEΛΑΤΗΣ 4	445563456	20,00 €
Πλήθος 4		Σύνολο 140,00 €

Φόρμα Φ.Π.Α.

Στην φόρμα αυτή μπορούμε να καταχωρήσουμε το ποσό του Φ.Π.Α. ανά περίοδο που μας ενδιαφέρει. (Εικόνα 7.23)

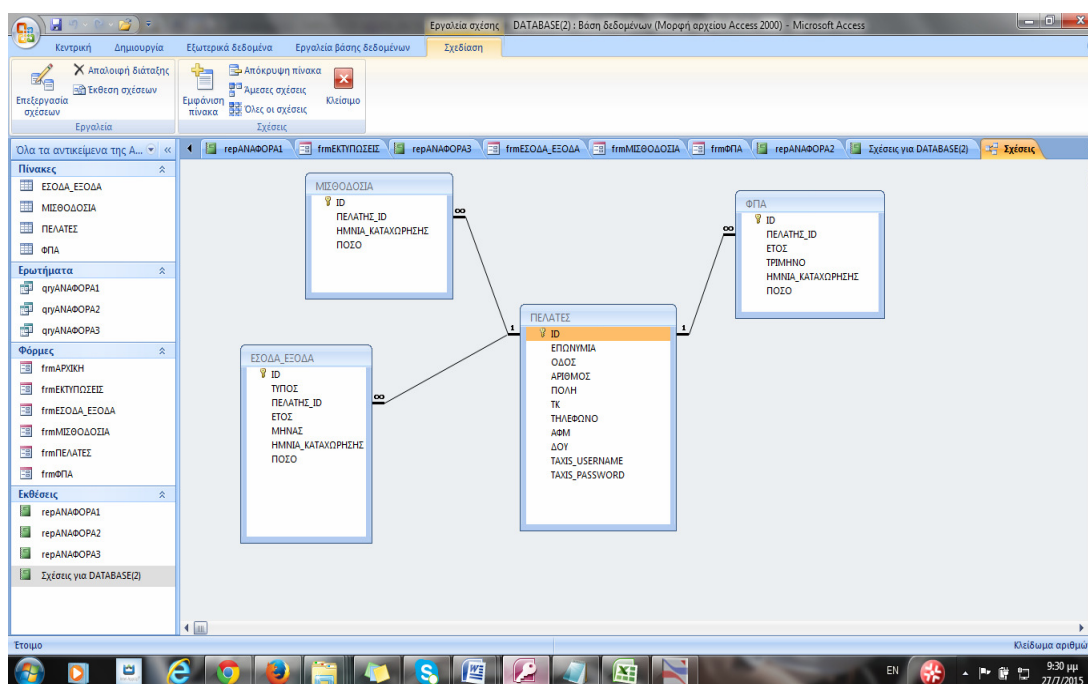
Εικόνα 7.23 Κατάσταση εσόδων – εξόδων ανά τρίμηνο



7.4.4. Σχέσεις των πινάκων

Στον πίνακα που ακολουθεί εμφανίζονται οι σχέσεις που υπάρχουν μεταξύ των πινάκων της βάσης μας (Εικόνα 7.24)

Εικόνα 7.24 Σχέσεις των πινάκων



7.4.5. Πρωτεύον κλειδί

Όταν ένας Εικόνα μιας βάσης δεδομένων της Microsoft Access διαθέτει ένα πεδίο ή ένα σύνολο πεδίων που χαρακτηρίζουν μοναδικά κάθε εγγραφή του πίνακα, αυτό μπορεί να οριστεί ως πρωτεύον κλειδί. Η επιλογή ενός πρωτεύοντος κλειδιού είναι μια από τις πιο σημαντικές αποφάσεις που θα λάβετε κατά το σχεδιασμό μιας νέας βάσης δεδομένων.

Μια βέλτιστη πρακτική στο σχεδιασμό βάσεων δεδομένων είναι η χρήση ενός εσωτερικά δημιουργημένου πρωτεύοντος κλειδιού. Το σύστημα διαχείρισης βάσης δεδομένων μπορεί να δημιουργήσει ένα μοναδικό αναγνωριστικό. Για παράδειγμα, μπορείτε να χρησιμοποιήσετε τον τύπο δεδομένων "Αυτόματη αρίθμηση" για να δημιουργήσετε ένα πεδίο με το όνομα "Αναγνωριστικό πελάτη". Ο τύπος δεδομένων "Αυτόματη αρίθμηση" θα αυξάνει κατά ένα τον αριθμό του πεδίου κάθε φορά που

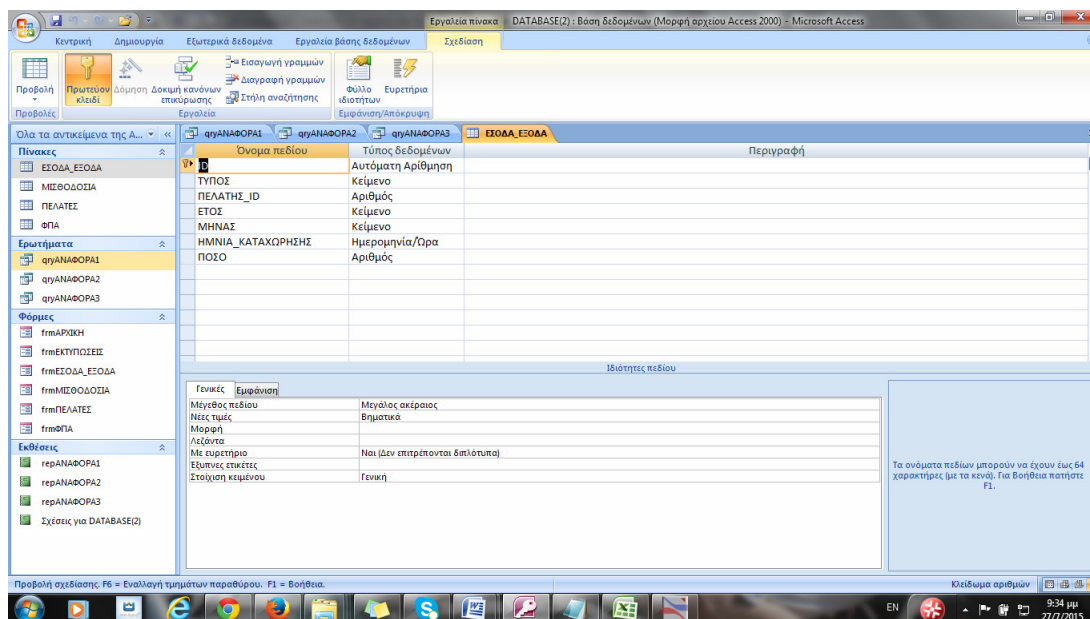
δημιουργείτε μια νέα εγγραφή. Ενώ ο αριθμός αυτός καθαυτός δεν έχει μεγάλη σημασία, προσφέρει έναν εύκολο τρόπο αναφοράς σε μια μεμονωμένη εγγραφή στα ερωτήματα.

Μια καλή επιλογή για πιθανό πρωτεύον κλειδί πρέπει να διαθέτει διάφορα χαρακτηριστικά:

- Αναγνωρίζει κάθε γραμμή μοναδικά
- Δεν είναι ποτέ κενό ή με μηδενική τιμή — περιέχει πάντα μια τιμή.
- Οι τιμές που περιέχει αλλάζουν σπάνια (ιδανικά, ποτέ)

Στον πίνακα που ακολουθεί εμφανίζεται ποιο είναι το πρωτεύον κλειδί στην βάση μας.(Εικόνα 7.24)

Εικόνα 7.25 πρωτεύον κλειδί



7.5. Κρυπτογράφηση και αποκρυπτογράφηση Βάσης Δεδομένων

Για την κρυπτογράφηση μιας βάσης δεδομένων σε Microsoft Access θα πρέπει πρώτα να δώσουμε προσοχή στο παρακάτω:

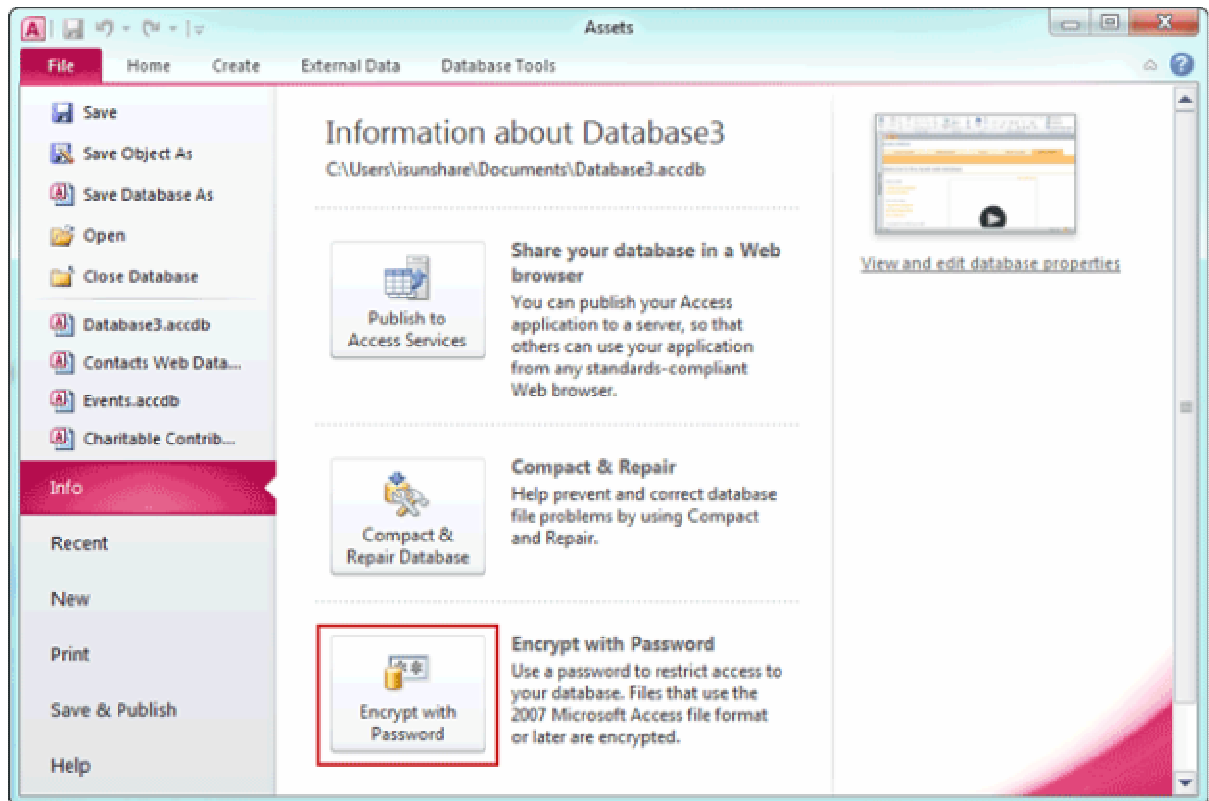
1. Η κρυπτογράφηση μιας βάσης δεδομένων σε Microsoft Access γίνεται σε επίπεδο Access, και όχι σε επίπεδο δεδομένων
2. Τα πάντα σε ένα αρχείο Microsoft Access database κρυπτογραφούνται, συμπεριλαμβανομένων των πινάκων, των φορμών, των index, κ.α
3. Η Microsoft Access χρησιμοποιεί τον αλγόριθμο RSA company για την κρυπτογράφηση της βάσης δεδομένων.
4. Η κρυπτογράφηση εμποδίζει κάποιον από το να χρησιμοποιήσει ένα αρχείο, να διαβάσει ή να γράψει δεδομένων σε ένα αρχείο Access MDB ή ACCDB file. Δεν σχετίζεται όμως με ασφάλεια δεδομένων.

Μέρος 1^ο : Κρυπτογράφηση

Είναι μια διαδικασία προσδιορισμού ενός κωδικού για τον περιορισμό της πρόσβασης στη βάση.

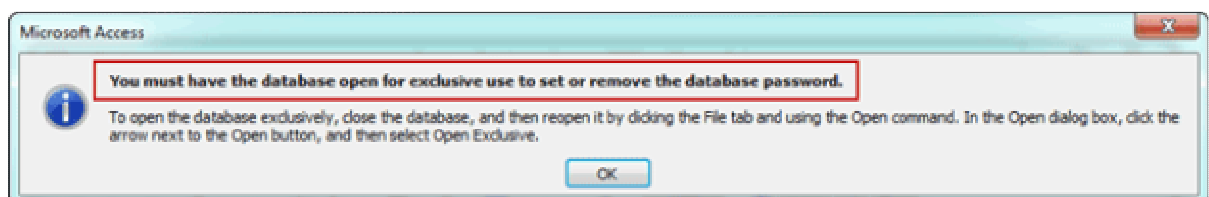
1. Όταν το αρχείο είναι ανοιχτό στη Microsoft Access, επιλέγουμε tab **File - Info**.²³

²³ (<http://www.isunshare.com/access-password/how-to-encrypt-and-decrypt-access-database.html>)

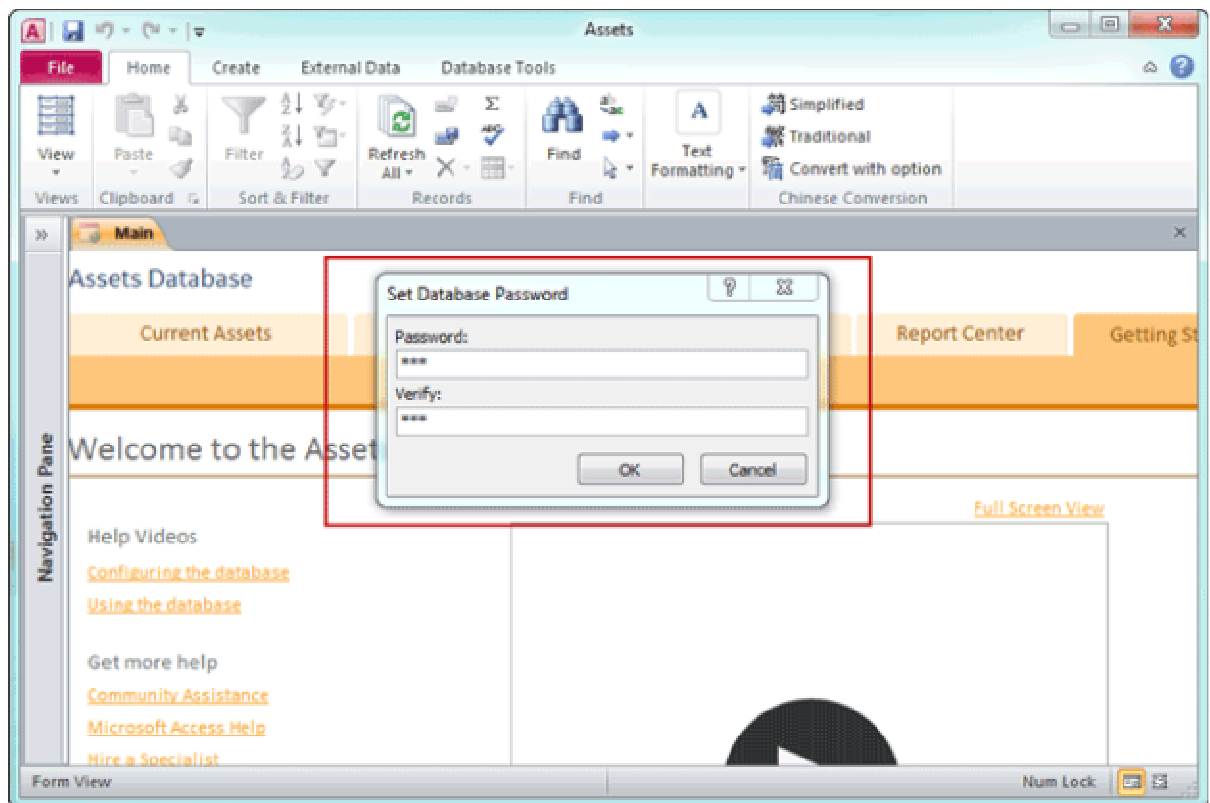


2. Επιλέγουμε **Encrypt with Password**.

Πολλές φορές εμφανίζεται το παρακάτω παράθυρό για να ανοίξουμε τη βάση σε **Open Exclusive mode** και πατάμε OK.

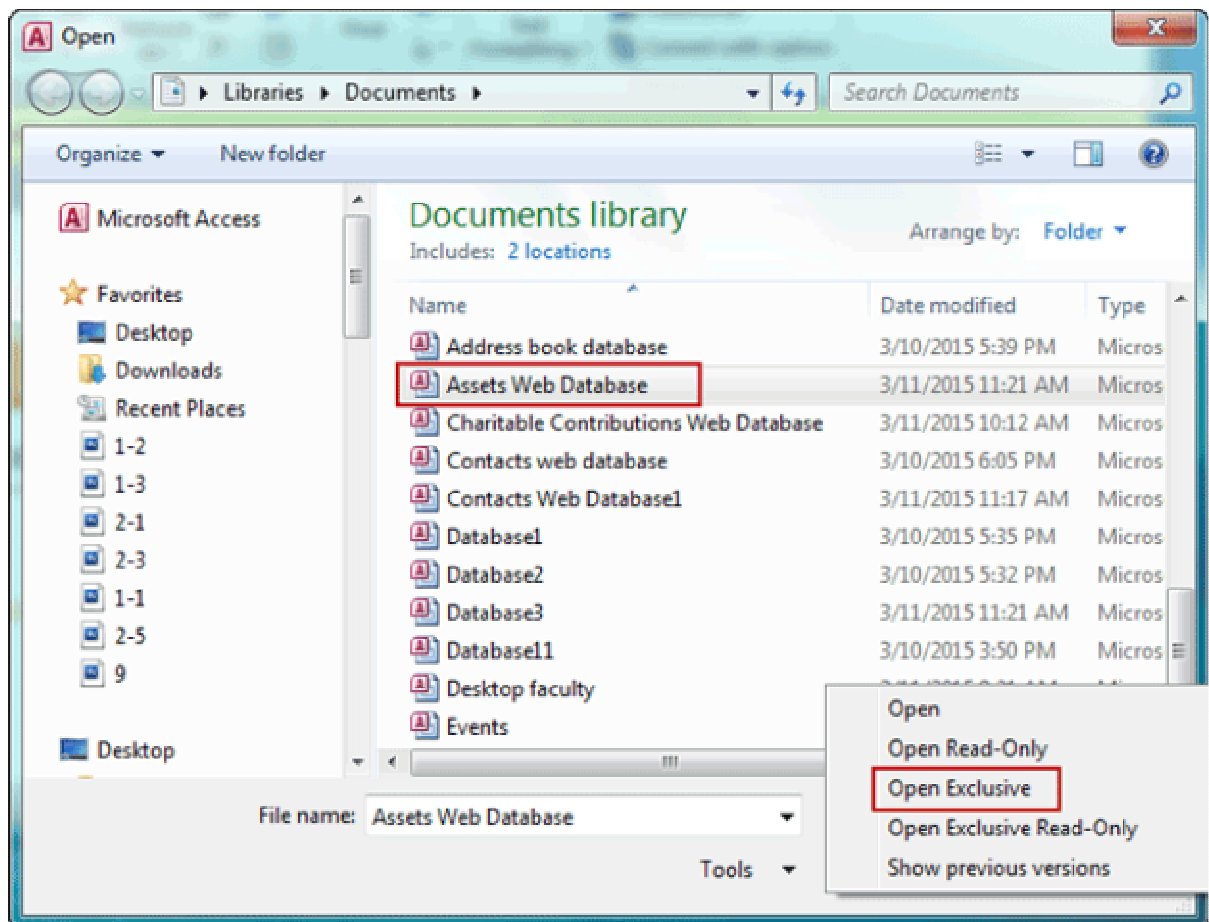


3. Στο παράθυρο **Set Database Password**, εισάγουμε τον κωδικό, και στη συνέχεια τον εισάγουμε δεύτερη φορά για επιβεβαίωση.



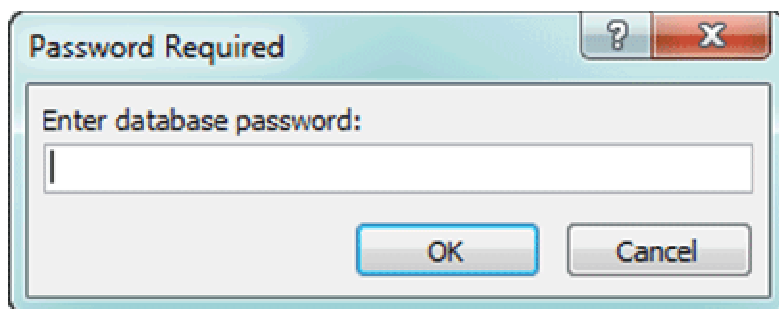
Μέρος 2^ο : Αποκρυπτογράφηση

1. Ανοίγουμε το αρχείο σε **Open Exclusive**.



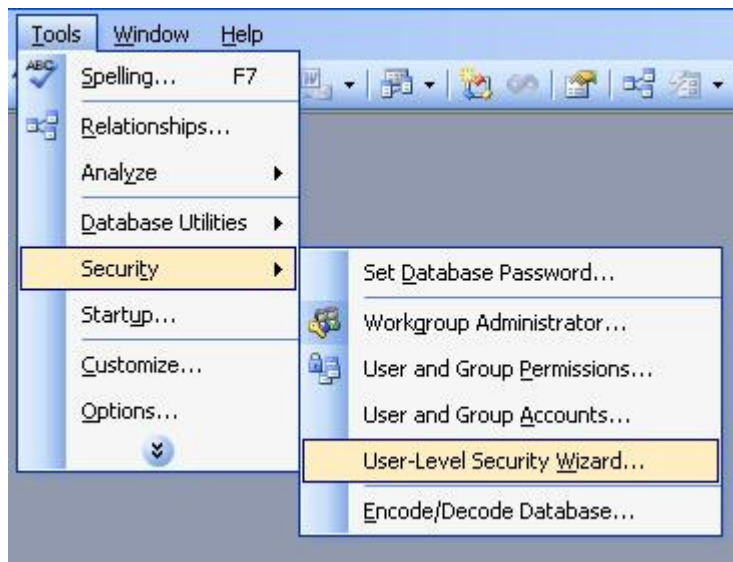
2. Μόλις επιλέξουμε την κρυπτογραφημένη βάση δεδομένων, εμφανίζεται ένα παράθυρο για να εισάγουμε τον κωδικό.

3. Εισάγουμε τον κωδικό και πατάμε OK.



7.6. Ασφάλεια σε επίπεδο χρηστών

Σε αυτή την παράγραφο θα δούμε πώς προσδιορίζουμε επίπεδα ασφαλείας για τους χρήστες στη Microsoft Access

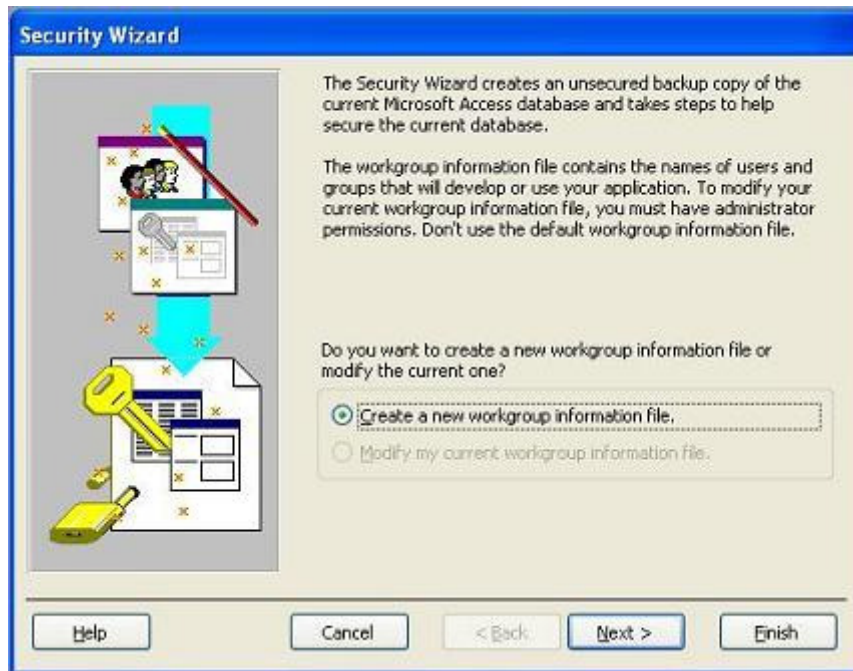


Η ασφάλεια χρήστη μας επιτρέπει να ελέγξουμε τον τύπο των δεδομένων που έχει πρόσβαση ο κάθε χρήστης και τις ενέργειες που μπορεί να εκτελέσει.²⁴

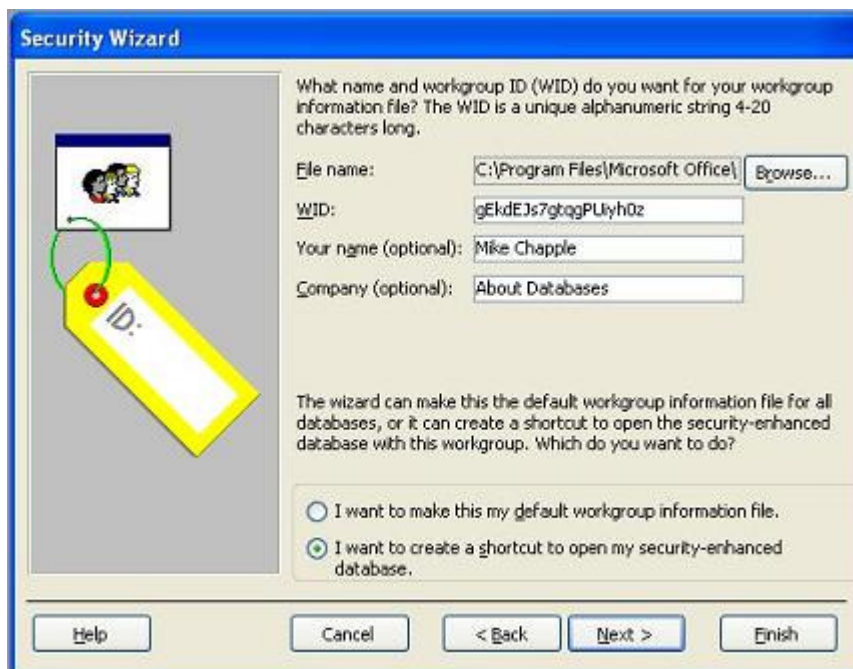
Το πρώτο βήμα είναι να ξεκινήσουμε τον οδηγό, όπως φαίνεται στην παραπάνω εικόνα.

Στη συνέχεια δημιουργούμε ένα νέο αρχείο **Workgroup Information File**.

²⁴ (<http://databases.about.com/od/tutorials/ss/usersecurity.htm#showall>)



Στη συνέχεια, αφού πατήσουμε Next, δίνουμε ένα όνομα και ένα κωδικό για το **Workgroup**.



Το επόμενο βήμα είναι να ορίσουμε την εμβέλεια της ασφάλειας. Επιλέγουμε σε ποια αντικείμενα της βάσης θα εφαρμοστεί η ασφάλεια χρηστών.



Επιλέγουμε όλα τα αντικείμενα (πίνακες, φόρμες, ερωτήματα) και προχωρούμε στο επόμενο βήμα.

Κατόπιν, στη συνέχεια, θα πρέπει να ορίσουμε ομάδες χρηστών (**User Groups**)



Καθώς επίσης και τα δικαιώματα που θα έχει κάθε ομάδα (**Users Group Permissions**)



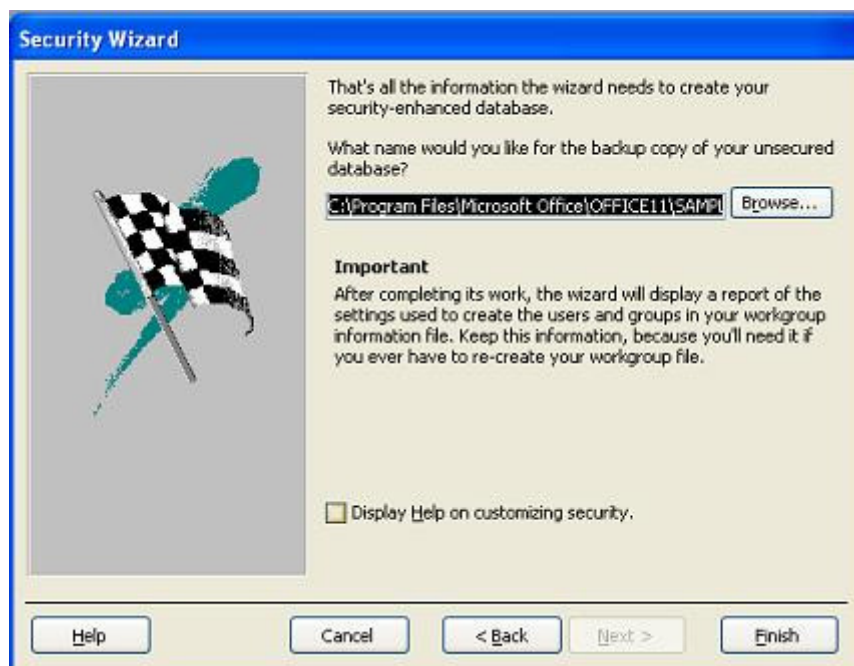
Στη συνέχεια, προσθέτουμε χρήστες (**Users**)



Και προσθέτουμε τους χρήστες στις ομάδες χρηστών.



Στο τελευταίο βήμα, δημιουργούμε ένα backup της βάσης δεδομένων.



ΣΥΜΠΕΡΑΣΜΑΤΑ

Η χρήση Βάσεων Δεδομένων στη μηχανογράφηση αντί εφαρμογών διαχείρισης αρχείων παρέχει τις εξής ευκολίες:

- **Συνέπεια δεδομένων και μη περιττή αποθήκευση τους.** Να μην υπάρχουν διπλές εγγραφές των ίδιων στοιχείων π.χ. εάν τα στοιχεία της διεύθυνσης ενός πελάτη αλλάξουν θα πρέπει να αλλάξουν για όλες τις πιθανές εφαρμογές της τράπεζας (δάνεια, πιστωτικές κάρτες, τραπεζικοί λογαριασμοί,...) ενώ αν είναι αποθηκευμένα και σε άλλο σημείο και δεν ενημερωθούν θα έχουμε ασυνέπεια δεδομένων.
- **Ευκολία πρόσβασης στα δεδομένα.** Υπάρχει γλώσσα ανάκτησης των δεδομένων και ειδικό περιβάλλον χρήσης.
- **Μη απομόνωση δεδομένων.** Τα δεδομένα βρίσκονται σε πίνακες και όχι διασκορπισμένα σε διάφορα αρχεία που σημαίνει την αναγκαιότητα κατασκευής νέων εφαρμογών για την πρόσκτηση των απαιτούμενων στοιχείων.

Αντιμετώπιση του προβλήματος της ταυτόχρονης πρόσβασης. Είναι ένα από τα μεγάλα πλεονεκτήματα των Βάσεων Δεδομένων γιατί παρέχουν επίβλεψη και ευκολίες κλειδώματος των κατάλληλων δεδομένων ώστε να μην ενημερώσουν παραπάνω από ένας χρήστες τις ίδιες εγγραφές και προκύψουν ασυνέπειες. Αυτή η επίβλεψη και ευκολίες δεν είναι εύκολο να εξασφαλιστούν με εφαρμογές διαχείρισης αρχείων.

- **Προβλήματα ασφάλειας.** Δεν πρέπει ο κάθε χρήστης μιας Βάσης Δεδομένων να μπορεί να βλέπει όλα τα δεδομένα π.χ. ο χρήστης της εφαρμογής μιας μισθοδοσίας θα πρέπει να έχει πρόσβαση μόνο στα στοιχεία που τον αφορούν. Αυτό δεν εξασφαλίζεται εύκολα σε ένα σύστημα διαχείρισης αρχείων όπου προστίθενται εφαρμογές και δεδομένα σύμφωνα με τις αναγκαιότητες κάθε στιγμής.

- **Ακεραιότητα δεδομένων.** Τα δεδομένα στη Βάση θα πρέπει να ικανοποιούν κάποιες συνθήκες π.χ. ο λογαριασμός καταθέσεων ενός πελάτη δε θα πρέπει να είναι κάτω από ένα συγκεκριμένο ποσό για παράδειγμα 20 € ή να μη γίνει η πώληση ενός προϊόντος αν δεν υπάρχει τουλάχιστον ένα τεμάχιο στην αποθήκη.

ΒΙΒΛΙΟΓΡΑΦΙΑ

Γιωτόπουλος Σ, Σημειώσεις Μαθήματος Βάσεις Δεδομένων, ΑΤΕΙ ΠΑΤΡΑΣ, Δεκέμβριος 2010

«Συστήματα Βάσεων Δεδομένων - Η Πλήρης Θεωρία των Βάσεων Δεδομένων -Silberschatz, Korth, Sudarshan - Εκδόσεις Μ.Γκιούρδας -4η Έκδοση»

Ευφυή Πληροφοριακά Συστήματα και Ευρετικές Μέθοδοι, Δρ. Ιωάννης Ζαχαράκης, Πανεπιστημιακές Εκδόσεις ΕΑΠ.

Korth F. Henry, Silberschatz Abraham, Database System Concepts, *McGraw-Hill, Inc*

Date C. J., An Introduction to Database Systems, *Addison-Wesley Publishing Company*

INTEPNET

(<http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>)

(<http://www.isunshare.com/access-password/how-to-encrypt-and-decrypt-access-database.html>)

(<http://databases.about.com/od/tutorials/ss/usersecurity.htm#showall>)