



**ΤΕΧΝΟΛΟΓΙΚΟ
ΕΚΠΑΙΔΕΥΤΙΚΟ
ΙΔΡΥΜΑ
ΔΥΤΙΚΗΣ ΕΛΛΑΔΑΣ**

ΤΜΗΜΑ ΛΟΓΙΣΤΙΚΗΣ ΚΑΙ ΧΡΗΜΑΤΟΟΙΚΟΝΟΜΙΚΗΣ

ΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ ΜΕ ΘΕΜΑ

**Ολοκληρωμένο Σύστημα Διαχείρισης Κέντρου
Ξένων Γλωσσών με χρήση OpenOffice (Base)**

ΑΘΑΝΑΣΙΟΥ ΧΡΥΣΟΥΛΑ Α.Μ. 14883

ΣΤΑΥΡΟΠΟΥΛΟΣ ΠΑΝΤΕΛΕΗΜΩΝ Α.Μ. 15121

ΤΣΙΡΙΚΟΣ ΒΑΣΙΛΕΙΟΣ Α.Μ. 14356

ΕΙΣΗΓΗΤΗΣ : ΚΟΜΠΟΘΡΕΚΑΣ ΑΡΙΣΤΟΤΕΛΗΣ

ΜΕΣΟΛΟΓΓΙ 2016

Πίνακας περιεχομένων

Πρόλογος	5
ΚΕΦΑΛΑΙΟ 1.....	7
Εισαγωγή στις Βάσεις Δεδομένων.....	7
Οργάνωση και Επεξεργασία Αρχείων	8
Εγγραφές, αρχεία και βάσεις δεδομένων	10
Εξέλιξη Βάσεων Δεδομένων	15
Η Αρχιτεκτονική των ΣΔΒΔ	16
Δομή και Είδη βάσεων δεδομένων	17
Το σχεσιακό μοντέλο	19
ΚΕΦΑΛΑΙΟ 2.....	21
Εισαγωγή στην OpenOffice.....	21
OpenOffice Base.....	21
ΚΕΦΑΛΑΙΟ 3.....	25
Οδηγός χρήσης της Βάσης Δεδομένων OpenOffice	25
Άνοιγμα της Εφαρμογής και Δημιουργία Βάσης Δεδομένων	25
Πίνακες.....	28
Σχέσεις.....	32
Ερωτήματα	34
Φόρμες	42
Αναφορές (ή Εκθέσεις)	50
ΚΕΦΑΛΑΙΟ 4.....	55
Σκοπός Δημιουργίας της Βάσης Δεδομένων για ένα Κέντρο Ξένων Γλωσσών.....	55
Ανάλυση Αναγκών-Απαιτήσεων	55
Κανονικοποίηση.....	55
Το Μοντέλο Οντοτήτων –Συσχετίσεων (Entity-Relationship Model).....	57
Βασικές έννοιες μοντέλου Οντοτήτων- Συσχετίσεων (E-R μοντέλου)	58
Διάγραμμα Οντοτήτων-Συσχετίσεων (E-R DIAGRAM).....	60
Υλοποίηση /Ανάλυση Πινάκων-Δεδομένων	63
Σχολιασμός των Σχέσεων των Πινάκων	68

Παραδείγματα Ερωτημάτων.....	70
Βιβλιογραφία.....	84

Πρόλογος

Παρόλο που έχουν περάσει λιγότερα από 50 χρόνια από τότε που κατασκευάστηκε ο πρώτος υπολογιστής, σήμερα οι «απόγονοί τους» έχουν εισχωρήσει σχεδόν σε κάθε δραστηριότητα της ζωής μας. Οι περισσότεροι άνθρωποι γνωρίζουν ότι οι υπολογιστές παίζουν σημαντικό ρόλο στη ζωή τους, αλλά λίγοι αντιλαμβάνονται πόσο διεισδυτικός είναι ο ρόλος αυτός. Εμπλέκονται σε όλες σχεδόν τις πτυχές της καθημερινής μας ζωής, από τον λογαριασμό του σούπερ-μάρκετ έως τον σχεδιασμό των τηλεφωνικών επικοινωνιών και από τη δημιουργία ενός τραπεζικού λογαριασμού έως τον έλεγχο των φωτεινών σηματοδοτών.

Οι βάσεις δεδομένων και η τεχνολογία βάσεων δεδομένων εξασκούν σημαντική επίδραση στην αυξανόμενη χρήση των υπολογιστών. Είναι εύλογο να ειπωθεί ότι οι βάσεις δεδομένων διαδραματίζουν ένα σημαντικό ρόλο σε όλες τις περιοχές όπου χρησιμοποιούνται υπολογιστές, συμπεριλαμβανομένων των επιχειρήσεων, του ηλεκτρονικού εμπορίου, της μηχανικής, της ιατρικής, της εκπαίδευσης, για να αναφέρουμε μερικές μόνο από αυτές.

Τα συστήματα βάσεων δεδομένων τα χρησιμοποιούμε για να μπορούμε να αποθηκεύσουμε, να επεξεργαστούμε αλλά και να εκμεταλλευτούμε αποδοτικά αυτόν τον τεράστιο όγκο πληροφοριών που αυξάνεται με αλματώδεις ρυθμούς καθημερινά.

Στόχος της παρούσας πτυχιακής εργασίας είναι η δημιουργία ενός ολοκληρωμένου συστήματος διαχείρισης ενός Κέντρου Ξένων Γλωσσών, με τη χρήση της βάσης δεδομένων του Open Office.

ΚΕΦΑΛΑΙΟ 1

Εισαγωγή στις Βάσεις Δεδομένων

Ως **βάση δεδομένων** ορίζουμε μια συλλογή από συστηματικά οργανωμένα και σχετιζόμενα μεταξύ τους δεδομένα. Ως δεδομένα ορίζονται στοιχεία, έννοιες και οδηγίες τυποποιημένα σε καθορισμένη μορφή, κατάλληλα για επεξεργασία από ανθρώπους ή μηχανές. Η συλλογή και αποθήκευση ενός τεράστιου όγκου δεδομένων δεν λύνει τελείως το πρόβλημα της σωστής οργάνωσης και ταξινόμησης τους. Τα δεδομένα θα πρέπει να οργανωθούν με τέτοιο τρόπο, ώστε να μπορούν να εντοπίζονται και να αξιοποιούνται εύκολα και γρήγορα τη στιγμή που τα χρειαζόμαστε. Ένας τηλεφωνικός κατάλογος για παράδειγμα θεωρείται βάση δεδομένων, καθώς αποθηκεύει και οργανώνει σχετιζόμενα τμήματα πληροφορίας, όπως είναι το όνομα και ο αριθμός τηλεφώνου. Αν όμως τα ονόματα του καταλόγου δεν είναι καταχωρημένα αλφαβητικά, αλλά εντελώς τυχαία, ένας τέτοιος κατάλογος θα περιείχε μια τεράστια ποσότητα δεδομένων, αλλά θα ήταν ουσιαστικά άχρηστος. Εκτός, λοιπόν, από τη μόνιμη αποθήκευση των δεδομένων, χρειαζόμαστε και κάποιους τρόπους ευέλικτης και αποδοτικής οργάνωσής τους. [1]

Η ανάγκη, λοιπόν, για τη διατήρηση και το χειρισμό μεγάλων ποσοτήτων πληροφοριών, καθώς και για μεγαλύτερη ταχύτητα και ακρίβεια στην επεξεργασία, οδήγησε στην εξέλιξη των βάσεων δεδομένων και στη δημιουργία των Συστημάτων Διαχείρισης Βάσης Δεδομένων (ΣΔΒΔ).

Μια **Βάση Δεδομένων** (ΒΔ) είναι ένα σύνολο αρχείων, με υψηλό βαθμό οργάνωσης, τα οποία είναι συνδεδεμένα μεταξύ τους με λογικές σχέσεις, έτσι ώστε να μπορούν να χρησιμοποιούνται από πολλές εφαρμογές και από πολλούς χρήστες ταυτόχρονα. Το Σύστημα Διαχείρισης Βάσης Δεδομένων (ΣΔΒΔ) είναι ουσιαστικά ένα πρόγραμμα λογισμικού υπολογιστών που σχεδιάζεται για να διαχειρίζονται όλες οι βάσεις δεδομένων που υπάρχουν σε σκληρό δίσκο ή ένα δίκτυο συστημάτων. Είναι δηλαδή ένα σύνολο από προγράμματα που έχουν να κάνουν με το χειρισμό της βάσης δεδομένων, όσον αφορά τη δημιουργία, τροποποίηση, διαγραφή στοιχείων, έλεγχο ασφάλειας κ.ά.

Με την πρόοδο της τεχνολογίας στους τομείς των επεξεργαστών, της μνήμης των υπολογιστών, του χώρου αποθήκευσης στους υπολογιστές και των δικτύων υπολογιστών, τα μεγέθη, οι δυνατότητες και η απόδοση των Βάσεων Δεδομένων και των Συστημάτων Διαχείρισης ΒΔ έχουν αυξηθεί κατά μεγάλες τάξεις μεγέθους. Η αξιοποίηση των βάσεων δεδομένων έχει εξαπλωθεί τόσο, ώστε οργανισμοί και

εταιρείες, από τις πιο μικρές έως τις πιο μεγάλες, εξαρτώνται σε μεγάλο βαθμό από τις ΒΔ για τη λειτουργία τους. [1]

Οργάνωση και Επεξεργασία Αρχείων

Τα δεδομένα τα οποία χρησιμοποιεί ο υπολογιστής στις διάφορες επεξεργασίες είναι συνήθως αποθηκευμένα στις μονάδες βοηθητικής μνήμης. Τα δεδομένα είναι οργανωμένα με διάφορες μεθόδους, για τις οποίες υπάρχουν και ανάλογες μέθοδοι προσπέλασης και επεξεργασίας. Οι προγραμματιστές και οι αναλυτές είναι αυτοί που σχεδιάζουν τον τρόπο συλλογής, οργάνωσης και αποθήκευσης των δεδομένων και επιπλέον τον τρόπο της επεξεργασίας τους στον υπολογιστή.

Τα δεδομένα μιας βάσης οργανώνονται για την αποθήκευση και την αξιοποίησή τους από διάφορες εφαρμογές στους δίσκους. Η **φυσική οργάνωση** αναφέρεται στον τρόπο αποθήκευσης και προσπέλασης στα δεδομένα αυτά. Το έργο αυτό αναλαμβάνει ο υπεύθυνος διαχείρισης της βάσης δεδομένων (data base administrator). Λαμβάνοντας υπόψη τους περιορισμούς, όπως επανάληψη δεδομένων, χρόνο και περιορισμούς στο χώρο αποθήκευσης, ο διαχειριστής της βάσης προσπαθεί να εφαρμόσει το λογικό σχεδιασμό μέσα στις φυσικές εγγραφές και στα αρχεία που αποθηκεύονται στη βάση δεδομένων.

Η **Λογική οργάνωση** αναφέρεται στον τρόπο απεικόνισης των δεδομένων στα προγράμματα εφαρμογών ή στους μεμονωμένους χρήστες. Ο λογικός σχεδιασμός είναι έργο του αναλυτή συστημάτων ή του αναλυτή εφαρμογών. Οι λογικές εγγραφές πρέπει να σχεδιάζονται ανεξάρτητα από τον τρόπο αποθήκευσης στα φυσικά μέσα.

Η επεξεργασία δεδομένων καλύπτει μια ευρεία γκάμα εφαρμογών, κάθε μια με τις δικές της ξεχωριστές απαιτήσεις για την αποθήκευση και τον χειρισμό των διαφορετικών ειδών δεδομένων. Κατ'αρχήν, τα δεδομένα που αποθηκεύονται και χρησιμοποιούνται από ένα σύστημα επεξεργασίας δεδομένων αντιπροσωπεύουν πληροφορίες για εκείνο το τμήμα του «κόσμου» με το οποίο σχετίζεται η συγκεκριμένη εφαρμογή. Για ένα κέντρο ξένων γλωσσών, για παράδειγμα, ο «κόσμος» αποτελείται από τους μαθητές, τους καθηγητές κτλ. και αυτές είναι οι **οντότητες** (entities) ή **αντικείμενα** (objects) τα οποία αντιπροσωπεύονται στο σύστημα διαχείρισής του από τα δεδομένα.

Κάθε οντότητα χαρακτηρίζεται από έναν αριθμό ιδιοτήτων όπως όνομα, διεύθυνση, αριθμός τηλεφώνου, e-mail κλπ. Οι ιδιότητες οι οποίες είναι σχετικές και για τις οποίες αποθηκεύονται τα δεδομένα, ονομάζονται **γνωρίσματα** (attributes).

Οντότητες του ίδιου είδους έχουν το ίδιο σύστημα γνωρισμάτων και σχηματίζουν ένα **σύνολο οντοτήτων** (entity set) ή μια **κλάση οντοτήτων** (entity class). Έτσι, για παράδειγμα, σε μια εφαρμογή διαχείρισης κέντρου ξένων γλωσσών το σύνολο όλων των μαθητών σχηματίζει ένα σύνολο οντοτήτων, αφού κάθε μαθητής έχει τα γνωρίσματα : όνομα, διεύθυνση, τηλέφωνο κ.ο.κ. [2]

Παρόλο που διάφορες οντότητες σ'ένα σύνολο οντοτήτων μπορούν να έχουν τις ίδιες τιμές για συγκεκριμένα γνωρίσματα (π.χ. διάφοροι μαθητές παρακολουθούν μάθημα στο ίδιο τμήμα), πρέπει να υπάρχει τουλάχιστον ένα γνώρισμα (ή συνδυασμός χαρακτηριστικών γνωρισμάτων) το οποίο έχει μια μοναδική τιμή για κάθε οντότητα μέσα στο σύνολο. Διαφορετικά, θα ήταν αδύνατο να διαχωρίσουμε δύο οντότητες με τις ίδιες ακριβώς τιμές για όλα τα γνωρίσματα, αφού αυτές θα ταυτίζονταν. Το γνώρισμα που χρησιμοποιείται για τη διάκριση των μελών ενός συνόλου οντοτήτων ονομάζεται **κλειδί** (key) του συνόλου οντοτήτων, και εξ'ορισμού η τιμή του είναι μοναδική για κάθε οντότητα. Πολλές φορές μια εγγραφή μπορεί να έχει παραπάνω από ένα κλειδιά. Τα κλειδιά στις περιπτώσεις αυτές διακρίνονται στο **πρωτεύον κλειδί** (primary key) , στο δευτερεύον κλειδί (secondary key) κ.λ. Για παράδειγμα ας πάρουμε ένα αρχείο μαθητών, για κάθε μαθητή έχουμε μια εγγραφή που έχει την παρακάτω δομή:

	Περιγραφή πεδίου	Μήκος
1	Αριθμός Μητρώου	5
2	Επώνυμο	30
3	Όνομα	20
4	Πατρώνυμο	20
5	Διεύθυνση κατοικίας	40
6	Τηλέφωνο	15
7	Λοιπά στοιχεία	50

Το πεδίο «Αριθμός Μητρώου» είναι πρωτεύον κλειδί και προσδιορίζει μόνο μια εγγραφή, ένα μαθητή. Το πεδίο «Επώνυμο» μπορεί να θεωρηθεί ως το δευτερεύον κλειδί.

Τέλος, οι οντότητες δεν είναι απομονωμένες, αλλά έχουν **σχέσεις** (relationships) μεταξύ τους. Αυτές οι σχέσεις είναι σημαντικές στον κόσμο των εφαρμογών με τον οποίο σχετίζονται και πρέπει να αναπαρίστανται στα δεδομένα τα οποία τον περιγράφουν.

Οι έννοιες που παρουσιάστηκαν παραπάνω, μας επιτρέπουν να μιλάμε για δεδομένα με τρόπους οι οποίοι είναι ανεξάρτητοι από μια συγκεκριμένη εφαρμογή.

Αποτελούν τη βάση των **μοντέλων δεδομένων** (data models) των οποίων ο στόχος είναι να περιγράψουν τυπικά τους τρόπους με τους οποίους τα δεδομένα δομούνται και χρησιμοποιούνται. Για να είναι γενικά χρήσιμο ένα μοντέλο δεδομένων πρέπει να ικανοποιεί τουλάχιστον τις ακόλουθες τρεις απαιτήσεις των δεδομένων:

1. **Δομή** (Structure): Το μοντέλο πρέπει να εξασφαλίζει τα μέσα αναπαράστασης των οντοτήτων, των γνωρισμάτων και των σχέσεων.
2. **Χειρισμός** (manipulation): Πρέπει να είναι δυνατό να χειρίζεται τις παραπάνω αναπαραστάσεις, να αντανακλά, για παράδειγμα, την πρόσθεση νέων οντοτήτων ή κάποιες αλλαγές στα χαρακτηριστικά γνωρίσματα.
3. **Ακεραιότητα** (integrity): Σε όλες τις εφαρμογές υπάρχουν πολυάριθμοι περιορισμοί οι οποίοι ισχύουν στις οντότητες, τα γνωρίσματα και τις σχέσεις. Για παράδειγμα, ένα μαθητής δεν μπορεί να παρακολουθεί δυο τμήματα ταυτόχρονα (την ίδια ώρα). Ένα μοντέλο δεδομένων πρέπει να είναι ικανό να εκφράζει αυτούς τους περιορισμούς (και η υλοποίηση πρέπει να είναι ικανή να τους επιβάλλει).

Τα μοντέλα δεδομένων απευθύνονται σε αυτές τις τρεις πλευρές με ποικίλους βαθμούς επιτυχίας. Το πιο αδύναμο σημείο στα περισσότερα μοντέλα είναι η ακεραιότητα. [2]

Εγγραφές, αρχεία και βάσεις δεδομένων

Πολύ πριν την ανάπτυξη τυπικών μοντέλων δεδομένων είχε αναγνωριστεί ότι την πληροφορία, σε μεγάλες ποσότητες, μπορεί κανείς να τη διαχειριστεί μόνο αν είναι δομημένη με κάποιο τρόπο. Δομές δεδομένων επινοήθηκαν για να διευκολύνουν την αποθήκευση και την ανάκληση μεγάλων ποσοτήτων δεδομένων στην και από την δευτερεύουσα μνήμη. Αυτές οι δομές χρησιμοποιούνται ακόμα στις εφαρμογές επεξεργασίας δεδομένων ανεξάρτητα από το εάν χρησιμοποιείται ένα μοντέλο δεδομένων ή όχι.

Η βασική δομική μονάδα στην επεξεργασία δεδομένων είναι η **εγγραφή** (record), η οποία είναι συλλογή πληροφοριών για μια συγκεκριμένη οντότητα, όπως ένα συγκεκριμένο άτομο, ένα είδος μιας αποθήκης ή ένας τραπεζικός λογαριασμός. Κάθε εγγραφή αποτελείται από έναν αριθμό **πεδίων** (fields), καθένα από τα οποία κρατά ένα και μόνο τμήμα της πληροφορίας για την οντότητα που περιγράφει η εγγραφή. Εγγραφές του ίδιου τύπου (που περιγράφουν οντότητες του ίδιου τύπου)

έχουν τα ίδια πεδία, παρόλο που οι πληροφορίες στα πεδία διαφέρουν φυσικά από η μια εγγραφή στην άλλη.

Εγγραφές του ίδιου τύπου που σχετίζονται μεταξύ τους καταχωρούνται ως **αρχεία** (files) στις μονάδες περιφερειακής αποθήκευσης. Σε μια τυπική εφαρμογή επεξεργασίας δεδομένων ένα αρχείο μπορεί να περιέχει χιλιάδες ή ακόμα και εκατοντάδες χιλιάδες εγγραφές.

Σε πολλές εφαρμογές οι εγγραφές σε ένα αρχείο πρέπει να είναι μοναδικές, δηλαδή δεν πρέπει να υπάρχουν διπλές. Σε τέτοιες περιπτώσεις, κάθε εγγραφή μπορεί να προσδιοριστεί από κάποιο ή από όλα τα πεδία της. Το πεδίο (ή συνδυασμός πεδίων), το οποίο προσδιορίζει μια εγγραφή ονομάζεται **κλειδί** (key).

Σύνολο αρχείων που συνδέονται μεταξύ τους μέσω μιας εφαρμογής ονομάζεται συνήθως **βάση δεδομένων** (data base). Ο όρος «βάση δεδομένων» συνήθως υπονοεί ότι τα συγκεκριμένα αρχεία συνδέονται πιο στενά: δηλαδή εγγραφές από ένα αρχείο αναφέρονται (πιθανότατα εμμέσως) σε εγγραφές ενός άλλου αρχείου.

Όπως έχει ήδη αναφερθεί, το λογισμικό που χρειάζεται για τη διαχείριση μιας βάσης δεδομένων ονομάζεται **Σύστημα Διαχείρισης Βάσεων Δεδομένων, ΣΔΒΔ** (Data Base Management System, DBMS). Ένα ΣΔΒΔ περιέχει προγράμματα τόσο για τη δημιουργία μιας βάσης δεδομένων, όσο και για την πρόσβαση και διαχείριση των δεδομένων που περιέχει. Ένα ΣΔΒΔ που βασίζεται στο σχεσιακό μοντέλο ονομάζεται, όπως θα αναμενόταν, **Σχεσιακό Σύστημα Διαχείρισης Βάσεων Δεδομένων** (Relational DBMS). [2]

Οι στοιχειώδεις πράξεις ή **δοσοληψίες** (transactions) που γίνονται πάνω στις εγγραφές είναι:

1. **Διάβασμα** (reading): Δηλαδή η μεταφορά μιας εγγραφής από το δευτερεύοντα χώρο αποθήκευσης στην κεντρική μνήμη, ώστε να είναι δυνατή η πρόσβαση στα πεδία της
2. **Γράψιμο** (writing): Δηλαδή η μεταφορά μιας εγγραφής από τη μνήμη στον δευτερεύοντα χώρο αποθήκευσης
3. **Πρόσθεση** (adding) μιας εγγραφής σε ένα αρχείο
4. **Διαγραφή** (deleting) μιας εγγραφής από ένα αρχείο
Οι λειτουργίες διάβασμα και γράψιμο συνδυάζονται στην επόμενη λειτουργία:
5. **Ενημέρωση** (updating) μιας εγγραφής: δηλαδή μεταβολή των πληροφοριών που περιέχονται σε ένα ή περισσότερα πεδία της

Η μεταφορά εγγραφών από το δευτερεύοντα χώρο αποθήκευσης στη μνήμη είναι ευθύνη του λειτουργικού συστήματος του υπολογιστή.

Η χρήση Βάσεων δεδομένων παρέχει τα παρακάτω πλεονεκτήματα:

1. Δυνατότητα ταυτόχρονης προσπέλασης πολλών χρηστών στα δεδομένα και έλεγχο της προσπέλασης.

Ένα βασικό κέρδος απ'τη χρήση Βάσης Δεδομένων είναι η δυνατότητα που παρέχει σε πολλούς χρήστες να έχουν ταυτόχρονη προσπέλαση στα δεδομένα, αλλά και η αυτοματοποίηση των ελέγχων που συνεπάγεται αυτή η ταυτόχρονη προσπέλαση.

2. Δυνατότητα ταχύτατης εξαγωγής απαντήσεων σε απλές ερωτήσεις.

Ένα μεγάλο πρόβλημα της οργάνωσης ενός μοντέλου του πραγματικού κόσμου χωρίς τη χρήση Βάσης Δεδομένων είναι η αδυναμία εξαγωγής απαντήσεων σε απλές ερωτήσεις, χωρίς να χρειάζεται αρκετή εργασία για να αντληθεί η απαιτούμενη πληροφορία.

3. Ευελιξία σε τυχόν αλλαγές και γενικότερα ευκολία παρακολούθησης των αλλαγών του μοντέλου του πραγματικού κόσμου.

Πολλές φορές το μοντέλο του πραγματικού κόσμου αλλάζει. Αυτό κατά κανόνα απαιτεί αλλαγές που τις περισσότερες φορές είναι πολύ δύσκολο να υλοποιηθούν σε ένα παραδοσιακό μοντέλο.

4. Υψηλή ποιότητα δεδομένων.

Η βάση δεδομένων χρησιμοποιεί τις δυνατότητες που παρέχει το σύστημα Διαχείρισης Βάσεων Δεδομένων και ελέγχει για πλεονασμό και παραβιάσεις της ορθότητας των δεδομένων, καθώς και για ταυτόχρονης προσπελάσεις και μη εξουσιοδοτημένες εγγραφές. Με αυτό τον τρόπο δια σφαλίζεται η ποιότητα των δεδομένων, δηλαδή ο χρήστης έχει μεγαλύτερη ασφάλεια ότι τα δεδομένα που παίρνει στις απαντήσεις του είναι σωστά.

Η αξιοπιστία των δεδομένων αποτελεί βασικό παράγοντα στην επιτυχία των εφαρμογών που υποστηρίζει μία βάση. Όταν χάνονται στοιχεία από τη βάση, όταν δεν συνδέονται οι πληροφορίες που συγγενεύουν και τις οποίες αναμένει ο χρήστης, όταν οι αναφορές περιλαμβάνουν αλληλοσυγκρουόμενα στοιχεία κλπ, ο χρήστης πολύ σύντομα χάνει την εμπιστοσύνη του στο λογισμικό της βάσης, αλλά και στο περιβάλλον της βάσης γενικά. Ο σχεδιαστής λοιπόν, πέραν των τεχνικών ανάλυσης οντοτήτων και λογικών τύπων εγγραφών, πρέπει να ερευνήσει διάφορα άλλα θέματα που αφορούν στο σύνολο της βάσης, ιδιαίτερα στη χρήση αυτής από διαφορετικές κατηγορίες χρηστών. [1]

Τα κύρια σημεία που είναι και τα επιθυμητά χαρακτηριστικά που πρέπει να διακρίνουν τη βάση είναι:

- a) Ακεραιότητα
- b) Συνέπεια
- c) Αποκατάσταση
- d) Ασφάλεια
- e) Αποδοτικότητα

ΑΚΕΡΑΙΟΤΗΤΑ

Μια βάση είναι ακέραια εφόσον η δομή και τα στοιχεία αυτής συμμορφώνονται με συγκεκριμένους κανόνες κάτω από όλες τις συνθήκες ενημέρωσης των καταχωρημένων δεδομένων. Το σχεσιακό περιβάλλον ανάπτυξης βάσεων δεδομένων είναι ιδιαίτερα ευέλικτο στην εφαρμογή κανόνων ακεραιότητας.

ΣΥΝΕΠΕΙΑ

Μια βάση έχει συνέπεια όταν κατά την ταυτόχρονη χρήση αυτής από διαφορετικούς χρήστες που υποβάλουν τη ίδια ερώτηση, τα αποτελέσματα που παράγονται είναι ίδια για όλους τους χρήστες. Η συνέπεια της βάσης επιτυγχάνεται με την τεχνική του κλειδώματος, όπου το αρχείο που ενημερώνει κάποιος χρήστης κλειδώνεται και, κατά συνέπεια, δεν είναι διαθέσιμο σε άλλους χρήστες για ταυτόχρονη επεξεργασία. Ο χρήστης που ζητά την χρήση κλειδωμένου αρχείου παίρνει συνήθως κάποιο επεξηγηματικό μήνυμα, ενώ το πρόγραμμα μιας εφαρμογής που προσπαθεί να ανοίξει ένα κλειδωμένο αρχείο παίρνει ως μήνυμα ένα αντίστοιχο κωδικό με τον οποίο ελέγχεται η παραπέρα ροή του προγράμματος.

ΑΠΟΚΑΤΑΣΤΑΣΗ

Με τον όρο αποκατάσταση εννοούμε τη δυνατότητα που παρέχει το περιβάλλον της βάσης και το σχετικό λογισμικό αυτής, να επαναφέρεται η βάση σε κατάσταση ακεραιότητας ύστερα από κάποιο σφάλμα, απροσδόκητη διακοπή ή γενικά κάποια λειτουργία που παραβιάζει την ακεραιότητα ή την συνέπεια των δεδομένων. Για να είμαστε σε θέση να παναφέρουμε τη βάση στην προηγούμενη κατάστασή της, χωρίς φυσικά την απώλεια δεδομένων ή/και συναλλαγών που έχουν γίνει, το ΣΔΒΔ πρέπει να διεξάγει διάφορους ελέγχους και να καταγράφει διάφορες πληροφορίες. Οι πληροφορίες αυτές καταγράφονται σε κατάλληλα ημερολογιακά αρχεία (log files).

ΑΣΦΑΛΕΙΑ

Μια βάση παρέχει ασφάλεια όταν προστατεύει τόσο τα αποθηκευμένα στοιχεία όσο και τη δομή της από εκούσια ή ακούσια, αλλά μη αποδεκτή τροποποίηση, απώλεια ή ανάκτηση των δεδομένων. Βασικός σκοπός της σχεδίασης εδώ είναι η προστασία των δεδομένων από μη εξουσιοδοτημένους χρήστες, με το ελάχιστο δυνατό κόστος. Κατά συνέπεια, το πρόβλημα μεταφέρεται στην υλοποίηση κατάλληλων μηχανισμών πρόσβασης στις εγγραφές της βάσης- ένα πρόβλημα που γίνεται ιδιαίτερα έκδηλο σε περιβάλλοντα πολλών χρηστών, πολλαπλής επεξεργασίας, ταυτόχρονων περιβάλλοντων και παράλληλης επεξεργασίας. Ένα μέσο επιβολής ασφάλειας είναι το λεγόμενο υπόσχημα, το οποίο είναι ένα υποσύνολο της βάσης με τα δικά του δομικά χαρακτηριστικά, στο οποίο έχει πρόσβαση μια κατηγορία χρηστών, συνήθως για μια συγκεκριμένη εφαρμογή. Η όψη είναι ένα άλλο μέσο επιβολής ασφάλειας, ενώ παράλληλα παρέχει στον χρήστη μόνο εκείνο το μέρος της βάσης που χρειάζεται κάθε φορά. Επιπλέον, κάθε όψη μπορεί να έχει το δικό της επίπεδο πρόσβασης καθώς και τη δική της ομάδα χρηστών.

ΑΠΟΔΟΤΙΚΟΤΗΤΑ

Η αποδοτικότητα μιας βάσης αναφέρεται τόσο στη λειτουργικότητα των στοιχείων που καταχωρούνται και των συσχετίσεων μεταξύ αυτών, όσο και στο χρόνο που απαιτείται για την εξυπηρέτηση των εφαρμογών που υποστηρίζονται από το σχήμα της βάσης. Κατά συνέπεια, η αποδοτικότητα μιας βάσης εξαρτάται από τους ακόλουθους παράγοντες:

1. Το λογικό σχήμα και την πληρότητα, καθώς και τη λειτουργικότητα που αυτό παρέχει στην εξυπηρέτηση των εφαρμογών
2. Το φυσικό σχήμα και τις δομές που παρέχει για βέλτιστη δυνατή καταχώριση των δεδομένων, δηλαδή τη βέλτιστη φυσική δόμηση των εγγραφών του λογικού σχήματος
3. Τον αποθηκευτικό χώρο που απαιτείται- που πρέπει να είναι ο ελάχιστος δυνατός- για την καταχώριση των στοιχείων του υπό διαπραγμάτευση οργανισμού
4. Τον χρόνο απόκρισης του συστήματος, όπου κάθε εφαρμογή μπορεί να έχει διαφορετικές απαιτήσεις

Σκοπός του σχεδιαστή είναι να βελτιστοποιήσει το λογικό και το φυσικό σχήμα, έχοντας υπόψιν του πάντοτε τις τέσσερις βασικές αρχές σχεδιασμού (εννοιολογική αρχή, αρχή της αυτάρκειας, λογική και φυσική ανεξαρτησία) και να μειώσει τόσο τον αποθηκευτικό χώρο όσο και το μέσο χρόνο απόκρισης. Αυτά πρέπει να τα

επιτυγχάνει, χωρίς να παραβιάζεται ή να παρεμποδίζεται η ακεραιότητα, η συνέπεια, η δυνατότητα αποκατάστασης και η ασφάλεια των δεδομένων.

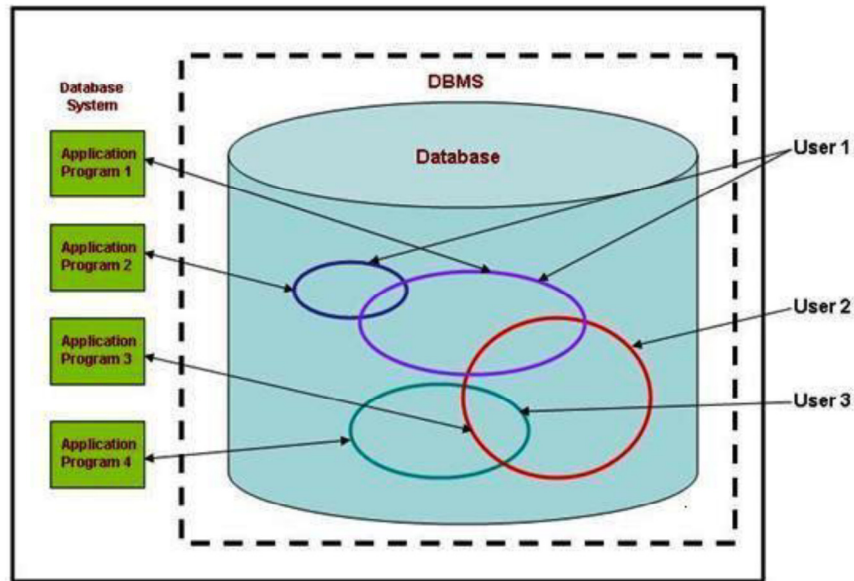
Τέλος, είναι αυτονόητο ότι μια καλοσχεδιασμένη βάση δεδομένων απαιτεί ένα καλοσχεδιασμένο σύστημα ηλεκτρονικού υπολογιστή, με τη σχετική απόδοση, ιδιαίτερα σε χρόνους μεταφοράς δεδομένων από και προς τους δίσκους και την κεντρική μνήμη. [1]

Εξέλιξη Βάσεων Δεδομένων

Η εισαγωγή του όρου «Βάση Δεδομένων» συνέπεσε με τη διαθεσιμότητα της άμεσης αποθήκευσης σε κατάλληλα μέσα (σκληρούς δίσκους και δισκέτες) από τα μέσα της δεκαετίας του 1960 και μετά. Ο όρος αντιπροσώπευε μια αντίθεση με τα *taped-based* συστήματα του παρελθόντος, επιτρέποντας την κοινή διαδραστική χρήση.

Στα πρώτα συστήματα ΒΔ, η αποτελεσματικότητα ήταν ίσως το κυριότερο μέλημα, αλλά είχαν ήδη αναγνωριστεί κι άλλοι στόχοι. Ένας από τους βασικότερους, ήταν να γίνουν τα δεδομένα ανεξάρτητα από τη λογική της εφαρμογής αποκλειστικά σε κάποια προγράμματα, έτσι ώστε να μπορούν να διατεθούν σε διαφορετικές εφαρμογές.

Από το 1970 και μετά, η τεχνολογία βάσεων δεδομένων έχει συμβαδίσει με τους αυξανόμενους πόρους που καθίστανται διαθέσιμοι μέσω της τεχνολογίας υπολογιστών και συγκεκριμένα μέσω της ταχείας αύξησης της χωρητικότητας και της ταχύτητας (με ταυτόχρονη μείωση των τιμών) των μέσων αποθήκευσης και της αύξησης της χωρητικότητας της κύριας μνήμης. [4]



Εικόνα 1: Σχεδιάγραμμα ΒΔ και ΣΔΒΔ

Η Αρχιτεκτονική των ΣΔΒΔ

Όπως είδαμε νωρίτερα, ένα ΣΔΒΔ (Σύστημα Διαχείρισης Βάσης Δεδομένων) έχει σαν αποστολή τη διαχείριση των δεδομένων των αρχείων της βάσης, δηλαδή 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 δημιουργούμε δεν απασχολούνται με τον τρόπο που είναι αποθηκευμένα τα δεδομένα, πόσο χώρο καταλαμβάνουν κ.ο.κ και αυτή η ιδιότητα είναι γνωστή ως *ανεξαρτησία δεδομένων*.

Αυτό σημαίνει πρακτικά ότι οποιαδήποτε αλλαγή στον τρόπο οργάνωσης των αρχείων της βάσης δεδομένων δε θα συνεπάγεται και αλλαγή στις εφαρμογές, ένα πρόβλημα που ταλαιπωρούσε πολύ τους προγραμματιστές παλαιότερων εποχών. Ακόμη η προσθήκη, η κατάργηση ή και η τροποποίηση κάποιων εφαρμογών δε θα έχει καμιά επίπτωση στον τρόπο οργάνωσης των αρχείων της βάσης δεδομένων. Στα ΣΔΒΔ έχει επικρατήσει η λεγόμενη αρχιτεκτονική των τριών επιπέδων (βαθμίδων) όπου τα τρία επίπεδα είναι τα εξής:

- Εσωτερικό επίπεδο (Internal level), αφορά στην αποθήκευση των αρχείων στο σκληρό δίσκο, δηλαδή την πραγματική ή φυσική κατάσταση τους.

- Εξωτερικό επίπεδο (external level), αφορά στους χρήστες είτε αυτοί είναι απλοί χειριστές, είτε προγραμματιστές ή και οι διαχειριστές της βάσης δεδομένων.
- Εννοιολογικό επίπεδο (conceptual level), ένα ενδιάμεσο επίπεδο που διασυνδέει τα δύο άλλα επίπεδα και αφορά τη λογική σχεδίαση των αρχείων της βάσης δεδομένων.

Δομή και Είδη βάσεων δεδομένων

Ο σχεδιασμός, η κατασκευή και η συντήρηση μιας σύνθετης βάσης δεδομένων απαιτεί ειδικές δεξιότητες: υπάρχει εξειδικευμένο προσωπικό που εκτελεί αυτές τις λειτουργίες και αναφέρονται ως προγραμματιστές εφαρμογών βάσεων δεδομένων και διαχειριστές ΒΔ. Οι γλώσσες των βάσεων δεδομένων είναι συγκεκριμένες για ένα μοντέλο δεδομένων, και σε πολλές περιπτώσεις για έναν συγκεκριμένο τύπο ΣΔΒΔ. Η πιο ευρέως υποστηριζόμενη γλώσσα βάσης δεδομένων είναι η SQL, η οποία έχει αναπτυχθεί για το σχεσιακό μοντέλο δεδομένων.

Οι χρήστες των εφαρμογών αντλούν τα στοιχεία που τους ενδιαφέρουν από τη βάση δεδομένων, χωρίς να είναι σε θέση να γνωρίζουν με ποιο τρόπο είναι οργανωμένα τα δεδομένα σ'αυτή. Το ΣΔΒΔ παίζει το ρόλο του μεσάζοντα ανάμεσα στο χρήστη και τη βάση δεδομένων.

Οι περισσότεροι γνωστές δομές των Βάσεων Δεδομένων είναι:

- Η ιεραρχική
- Η δικτυωτή
- Η σχεσιακή

Κάθε δομή έχει διαφορετικό τρόπο οργάνωσης και χρήσης των δεδομένων. Οι ιεραρχικές και δικτυωτές δομές χρησιμοποιούνται κυρίως σε μεγάλους και σε μεσαίου μεγέθους υπολογιστές. Η σχεσιακή δομή, που είναι και η νεότερη, είναι δυνατόν να χρησιμοποιηθεί σε κάθε τύπο υπολογιστή.

Το ιεραρχικό μοντέλο έχει μια ιεραρχική δομή που θυμίζει δέντρο. Οι οντότητες μοιάζουν με απολήξεις από κλαδιά δένδρων και τοποθετούνται σε επίπεδα ιεραρχίας. Τα κλαδιά παριστάνουν τις συσχετίσεις ανάμεσα στις οντότητες. Από μια οντότητα που βρίσκεται σε ένα ανώτερο επίπεδο εκκινούν πολλά κλαδιά, καθένα από τα οποία καταλήγει σε μια οντότητα που βρίσκεται σε ένα χαμηλότερο επίπεδο,

αλλά σε κάθε οντότητα που βρίσκεται σε ένα χαμηλότερο επίπεδο αντιστοιχεί μία και μόνο μία οντότητα που βρίσκεται σε ένα ανώτερο επίπεδο. Το μοντέλο αυτό ήταν το πρώτο που εμφανίστηκε, αλλά σήμερα θεωρείται δύσχρηστο και ξεπερασμένο.

Στο δικτυωτό μοντέλο τα στοιχεία τοποθετούνται σε ένα επίπεδο ιεραρχίας, αλλά κάθε στοιχείο μπορεί να συσχετιστεί με πολλά στοιχεία, είτε σε ένα κατώτερο ή σε ένα ανώτερο επίπεδο.

Το σχεσιακό μοντέλο έχει επικρατήσει σήμερα στην αναπαράσταση των δεδομένων, καθώς διαθέτει σημαντικά πλεονεκτήματα ως προς τα άλλα δύο, και οι βάσεις δεδομένων που σχεδιάζονται σύμφωνα με αυτό αποκαλούνται σχεσιακές. [1]

Το σχεσιακό μοντέλο

Η πρώτη γενιά των συστημάτων βάσεων δεδομένων ήταν βάσεις πλοήγησης (navigational), όπου οι εφαρμογές συνήθως είχαν πρόσβαση σε δεδομένα ακολουθώντας δείκτες από μια εγγραφή σε μια άλλη. Το **Σχεσιακό Μοντέλο** (Relational Model) αναπτύχθηκε γύρω στο 1970 από τον Ted Codd, ο οποίος επέμενε ότι οι αιτήσεις ενός χρήστη θα πρέπει να αναζητούν δεδομένα με βάση το περιεχόμενο και όχι ακολουθώντας συνδέσμους. Το σχεσιακό μοντέλο αποτελεί τη βάση των περισσότερων σύγχρονων συστημάτων βάσεων δεδομένων. Έχει επικρατήσει, καθώς διαθέτει σημαντικά πλεονεκτήματα. Πιο συγκεκριμένα, με τις σχεσιακές βάσεις δεδομένων διαθέτουμε έναν σαφή, απλό και εύκολα κατανοητό τρόπο για να μπορέσουμε να αναπαραστήσουμε και να διαχειριστούμε τα δεδομένα μας. Υστερούν μόνο σε ταχύτητα υπολογισμών και σε χώρο 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.

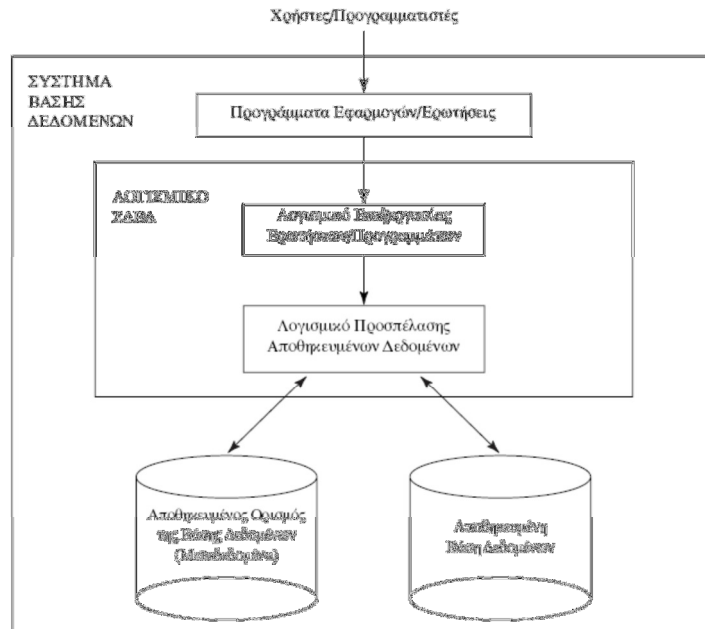
Ένας πίνακας αποτελείται από γραμμές και στήλες, όπου τοποθετούμε τα στοιχεία σε οριζόντια και κάθετη μορφή. Η κάθε στήλη του πίνακα χαρακτηρίζει κάποια ιδιότητα της οντότητας και αποκαλείται **χαρακτηριστικό** ή **πεδίο**, ενώ η κάθε γραμμή του πίνακα περιέχει όλες τις πληροφορίες (στήλες) που αφορούν σε ένα στοιχείο της οντότητας και αποκαλείται **πλειάδα** ή **εγγραφή**. Κάθε πεδίο του πίνακα μπορεί να πάρει ορισμένες μόνο τιμές, οι οποίες μπορεί να καθορίζονται από τον τύπο δεδομένων της ιδιότητας, όπως ονόματα ή αριθμοί για παράδειγμα, ή και από αυτό που εκφράζει, όπως το ότι δεν μπορούμε να έχουμε αρνητικό βάρος ή αρνητικό Α.Φ.Μ. Το σύνολο των αποδεκτών τιμών μιας οντότητας αποκαλείται **πεδίο ορισμού**.

Πολλά πρόσφατα συστήματα διαχείρισης βάσεων δεδομένων έχουν στηριχθεί το σχεσιακό μοντέλο, ελάχιστα όμως από αυτά το υλοποιούν απόλυτα. Για να είναι ένα ΣΔΒΔ πλήρως σχεσιακό πρέπει να υλοποιεί και τις τρεις πλευρές του μοντέλου – δομή, χειρισμός, ακεραιότητα – όπως περιγράφηκαν πιο πάνω. Τα σημεία στα οποία αποτυγχάνουν τα περισσότερα σημερινά συστήματα είναι

1. Αποτυχία να μεταχειρίζονται όλα τα δεδομένα σας σχέσεις
2. Αποτυχία να υλοποιήσουν όλες τις σχεσιακές πράξεις

3. Αποτυχία να επιβάλλουν τους περιορισμούς ακεραιότητας

Παρόλα αυτά, το σχεσιακό μοντέλο είναι η πιο κοινή βάση για ένα σύγχρονο ΣΔΒΔ και πρέπει να αναμένεται ότι όλο και περισσότερα συστήματα στο μέλλον θα είναι πλήρως σχεσιακά.



Εικόνα 2: Σύστημα βάσης Δεδομένων

(Elmasri and Shamkant B. Navathe Ελληνική Εκδόσεις Δίαυλος, Επιμέλεια Μ.Χατζόπουλος)

Για να μπορέσουμε να προσθέσουμε, να διαγράψουμε ή να τροποποιήσουμε τα στοιχεία που περιέχονται σε μια βάση δεδομένων χρησιμοποιούμε ειδικές γλώσσες προγραμματισμού, που αποκαλούνται γλώσσες ερωταπαντήσεων (Query Languages). Μια φιλική προς τον χρήστη γλώσσα προγραμματισμού, για να μπορούμε να υποβάλλουμε ερωτήματα σε σχεσιακές βάσεις δεδομένων και να λαμβάνουμε απαντήσεις είναι η QBE (Query By Example), η οποία χρησιμοποιεί φόρμες για τη γραφική απεικόνιση των ερωτημάτων μας.

Η γλώσσα που αποτελεί σήμερα ένα διεθνές πρότυπο για την επικοινωνία των χρηστών με τα σχεσιακά Συστήματα Διαχείρισης Βάσης Δεδομένων είναι η **SQL** (Structured Query Language) ή Δομημένη Γλώσσα Ερωτημάτων. Μπορεί να λειτουργήσει αυτόνομα, αλλά και σε συνεργασία με άλλες γλώσσες προγραμματισμού.

ΚΕΦΑΛΑΙΟ 2

Εισαγωγή στην OpenOffice

Το Apache OpenOffice είναι ένα ολοκληρωμένο πακέτο εφαρμογών γραφείου ανοιχτού κώδικα. Περιλαμβάνει: Επεξεργαστή Κειμένου (Writer), Υπολογιστικό Φύλλο (Calc), Δημιουργό Παρουσιάσεων (Impress), Πρόγραμμα Σχεδίασης (Draw), Βάση Δεδομένων (Base) και Επεξεργαστή Εξισώσεων (Math). Λειτουργεί σε όλες τις γνώστες πλατφόρμες λειτουργικών συστημάτων: Windows, Linux και Mac, ενώ υποστηρίζει άψογα και την Ελληνική γλώσσα.

Διαθέτει την ίδια λειτουργικότητα με άλλα δημοφιλή προγράμματα εφαρμογών γραφείου (όπως το MS Office) και έτσι αποτελεί την ιδανική λύση για επιχειρήσεις, σχολεία και οργανισμούς, καθώς μπορούν να το αποκτήσουν ελεύθερα. Το OpenOffice μπορεί να διαβάζει και να γράφει αρχεία Word, Excel και PowerPoint.

OpenOffice Base

Το OpenOffice Base είναι το πρόγραμμα χειρισμού βάσεων δεδομένων στην ομάδα προγραμμάτων **Apache OpenOffice** – δηλαδή ένα πρόγραμμα που παρέχει εξειδικευμένες λειτουργίες για την αποθήκευση, τον χειρισμό και την ανάλυση μεγάλων ποσοτήτων με δεδομένα, τα οποία είναι αποθηκευμένα σε πίνακες. Το Apache OpenOffice σχεδιάστηκε με σκοπό να αποτελέσει μια συμβατή και ελεύθερη εναλλακτική του Microsoft Office. Το OpenOffice Base αποτελεί, όπως ειπώθηκε, διαχείριση βάσης δεδομένων αντίστοιχη του Access της Microsoft.

Ένα αρχείο βάσης δεδομένων της Base περιέχει πολλά διαφορετικά είδη αντικειμένων, ομαδοποιημένα κατά κατηγορίες (τύπους) αντικειμένων. Οι κατηγορίες αυτές είναι οι εξής:

- **Πίνακες** Οι πίνακες (tables) είναι τα σημαντικότερα δομικά στοιχεία μιας βάσης δεδομένων. Στην πραγματικότητα, όλα τα δεδομένα που υπάρχουν μέσα στη βάση δεδομένων είναι καταχωρημένα σε κάποιον πίνακα – επειδή μόνο στους πίνακες μπορούν να αποθηκευτούν δεδομένα. Όλα τα υπόλοιπα αντικείμενα της βάσης δεδομένων αναφέρονται, έμμεσα ή άμεσα, στους πίνακες για να πάρουν από αυτούς δεδομένα. Έτσι, είναι θεμελιώδες να καταστρώνει κανείς σωστά τη διαμόρφωση των πινάκων, αφού αποτελούν τον αποθηκευτικό χώρο.

- **Ερωτήματα** Τα ερωτήματα (queries) παίρνουν τα δεδομένα από τους πίνακες και τα αναλύουν, τα συγκρίνουν ή τα παρουσιάζουν με διαφορετικούς τρόπους. Με άλλα λόγια, τα ερωτήματα εκτελούν καθήκοντα «αποθηκάρου» στη βάση δεδομένων- αναλαμβάνουν να εντοπίσουν μέσα στην πληθώρα δεδομένων μόνο τα συγκεκριμένα στοιχεία που μας ενδιαφέρουν.
- **Φόρμες** Οι φόρμες (forms) είναι τα εργαλεία που βοηθούν στην καταχώριση ή τη διόρθωση των δεδομένων. Είναι το ηλεκτρονικό αντίστοιχο των έντυπων φορμών που καλούμαστε να συμπληρώσουμε στις διάφορες υπηρεσίες: παρέχουν κουτιά (πλαίσια) μέσα στα οποία θα πρέπει να καταχωρηθούν, με συγκεκριμένο τρόπο και σε συγκεκριμένο χώρο, τα διάφορα στοιχεία.
- **Αναφορές** Οι αναφορές (reports) είναι το εργαλείο εκτύπωσης των δεδομένων- συλλέγουν τα δεδομένα από τους πίνακες ή από τα ερωτήματα και στη συνέχεια τα παρουσιάζουν με τέτοιο τρόπο ώστε αυτά να μπορούν να τυπωθούν σε ένα φύλλο χαρτί. Με άλλα λόγια, οι εκθέσεις είναι η υπηρεσία «παραδόσεων» της βάσης δεδομένων, αφού αναλαμβάνουν να ετοιμάσουν και να παραδώσουν τα δεδομένα στη μορφή που θέλει ο παραλήπτης.

Όπως έχει ήδη αναφερθεί, οι πίνακες αποτελούν τον σημαντικότερο τύπο αντικειμένων σε μια βάση δεδομένων, καθώς μόνο σε αυτούς αποθηκεύονται τα δεδομένα. Άρα, όσο πιο καλοσχεδιασμένοι είναι οι πίνακες, τόσο ευκολότερα αλλά και σωστότερα γίνεται η εισαγωγή των πληροφοριών.

Ο πίνακας δεν είναι τίποτα περισσότερο από μια συλλογή δεδομένων που εμφανίζονται σε στήλες και γραμμές. Κάθε γραμμή αποτελεί ένα «αυτοτελές» τμήμα δεδομένων, που περιγράφει πλήρως το αντίστοιχο στοιχείο και ονομάζεται στην ορολογία των βάσεων δεδομένων *εγγραφή* (record). Για παράδειγμα, η εγγραφή ενός εργαζόμενου στη βάση δεδομένων των υπαλλήλων μιας εταιρίας περιέχει όλες τις επιμέρους πληροφορίες που αφορούν τον υπάλληλο αυτό, όπως είναι το όνομά του, το επίθετο, η ημερομηνία γέννησής του κ.τ.λ.

Κάθε στήλη περιέχει, υποχρεωτικά, δεδομένα ίδιου τύπου – για παράδειγμα μόνο ημερομηνίες γέννησης, μόνο ονόματα κ.ο.κ. Από την άποψη της εμφάνισης, το τελικό αποτέλεσμα μοιάζει με τους πίνακες δεδομένων των υπολογιστικών φύλλων, με τη διαφορά ότι εδώ οι κανόνες είναι πολύ πιο αυστηροί- για παράδειγμα, αν κάποιος προσπαθήσει να καταχωρίσει κείμενο σε ένα πεδίο ημερομηνιών, η Base απλώς δε θα του το επιτρέψει.

Επιπλέον, παρά το ότι κάθε πίνακας αποτελεί μια αυτόνομη οντότητα, η Base μας επιτρέπει να δημιουργήσουμε συνδέσεις μεταξύ των πινάκων, οι οποίες ονομάζονται *σχέσεις* (relationships). Οι σχέσεις είναι ο μηχανισμός που μας δίνει τη δυνατότητα να ξεφύγουμε από τις επίπεδες βάσεις δεδομένων. Όταν δημιουργούμε σχέσεις, δηλώνουμε στην base ότι οι πληροφορίες ενός πεδίου του πίνακα θα πρέπει να χρησιμοποιηθούν ως μέσο ταύτισης για την εμφάνιση πληροφοριών από έναν άλλο πίνακα. Άλλωστε, μια από τις μεγαλύτερες χρησιμότητες μιας σχεσιακής βάσης δεδομένων (όπως η OpenOffice Base) είναι το ότι μπορούμε να συσχετίζουμε τους διάφορους πίνακες που υπάρχουν σε αυτή, ώστε να αποφεύγεται η άσκοπη επανάληψη πληροφοριών.

Όταν δημιουργείται μια σχέση, δηλώνεται ότι το στοιχείο που καταχωρείται σε ένα συγκεκριμένο πεδίο ενός πίνακα θα πρέπει να χρησιμοποιηθεί ως *μοναδικό αναγνωριστικό* που προσδιορίζει μια συγκεκριμένη εγγραφή από έναν άλλο πίνακα. Η ύπαρξη μοναδικών αναγνωριστικών είναι κάτι πολύ συνηθισμένο στη σημερινή εποχή. Για παράδειγμα, ο αριθμός ταυτότητας, το ΑΦΜ, ο αριθμός διαβατηρίου και άλλοι παρόμοιοι κωδικοί είναι πάντοτε μοναδικοί, ώστε να επιτρέπουν αυτή ακριβώς τη συσχέτιση.

Ο πίνακας από τον οποίο παίρνουμε τις μοναδικές τιμές ονομάζεται *πρωτεύων πίνακας* ενώ ο άλλος πίνακας ονομάζεται *συσχετισμένος*. Ο πιο συνηθισμένος τύπος σχέσης είναι η *μονοσήμαντη* σχέση (σχέση ένα προς πολλά). Σε αυτή τη σχέση, το κάθε στοιχείο του πρωτεύοντος πίνακα μπορεί να συνδεθεί με πολλά στοιχεία του συσχετισμένου πίνακα. Μια άλλη περίπτωση, πολύ λιγότερο συνηθισμένη, είναι η σχέση «ένα προς ένα». Ο λόγος για τον οποίο αυτές οι σχέσεις δεν είναι συνηθισμένες είναι το ότι συνήθως μπορούμε να έχουμε εξίσου καλά τα δεδομένα αυτά σε έναν μόνο πίνακα.

Αφού δημιουργηθούν οι πίνακες, πρέπει τα δεδομένα να καταχωρηθούν σε αυτούς. Όσο περισσότερες πληροφορίες καταχωρούνται στη βάση δεδομένων, τόσο πιο δύσκολος γίνεται ο εντοπισμός και η ανάκτηση συγκεκριμένων πληροφοριών. Στις περισσότερες περιπτώσεις, δε χρειαζόμαστε το πλήρες σύνολο των δεδομένων, αλλά μόνο ένα μικρό υποσύνολο που ικανοποιεί συγκεκριμένα κριτήρια. Αυτό το πρόβλημα καλύπτεται από τα *ερωτήματα* (queries).

Τα ερωτήματα είναι στην πραγματικότητα κώδικας γραμμένος σε μια ειδική γλώσσα προγραμματισμού, που ονομάζεται SQL (Structured Query Language, Δομημένη Γλώσσα Ερωτημάτων). Η Base όμως, κάνει σχεδόν περιττή την εκμάθηση του προγραμματισμού σε γλώσσα SQL, αφού μας επιτρέπει να

δημιουργήσουμε τα ερωτήματά μας χωρίς να γράψουμε ούτε μια γραμμή κώδικα- το μόνο που κάνουμε είναι η επιλογή πεδίων και η καταχώριση κριτηρίων σε ένα πλέγμα σχεδίασης ερωτήματος.

Αν και τα δεδομένα μπορούν να καταχωρούνται ή να εμφανίζονται στην προβολή φύλλου δεδομένων, αναμφίβολα η χρήση μιας φόρμας έχει το πλεονέκτημα ότι δημιουργεί ένα πιο ευχάριστο και οικείο περιβάλλον εργασίας με τα δεδομένα, και επιπλέον επιτρέπει την εστίαση της προσοχής σε μια συγκεκριμένη εγγραφή του πίνακα-αντί σε ολόκληρο τον πίνακα.

Η φόρμα είναι στην πραγματικότητα ένα αντικείμενο της βάσης δεδομένων που παίρνει απλώς τα δεδομένα από κάποιον πίνακα ή ερώτημα και τα εμφανίζει στην οθόνη με έναν πιο βολικό τρόπο. Έτσι, η σημαντικότερη χρησιμότητα των φορμών είναι το να διευκολύνουν αυτούς που θα αναλάβουν την καταχώριση των δεδομένων, και γι'αυτό είναι κυρίως προσανατολισμένες στην εμφάνιση στην οθόνη.

Οι αναφορές είναι το κατεξοχήν εργαλείο εκτύπωσης πληροφοριών από τη βάση δεδομένων – δηλαδή ο τελικός τους στόχος είναι η παραγωγή εντύπων και όχι τόσο η εμφάνιση στην οθόνη. Μεταξύ άλλων, οι εκθέσεις μας επιτρέπουν :

- Να ομαδοποιούμε τις πληροφορίες σε κατηγορίες και υποκατηγορίες, και να αποφεύγουμε έτσι την επαναλαμβανόμενη εμφάνιση των ίδιων πληροφοριών σε όλες τις εγγραφές.
- Να καθορίζουμε διαφορετική μορφοποίηση για κάθε πεδίο.
- Να κάνουμε συγκεντρωτικούς υπολογισμούς ανά κατηγορία στοιχείων.
- Να δημιουργούμε εντυπωσιακές κεφαλίδες και υποσέλιδα, που θα διακοσμούν, αλλά και θα κάνουν πιο κατανοητή την έκθεσή μας.

ΚΕΦΑΛΑΙΟ 3

Οδηγός χρήσης της Βάσης Δεδομένων OpenOffice

Άνοιγμα της Εφαρμογής και Δημιουργία Βάσης Δεδομένων

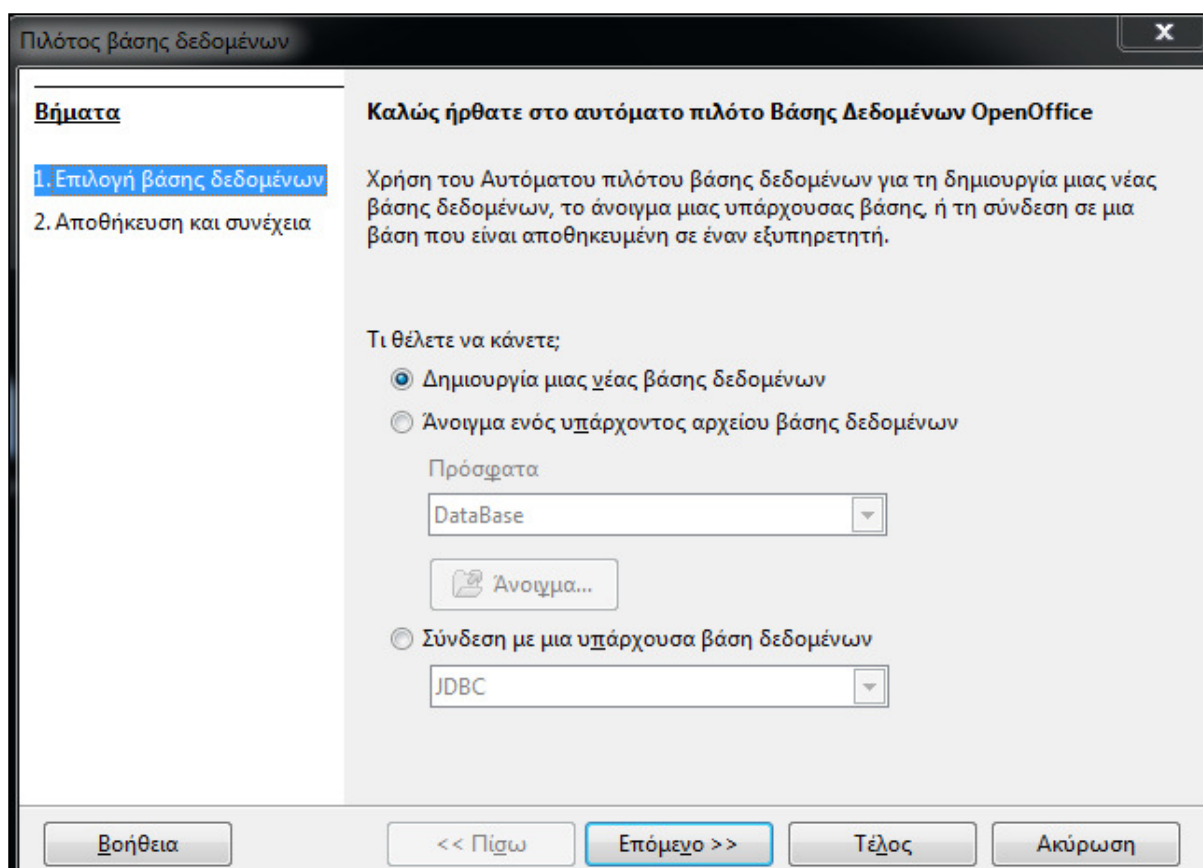
Δύο βασικοί τρόποι με τους οποίους μπορούμε να εκκινήσουμε το πρόγραμμα OpenOffice Base είναι οι παρακάτω:

- ✓ Πατάμε στο κουμπί **«Έναρξη»** της γραμμής εργασιών, κατόπιν **«Όλα τα προγράμματα»** και στη συνέχεια στο φάκελο **«OpenOffice»**. Τέλος, κλοκάρουμε την εντολή **«Βάση δεδομένων»**.
- ✓ Αν έχουμε μια συντόμευση για την Base στην επιφάνεια εργασίας, διπλοκλικάρουμε στο εικονίδιο της συντόμευσης.

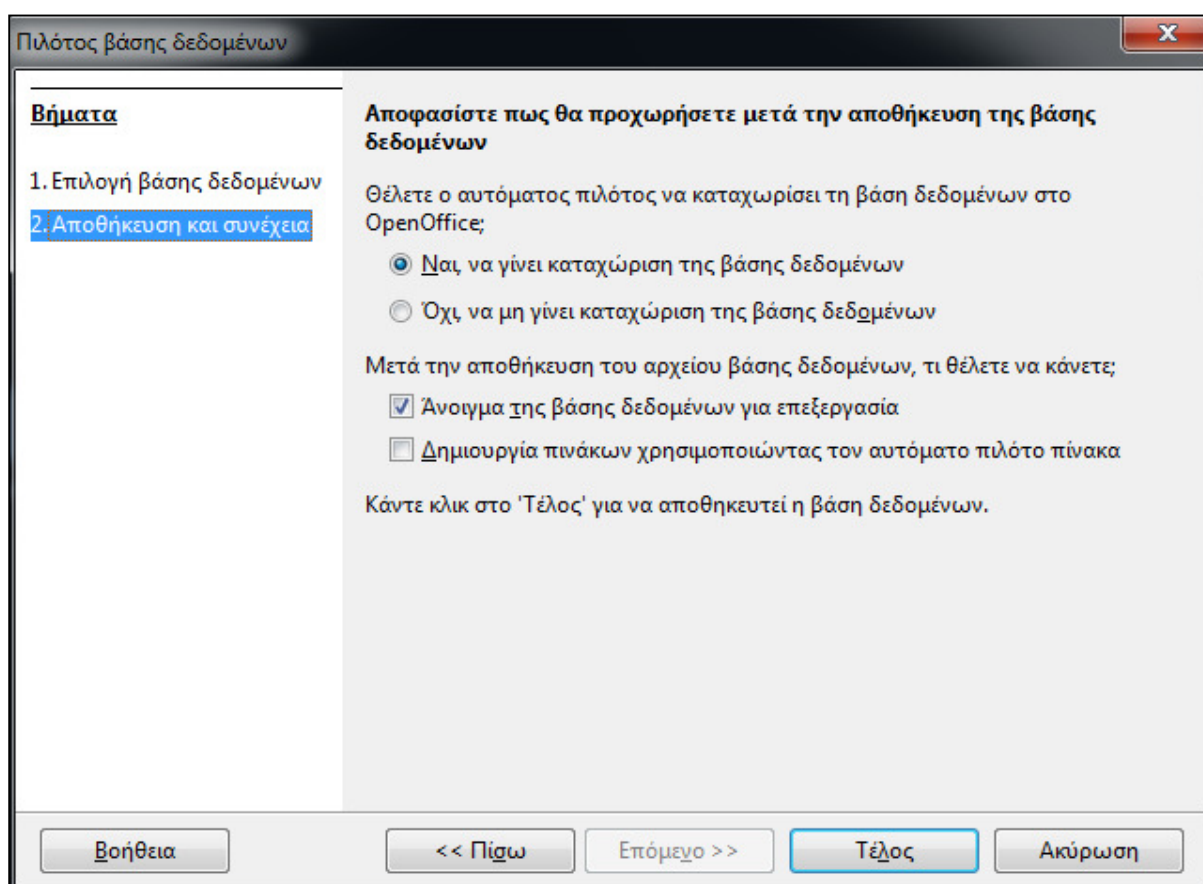
Ανοίγοντας το πρόγραμμα, δίνετε η δυνατότητα επιλογής να δημιουργηθεί μια νέα βάση δεδομένων ή να ανοιχθεί μια ήδη υπάρχουσα, ή να γίνει σύνδεση με υπάρχουσα βάση δεδομένων. Όσον αφορά το άνοιγμα μιας υπάρχουσας βάσης δεδομένων, μπορούμε να επιλέξουμε κάποια που χρησιμοποιήθηκε πρόσφατα ή να ανοίξουμε κάποια αποθηκευμένα στον υπολογιστή μας. Η σύνδεση με ήδη υπάρχουσα βάση δεδομένων, απευθύνεται σε πιο έμπειρους χρήστες και δε θα αναλυθεί εκτενώς στα πλαίσια της παρούσας πτυχιακής εργασίας.

Με δεδομένο ότι στόχος μας είναι η δημιουργία μιας νέας βάσης δεδομένων, επιλέγουμε «Δημιουργία μιας νέας βάσης δεδομένων», όπως φαίνεται και στην παρακάτω φωτογραφία (Εικόνα 3), πατάμε στο εικονίδιο «Επόμενο» και συνεχίζουμε. [5]

Στο δεύτερο παράθυρο, δίνεται η επιλογή να καταχωρήσουμε τη βάση δεδομένων στο OpenOffice.org και στη συνέχεια επιλέγουμε αν, μετά την αποθήκευση, μπορούμε να επεξεργαστούμε τη βάση δεδομένων ανοίγοντας τη (Εικόνα 4).



Εικόνα 3



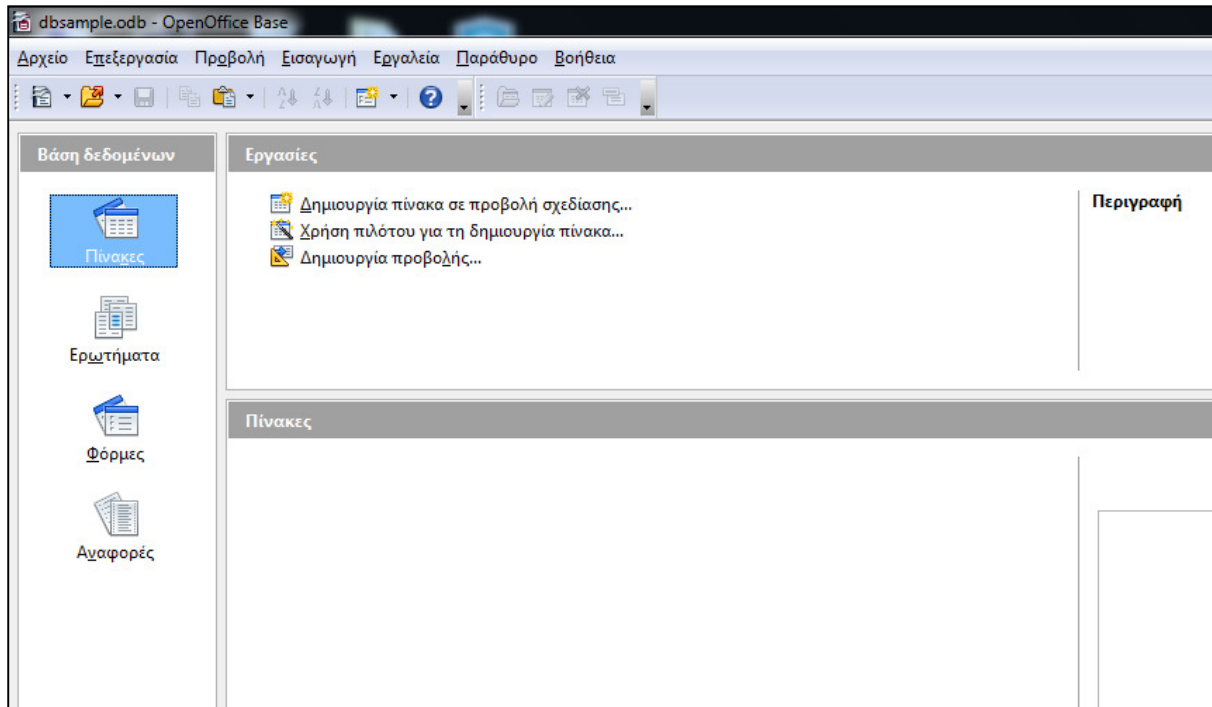
Εικόνα 4

Αφού γίνει η επιλογή, κάνουμε κλικ στο «Τέλος» και η βάση αποθηκεύεται, ή πατώντας το «Πίσω» επιστρέφουμε στην προηγούμενη καρτέλα. Αν πατήσουμε «Τέλος», επιλέγουμε πού θα αποθηκευτεί η βάση δεδομένων (π.χ. σε ποιο φάκελο) και με τι ονομασία, και με το «save» προχωράμε στο επόμενο βήμα. Σε αντίθεση με άλλες εφαρμογές που δουλεύουμε πρώτα και αποθηκεύουμε στο τέλος αν θέλουμε, εδώ δεν μπορούμε να ξεκινήσουμε αν δεν αποθηκεύσουμε τη βάση δεδομένων μας.

Στην κορυφή της οθόνης της OpenOffice Base υπάρχει μια γραμμή μενού, με επιλογές και ρυθμίσεις (αρχείο, επεξεργασία, προβολή, εισαγωγή, εργαλεία, παράθυρο, βοήθεια). Στο δεξιό άκρο της γραμμής αυτής βρίσκεται το κουμπί του κλεισίματος, το οποίο κλείνει το ενεργό παράθυρο.

Ακριβώς από κάτω ακολουθεί μια μικρή γραμμή εργαλείων με τα οποία μπορούν να εκτελεστούν οι σημαντικότερες εργασίες χειρισμού των αντικειμένων ή αλλαγής του τρόπου εμφάνισής τους στην οθόνη.

Στη συνέχεια, προχωρώντας στο κυρίως παράθυρο, δίδεται η επιλογή για τη δημιουργία διαφορετικών χαρακτηριστικών των δεδομένων. Στην αριστερή πλευρά του παραθύρου έχουμε τα εικονίδια «Πίνακες» (Tables), «Ερωτήματα» (Queries), «Φόρμες» (Forms) και «Αναφορές» (Reports). [5]



Εικόνα 5: Επιλογή για τη δημιουργία διαφορετικών χαρακτηριστικών δεδομένων της βάσης

Πίνακες

Πατώντας στο κουμπί «Πίνακες» ξεκινάμε τη δημιουργία πινάκων. Μπορούμε να επιλέξουμε αν θέλουμε να δημιουργήσουμε έναν πίνακα σε **προβολή σχεδίασης**, **προβολή φύλλου δεδομένων** ή να χρησιμοποιήσουμε **οδηγό πινάκων**. Αν θέλουμε να έχουμε τον πλήρη έλεγχο, από την αρχή μέχρι το τέλος της διαδικασίας, επιλέγουμε την προβολή σχεδίασης, για να δημιουργήσουμε μόνοι μας τον πίνακα. Στο σημείο αυτό είναι σημαντικό να αναφερθεί ότι, ακόμα και αν αρχικά χρησιμοποιηθεί κάποια από τις υπόλοιπες επιλογές, πιθανότατα θα χρειαστεί και πάλι να καταφύγουμε στην προβολή σχεδίασης για τις απαραίτητες μικρορυθμίσεις.

Για τη δημιουργία ενός πίνακα σε προβολή φύλλου δεδομένων, ακολουθούμε τα παρακάτω βήματα:

- ✓ Στο πλαίσιο διαλόγου **Πίνακες**, διαλέγουμε την επιλογή **Προβολή φύλλου δεδομένων** για να εμφανιστεί ένα ενδεικτικό φύλλο δεδομένων.
- ✓ Δίνεται η δυνατότητα να προσθέσουμε πεδίο και να πληκτρολογήσουμε το όνομά του.
- ✓ Η Base δίνει στα πεδία ενδεικτικά ονόματα. Συμπληρώνουμε τις εγγραφές και σε περίπτωση που δεν χρειαζόμαστε κάποιο πεδίο, το αφήνουμε κενό.
- ✓ Με την επεξεργασία μας δίνεται η δυνατότητα αναίρεσης, αλλά και αντιγραφής, αποκοπής, επικόλλησης κτλ., ενώ πατώντας στο «Αρχείο» (πάνω και δεξιά στο αντίστοιχο παράθυρο) μπορούμε να το αποθηκεύσουμε ή να το καταργήσουμε.

Ο **αυτόματος πιλότος** για πίνακες μας βοηθά να δημιουργήσουμε έναν πίνακα για τη βάση δεδομένων μας, χρησιμοποιώντας κατηγορία πίνακα, αλλά και κάποιο δείγμα πίνακα για την επιλογή των πεδίων που θέλουμε να περιέχει. Μπορούμε να συμπεριλάβουμε πεδία από περισσότερα από ένα δείγματα πινάκων, αν το επιθυμούμε.

Αφού επιλέξουμε τα πεδία που θέλουμε, τα προσθέτουμε στα επιλεγμένα πατώντας στο βελάκι (>). Στη συνέχεια, επιλέγοντας είτε το βήμα 2. **Ορισμός τύπων και μορφοποίησης** είτε πατώντας την επιλογή **Επόμενο** πληκτρολογούμε το όνομα του πεδίου και καθορίζουμε το είδος των πληροφοριών που θα αποθηκεύονται στο συγκεκριμένο πεδίο του πίνακα. Με τον ίδιο τρόπο προχωρούμε στο στάδιο 3, όπου ορίζουμε το πρωτεύον κλειδί ενώ στο βήμα 4 ολοκληρώνεται η δημιουργία του πίνακα. Πιο συγκεκριμένα, δίνουμε όνομα στον πίνακα και αμέσως μετά επιλέγουμε

αν θέλουμε να συνεχίσουμε εισάγωντας άμεσα τα δεδομένα. Πατώντας πάνω στο Τέλος βλέπουμε τον πίνακά μας ολοκληρωμένο.

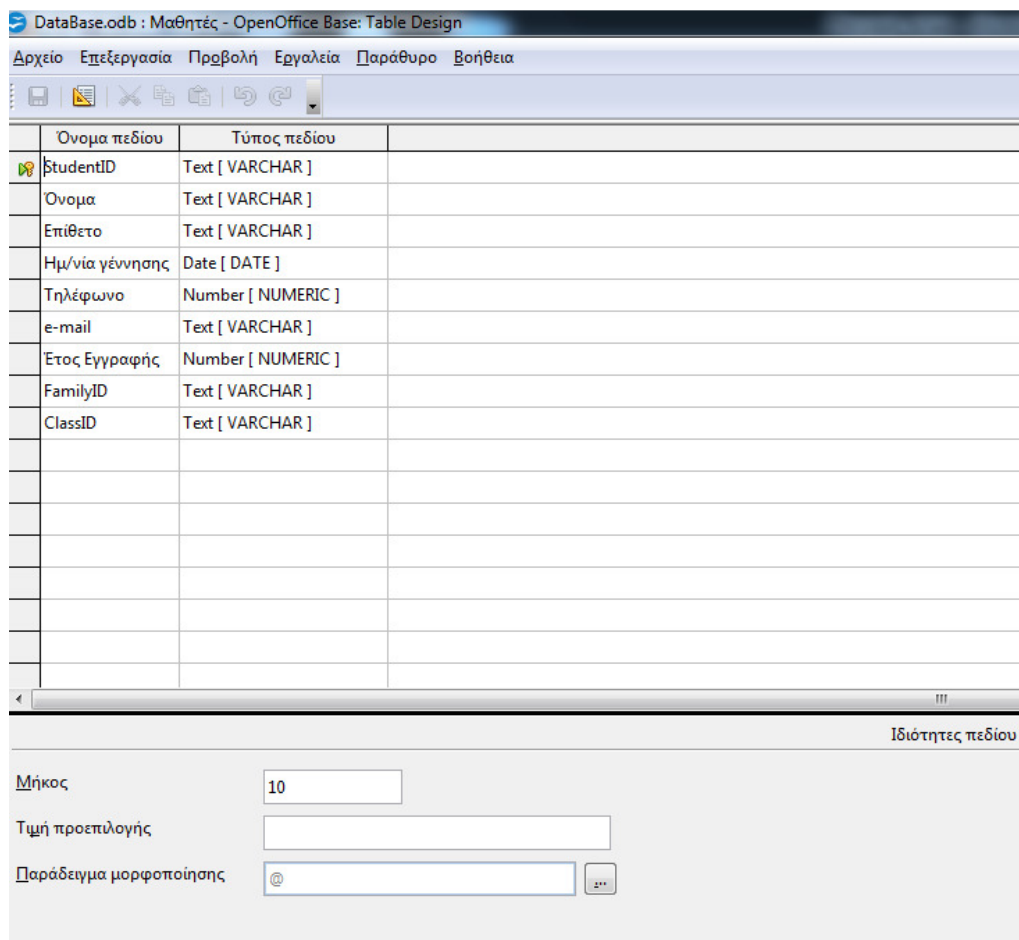
Για τη δημιουργία ενός πίνακα σε προβολή Σχεδίασης ακολουθούμε τα παρακάτω βήματα:

- ✓ Στο πλαίσιο διαλόγου **Πίνακες** επιλέγουμε τη *Δημιουργία πίνακα σε προβολή σχεδίασης*. Το παράθυρο της προβολής Σχεδίασης που εμφανίζεται περιέχει ένα πλέγμα γραμμών και στηλών, όπου πρέπει να καθορίσουμε το όνομα του πεδίου και τον τύπο των δεδομένων του πεδίου. Επιπλέον, μπορούμε να καθορίσουμε και μια προαιρετική περιγραφή του πεδίου, η οποία θα εξηγήσει τη σκοπιμότητά του στη βάση δεδομένων.
- ✓ Πατάμε στο πρώτο κενό κελί της στήλης **Όνομα πεδίου** και πληκτρολογούμε ένα όνομα για το πεδίο. Παρότι παρέχεται ελευθερία από την OpenOffice Base ως προς τα ονόματα των πεδίων, καλό είναι να χρησιμοποιούνται σύντομα ονόματα πεδίων.
- ✓ Αφού πληκτρολογηθεί το όνομα του πεδίου, περνάμε στη διπλανή στήλη *Τύπος πεδίου* και καθορίζουμε το είδος των πληροφοριών που θα αποθηκεύονται στο συγκεκριμένο πεδίο του πίνακα. Οι πιο συνηθισμένοι τύποι δεδομένων είναι οι ακόλουθοι:

Τύπος	Χρήση
Κείμενο	Για οποιαδήποτε μορφής καταχωρίσεις κειμένου. Σε αυτόν τον τύπο δεδομένων συνήθως χρειάζεται να προσδιορίσουμε μόνο πόσο είναι το μέγιστο επιτρεπτό μήκος του κειμένου.
Αριθμός	Για αριθμητικές πληροφορίες. Σε αυτόν τον τύπο δεδομένων μπορούμε να καθορίσουμε αν ο αριθμός θα είναι ακέραιος ή πραγματικός αριθμός (δηλαδή αριθμός με δεκαδικά ψηφία).
Ναι/Όχι	Λογικά πεδία, στα οποία μπορεί να αποθηκευτεί μια από τις δύο διαθέσιμες τιμές ναι/όχι.

- ✓ Προαιρετικά, πληκτρολογούμε στο διπλανό κελί (*Περιγραφή*) ένα περιγραφικό σχόλιο σχετικά με τη χρησιμότητα του πεδίου στη βάση δεδομένων.

- ✓ Τέλος, στο κάτω μέρος του πλαισίου διαλόγου υπάρχει πίνακας με τις ιδιότητες του πεδίου και επιλέγουμε τις ιδιότητες που θέλουμε να ισχύουν για το κάθε πεδίο. Οι διαθέσιμες ιδιότητες ποικίλλουν ανάλογα με τον τύπο δεδομένων του πεδίου. Για να καθορίσουμε την τιμή μιας ιδιότητας, πατάμε απλώς στο κενό πλαίσιο που υπάρχει στα δεξιά από το όνομά της. Πολλές ιδιότητες διαθέτουν πτυσσόμενους καταλόγους από τους οποίους μπορούμε να διαλέξουμε μια ρύθμιση και εμφανίζονται πατώντας το βέλος του καταλόγου, που βρίσκεται στα δεξιά του κενού πλαισίου. Σε άλλες περιπτώσεις, όταν πατάμε σε ένα πλαίσιο ιδιότητας, εμφανίζεται στα δεξιά του πλαισίου ένα κουτί με αποσιωπητικά [...]. Αυτό το κουμπί μας επιτρέπει να ξεκινήσουμε ένα κατάλληλο βοηθητικό εργαλείο, το οποίο θα μας καθοδηγήσει στα βήματα για τη ρύθμιση της τιμής αυτής της ιδιότητας.



Εικόνα 6 : Πεδία του πίνακα «Μαθητές»

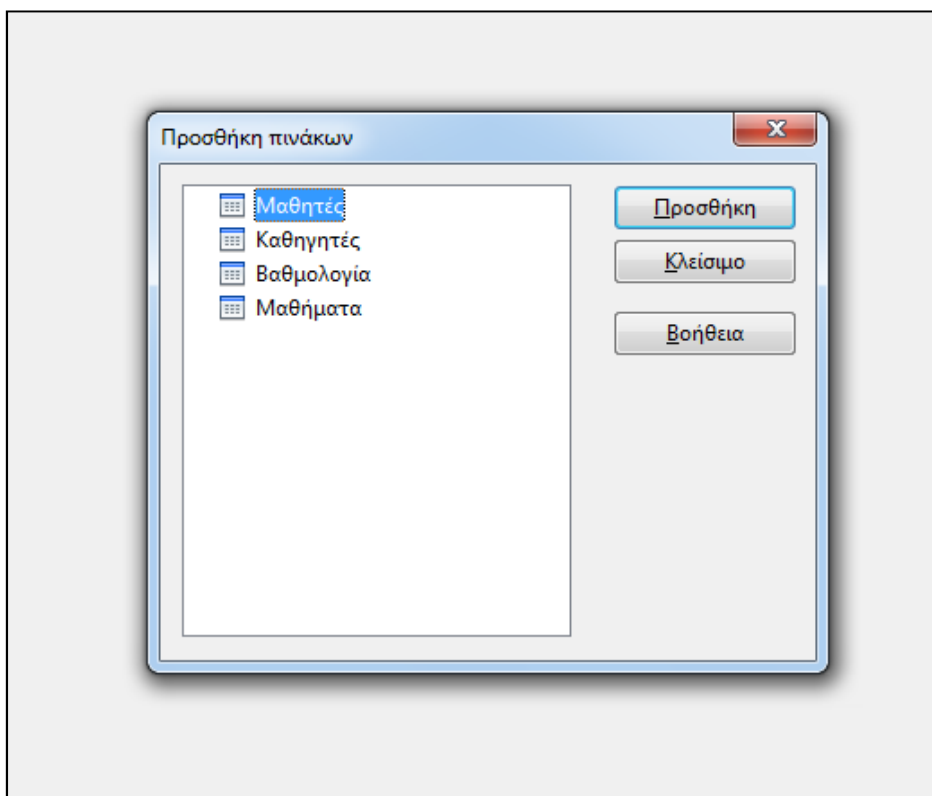
Οι σημαντικότερες ιδιότητες των πεδίων είναι οι ακόλουθες :

- **Μήκος** Στα πεδία κειμένου, η ιδιότητα αυτή έχει προφανή λειτουργία – απλώς καθορίζουμε πόσοι θα είναι οι μέγιστοι επιτρεπόμενοι χαρακτήρες για αυτό το πεδίο.
- **Απαιτείται εισαγωγή** Με αυτή την ιδιότητα μπορούμε να υποχρεώσουμε την Base να μη δέχεται την καταχώριση μιας εγγραφής χωρίς να υπάρχει τιμή σε ένα συγκεκριμένο πεδίο. Για παράδειγμα, να μη δέχεται της καταχώριση νέου πελάτη-μαθητή όταν είναι κενό το πεδίου του τηλεφώνου του.
- **Τιμή προεπιλογής** Όταν γνωρίζουμε από πριν ότι στην πλειονότητα των περιπτώσεων θα καταχωρούμε σε ένα πεδίο μια συγκεκριμένη τιμή, μπορούμε να την ορίσουμε ως τιμή προεπιλογής, ώστε να αποφεύγουμε τον κόπο πληκτρολόγησής της. Για παράδειγμα, για μια βάση δεδομένων με τα ονόματα των μαθητών-πελατών, μπορεί να καταχωρηθεί ως τιμή προεπιλογής στο πεδίο **Πόλη** η πόλη όπου κατοικούμε, αφού πιθανότατα εκεί κατοικούν και οι περισσότεροι μαθητές μας.
- **Μορφοποίηση πεδίου** Η ιδιότητα αυτή ρυθμίζει τον τρόπο εμφάνισης των δεδομένων.
 - ✓ Στη συνέχεια, καθορίζουμε το πρωτεύον κλειδί (όταν υπάρχει). Επιλέγουμε το πεδίο που θέλουμε να ορίσουμε ως πρωτεύον κλειδί, και κάνουμε δεξί κλικ πάνω στο πράσινο βελάκι που βρίσκεται αριστερά από το πεδίο. Ανοίγει πεδίο με επιλογές, και πατάμε στο **Πρωτεύον κλειδί**. Στα αριστερά του ονόματος του πεδίου, θα εμφανιστεί το εικονίδιο του κλειδιού.
 - ✓ Για την αποθήκευση του πίνακα, μπορούμε είτε να πατήσουμε στη δισκέτα αποθήκευσης στη γραμμή εργαλείων, είτε να πατήσουμε στο **Αρχείο** (βρίσκεται αριστερά και πάνω στο παράθυρο) και να επιλέξουμε την αποθήκευση. Όποιον τρόπο και να επιλέξουμε, θα μας ζητηθεί να δώσουμε όνομα στον πίνακα. [5]

Σχέσεις

Αφού ολοκληρωθεί η δημιουργία καθενός από τους πίνακες που χρειαζόμαστε, θα χρειαστεί πιθανότατα να δημιουργηθούν **σχέσεις**, ώστε να καθοριστεί ο τρόπος σύνδεσης του κάθε πίνακα με τους υπόλοιπους. Άλλωστε, μια από τις μεγαλύτερες χρησιμότητες μιας σχεσιακής βάσης δεδομένων (όπως η OpenOffice Base) είναι το ότι μπορούμε να συσχετίζουμε τους διάφορους πίνακες που υπάρχουν σε αυτή, ώστε να αποφεύγεται η άσκοπη επανάληψη πληροφοριών.

Για να ανοίξουμε και να δουλέψουμε το παράθυρο **Σχέσεις**, πατάμε **Εργαλεία** στη Γραμμή Μενού και στη συνέχεια στην αντίστοιχη γραμμή **Σχέσεις** που δίδεται ως επιλογή. Αυτόματα ανοίγει νέο παράθυρο – με το όνομα Προσθήκη πινάκων. Στο πλαίσιο αυτό, παρατίθενται όλα τα ονόματα των πινάκων της βάσης δεδομένων. Επιλέγουμε έναν-έναν τους πίνακες που μας ενδιαφέρει να συνδεθούν μεταξύ τους με κάποιο τρόπο και κλικάρωντας στην **Προσθήκη** ενημερώνουμε το παράθυρο των σχέσεων. Όταν τελειώσουμε με την προσθήκη πινάκων, πατάμε το κουμπί **Κλείσιμο** του πλαισίου διαλόγου. Για να δημιουργηθεί μια νέα σχέση, σύρουμε το πεδίο από τον πρωτεύοντα πίνακα (δηλαδή τον πίνακα από τον οποίο παίρνουμε τις μοναδικές τιμές) πάνω στο αντίστοιχο πεδίο του συσχετισμένου πίνακα.



Εικόνα 7 : Προσθήκη των επιθυμητών πινάκων για τη δημιουργία σχέσεων

Ο πιο συνηθισμένος τύπος σχέσης είναι η μονοσήμαντη σχέση (σχέση ένα προς πολλά). Σε αυτή τη σχέση, το κάθε στοιχείο του πρωτεύοντος πίνακα (για παράδειγμα ο κάθε μαθητής) μπορεί να συνδεθεί με πολλά στοιχεία του συσχετισμένου πίνακα (για παράδειγμα, με πολλά τμήματα αφού μπορεί να παρακολουθεί μαθήματα διαφορετικών ξένων γλωσσών). Μια άλλη περίπτωση, πολύ πιο ασυνήθιστη, είναι η σχέση ένα προς ένα (αμφιμονοσήμαντη σχέση). Ο λόγος για τον οποίο αυτού του είδους οι σχέσεις δεν είναι συνηθισμένες είναι το ότι συνήθως μπορούμε εξίσου καλά να έχουμε τα δεδομένα σε ένα μόνο πίνακα. Για παράδειγμα, μπορούν όλα τα στοιχεία για τους υπαλλήλους να διατηρούνται σε έναν μόνο πίνακα. Αν όμως, για λόγους π.χ. προστασίας προσωπικών δεδομένων, επιθυμούμε να κρατήσουμε ξεχωριστά κάποιες πληροφορίες (όπως το μισθό του καθενός), μπορούμε να δημιουργήσουμε δυο πίνακες με συσχέτιση ένα προς ένα.

Για να διαγραφεί μια σχέση ή για να γίνει επεξεργασία της, κάνουμε δεξί κλικ πάνω στην αντίστοιχη γραμμή συσχέτισης και επιλέγουμε delete ή edit αντίστοιχα. Όταν μια σχέση έχει δημιουργηθεί, δε μπορεί να διαγραφεί κανένα από τα πεδία που συμμετέχουν στη σχέση αυτή. Για να μπορέσουν να διαγραφούν, θα πρέπει πρώτα να διαγραφεί η ίδια η σχέση, με τη μέθοδο που ήδη αναφέρθηκε.

Όπως έχει ήδη αναφερθεί, με την επιλογή edit μπορούμε να δούμε τις ιδιότητες μιας σχέσης, καθώς και να την επεξεργαστούμε. Αφού γίνουν οι αλλαγές που θέλουμε, πατάμε OK για να ενεργοποιηθούν. [5]

Ερωτήματα

Τα ερωτήματα αποτελούν ένα ξεχωριστό είδος αντικειμένων της βάσης δεδομένων. Για να δούμε τα διαθέσιμα ή για να δημιουργήσουμε νέα ερωτήματα, πρέπει πρώτα να πατήσουμε το κουμπί **Ερωτήματα**, στο παράθυρο της βάσης δεδομένων. Με τα ερωτήματα μπορούμε να κάνουμε πολύ περισσότερα πράγματα από την απλή επιλογή συγκεκριμένων τμημάτων δεδομένων. Μπορούμε να δημιουργήσουμε ερωτήματα που ενημερώνουν ή διαγράφουν εγγραφές, ή ακόμα και δημιουργούν νέους πίνακες. Πάντως, το πιο συνηθισμένο είδος ερωτημάτων είναι τα ερωτήματα επιλογής εγγραφών, και σε αυτά θα δώσουμε το μεγαλύτερο βάρος- άλλωστε ο τρόπος δημιουργίας και των υπόλοιπων τύπων ερωτημάτων είναι παρόμοιος.

Για να δημιουργήσουμε ένα νέο ερώτημα, φροντίζουμε να ενεργοποιήσουμε το «κουμπί» **Ερωτήματα**. Η OpenOffice Base παρέχει τρεις επιλογές δημιουργίας ερωτήματος, τη δημιουργία ερωτήματος σε προβολή SQL, τη χρήση πιλότου για τη δημιουργία ερωτήματος και τη Δημιουργία ερωτήματος σε προβολή σχεδίασης.

A) Εισαγωγή στην SQL

Η SQL είναι μια γλώσσα υπολογιστή η οποία μοιάζει περισσότερο με τα Αγγλικά, αλλά που αναγνωρίζουν προγράμματα βάσεων δεδομένων. Κάθε ερώτημα που εκτελείτε χρησιμοποιεί την SQL στο παρασκήνιο. Η αρχική έκδοση της γλώσσας ονομάστηκε SEQUEL (Structured English Query Language) και σχεδιάστηκε από ερευνητική ομάδα της IBM το 1974. Η εταιρεία Oracle χρησιμοποίησε για πρώτη φορά τη γλώσσα σε εμπορικό ΣΔΒΔ το 1979. [3]

Η γλώσσα SQL έχει σημαντικά **πλεονεκτήματα**, με βασικότερα τα εξής:

- Υποστηρίζεται από κάθε σχεσιακό σύστημα, οπότε η διατύπωση των ερωτημάτων είναι ανεξάρτητη του ΣΔΒΔ
- Πολλές γλώσσες προγραμματισμού έχουν επεκταθεί ώστε να υποστηρίζουν διατύπωση ερωτημάτων σε SQL, και
- Έχει απλή σύνταξη και αποδεσμεύει το χρήστη από λεπτομέρειες υλοποίησης.

Η SQL στηρίζεται στο σχεσιακό λογισμό πλειάδων και τη σχεσιακή άλγεβρα. Υποδιαιρείται σε δυο ξεχωριστές υπογλώσσες, τη γλώσσα ορισμού δεδομένων (DDL:Data Definition Language) και τη γλώσσα χειρισμού δεδομένων (DML:Data Manipulation Language).

Είναι δηλωτική γλώσσα υψηλού επιπέδου, δηλαδή ορίζουμε τι θέλουμε να γίνει και όχι το πώς. Χρησιμοποιεί όρους όπως «πίνακας», «γραμμή», «στήλη», οι οποίοι αντιστοιχούν στις έννοιες «σχέση», «πλειάδα» και «χαρακτηριστικό» αντίστοιχα.

Μια χαρακτηριστική ερώτηση σε SQL έχει την εξής μορφή:

select A_1, A_2, \dots, A_n	(ονόματα γνωρισμάτων)
from R_1, R_2, \dots, R_m	(ονόματα σχέσεων)
where P	(συνθήκη)

Η εντολή **select** αντιστοιχεί στην πράξη της προβολής της σχεσιακής άλγεβρας, δηλαδή ποια γνωρίσματα θέλουμε να υπάρχουν στο αποτέλεσμα της ερώτησης. Η εντολή **from** αντιστοιχεί στην πράξη του καρτεσιανού γινομένου της σχεσιακής άλγεβρας, δηλαδή απαντά στο ποιες σχέσεις θα χρησιμοποιηθούν για τον υπολογισμό του αποτελέσματος. Το **where** αντιστοιχεί στη συνθήκη της επιλογής στη σχεσιακή άλγεβρα. Πιο συγκεκριμένα το κατηγορήμα P έχει γνωρίσματα των σχέσεων που εμφανίζονται στο **from**.

Παρακάτω αναφέρονται συνοπτικά κάποιες βασικές εντολές-συναρτήσεις της SQL. Αρχικά, έχει πέντε built-in συναθροιστικές συναρτήσεις:

- Μέσος όρος: $avg(A)$ (μόνο σε αριθμούς) A γνώρισμα
- Ελάχιστο : $min(A)$
- Μέγιστο : $max(A)$
- Άθροισμα : $sum(A)$ (μόνο σε αριθμούς)
- Πλήθος : $count(A)$

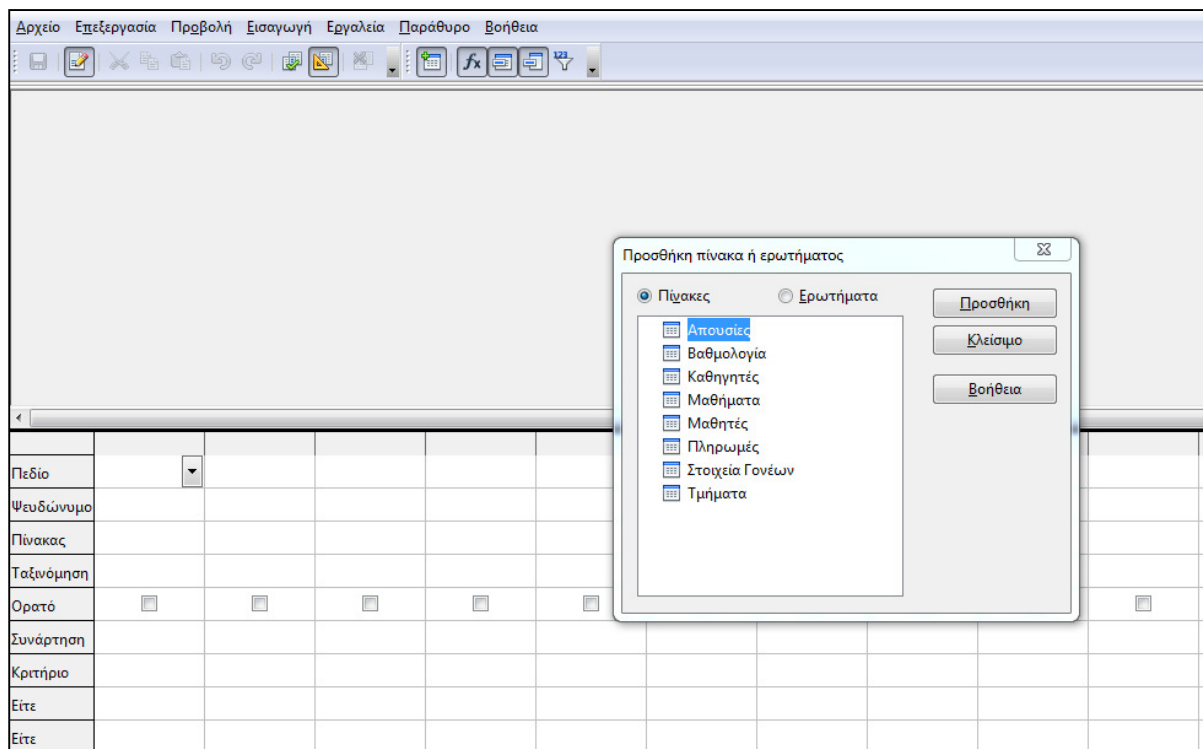
Αν θέλουμε να απαλείψουμε διπλές εμφανίσεις χρησιμοποιούμε τη λέξη-κλειδί **distinct** στην αντίστοιχη έκφραση. Οι συναρτήσεις μπορούν να εφαρμοστούν όχι μόνο σε ένα σύνολο από πλειάδες, αλλά και σε ομάδες από σύνολα πλειάδων. Οι ομάδες προσδιορίζονται χρησιμοποιώντας το **group by**. Τέλος, μια συνθήκη μπορεί να εφαρμοστεί σε μια συγκεκριμένη ομάδα από πλειάδες χρησιμοποιώντας το **having**. Η συνθήκη του **having** εφαρμόζεται αφού σχηματιστούν οι ομάδες και υπολογιστούν οι συναθροιστικές συναρτήσεις.

Τύποι πεδίου ορισμού στην SQL

- `char(n)`. Σταθερού μήκους συμβολοσειρά, με μήκος n (το n καθορίζεται από το χρήστη).
- `varchar(n)`. Μεταβλητού μήκους συμβολοσειρά, με μέγιστο μήκος n (το n καθορίζεται από το χρήστη).
- `int`. Ακέραιος (το εύρος του εξαρτάται από τη μηχανή).
- `smallint`. Μικρός ακέραιος (υποσύνολο των ακέραιων, το εύρος του εξαρτάται από τη μηχανή).
- `numeric(p,d)`. Αριθμός σταθερής υποδιαστολής, με ακρίβεια p ψηφίων (το p καθορίζεται από το χρήστη), με d ψηφία στα δεξιά της υποδιαστολής.
- `real, double precision`. Αριθμός κινητής υποδιαστολής απλής και διπλής ακρίβειας, αντίστοιχα (η ακρίβειά του εξαρτάται από τη μηχανή).
- `float(n)`. Αριθμός κινητής υποδιαστολής. Ο χρήστης καθορίζει την ακρίβεια τουλάχιστον n ψηφίων.

B) Παράθυρο Προβολής Σχεδίασης Ερωτημάτων

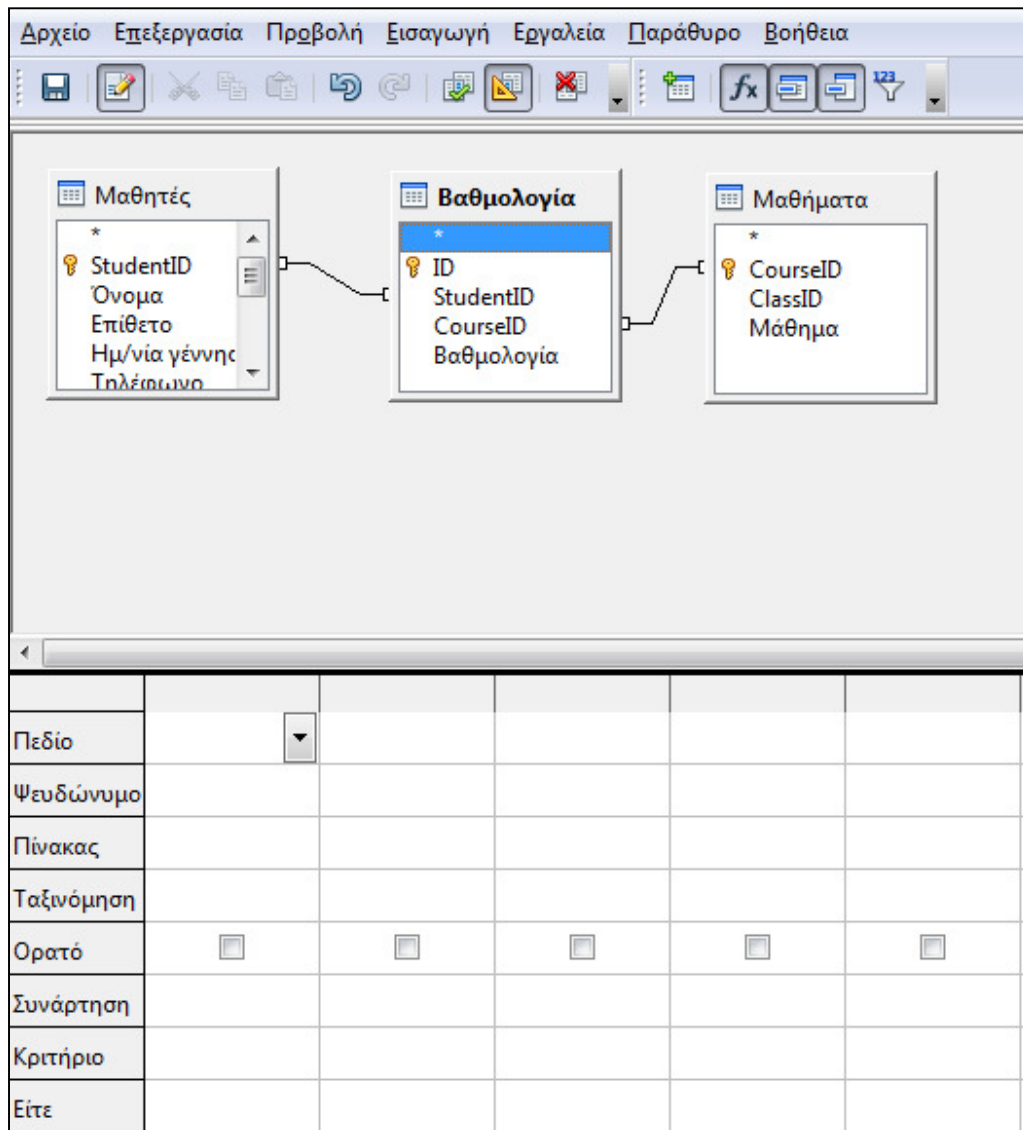
Για να δημιουργήσουμε ένα ερώτημα σε προβολή σχεδίασης, επιλέγουμε τη **Δημιουργία ερωτήματος σε προβολή σχεδίασης**. Τότε θα ανοίξει το πλαίσιο διαλόγου *Προσθήκη πίνακα ή ερωτήματος*.



Εικόνα 8 : Προσθήκη των επιθυμητών πινάκων για τη δημιουργία ενός ερωτήματος

Σε αυτό το πλαίσιο διαλόγου πρέπει να καθοριστεί ποιοι πίνακες περιέχουν τα πεδία τα οποία θέλουμε να χρησιμοποιήσουμε στο ερώτημά μας. Στα αποτελέσματα ή τα κριτήρια του ερωτήματος μπορούμε να χρησιμοποιήσουμε πεδία από έναν ή περισσότερους πίνακες. Επιλέγουμε καθέναν από τους πίνακες που θέλουμε να συμπεριλάβουμε στο ερώτημά μας και πατάμε το κουμπί **Προσθήκη**. Μπορούμε επίσης να προσθέσουμε και πεδία από άλλα υπάρχοντα ερωτήματα. Για να το κάνουμε αυτό, πατάμε στην καρτέλα **Ερωτήματα** του πλαισίου διαλόγου *Προσθήκη πίνακα ή ερωτήματος* και προσθέτουμε όποια ερωτήματα χρειαζόμαστε. Αφού προσθέσουμε ό,τι χρειαζόμαστε, πατάμε το **Κλείσιμο**.

Αμέσως εμφανίζεται το πλέγμα ερωτήματος. Οι πίνακες που επιλέχθηκαν εμφανίζονται στο πάνω μέρος του ερωτήματος, ενώ για κάθε έναν πίνακα εμφανίζεται και η αντίστοιχη λίστα πεδίων, στο πάνω μέρος του παραθύρου. Επιπλέον, εμφανίζονται και οι γραμμές σύνδεσης που δηλώνουν τις σχέσεις που υπάρχουν μεταξύ των πινάκων. Προς το παρόν όμως, το κάτω μέρος του πλέγματος είναι κενό. [5]



Εικόνα 9 : Πλέγμα ερωτήματος

Πατώντας στο πρώτο κενό κελί της γραμμής **Πεδίο** επιλέγουμε ένα όνομα πεδίου από τον κατάλογο. Το κελί της γραμμής **Πίνακας** η OpenOffice Base θα το συμπληρώσει αυτόματα με το όνομα του πίνακα (ή του ερωτήματος) από το οποίο προέρχεται το πεδίο. Επίσης, θα ενεργοποιήσει το πλαίσιο ελέγχου της γραμμής **Ορατό** ώστε τα περιεχόμενα του πεδίου αυτού να συμπεριλαμβάνονται στα αποτελέσματα του ερωτήματος. Καθορίζουμε κριτήρια για να επιλέξουμε ποια ακριβώς δεδομένα χρειαζόμαστε και πατάμε τη δισκέτα για **Αποθήκευση** του ερωτήματος στη βάση δεδομένων.

Μπορούμε, φυσικά, να δημιουργήσουμε ένα ερώτημα το οποίο δε θέτει καθόλου κριτήρια (δηλαδή περιορισμούς) ως προς την επιλογή δεδομένων. Στην περίπτωση αυτή, το ερώτημά μας θα χρησιμοποιείται απλώς και μόνο για τη συγκέντρωση πεδίων που ενδεχομένως βρίσκονται σε πολλούς διαφορετικούς πίνακες (εκμεταλλευόμενο τις σχέσεις που καθορίστηκαν στη βάση δεδομένων). Όμως, ένα από τα μεγαλύτερα πλεονεκτήματα των ερωτημάτων είναι ότι μπορούμε σε αυτά να καθορίζουμε κριτήρια ως προς τα δεδομένα, ώστε να αποκλείονται από τα αποτελέσματα του ερωτήματος τα ακατάλληλα δεδομένα.

Πατώντας στο κελί **Κριτήρια** στο πλέγμα ερωτήματος, καταχωρούμε την παράσταση κριτηρίου. Η παράσταση του κριτηρίου, σε συνδυασμό με την τιμή του πεδίου, σχηματίζουν μια *λογική παράσταση*. Για παράδειγμα, αν η παράσταση κριτηρίου είναι **>15** (π.χ. για τη βαθμολογία ενός μαθητή), όταν σε μια εγγραφή η τιμή του αντίστοιχου πεδίου είναι 18 (οπότε η παράσταση γίνεται $18 > 15$, που είναι αληθής) τότε η αντίστοιχη εγγραφή εμφανίζεται στα αποτελέσματα. Αντίθετα, αν η τιμή του πεδίου είναι 10 (άρα και η παράσταση $10 > 15$ είναι ψευδής) η αντίστοιχη εγγραφή δεν εμφανίζεται στα αποτελέσματα. Ο επόμενος πίνακας δείχνει κάποιους βασικούς τελεστές και τις λέξεις-κλειδιά που μπορούν να χρησιμοποιηθούν στις παραστάσεις κριτηρίων, μέσα από μερικά απλά παραδείγματα.

Τελεστής	Σημασία	Παράδειγμα	Αποτέλεσμα
>	Μεγαλύτερο από	>20	Είναι αληθής όταν το πεδίο περιέχει αριθμό μεγαλύτερο από το 20
<	Μικρότερο από	<20	Είναι αληθής όταν το πεδίο περιέχει έναν αριθμό μικρότερο από το 20
=	Ίσο με	=120	Είναι αληθής μόνο

			όταν το πεδίο έχει ακριβώς τιμή 120
<=	Μικρότερο ή ίσο	<=50	Είναι αληθής όταν η τιμή αυτού του πεδίου είναι μικρότερη ή ίση με 50
<>	Όχι ίσο με	<>'Κυριακίδης'	Είναι αληθής όταν το πεδίο περιέχει οποιοδήποτε άλλο κείμενο εκτός από 'Κυριακίδης'. (Το Base προσθέτει μόνο του τα " μέσα στα οποία περικλείονται οι χαρακτήρες.
AND	Λογικό ΚΑΙ	>10 AND <20	Είναι αληθής όταν η τιμή του πεδίου είναι μεγαλύτερη από 10 και μικρότερη από 20.
OR	Λογικό Ή	<10 OR >20	Είναι αληθής όταν η τιμή του πεδίου είναι είτε μικρότερη από 10 είτε μεγαλύτερη από 20.
BETWEEN	Μεταξύ	BETWEEN 10 AND 20	Είναι αληθής όταν το πεδίο περιέχει τιμή μεγαλύτερη ή ίση του 10 ΚΑΙ μικρότερη ή ίση του 20.
NOT	Όχι	NOT 10	Είναι αληθής ότι το πεδίο περιέχει οποιαδήποτε άλλη τιμή εκτός από το 10.

Μπορούμε να θέσουμε κριτήρια για περισσότερα από ένα πεδία. Όταν τα κριτήρια αυτά βρίσκονται στην ίδια γραμμή του πλέγματος ερωτήματος, συνδυάζονται μεταξύ τους με τον τελεστή *AND* (και). Στην περίπτωση αυτή, για να θεωρηθεί ότι μια εγγραφή ικανοποιεί τα κριτήρια, θα πρέπει να ικανοποιεί όλα τα κριτήρια της γραμμής (δηλαδή να είναι αληθής σε όλα τα κριτήρια).

Κάτω από τη γραμμή **Κριτήρια** υπάρχουν και τέσσερις γραμμές **Είτε**. Στις γραμμές αυτές μπορούμε να καταχωρήσουμε και άλλα σύνολα κριτηρίων, τα οποία εξετάζονται *εναλλακτικά* ως προς αυτό της γραμμής **Κριτήρια** (με άλλα λόγια, τα σύνολα αυτά συνδέονται με τον τελεστή OR (είτε)).

Αφού ολοκληρωθεί ο σχεδιασμός του ερωτήματός μας (ή και κατά τη διάρκεια του σχεδιασμού, αν θέλουμε να ελέγχουμε ενδιάμεσα τα αποτελέσματα), έρχεται η στιγμή να εκτελέσουμε. Επιλέγοντας την **επεξεργασία** από το μενού, κλικάρουμε στην επιλογή **εκτέλεση ερωτήματος**. [5]

Φόρμες

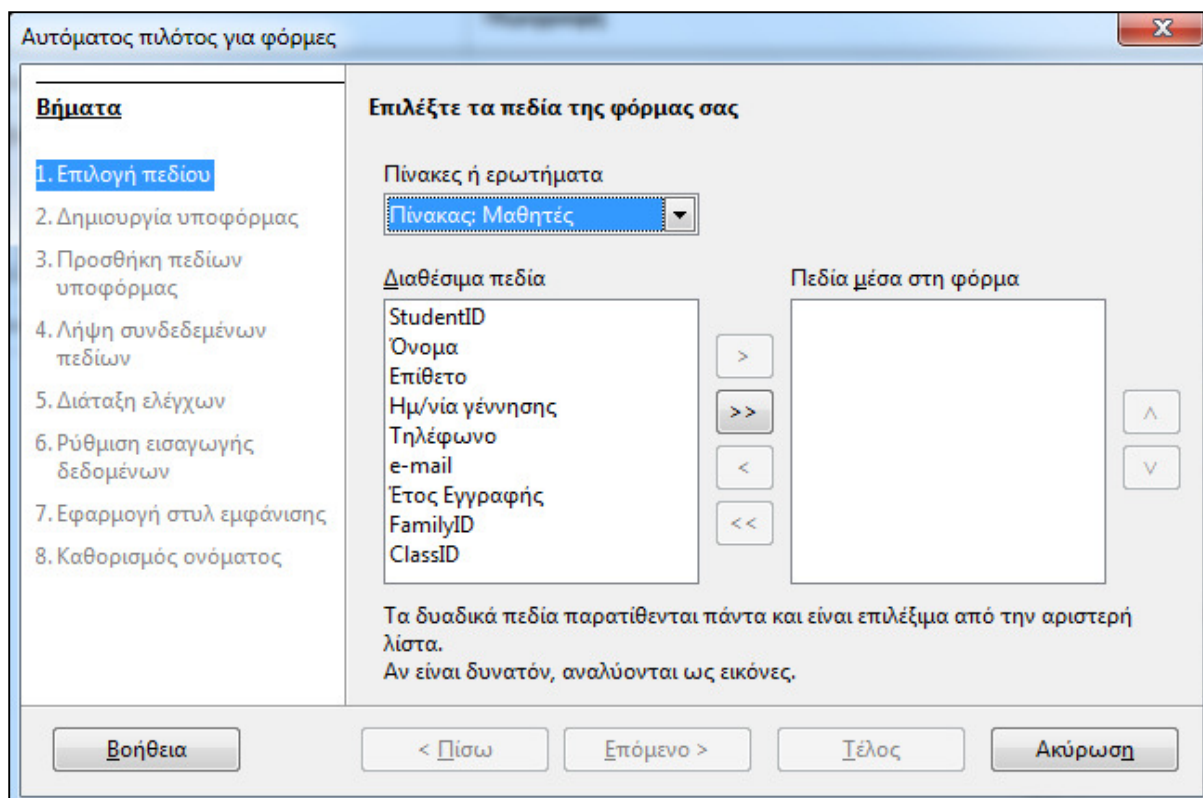
Αν και μπορούμε να καταχωρήσουμε ή να εμφανίσουμε τα δεδομένα μας στην προβολή φύλλου δεδομένων των πινάκων, αναμφίβολα οι περισσότεροι άνθρωποι θα εύρισκαν πιο βολική τη χρήση μιας φόρμας. Η φόρμα είναι στην πραγματικότητα, όπως έχει ήδη αναφερθεί, ένα αντικείμενο της βάσης δεδομένων που παίρνει απλώς τα δεδομένα από κάποιον πίνακα (ή από κάποιο ερώτημα) και τα εμφανίζει στην οθόνη με έναν πιο βολικό τρόπο.

Ουσιαστικά, η δημιουργία μιας φόρμας θα ήταν πάρα πολύ δύσκολη, αν η OpenOffice Base δε διέθετε μερικά εξαιρετικά χρήσιμα βοηθήματα (αυτόματος πιλότος για φόρμες) που, στις περισσότερες περιπτώσεις, κάνουν όλη τη δουλειά για λογαριασμό μας.

Για να δημιουργηθεί μια φόρμα, επιλέγουμε το εικονίδιο **Φόρμες**. Όπως συμβαίνει συνήθως, μας δίνεται η δυνατότητα είτε να χρησιμοποιήσουμε προβολή σχεδίασης, για να δημιουργήσουμε μια φόρμα χωρίς «εξωτερική βοήθεια», είτε να διαλέξουμε τη χρήση πιλότου για τη δημιουργία της φόρμας. Η δεύτερη επιλογή είναι και η πιο εύχρηστη για αρχάριους, γι αυτό και είναι αυτή που θα αναλυθεί.

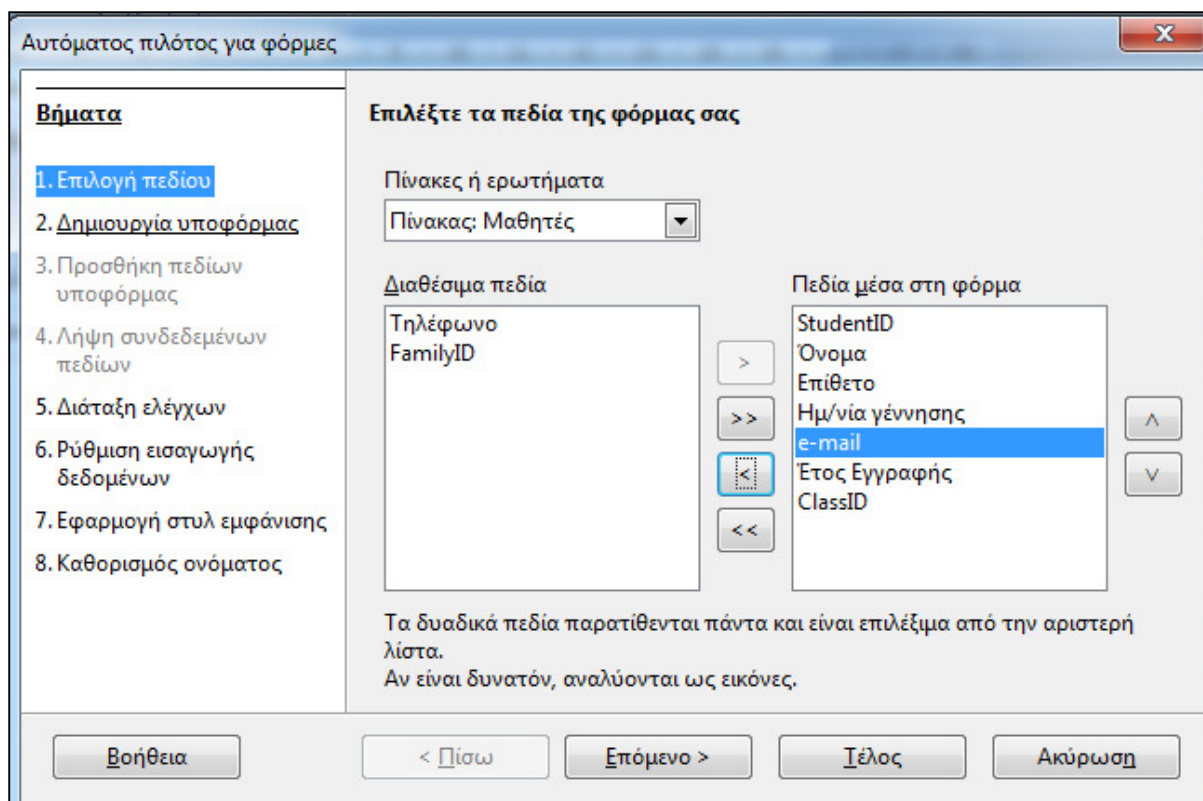
Επιλέγοντας τον **Αυτόματο πιλότο για φόρμες** ανοίγει ένα νέο παράθυρο, στα αριστερά του οποίου αναφέρονται τα 8 βήματα για τη δημιουργία της φόρμας :

1. Επιλογή πεδίου
2. Δημιουργία υποφόρμας
3. Προσθήκη πεδίων υποφόρμας
4. Λήψη συνδεδεμένων πεδίων
5. Διάταξη ελέγχων
6. Ρύθμιση εισαγωγής δεδομένων
7. Εφαρμογή στυλ εμφάνισης
8. Καθορισμός ονόματος



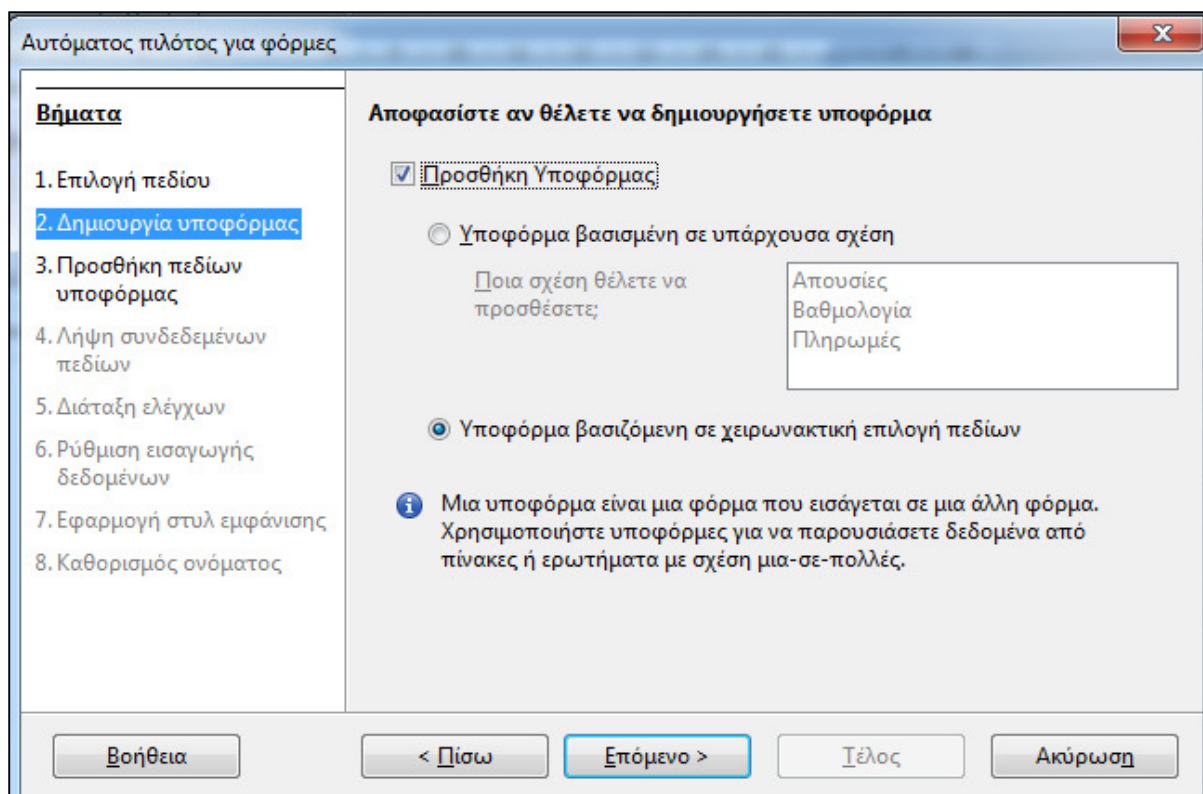
Εικόνα 10 : Προβολή του αρχικού πλαισίου που εμφανίζεται όταν επιλεγεί η χρήση αυτόματου πιλότου για τη δημιουργία μιας φόρμας

Στο πρώτο πλαίσιο διαλόγου του Οδηγού, πρέπει να επιλέξουμε τα πεδία που θέλουμε να εμφανίζονται στη φόρμα, από όποιον πίνακα ή όποιο ερώτημα θέλουμε. Για να το κάνουμε αυτό, επιλέγουμε από τον πτυσσόμενο κατάλογο **Πίνακες ή ερωτήματα** τον πίνακα ή το ερώτημα που περιέχει τα πεδία, επιλέγουμε τα επιθυμητά πεδία στον κατάλογο **Διαθέσιμα πεδία** και στη συνέχεια χρησιμοποιούμε τα τέσσερα κουμπιά που υπάρχουν στη μέση του πλαισίου διαλόγου για να μεταφέρουμε τα επιλεγμένα πεδία από και προς τον κατάλογο **Πεδία μέσα στη φόρμα**.



Εικόνα 11 : Επιλογή των επιθυμητών πεδίων που θα εμφανίζονται στη φόρμα

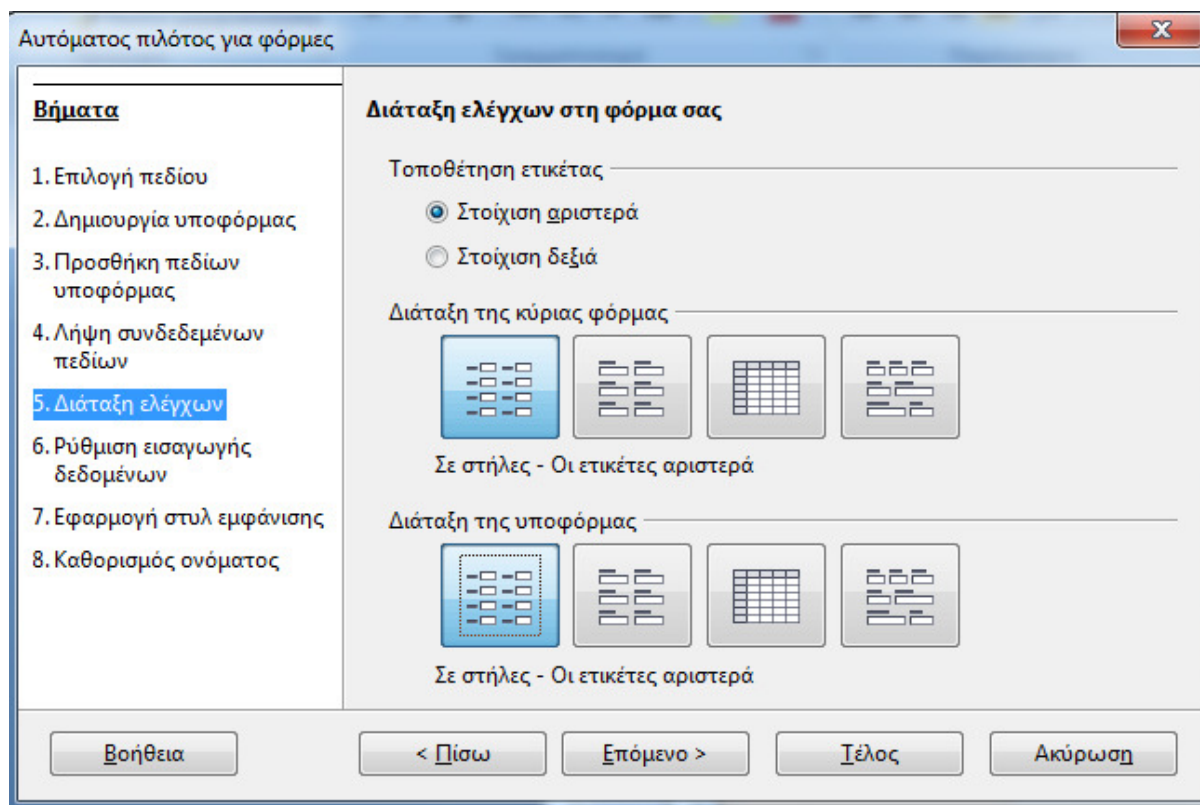
Όταν ολοκληρωθεί η επιλογή πεδίων, πατάμε το κουμπί *Επόμενο* για να εμφανιστεί το δεύτερο πλαίσιο διαλόγου του Οδηγού. Στο σημείο αυτό, μας δίδεται η δυνατότητα να αποφασίσουμε αν θέλουμε να δημιουργήσουμε μια υποφόρμα, δηλαδή μια φόρμα που εισάγεται σε μια άλλη. Οι υποφόρμες χρησιμοποιούνται για να παρουσιάσουμε δεδομένα από πίνακες ή ερωτήματα με σχέση ένα προς πολλά. Αν θελήσουμε να δημιουργηθεί υποφόρμα, πατώντας το κουμπί *Επόμενο* μας δίνεται η δυνατότητα να προσθέσουμε τα πεδία της (τρίτο πλαίσιο διαλόγου), ακριβώς όπως τα επιλέξαμε και στο πρώτο πλαίσιο διαλόγου για τη βασική φόρμα.



Εικόνα 12 : Επιλογή για τη δημιουργία υποφόρμας

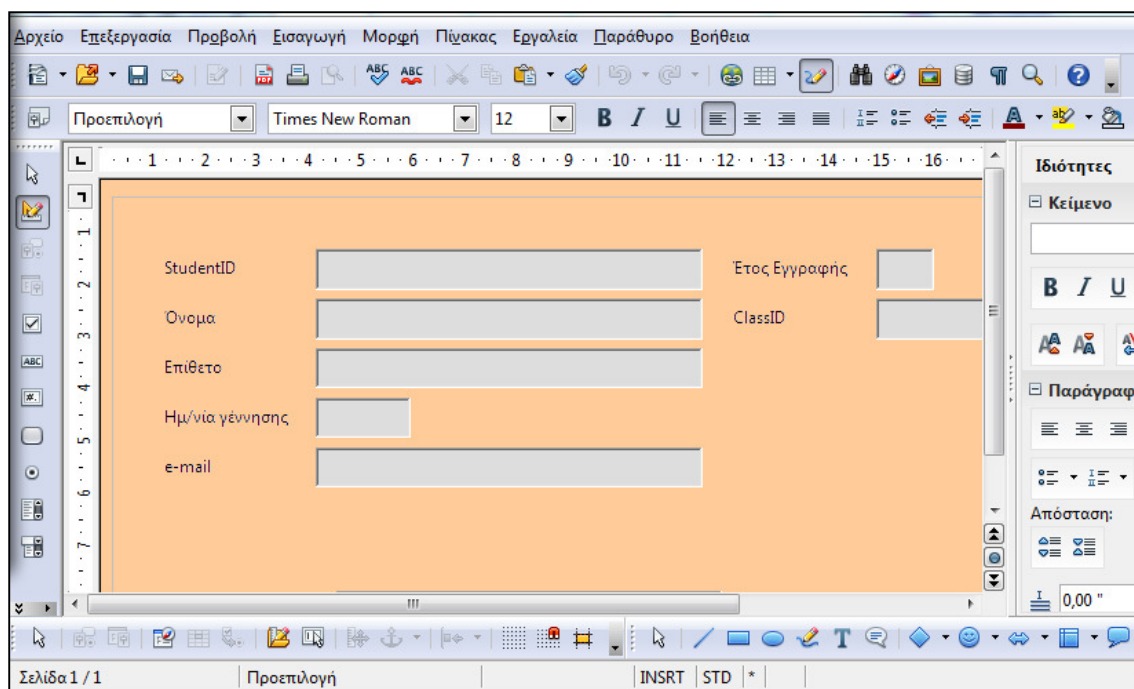
Αν δεν επιλέξουμε τη δημιουργία υποφόρμας, οδηγούμαστε αυτομάτως στο τέταρτο πλαίσιο διαλόγου, όπου επιλέγουμε τους συνδέσμους μεταξύ των φορμών μας.

Με τη διάταξη ελέγχων στο πέμπτο πλαίσιο, μπορούμε ουσιαστικά να επιλέξουμε τη μορφή με την οποία θα εμφανίζονται, τόσο η φόρμα όσο και η υποφόρμα που δημιουργούμε. [5]



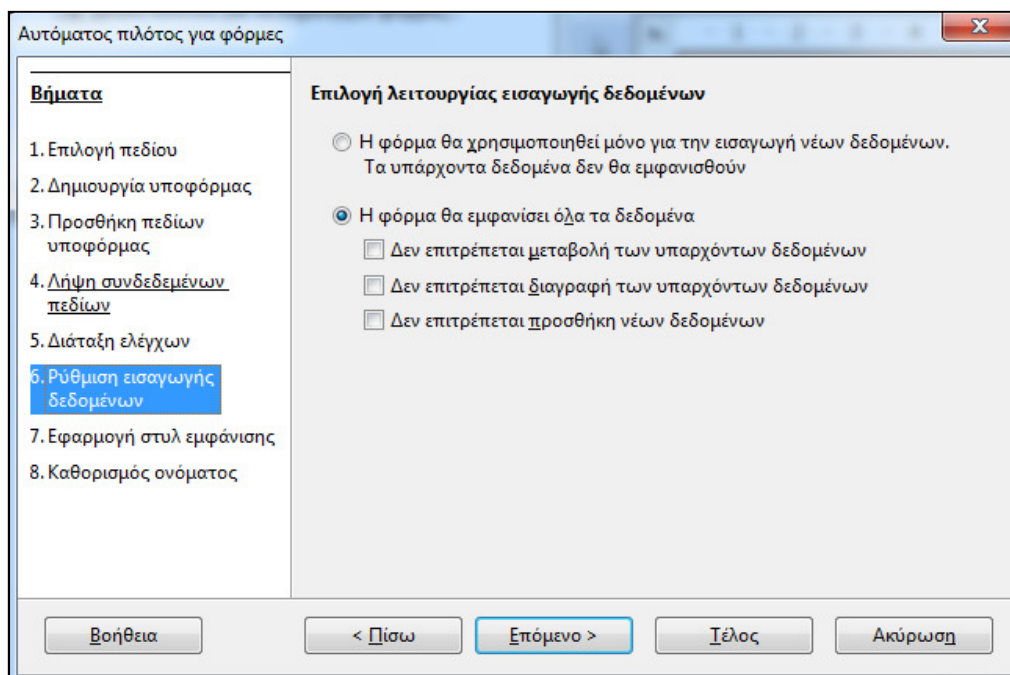
Εικόνα 13 : Διάταξη ελέγχων στη φόρμα

Όπως φαίνεται και στην εικόνα 13 μπορούμε να επιλέξουμε τη στοίχιση της ετικέτας (αριστερά ή δεξιά), και τη διάταξη της φόρμας και της υποφόρμας. Ο πιο απλός τρόπος για να αποφασίσουμε για την κατάλληλη διάταξη είναι να δοκιμάσουμε τις διάφορες επιλογές και να παρατηρήσουμε το δείγμα φόρμας που φαίνεται στο πλαίσιο προεπισκόπησης. (Παράδειγμα Εικόνα 14).



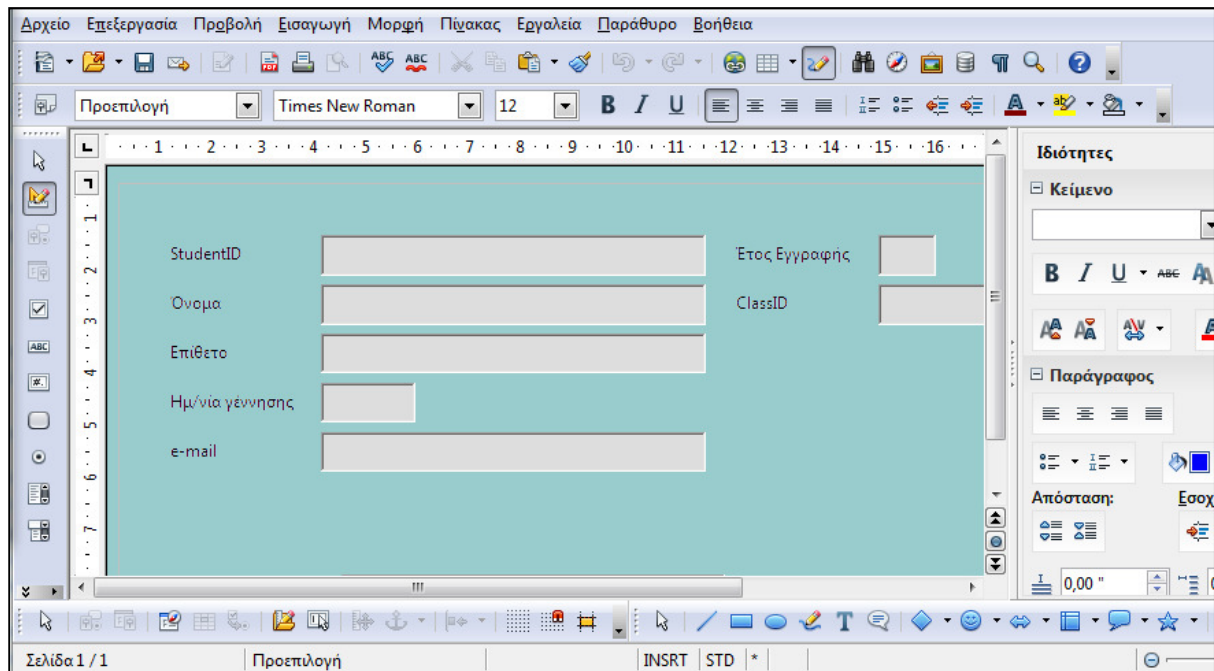
Εικόνα 14 : Προεπισκόπηση μιας επιλεγόμενης διάταξης

Στη συνέχεια, στο πλαίσιο διαλόγου *Ρύθμιση εισαγωγής δεδομένων* μπορούμε να επιλέξουμε είτε τη χρήση της φόρμας μόνο για εισαγωγή νέων δεδομένων-χωρίς να εμφανίζονται τα υπάρχοντα-είτε η φόρμα να εμφανίζει όλα τα δεδομένα- με δυνατότητα ή όχι μεταβολής ή διαγραφής των υπάρχοντων και προσθήκης νέων δεδομένων.



Εικόνα 15 : Επιλογή του τρόπου εισαγωγής (ή μη) των δεδομένων

Η εφαρμογή *στυλ εμφάνισης*, το έβδομο πλαίσιο διαλόγου, μας επιτρέπει να επιλέξουμε αν το πεδίο μας θα έχει περίγραμμα και αν ναι τι είδους (όψη 3D ή επίπεδη) καθώς και τη ρύθμιση χαρακτηριστικών όπως το χρώμα του φόντου. Κι εδώ ο καλύτερος τρόπος για να επιλέξουμε το κατάλληλο στυλ είναι να παρατηρήσουμε το πλαίσιο προεπισκόπησης.

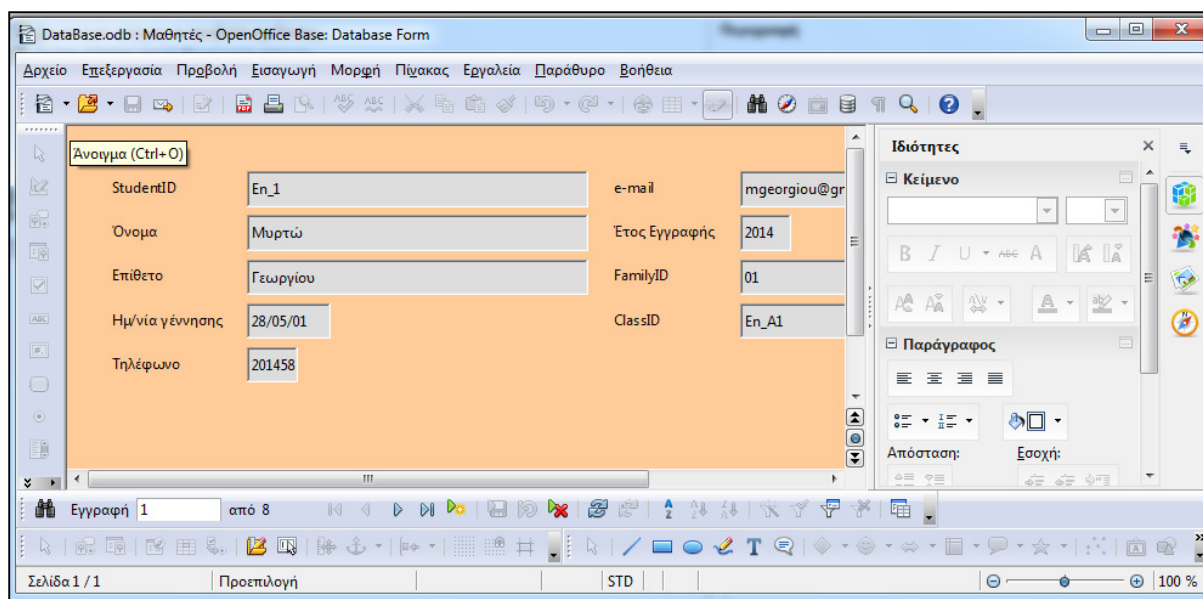


Εικόνα 16 : Πλαίσιο προεπισκόπησης του επιλεγόμενου στυλ εμφάνισης

Στο τελευταίο πλαίσιο διαλόγου του Οδηγού καθορίζουμε το όνομα με το οποίο θα αποθηκευτεί η φόρμα στη βάση δεδομένων και επιλέγουμε αν θέλουμε να ανοίξουμε κατευθείαν τη φόρμα ή αν προτιμούμε να την ανοίξουμε πρώτα και να την τροποποιήσουμε. Διαλέγουμε την επιθυμητή επιλογή και πατάμε το κουμπί **Τέλος**.

Αφού δημιουργηθεί η φόρμα, η εργασία σε αυτή για την καταχώριση δεδομένων δε διαφέρει σε τίποτα από την καταχώριση δεδομένων στα πλαίσια διαλόγου. Ουσιαστικά, το μόνο που χρειάζεται να κάνουμε είναι να πληκτρολογήσουμε τιμές στα πλαίσια κειμένου, να επιλέξουμε τιμές από τους πτυσσόμενους καταλόγους ή να συμπληρώσουμε με όποιον άλλο κατάλληλο τρόπο τα απαραίτητα στοιχεία.

Τόσο στο πάνω όσο και στο κάτω μέρος της φόρμας υπάρχει γραμμή εργαλείων, που επιτρέπει τη δημιουργία νέων εγγραφών, τη διαγραφή τους, την ταξινόμησή τους αλφαβητικά (από το Α στο Ζ και αντίστροφα) κ.ά.



Εικόνα 17 : Εισαγωγή δεδομένων στη φόρμα

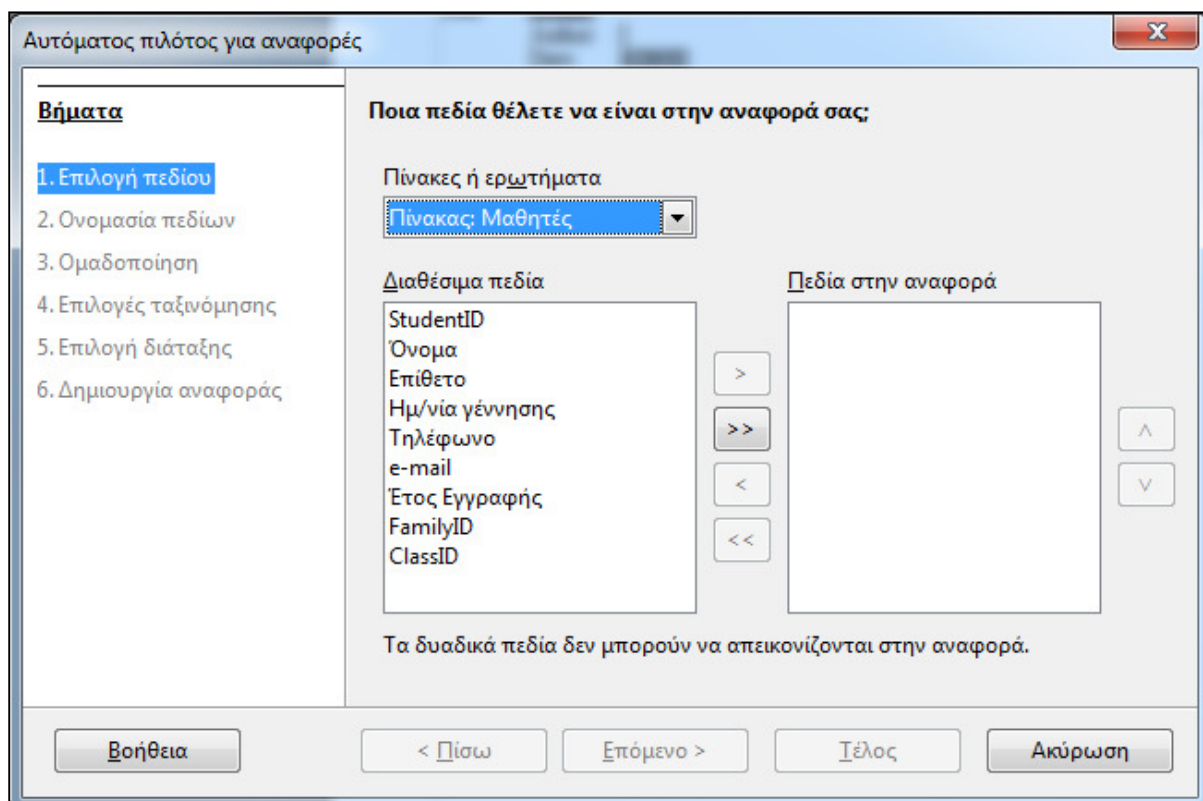
Ο ευκολότερος τρόπος για να προσθέσουμε χειριστήρια στη φόρμα μας, είναι να επιλέξουμε **Προσαρμογή εργαλειοθήκης**. Τα κουμπιά της εργαλειοθήκης αντιστοιχούν στα διάφορα είδη χειριστηρίων που μπορούμε να προσθέσουμε στις φόρμες μας. Θα πρέπει βέβαια, ο τύπος χειριστηρίου να είναι κατάλληλος για την απεικόνιση δεδομένων αυτού του τύπου πεδίου που επιλέξαμε- κάτι που δε συμβαίνει πάντα. Για παράδειγμα, δεν μπορούμε να χρησιμοποιήσουμε ένα κουμπί εντολής για την απεικόνιση αριθμητικών δεδομένων.

Αν επιλέξουμε να ανοίξουμε το αρχείο της φόρμας για επεξεργασία, μπορούμε με τη γραμμή εργαλείων μορφοποίησης να πετύχουμε τις συνηθισμένες «αισθητικές» παρεμβάσεις στην εμφάνιση των χειριστηρίων- όπως είναι η αλλαγή στη γραμματοσειρά, το μέγεθος των χαρακτήρων ή το χρώμα του αλλά μπορούν να τροποποιηθούν και πολλές άλλες ιδιότητες, όπως το ύψος και πλάτος του πεδίου, το χρώμα του φόντου, το στυλ του περιγράμματος κ.ά. [5]

Αναφορές (ή Εκθέσεις)

Οι αναφορές ή εκθέσεις έχουν πολλές ομοιότητες με τις φόρμες. Στην πραγματικότητα, όπως και οι φόρμες έτσι και οι εκθέσεις παίρνουν απλώς τα δεδομένα από κάποιον πίνακα ή από κάποιο ερώτημα και τα εμφανίζουν στην οθόνη με διαφορετικό τρόπο. Επιπλέον, και οι εκθέσεις αποτελούνται από χειριστήρια- έτσι ο τρόπος δημιουργίας και τροποποίησης μιας έκθεσης δε διαφέρει σημαντικά από τον τρόπο δημιουργίας μιας φόρμας, αν εξαιρέσουμε τις πρόσθετες λειτουργίες ομαδοποίησης που παρέχουν οι εκθέσεις.

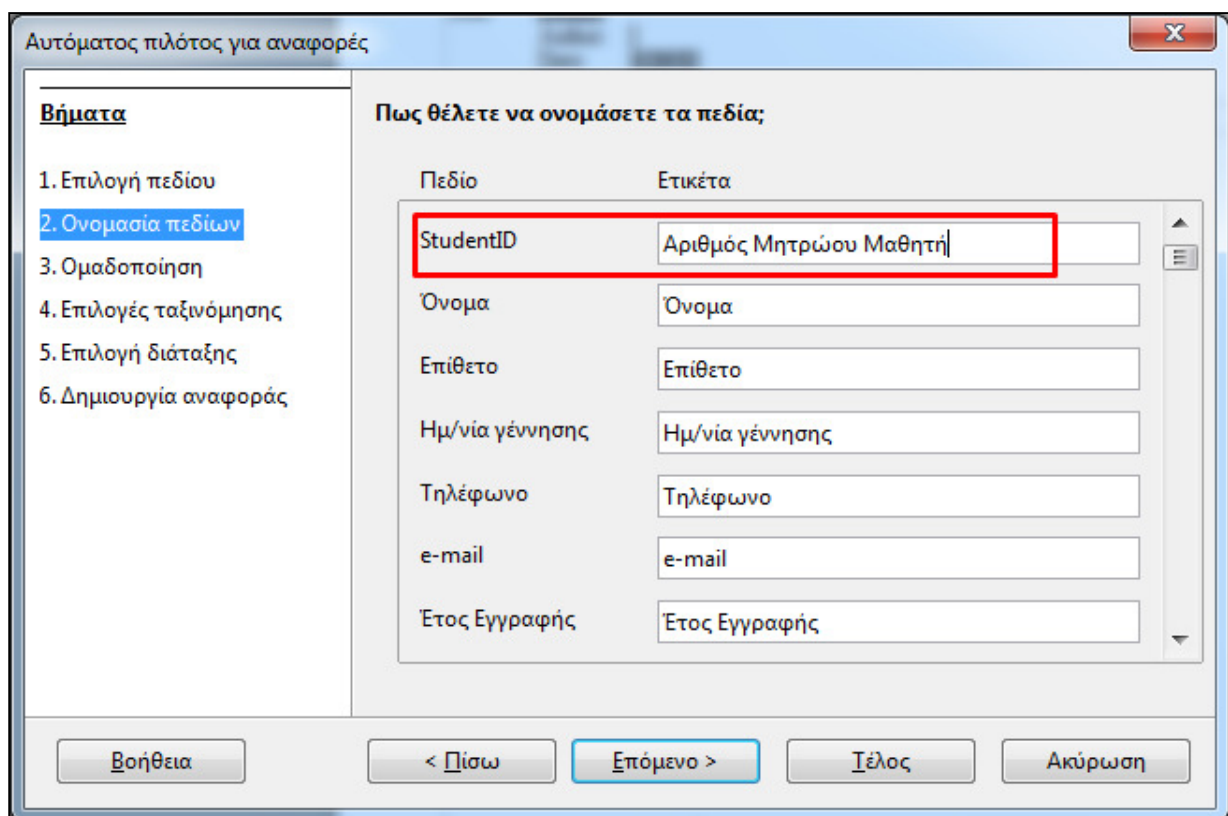
Η OpenOffice Base παρέχει ένα εξαιρετικά ευέλικτο και δυναμικό εργαλείο για τη δημιουργία εκθέσεων- τον *πιλότο* για τη δημιουργία εκθέσεων. Για να δημιουργήσουμε μια έκθεση, ενεργοποιούμε την επιλογή *χρήση πιλότου για τη δημιουργία αναφοράς*.



Εικόνα 18 : Προβολή του αρχικού πλαισίου όταν επιλεγεί η χρήση πιλότου για τη δημιουργία αναφοράς

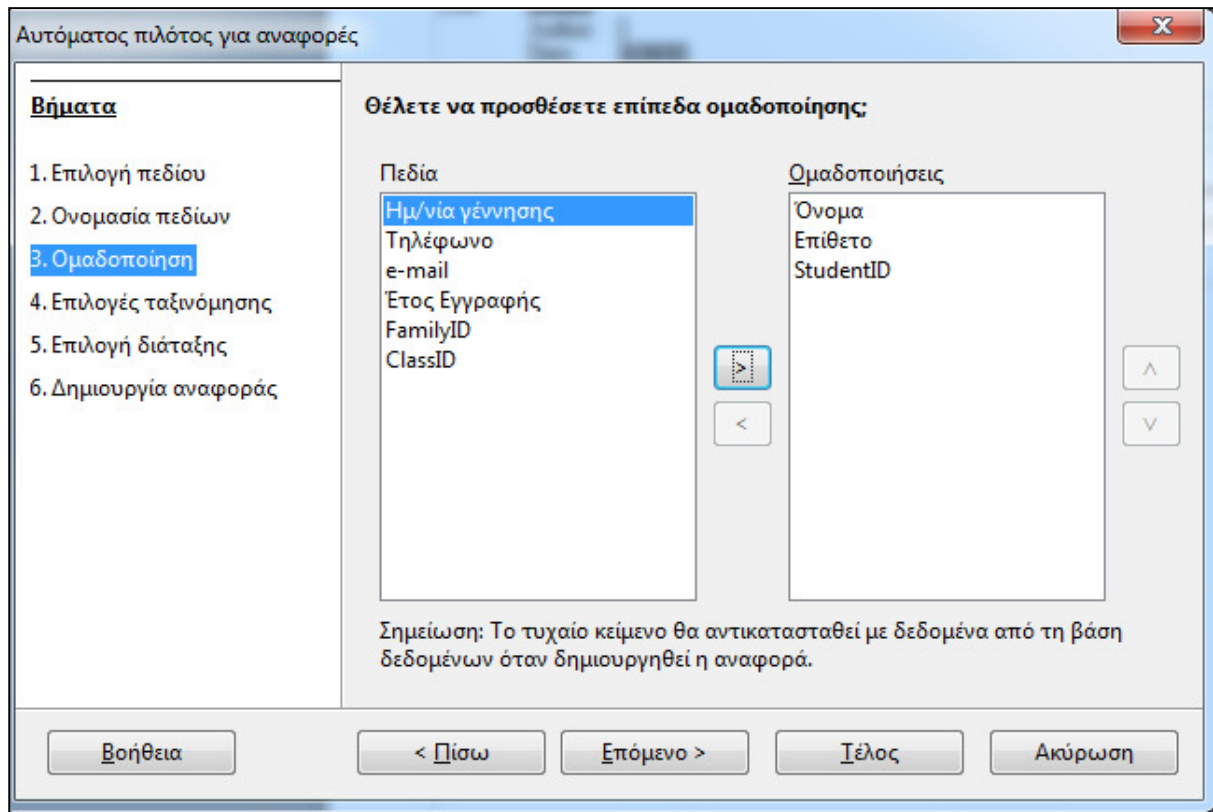
Στο πρώτο πλαίσιο διαλόγου του πιλότου πρέπει να επιλέξουμε τα πεδία που θέλουμε να συμπεριληφθούν στην αναφορά. Όπως συμβαίνει και σε άλλους, αντίστοιχους πιλότους στην OpenOffice Base, έχουμε τη δυνατότητα να επιλέξουμε πεδία από τους πίνακες ή τα ερωτήματα. Επιλέγουμε πρώτα τον πίνακα ή το ερώτημα στον κατάλογο **Πίνακες ή ερωτήματα** και στη συνέχεια μεταφέρουμε τα επιθυμητά πεδία στον κατάλογο **Πεδία στην αναφορά**, χρησιμοποιώντας τα κουμπιά που βρίσκονται στο μέσο του πλαισίου διαλόγου. Όταν ολοκληρωθεί η επιλογή των πεδίων, πατάμε το κουμπί **Επόμενο** για να εμφανιστεί το επόμενο πλαίσιο διαλόγου.

Στο δεύτερο πλαίσιο διαλόγου, μας δίνεται η δυνατότητα να ονομάσουμε διαφορετικά τα πεδία που έχουμε επιλέξει-η αρχική ονομασία είναι το όνομα του αντίστοιχου πεδίου.



Εικόνα 19 : Παρουσίαση της δυνατότητας αλλαγής του αρχικού ονόματος ενός πεδίου

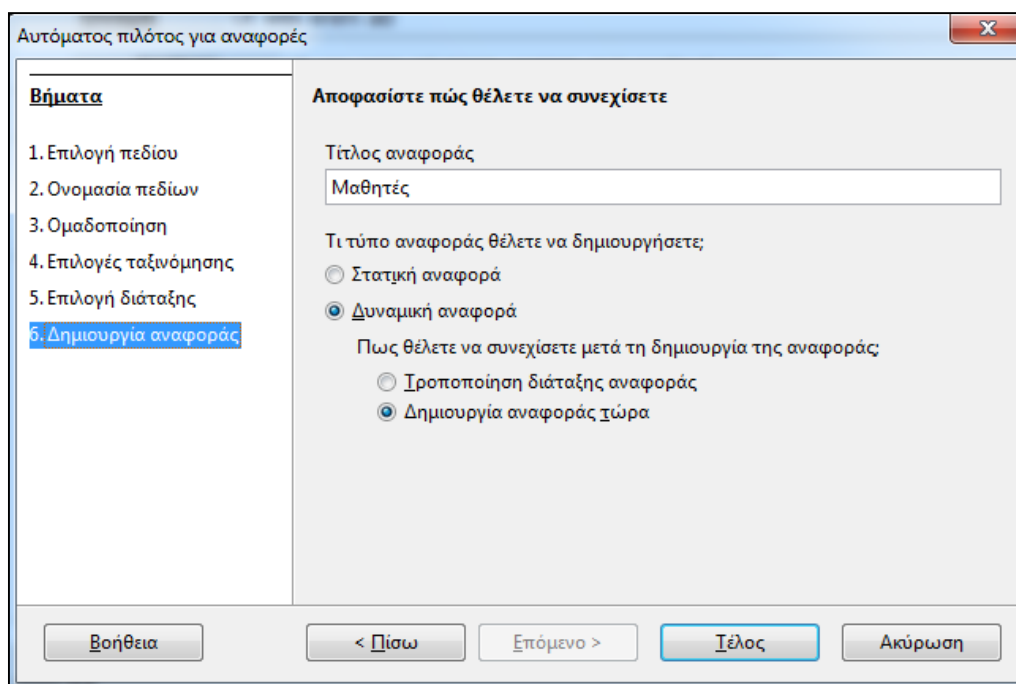
Όπως φαίνεται και στην Εικόνα 20, στο τρίτο πλαίσιο διαλόγου, δύναται η δυνατότητα επιλογής των πεδίων που θέλουμε να ομαδοποιηθούν.



Εικόνα 20 : Επιλογή πεδίων που θέλουμε να ομαδοποιηθούν

Η ομαδοποίηση εξαρτάται από το είδος της σχέσεις που υπάρχει μεταξύ των πεδίων. Στο πλαίσιο προεπισκόπησης θα εμφανιστεί τυχαίο κείμενο, το οποίο θα αντικατασταθεί με δεδομένα από τη βάση δεδομένων όταν δημιουργηθεί η αναφορά (Εικόνα 21). [5]

προτιμάμε να συνεχίσουμε τις τροποποιήσεις (*Τροποποίηση Διάταξης αναφοράς*). Αφού κάνουμε τις επιλογές μας, πατάμε στο κουμπί **Τέλος**.



Εικόνα 22 : Εμφάνιση επιλογών στο τελευταίο πλαίσιο διαλόγου

Όπως και στις φόρμες, μπορούμε με τις γραμμές εργαλείων-ιδιοτήτων να κάνουμε παρεμβάσεις στη γραμματοσειρά, το μέγεθος χαρακτήρων και το χρώμα τους, αλλά μπορούμε και να καθορίζουμε ποια στοιχεία θα εμφανίζονται ως κεφαλίδες ή υποσέλιδα, καθώς και την προσθήκη υπομνημάτων και διαγραμμάτων ροής. Και οι δυνατότητες είναι απεριόριστες.

Για να εμφανίσουμε και να τυπώσουμε μια αναφορά, διπλοπατάμε πάνω στην αναφορά στο παράθυρο της βάσης δεδομένων. Επιλέγουμε προεπισκόπηση εκτύπωσης από το μενού **Αρχείο** για να ελέγξουμε την εμφάνισή της. Από τη **Μορφή**, πατώντας **Στυλ και Μορφοποίηση** μπορούμε να τροποποιήσουμε ότι θέλουμε, και αφού είμαστε σίγουροι για το τελικό αποτέλεσμα- αφού διαπιστώσουμε και σε πόσες σελίδες εκτείνεται η αναφορά μας- την τυπώνουμε, πατώντας στο κουμπί **Εκτύπωση** της γραμμής εργαλείων ή δίνοντας τη διαταγή **Εκτύπωση** από το μενού **Αρχείο**. [5]

ΚΕΦΑΛΑΙΟ 4

Σκοπός Δημιουργίας της Βάσης Δεδομένων για ένα Κέντρο Ξένων Γλωσσών

Η δημιουργία της βάσης δεδομένων για ένα Κέντρο Ξένων Γλωσσών έχει ως σκοπό την πλήρη και ολοκληρωμένη ηλεκτρονική διαχείριση όλων των στοιχείων: μαθητές, καθηγητές, τμήματα, μαθήματα, βαθμολογίες, απουσίες, πληρωμές να βρίσκονται στον υπολογιστή του χρήστη/διαχειριστή, καλά οργανωμένα και τακτοποιημένα και στη διάθεσή του οποιαδήποτε στιγμή.

Ανάλυση Αναγκών-Απαιτήσεων

Για να είναι επιτυχές ένα σύστημα διαχείρισης, θα πρέπει να είναι προσαρμοσμένο στις ανάγκες, τις απαιτήσεις, αλλά και τις προσδοκίες του τελικού χρήστη. Αυτό σημαίνει ότι το ζητούμενο είναι η πραγματική επιθυμία του χρήστη, τι ακριβώς περιμένει από το σύστημα, πόσο φιλικό είναι προς τον ίδιο και κατά πόσο ικανοποιεί τους σκοπούς για τους οποίους υλοποιήθηκε.

Ερωτήματα που σε αρχικό στάδιο θέλουμε να απαντά η βάση δεδομένων είναι π.χ. η εύρεση των ονομάτων των μαθητών που αρίστευσαν σε συγκεκριμένη αξιολόγηση, η εύρεση των μαθητών με περισσότερες από τις επιτρεπόμενες απουσίες, καθώς και τα στοιχεία των κηδεμόνων τους για άμεση ενημέρωση.

Η συγκεκριμένη βάση δεδομένων απευθύνεται σε μια μόνο κατηγορία χρηστών, σε αυτούς που αποτελούν τη γραμματεία του κέντρου ξένων γλωσσών. Οι χρήστες μπορούν να τελέσουν όλες τις επιτρεπτές ενέργειες με τα στοιχεία της βάσης. Πιο συγκεκριμένα, ο εκάστοτε γραμματέας του φροντιστηρίου θα μπορεί να κάνει εισαγωγή των στοιχείων των μαθητών, καθηγητών κτλ. καθώς και να μεταβάλλει ή να διαγράψει στοιχεία που έχουν αλλάξει ή έχει γίνει λανθασμένη εισαγωγή αυτών.

Κανονικοποίηση

Τα προβλήματα που είναι πιθανό να εμφανιστούν κατά τη διαδικασία της υλοποίησης του σχεδιασμού μιας βάσης δεδομένων είναι η περιττή επανάληψη πληροφοριών, καθώς όταν ένας αρχάριος χρήστης αρχίζει να δημιουργεί μια Βάση Δεδομένων, συνήθως έχει την τάση να τοποθετεί όλες τις πληροφορίες που του χρειάζονται σε έναν πίνακα. Ας υποθέσουμε, για παράδειγμα, ότι θέλουμε να

καταχωρίσουμε τα στοιχεία των μαθητών μιας τάξης και τις βαθμολογίες τους σε μια Βάση Δεδομένων. Κάθε φορά που θα θέλαμε να καταχωρίσουμε για ένα μαθητή τη βαθμολογία ενός μαθήματος στον πίνακα, θα έπρεπε να επαναλαμβάνουμε μια σειρά από πληροφορίες, όπως StudentID, Όνομα, Επίθετο, Τηλέφωνο κ.τ.λ. Η επανάληψη ορισμένων δεδομένων, όμως, κοστίζει σε χρόνο, κόπο αλλά αυξάνει και τον κίνδυνο να γίνει κάποιο λάθος. Επίσης, σε περίπτωση που θελήσουμε να αλλάξουμε τη διεύθυνση ενός μαθητή ή τον αριθμό τηλεφώνου του, θα πρέπει να βρίσκουμε και να ενημερώνουμε όλες τις εγγραφές στις οποίες αυτός εμφανίζεται.

Προσπαθώντας επομένως να λύσουμε το πρόβλημα με το «στρίμωγμα» όλων των δεδομένων σε έναν πίνακα, οδηγούμαστε σε αδιέξοδο. Εξ'άλλου αν είναι να τοποθετήσουμε όλα τα δεδομένα σε έναν πίνακα, δεν είναι ανάγκη να χρησιμοποιήσουμε την Base και τις βάσεις δεδομένων, αλλά είναι πιο εύκολο να τα καταχωρίσουμε σε ένα λογιστικό φύλλο, τύπου Excel.

Το αδιέξοδο μπορεί να λυθεί, αν το προσεγγίσουμε από μια διαφορετική οπτική γωνία. Να διαχωρίσουμε τα δεδομένα σε διαφορετικούς πίνακες. Δηλαδή, ένα πίνακα ο οποίος θα περιέχει τα στοιχεία των μαθητών, έναν πίνακα με τα μαθήματα και ένα πίνακα με τις βαθμολογίες. Αυτή η διαδικασία ονομάζεται κανονικοποίηση.

Η κανονικοποίηση (normalization) είναι μια διαδικασία στη διάρκεια της οποίας διαχωρίζουμε τις πληροφορίες σε διάφορους πίνακες και αφαιρούμε εκείνα τα δεδομένα τα οποία πλεονάζουν από τη Β.Δ. Η διαδικασία της κανονικοποίησης μπορεί να συνοψιστεί σε μερικούς απλούς κανόνες:

1. Σε έναν πίνακα κάθε εγγραφή θα πρέπει να είναι μοναδική. Επομένως δεν πρέπει να υπάρχουν ποτέ δυο όμοιες εγγραφές σε έναν πίνακα. Αυτό επιτυγχάνεται, όπως είπαμε και προηγουμένως, με τη βοήθεια κάποιου πρωτεύοντος κλειδιού.
2. Σε έναν πίνακα δεν πρέπει να υπάρχουν επαναλαμβανόμενα πεδία όπως *Μάθημα 1*, *Μάθημα 2*, *Μάθημα 3* κτλ., δηλαδή το κάθε πεδίο πρέπει να περιέχει ένα μοναδικό είδος πληροφορίας.
3. Εφόσον κάθε πεδίο πρέπει να περιέχει ένα μοναδικό είδος πληροφοριών, δεν πρέπει να ολισθήσουμε στο άλλο άκρο και να συγχωνεύσουμε δυο πεδία σε ένα. Για παράδειγμα, σε ένα πίνακα πελατών δεν μπορούμε να συγχωνεύσουμε τα πεδία *διεύθυνση* και *πόλη*, διότι πρόκειται για δυο διαφορετικές ιδιότητες ενός θέματος (πελάτες).
4. Ένας πίνακας θα πρέπει να περιέχει όλα εκείνα τα πεδία τα οποία θα περιγράφουν πλήρως ένα θέμα (π.χ. Μαθητές). Ταυτόχρονα όμως πεδία τα

οποία δεν έχουν άμεση σχέση με το συγκεκριμένο θέμα (π.χ. Μαθήματα ή Βαθμοί) θα πρέπει να μεταφερθούν σε άλλους πίνακες.

5. Σε οποιοδήποτε πεδίο ενός πίνακα θα πρέπει να μπορούμε να κάνουμε τροποποιήσεις, δίχως να επηρεάζεται κάποιο άλλο πεδίο, με εξαίρεση το πεδίο του πρωτεύοντος κλειδιού.

Είναι απαραίτητο λοιπόν να αφιερωθεί στην αρχή αρκετός χρόνος για τη σωστή ανάλυση του προβλήματος και την εφαρμογή των παραπάνω κανόνων στη σχεδίαση των πινάκων.

Το Μοντέλο Οντοτήτων -Συσχετίσεων (Entity-Relationship Model)

Το μοντέλο οντοτήτων-συσχετίσεων (μοντέλο Ο/Σ –ER model) είναι ένα αφαιρετικό, ιδεατό μοντέλο δεδομένων, τα οποία έχουν καθορισμένη δομή. Στη μηχανική λογισμικού χρησιμοποιείται για να παρέχει ένα εννοιολογικό σχήμα κατά τη σχεδίαση βάσεων δεδομένων, ως μοντέλο δεδομένων ενός συστήματος και των απαιτήσεών του με top-down προσέγγιση. Ένα διάγραμμα που δημιουργείται με αυτή τη διαδικασία σχεδίασης καλείται διάγραμμα οντοτήτων-συσχετίσεων. Προτάθηκε αρχικά το 1976 από τον Peter Chen, ωστόσο στη συνέχεια επινοήθηκαν πολλές παραλλαγές της διαδικασίας.

Χρησιμοποιείται στο πρώτο στάδιο σχεδίασης ενός συστήματος πληροφοριών κατά την ανάλυση των απαιτήσεών του. Σκοπός του είναι να περιγράψει τις αναγκαίες πληροφορίες, οι οποίες πρόκειται να αποθηκευτούν στη βάση δεδομένων ή τον τύπο τους. Η μοντελοποίηση δεδομένων γίνεται για την περιγραφή των χρησιμοποιούμενων όρων και των σχέσεών τους σε ένα ορισμένο τομέα ενδιαφέροντος. Στην περίπτωση σχεδιασμού ενός συστήματος πληροφοριών, που στηρίζεται σε μια βάση δεδομένων, το εννοιολογικό μοντέλο δεδομένων χαρτογραφείται σε προχωρημένο στάδιο, σε ένα λογικό μοντέλο δεδομένων, όπως το σχεσιακό μοντέλο δεδομένων. Το στάδιο αυτό ονομάζεται συνήθως στάδιο λογικού σχεδιασμού. Ύστερα, κατά τη διάρκεια του φυσικού σχεδιασμού το λογικό μοντέλο χαρτογραφείται σε κάποιο φυσικό μοντέλο. Ας σημειωθεί ότι ορισμένες φορές και οι δύο φάσεις αναφέρονται ως «**φυσικός σχεδιασμός**».

Οι λεπτομέρειες για το πώς να σχεδιαστούν τα διαγράμματα ποικίλλουν ελαφρώς από μια μέθοδο σε άλλη, αλλά όλες έχουν τα ίδια βασικά στοιχεία. Τους τύπους, τις ιδιότητες και τις σχέσεις οντοτήτων, αυτές οι τρεις κατηγορίες θεωρούνται επαρκείς για να διαμορφώσουν τα ουσιαστικά στατικά μέρη των αναγκών επεξεργασίας πληροφοριών οποιασδήποτε οργάνωσης σε μια οποιαδήποτε βάση δεδομένων.

Στην πρώτη φάση της σχεδίασης της βάσης δεδομένων χρησιμοποιείται η σχεδίαση σε διαγράμματα, κατά τη διάρκεια της οποίας γίνεται ανάλυση των απαιτήσεων από το σχεδιαστή προκειμένου να περιγραφούν οι απαιτήσεις πληροφοριών ή το είδος αυτών που θα αποθηκευτεί στη ΒΔ.

Βασικές έννοιες μοντέλου Οντοτήτων- Συσχετίσεων (E-R μοντέλου)

Βάση των E-R μοντέλων είναι η κατηγοριοποίηση των αντικειμένων και των σχέσεών τους μεταξύ τους.

Οντότητα (Entity) είναι ένα αντικείμενο ενδιαφέροντος στον πραγματικό κόσμο, το οποίο ξεχωρίζει από τα υπόλοιπα. Μια Οντότητα λειτουργεί αφαιρετικά σε ένα πολύπλοκο τομέα. Οντότητες μπορεί να είναι άνθρωποι, μέρη, αντικείμενα, γεγονότα, έννοιες κτλ.

Ιδιότητα

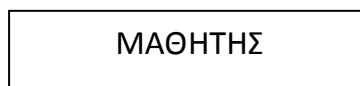
Ένας μαθητής μπορεί να έχει το πεδίο «μάθημα», το οποίο όμως δέχεται ως τιμές ένα σύνολο μαθημάτων που αυτός παρακολουθεί. Κάθε οντότητα έχει διάφορα στοιχεία που την προσδιορίζουν. Ένα τέτοιο στοιχείο ονομάζεται **ιδιότητα, χαρακτηριστικό** ή **πεδίο** της οντότητας. Κάθε οντότητα (εκτός αν είναι μια αδύνατη οντότητα) πρέπει να έχει κατά ελάχιστο μοναδική ιδιότητα, με την οποία και αναγνωρίζεται κάθε της στιγμιότυπο, το οποίο καλείται πρωτεύον κλειδί της οντότητας.

Οι ιδιότητες χωρίζονται σε :

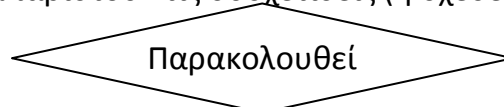
- **Μονότιμες**, τα οποία έχουν μόνο μία τιμή και
- **Πλειότιμες**, τα οποία έχουν σύνολο από τιμές

Στο σημείο αυτό γίνεται η επεξήγηση των συμβόλων ενός διαγράμματος Ο/Σ (E-R):

1. Γραμμές: για τη σύνδεση των οντοτήτων με τα γνωρίσματα και με τις συσχετίσεις.
2. Ορθογώνια: που αναπαριστούν τα σύνολα των οντοτήτων



3. Ρόμβοι: που αναπαριστούν τις συσχετίσεις (ή σχέσεις)



4. Ελλείψεις: που αναπαριστούν τα γνωρίσματα (ή ιδιότητες). Οι διπλές ελλείψεις εκφράζουν τα πλειότιμα γνωρίσματα, ενώ οι διακεκομμένες τα παραγόμενα γνωρίσματα.

ΟΝΟΜΑ

5. Υπογράμμιση: γνωρίσματα που είναι πρωτεύοντα κλειδιά
6. Ασθενείς οντότητες: (σχεδιάζονται με διπλό παραλληλόγραμμο) όταν μια οντότητα χρειάζεται να «δανειστεί» το κλειδί μιας άλλης.
7. Συμμετοχή ενός συνόλου οντοτήτων σε ένα σύνολο συσχετίσεων: **A)Ολική συμμετοχή**, (σχεδιάζεται με διπλή γραμμή) κάθε οντότητα του συνόλου οντοτήτων συμμετέχει σε τουλάχιστον μια συσχέτιση στο σύνολο των συσχετίσεων. Π.χ. Κάθε δάνειο πρέπει να σχετίζεται με ένα πελάτη μέσω της συσχέτισης δανειζόμενος. **B)Μερική συμμετοχή**, όταν ορισμένες οντότητες μπορεί να μην συμμετέχουν σε κάθε συσχέτιση του συνόλου συσχετίσεων. Π.χ. Μερική η συμμετοχή του πελάτη στο δανειζόμενος.
8. Η πληθικότητα (cardinality): περιγράφει τον αριθμό των στιγμιότυπων ενός τύπου οντοτήτων που μπορούν να αντιστοιχίζονται με μία οντότητα ενός άλλου τύπου σε μια συσχέτιση. Πιο συγκεκριμένα, κατά τη γραφική συσχέτιση δυο πινάκων χρησιμοποιούνται τα γράμματα **M** και **N** και ο αριθμός **1**. Πρόκειται για τον τρόπο που δηλώνεται ο τύπος της σχέσης ανάμεσα σε δυο οντότητες. Η σχέση είναι δυο δρόμων και δηλώνει σε κάθε εισαγωγή του κάθε πίνακα πόσες εγγραφές του άλλου πίνακα σχετίζονται με τη συγκεκριμένη εφαρμογή.

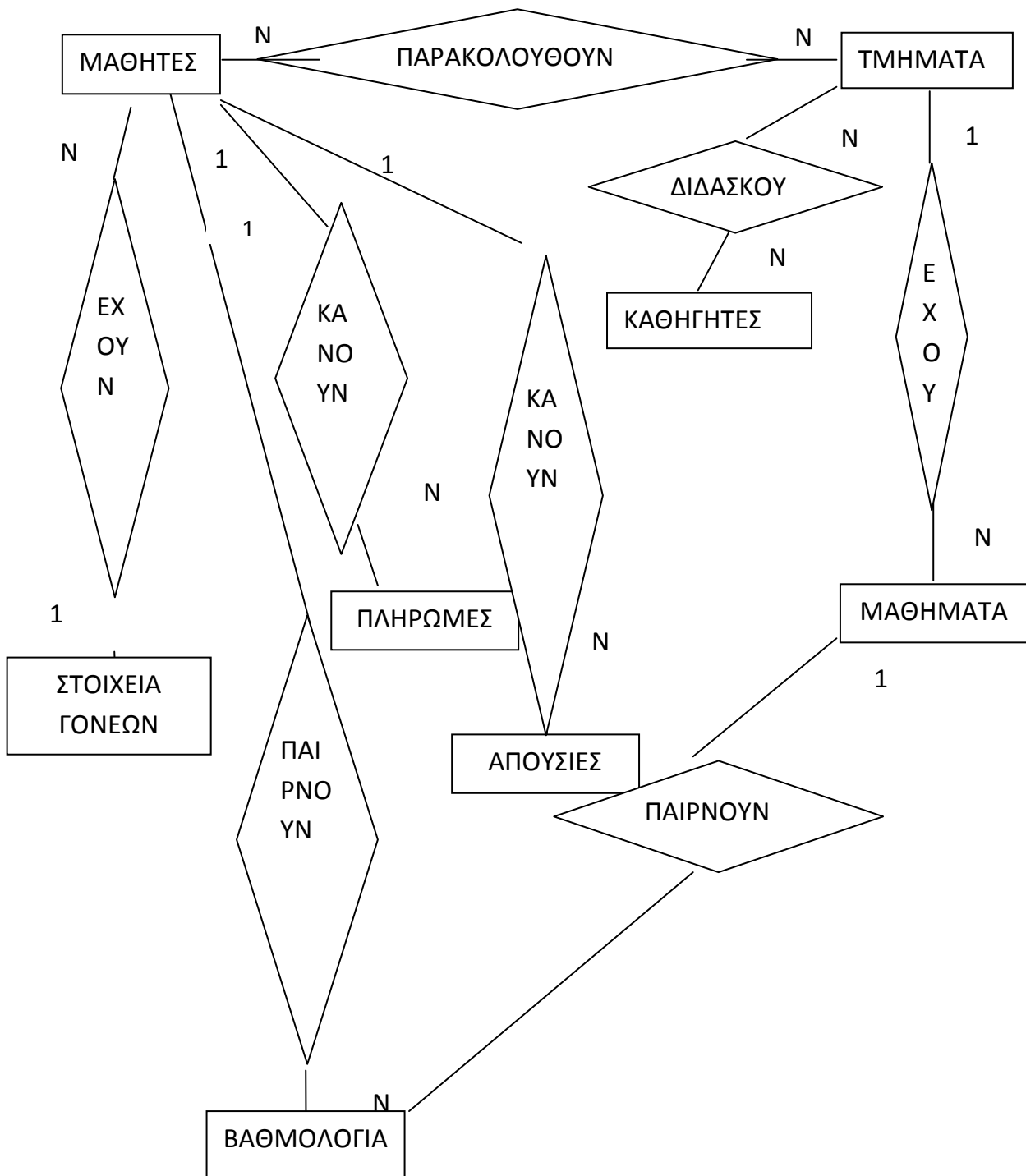
Για παράδειγμα, για τη σχέση ανάμεσα στους πίνακες **Στοιχεία Γονέων** και **Μαθητές** έχουμε μια σχέση 1:N, αφού από την πλευρά του πίνακα **Στοιχεία Γονέων**, κάθε **FamilyID** μπορεί να αντιστοιχεί σε περισσότερους **Μαθητές**, όπως π.χ. στην περίπτωση αδερφών.

Γενικά, οι σχέσεις μεταξύ οντοτήτων μπορούν να χωριστούν στις εξής κατηγορίες

- ✓ **1 προς 1** (*one-to-one*)
- ✓ **1 προς N** (*one-to-many*)
- ✓ **N προς 1** (*many-to-one*)
- ✓ **M προς N** (*many-to-many*)

όπου N, M συμβολίζουν την ύπαρξη της έννοιας "πολλά" στο συσχετισμό των οντοτήτων. [6]

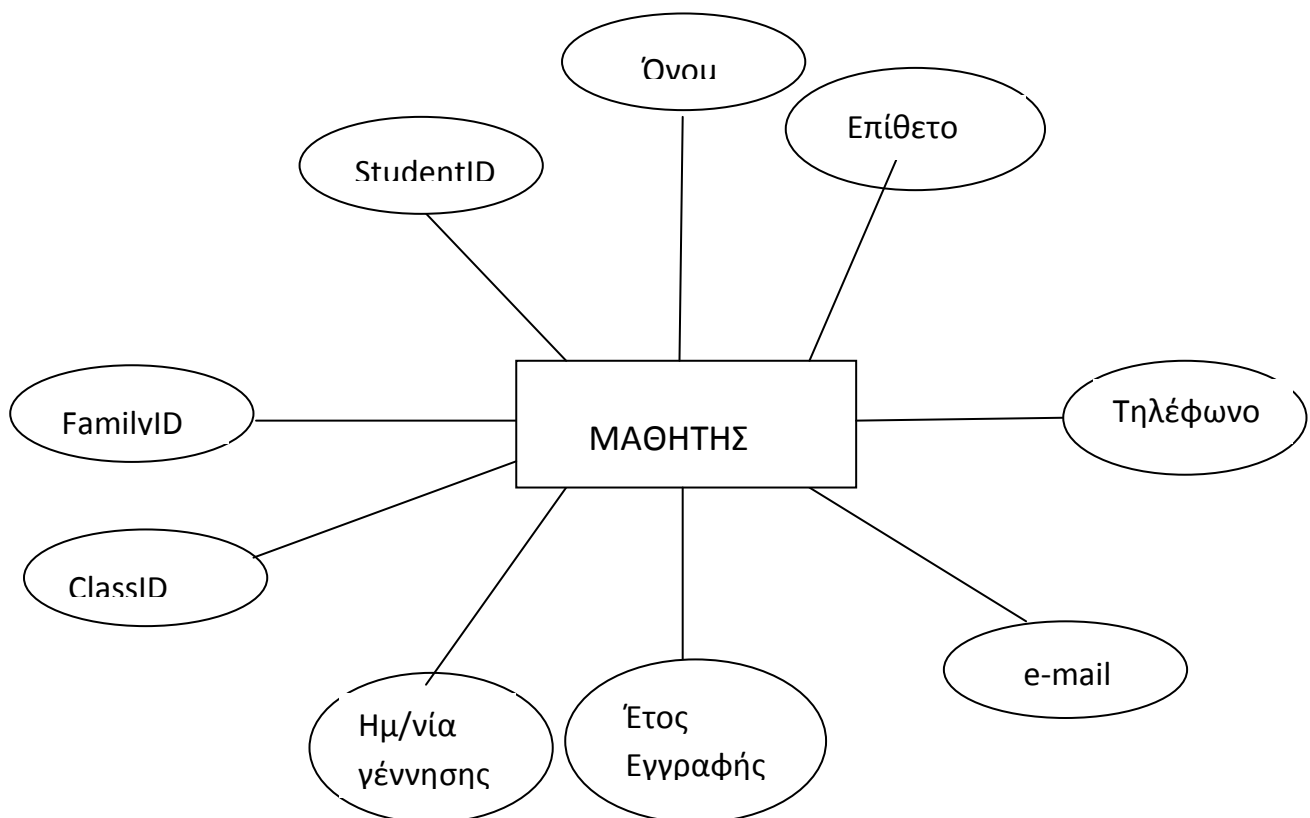
Διάγραμμα Οντοτήτων-Συσχετίσεων (E-R DIAGRAM)



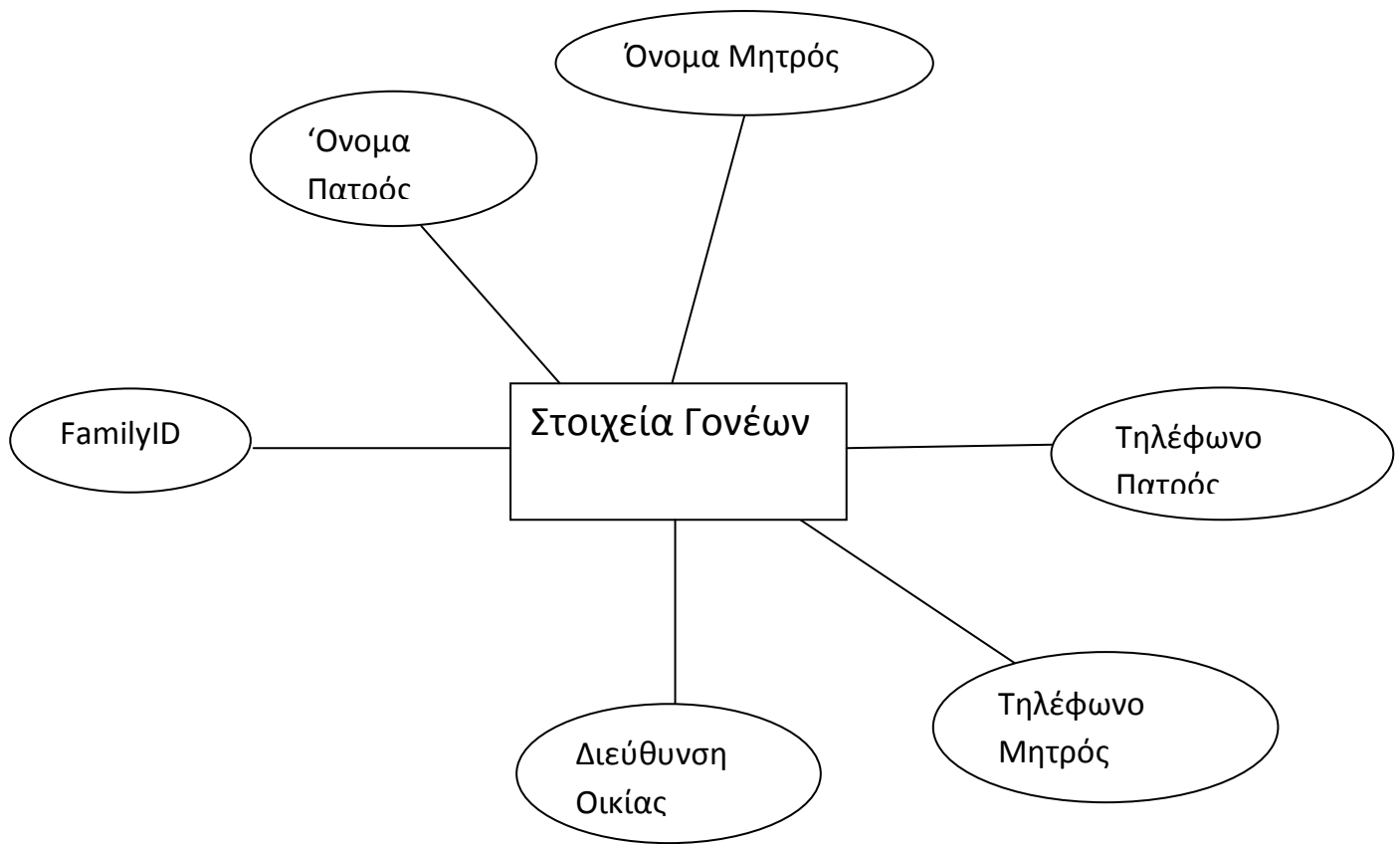
Στο διάγραμμα παρουσιάζεται το διάγραμμα οντοτήτων της βάσης δεδομένων διαχείρισης κέντρου ξένων γλωσσών. Σε αυτό το σχήμα καταλήξαμε προκειμένου να είναι ορθές οι πληθικότητες, όπως δόθηκαν από τις απαιτήσεις, και να είναι εύκολη η πραγματοποίηση των απαραίτητων ειδικών διεργασιών.

Στο διάγραμμα E-R φαίνονται λοιπόν οι σχέσεις που συνδέουν τους οκτώ διαφορετικούς πίνακες της βάσης δεδομένων του Κέντρου Ξένων Γλωσσών. Το διάγραμμα θα ήταν πιο ολοκληρωμένο αν γύρω από κάθε οντότητα αναπαριστάνονταν με ελλείψεις οι ιδιότητές της. Λόγω έλλειψης χώρου, ακολουθούν οι αναπαραστάσεις των βασικών οντοτήτων με τα χαρακτηριστικά τους.

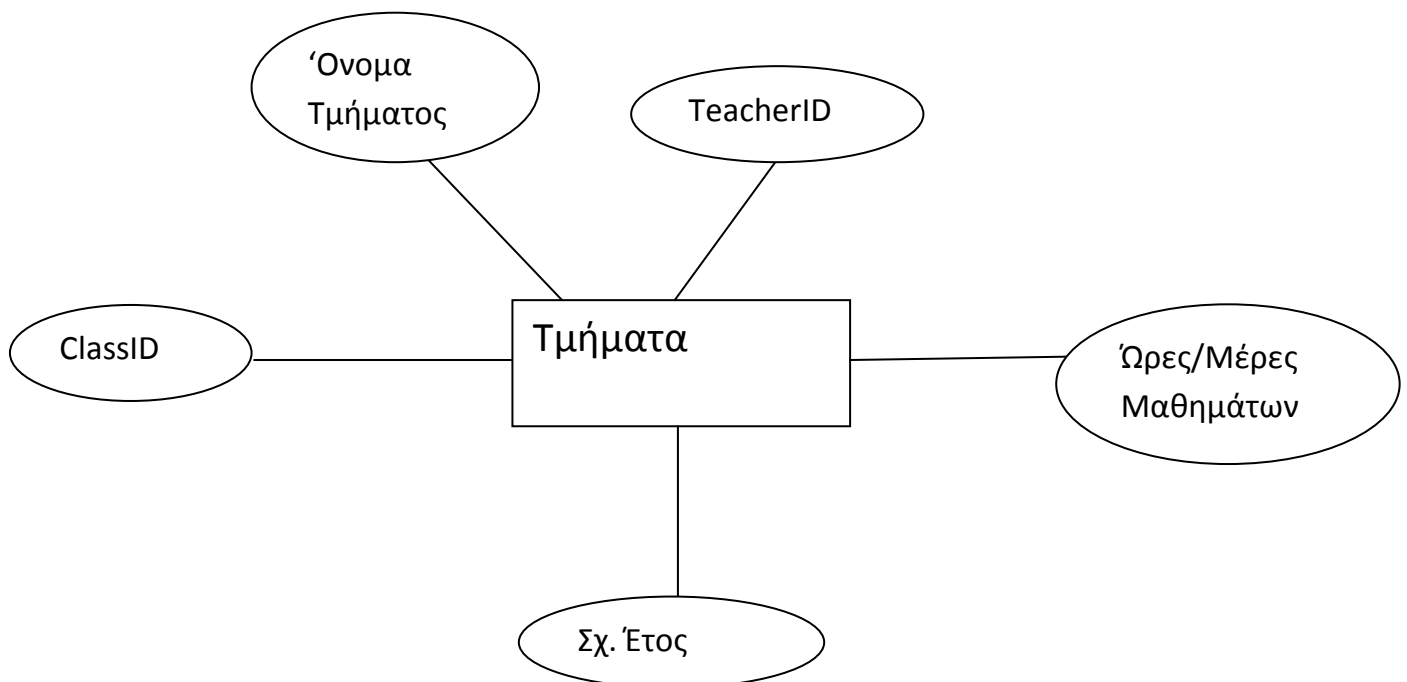
Στο πρώτο αυτό σχήμα παρουσιάζονται οι ιδιότητες που χαρακτηρίζουν κάθε οντότητα «Μαθητή». Πιο συγκεκριμένα, όπως βλέπουμε, κάθε μαθητής έχει ένα μοναδικό StudentID, που αποτελεί και το πρωτεύον κλειδί, και χαρακτηρίζεται και από τις υπόλοιπες ιδιότητες που αναφέρονται μέσα στις ελλείψεις.



Στη συνέχεια, στο σχήμα που ακολουθεί, φαίνονται καθαρά τα χαρακτηριστικά του πίνακα «Στοιχεία Γονέων», με το FamilyID να αποτελεί το πρωτεύον κλειδί του συγκεκριμένου πίνακα.



Ως τελευταίο παράδειγμα, ακολουθούν οι ιδιότητες του πίνακα «Τμήματα». Αντίστοιχα δημιουργούμε τα σχέδια για το διάγραμμα E-R για όλους τους πίνακες της βάσης.



Υλοποίηση /Ανάλυση Πινάκων-Δεδομένων

Σε αυτό το κεφάλαιο παρουσιάζεται η υλοποίηση του σχεδιασμού της βάσης δεδομένων και του αντίστοιχου συστήματος διαχείρισης.

Στο σημείο αυτό γίνεται μια εκτενής αναφορά στους πίνακες της βάσης δεδομένων Κέντρου Ξένων Γλωσσών, καθώς και στα γνωρίσματά τους, τα στοιχεία δηλαδή που κρατούνται στη βάση, καθώς και στους περιορισμούς που τίθενται σε αυτά, στις τιμές δηλαδή που μπορεί να πάρει κάθε γνώρισμα. Στο σημείο αυτό είναι σημαντικό να σημειωθεί ότι ο πίνακας “Students Data” σχετίζεται σχεδόν με όλους τους άλλους μέσω σχέσεων (relationships), οι οποίες θα αναπτυχθούν αργότερα. Ο πίνακας αυτός αποτελεί μια πατρική, κατά κάποιον τρόπο, οντότητα για τις υπόλοιπες, καθώς περιέχει ξένα κλειδιά, τα οποία είναι πρωτεύοντα σε άλλους πίνακες. Αυτό συμβαίνει για την υλοποίηση των σχέσεων μεταξύ των δεδομένων των οντοτήτων της βάσης.

Πίνακας “Μαθητές”

Ο βασικός αυτός πίνακας περιέχει καταρχήν ως γνώρισμα το “**StudentID**”, δηλαδή τον Αριθμό Μητρώου κάθε μαθητή. Ο κωδικός αυτός αποδίδεται σε κάθε φοιτητή με την εγγραφή του στο Κέντρο Ξένων Γλωσσών και είναι μοναδικός για τον καθένα. Για το λόγο αυτό, μπορούμε να τον χρησιμοποιήσουμε ως πρωτεύον κλειδί του πίνακα. Το πεδίο αυτό μπορεί να λάβει μεταβλητού μήκους αλφαριθμητικές τιμές, με μέγιστο μήκος 10 χαρακτήρων. Τα δεδομένα όλου του πίνακα κατά την εξαγωγή τους θα έχουν στοίχιση οριζόντια και αριστερά (η επιλογή έγινε από τις «ιδιότητες πεδίου», στο «Παράδειγμα μορφοποίησης».

Τα γνωρίσματα “**Όνομα**” και “**Επίθετο**” του μαθητή απαιτούν εισαγωγή και αποθηκεύονται στα αντίστοιχα πεδία με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 20 και 50 χαρακτήρων αντίστοιχα. Στο πεδίο “**Ημερομηνία γέννησης**” εισάγεται η ημερομηνία γέννησης του μαθητή. Το πεδίο επιτρέπεται να λάβει μόνο αριθμητικές τιμές με τη μορφή 28/05/99, ενώ γίνεται και έλεγχος για την ορθότητα των τιμών, δηλαδή δεν επιτρέπεται η εισαγωγή μήνα με τιμή μικρότερη από 1 και μεγαλύτερη από 12 κ.ο.κ. Στο πεδίο αυτό δεν είναι απαραίτητο να γίνει εισαγωγή.

Το πεδίο “**Τηλέφωνο**” λαμβάνει μόνο αριθμητικές τιμές, με μέγιστο μήκος 10 χαρακτήρων. Είναι υποχρεωτική η εισαγωγή (δεν επιτρέπονται τιμές null).

Τα πεδία “**e-mail**” και “**FamilyID**” απαιτούν εισαγωγή και αποθηκεύονται στα αντίστοιχα πεδία με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 50 και 10

χαρακτήρων αντίστοιχα, ενώ το “**Έτος Εγγραφής**” απαιτεί εισαγωγή και λαμβάνει μόνο αριθμητικές τιμές με μέγιστο μήκος 4 χαρακτήρων. Τέλος, το πεδίο “**ClassID**” δεν απαιτεί εισαγωγή και λαμβάνει αλφαριθμητικές τιμές με μέγιστο μήκος 30 χαρακτήρων.

Πίνακας “Στοιχεία Γονέων”

Ο πίνακας αυτός περιέχει ως βασικό γνώρισμα το “**FamilyID**”, δηλαδή έναν ξεχωριστό κωδικό για κάθε οικογένεια (κηδεμόνες), που αποτελεί και το πρωτεύον κλειδί του πίνακα. Ο κωδικός αυτός αποδίδεται με την εγγραφή του πρώτου φοιτητή από κάθε οικογένεια με την εγγραφή του στο Κέντρο Ξένων Γλωσσών. Δυσω αδέρφια έχουν ίδιο αυτόν τον κωδικό. Το πεδίο αυτό μπορεί να λάβει μεταβλητού μήκους αλφαριθμητικές τιμές, με μέγιστο μήκος 10 χαρακτήρων. Τα δεδομένα όλου του πίνακα κατά την εξαγωγή τους θα έχουν στοίχιση οριζόντια και αριστερά (η επιλογή έγινε από τις «ιδιότητες πεδίου», στο «Παράδειγμα μορφοποίησης».

Τα γνωρίσματα “**Όνομα Πατρός**” και “**Όνομα Μητρός**” του μαθητή απαιτούν εισαγωγή και αποθηκεύονται στα αντίστοιχα πεδία με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 20 χαρακτήρων. Τα πεδία “**Τηλέφωνο Πατρός**” και “**Τηλέφωνο Μητρός**” λαμβάνουν μόνο αριθμητικές τιμές, με μέγιστο μήκος 10 χαρακτήρων. Δεν είναι υποχρεωτική η εισαγωγή (επιτρέπονται τιμές null).

Το πεδίο “**Διεύθυνση Οικίας**” δεν απαιτεί εισαγωγή και αποθηκεύεται στο αντίστοιχα πεδίο με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 30 χαρακτήρων.

Πίνακας “Τμήματα”

Ο πίνακας αυτός περιέχει ως βασικό γνώρισμα το “**ClassID**”, δηλαδή έναν ξεχωριστό κωδικό για κάθε τμήμα, που αποτελεί και το πρωτεύον κλειδί του πίνακα. Το πεδίο αυτό μπορεί να λάβει μεταβλητού μήκους αλφαριθμητικές τιμές, με μέγιστο μήκος 20 χαρακτήρων. Τα δεδομένα όλου του πίνακα κατά την εξαγωγή τους θα έχουν στοίχιση οριζόντια και αριστερά (η επιλογή έγινε από τις «ιδιότητες πεδίου», στο «Παράδειγμα μορφοποίησης».

Τα γνωρίσματα “**TeacherID**” και “**Ωρες/Μέρες Μαθημάτων**” απαιτούν εισαγωγή και αποθηκεύονται στα αντίστοιχα πεδία με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 20 και 50 χαρακτήρων αντίστοιχα. Το πεδίο “**Όνομα Τμήματος**” δεν απαιτεί εισαγωγή και αποθηκεύεται στο αντίστοιχα πεδίο με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 50 χαρακτήρων.

Πίνακας “Καθηγητές”

Ο πίνακας αυτός περιέχει καταρχήν ως γνώρισμα το “**TeacherID**”, δηλαδή τον κωδικό μητρώου κάθε καθηγητή. Ο κωδικός αυτός αποδίδεται σε κάθε καθηγητή με την πρόσληψή του στο Κέντρο Ξένων Γλωσσών και είναι μοναδικός για τον καθένα. Για το λόγο αυτό, μπορούμε να τον χρησιμοποιήσουμε ως πρωτεύον κλειδί του πίνακα. Το πεδίο αυτό μπορεί να λάβει μεταβλητού μήκους αλφαριθμητικές τιμές, με μέγιστο μήκος 20 χαρακτήρων. Τα δεδομένα όλου του πίνακα κατά την εξαγωγή τους θα έχουν στοίχιση οριζόντια και αριστερά (η επιλογή έγινε από τις «ιδιότητες πεδίου», στο «Παράδειγμα μορφοποίησης»).

Τα γνωρίσματα “**Όνομα**” και “**Επίθετο**” του κάθε καθηγητή απαιτούν εισαγωγή και αποθηκεύονται στα αντίστοιχα πεδία με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 20 και 50 χαρακτήρων αντίστοιχα. Το πεδίο “**Τηλέφωνο**” λαμβάνει μόνο αριθμητικές τιμές, με μέγιστο μήκος 10 χαρακτήρων. Είναι υποχρεωτική η εισαγωγή (δεν επιτρέπονται τιμές null).

Τα πεδία “**e-mail**” και “**Διεύθυνση**” απαιτούν εισαγωγή και αποθηκεύονται στα αντίστοιχα πεδία με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 30 και 50 χαρακτήρων αντίστοιχα, ενώ το πεδίο “**Αρ. Λογαριασμού**” απαιτεί εισαγωγή και λαμβάνει μεταβλητού μήκους αλφαριθμητικές τιμές, με μέγιστο μήκος 30 χαρακτήρων.

Πίνακας “Βαθμολογία”

Σε αυτόν τον πίνακα, βασικό γνώρισμα αποτελεί το “**ID**”. Ο κωδικός αυτός χρησιμοποιείται ως πρωτεύον κλειδί του πίνακα. Το πεδίο αυτό μπορεί να λάβει μεταβλητού μήκους αλφαριθμητικές τιμές, με μέγιστο μήκος 10 χαρακτήρων. Τα δεδομένα όλου του πίνακα κατά την εξαγωγή τους θα έχουν στοίχιση οριζόντια και αριστερά (η επιλογή έγινε από τις «ιδιότητες πεδίου», στο «Παράδειγμα μορφοποίησης»).

Το γνωρίσματα “**CourseID**” δεν απαιτεί εισαγωγή και αποθηκεύεται στα αντίστοιχα πεδία με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 30 χαρακτήρων.

Στο πεδίο “**StudentID**” δεν είναι απαραίτητο να γίνει εισαγωγή και αποθηκεύεται στα αντίστοιχα πεδία με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 10 χαρακτήρων.

Το πεδίο “**Βαθμολογία**” λαμβάνει μόνο αριθμητικές τιμές, με μέγιστο μήκος 4 χαρακτήρων. Δεν είναι υποχρεωτική η εισαγωγή (δεν επιτρέπονται τιμές null).

Πίνακας “Μαθήματα”

Σε αυτόν τον πίνακα, βασικό γνώρισμα αποτελεί το “**CourseID**”. Ο κωδικός αυτός χρησιμοποιείται ως πρωτεύον κλειδί του πίνακα. Το πεδίο αυτό μπορεί να λάβει μεταβλητού μήκους αλφαριθμητικές τιμές, με μέγιστο μήκος 10 χαρακτήρων. Τα δεδομένα όλου του πίνακα κατά την εξαγωγή τους θα έχουν στοίχιση οριζόντια και αριστερά (η επιλογή έγινε από τις «ιδιότητες πεδίου», στο «Παράδειγμα μορφοποίησης»).

Το γνώρισμα “**ClassID**” απαιτεί εισαγωγή και αποθηκεύεται στα αντίστοιχα πεδία με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 30 χαρακτήρων.

Στο πεδίο “**Μάθημα**” δεν είναι απαραίτητο να γίνει εισαγωγή και αποθηκεύεται στα αντίστοιχα πεδία με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 30 χαρακτήρων.

Πίνακας “Πληρωμές”

Σε αυτόν τον πίνακα, βασικό πεδίο αποτελεί το “**ID**”. Ο κωδικός αυτός χρησιμοποιείται ως πρωτεύον κλειδί του πίνακα. Το πεδίο αυτό μπορεί να λάβει μεταβλητού μήκους αλφαριθμητικές τιμές, με μέγιστο μήκος 10 χαρακτήρων. Τα δεδομένα όλου του πίνακα κατά την εξαγωγή τους θα έχουν στοίχιση οριζόντια και αριστερά (η επιλογή έγινε από τις «ιδιότητες πεδίου», στο «Παράδειγμα μορφοποίησης»).

Το γνώρισμα “**StudentD**” απαιτεί εισαγωγή και αποθηκεύεται στα αντίστοιχα πεδία με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 50 χαρακτήρων.

Στο πεδίο “**Σχ. Έτος**” είναι απαραίτητο να γίνει εισαγωγή και λαμβάνει αλφαριθμητικές τιμές, με μέγιστο μήκος 20 χαρακτήρων.

Τα υπόλοιπα πεδία δεν απαιτούν εισαγωγή, αποτελούν λογικά πεδία, στα οποία μπορεί να αποθηκευτεί μια από τις δύο διαθέσιμες τιμές ναι/όχι, για να υπάρχει ενημέρωση αν έχει πληρωθεί η εγγραφή και ο εκάστοτε μήνας.

Πίνακας “Απουσίες”

Σε αυτόν τον πίνακα, βασικό πεδίο αποτελεί το “**ID**”. Ο κωδικός αυτός χρησιμοποιείται ως πρωτεύον κλειδί του πίνακα. Το πεδίο αυτό μπορεί να λάβει μεταβλητού μήκους αλφαριθμητικές τιμές, με μέγιστο μήκος 10 χαρακτήρων. Τα δεδομένα όλου του πίνακα κατά την εξαγωγή τους θα έχουν στοίχιση οριζόντια και αριστερά (η επιλογή έγινε από τις «ιδιότητες πεδίου», στο «Παράδειγμα μορφοποίησης»).

Το γνωρίσματα “**StudentD**” απαιτεί εισαγωγή και αποθηκεύεται στα αντίστοιχα πεδία με μεταβλητού μήκους αλφαριθμητικό με μέγιστο μήκος 50 χαρακτήρων.

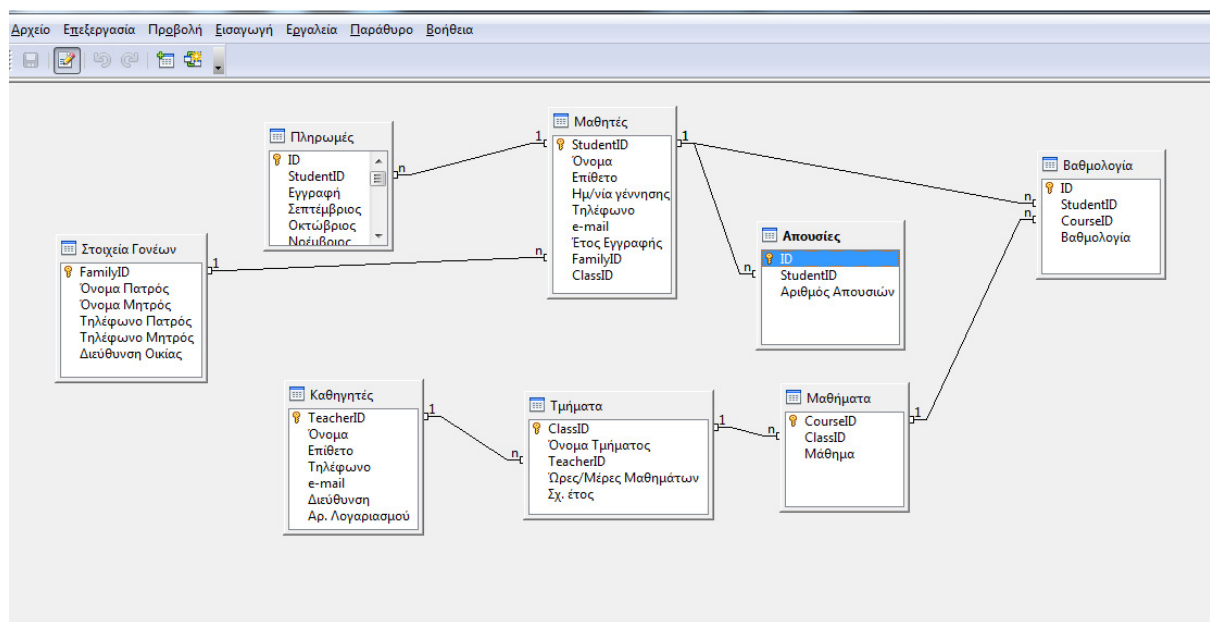
Στο πεδίο “**Αριθμός Απουσιών**” δεν είναι απαραίτητο να γίνει εισαγωγή και λαμβάνει μόνο αριθμητικές τιμές, με μέγιστο μήκος 50 χαρακτήρων.

Σχολιασμός των Σχέσεων των Πινάκων

Μετά την ολοκλήρωση της δημιουργίας των πινάκων, ορίσαμε τις **συσχετίσεις ή σχέσεις** μεταξύ τους.

Όπως έχει ήδη αναφερθεί, για να ανοίξουμε και να δουλέψουμε το παράθυρο **Σχέσεις**, πατάμε **Εργαλεία** στη Γραμμή Μενού και στη συνέχεια στην αντίστοιχη γραμμή **Σχέσεις** που δίδεται ως επιλογή. Αυτόματα ανοίγει νέο παράθυρο – με το όνομα Προσθήκη πινάκων. Στο πλαίσιο αυτό, παρατίθενται όλα τα ονόματα των πινάκων της βάσης δεδομένων. Επιλέγουμε έναν-έναν τους πίνακες που μας ενδιαφέρει να συνδεθούν μεταξύ τους. Για να δημιουργηθεί μια νέα σχέση, σύρουμε το πεδίο από τον πρωτεύοντα πίνακα (δηλαδή τον πίνακα από τον οποίο παίρνουμε τις μοναδικές τιμές) πάνω στο αντίστοιχο πεδίο του συσχετισμένου πίνακα.

Στην εικόνα που ακολουθεί, φαίνεται η διασύνδεση των πινάκων της Βάσης Δεδομένων του Κέντρου Ξένων Γλωσσών σε περιβάλλον *OpenOffice Base*.



Εικόνα 23 : Διασύνδεση των πινάκων της Βάσης Δεδομένων

Πιο συγκεκριμένα, ο πίνακας «Μαθητές» έχει μια σχέση 1:N με τον πίνακα «Πληρωμές», αφού κάθε μαθητής πρέπει να πληρώσει για την εγγραφή, αλλά και κάθε σχολικό μήνα, ακόμα και για μια ακόμα ξένη γλώσσα (όταν προστεθεί αργότερο-το παράδειγμα έχει μόνο τη διδασκαλία της Αγγλικής). Σχέση 1:N είναι και η σχέση του πίνακα «Μαθητές» με τον πίνακα «Βαθμολογία», αφού κάθε μαθητής μπορεί να βαθμολογηθεί σε περισσότερα από ένα μαθήματα. Την ίδια στιγμή, η σχέση του πίνακα «Μαθητές» με τον πίνακα «Στοιχεία Γονέων» είναι σχέση N:1, με δεδομένο ότι πολλοί μαθητές (π.χ. τα αδέρφια) μπορεί να έχουν τα ίδια στοιχεία γονέων. Τέλος, σχέση τύπου 1:N παρουσιάζει ο συγκεκριμένος πίνακας και με τον πίνακα «Απουσίες».

Ο πίνακας «Μαθήματα» έχει σχέση τύπου 1:N με τον πίνακα «Βαθμολογία», ενώ τύπου N:1 με τον πίνακα «Τμήματα», αφού κάθε τμήμα μπορεί να έχει πολλά, διαφορετικά μαθήματα. Τέλος, ο πίνακας «Καθηγητές» εμφανίζει σχέση τύπου 1:N με τον πίνακα «Τμήματα», διότι κάθε καθηγητής μπορεί να διδάσκει σε περισσότερα από ένα τμήματα.

Παραδείγματα Ερωτημάτων

Τα ερωτήματα δημιουργήθηκαν για να εξάγουν και να προβάλλουν συγκεκριμένα δεδομένα μέσα από τους πίνακες της βάσης δεδομένων. Ανάλογα με το τι χρειαζόταν σε κάθε περίπτωση για να προβληθεί σαν αποτέλεσμα, δημιουργήθηκε και το κατάλληλο ερώτημα. Στη συνέχεια, παρουσιάζουμε τρία ενδεικτικά παραδείγματα ερωτημάτων, που θα μπορούσε να απαντά η συγκεκριμένη βάση δεδομένων. Κάθε ένα ερώτημα παρουσιάζεται αναλυτικά παρακάτω.

Ερώτημα «Αριστούχοι»

Το ερώτημα «**Αριστούχοι**» δημιουργήθηκε για να προβάλλει τα δεδομένα των μαθητών που άριστευσαν σε συγκεκριμένα μάθημα. Οι πίνακες από τους οποίους ελήφθησαν τα δεδομένα είναι οι πίνακες «Μαθητές», «Μαθήματα» και «Βαθμολογία».

Τα πεδία που θέλουμε να εμφανίζονται είναι τα «StudentID», «Όνομα» και «Επίθετο» μαθητή, το «CourseID» και η αντίστοιχη βαθμολογία του μαθήματος. Κριτήρια για το ερώτημα, όπως φαίνεται και στην παραπάνω εικόνα, αποτελούν ο κωδικός αριθμός του μαθήματος και το κριτήριο «>85» που θέλουμε να ισχύει για το πεδίο «Βαθμολογία». Πατώντας στο εικονίδιο «Εκτέλεση ερωτήματος» εμφανίζονται οι μαθητές που άριστευσαν-είχαν βαθμό μεγαλύτερο από 85 (>85)-στο μάθημα με κωδικό «En_GR_A1», πληρούν δηλαδή τα κριτήρια που θέσαμε.

The screenshot shows the Microsoft Access interface with three tables defined:

- Μαθητές**: Primary key StudentID, fields include Όνομα, Επίθετο, Ημ/νία γέννησης, Τηλέφωνο, e-mail, Έτος Εγγραφής, FamilyID, ClassID.
- Μαθήματα**: Primary key CourseID, fields include ClassID, Μάθημα.
- Βαθμολογία**: Primary key ID, fields include StudentID, CourseID, Βαθμολογία.

Relationships are shown between tables: one-to-many between Μαθητές and Βαθμολογία, and one-to-many between Μαθήματα and Βαθμολογία.

Πεδίο	StudentID	Όνομα	Επίθετο	CourseID	Βαθμολογία
Ψευδώνυμο					
Πίνακας	Μαθητές	Μαθητές	Μαθητές	Μαθήματα	Βαθμολογία
Ταξινόμηση					
Ορατό	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Συνάρτηση					
Κριτήριο				'En_GR_A1'	> 85
Είτε					

Εικόνα 24 : Εμφάνιση των πεδίων που χρειάζονται για το ερώτημα «Αριστούχοι»

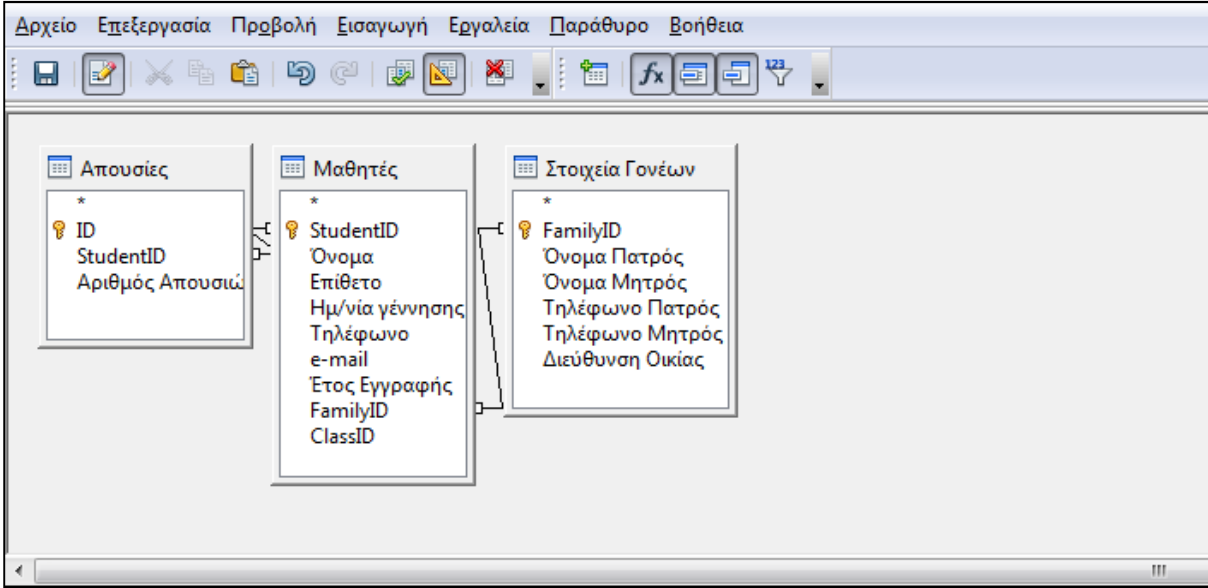
StudentID	Όνομα	Επίθετο	CourseID	Βαθμολογία
En_3	Γεράσιμος	Διαμαντάκος	En_GR_A1	89

Πεδίο	StudentID	Όνομα	Επίθετο	CourseID	Βαθμολογία
Ψευδώνυμο					
Πίνακας	Μαθητές	Μαθητές	Μαθητές	Μαθήματα	Βαθμολογία
Ταξινόμηση					
Ορατό	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Συνάρτηση					
Κριτήριο				'En_GR_A1'	> 85

Εικόνα 25 : Εμφάνιση των αποτελεσμάτων στο ερώτημα «Απιστούχι»

Ερώτημα «Αδικαιολόγητες Απουσίες»

Το ερώτημα «Αδικαιολόγητες Απουσίες» δημιουργήθηκε για να προβάλει τα δεδομένα των μαθητών που έχουν απουσιάσει αρκετές φορές, καθώς και τα στοιχεία επικοινωνίας των γονέων τους, προκειμένου να ενημερωθούν. Οι πίνακες από τους οποίους ελήφθησαν τα δεδομένα είναι οι πίνακες «Μαθητές», «Απουσίες» και «Στοιχεία Γονέων».



Πεδίο	StudentID	Όνομα	Επίθετο	Αριθμός Απουσιών	Τηλέφωνο Πατρός	Τηλέφωνο Μητρός
Ψευδώνυμο						
Πίνακας	Μαθητές	Μαθητές	Μαθητές	Απουσίες	Στοιχεία Γονέων	Στοιχεία Γονέων
Ταξινόμηση						
Ορατό	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Συνάρτηση						
Κριτήριο				>= 2		
Είτε						

Εικόνα 26 : Εμφάνιση των πεδίων που χρειάζονται για το ερώτημα «Αδικαιολόγητες Απουσίες»

Τα πεδία που θέλουμε να εμφανίζονται είναι τα «StudentID», «Όνομα» και «Επίθετο» μαθητή, ο «Αριθμός Απουσιών» και τα «Τηλέφωνο Πατρός» και «Τηλέφωνο Μητρός». Κριτήριο για το ερώτημα, όπως φαίνεται και στην παραπάνω εικόνα, αποτελεί ο αριθμός των απουσιών. Στο συγκεκριμένο παράδειγμα έχουμε ως κριτήριο ο αριθμός των απουσιών να είναι μεγαλύτερος ή ίσος με δύο (≥ 2). Πατώντας στο εικονίδιο «Εκτέλεση ερωτήματος» εμφανίζονται οι μαθητές που έχουν λήψει από δυο φορές και πάνω.

StudentID	Όνομα	Επίθετο	Αριθμός Απουσιών	Τηλέφωνο Πατρός	Τηλέφωνο Μητρός
En_4	Γεωργία	Μαύρου	2		2105896548
En_8	Βασίλειος	Παππάς	2	6985478521	

Εικόνα 27:Εμφάνιση των αποτελεσμάτων-απαντήσεων στο ερώτημα «Αδικαιολόγητες Απουσίες»

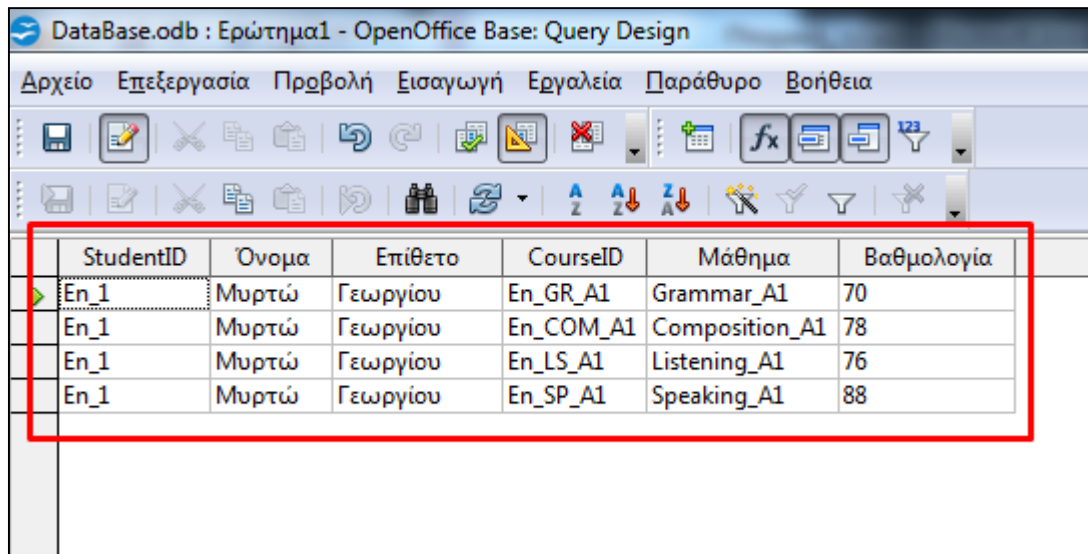
Ερώτημα «Βαθμολογία Τριμήνου»

Με το τρίτο ερώτημα, «**Βαθμολογία Τριμήνου**», δίνεται η δυνατότητα να συγκεντρώσουμε σε μια έκθεση τις βαθμολογίες του κάθε μαθητή σε όλα τα μαθήματα, για να κατατεθεί βαθμολογία για το αντίστοιχο τρίμηνο. Οι πίνακες από τους οποίους ελήφθησαν τα δεδομένα είναι οι πίνακες «Μαθητές», «Μαθήματα» και «Βαθμολογία». Τα πεδία που θέλουμε να εμφανίζονται είναι τα «StudentID», «Όνομα» και «Επίθετο» μαθητή, το «CourseID» , το «Μάθημα» και η αντίστοιχη «Βαθμολογία».

Πεδίο	StudentID	Όνομα	Επίθετο	CourseID	Μάθημα	Βαθμολογία
Ψευδώνυμο						
Πίνακας	Μαθητές	Μαθητές	Μαθητές	Βαθμολογία	Μαθήματα	Βαθμολογία
Ταξινόμηση						
Ορατό	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Συνάρτηση						
Κριτήριο	'Επ_1'					

Εικόνα 28:Εμφάνιση των πεδίων που χρειάζονται για το ερώτημα «Βαθμολογία Τριμήνου»

Στο ερώτημα αυτό μοναδικό κριτήριο αποτελεί ο κωδικός «StudentID», του οποίου θέλουμε να συγκεντρώσουμε τις πληροφορίες. Κάνουμε το ίδιο για κάθε μαθητή.



The screenshot shows the OpenOffice Base Query Design interface. The title bar reads "DataBase.odb : Ερώτημα1 - OpenOffice Base: Query Design". The menu bar includes "Αρχείο", "Επεξεργασία", "Προβολή", "Εισαγωγή", "Εργαλεία", "Παράθυρο", and "Βοήθεια". Below the menu is a toolbar with various icons. The main area displays a table with the following data:

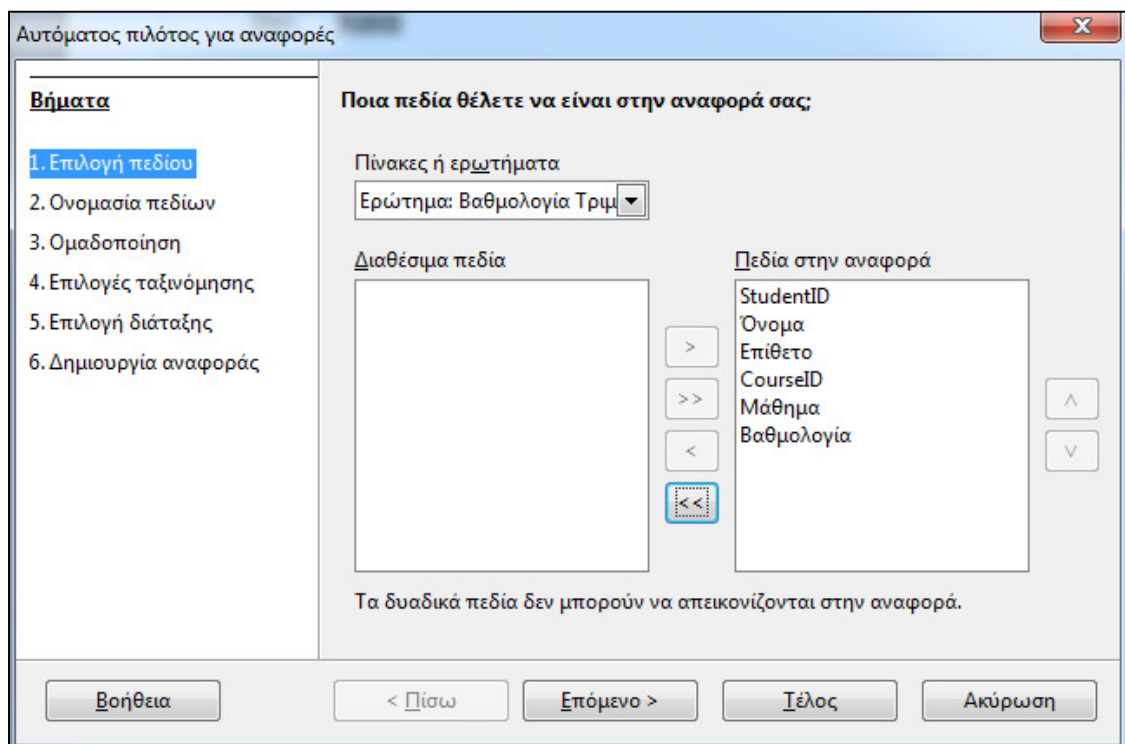
StudentID	Όνομα	Επίθετο	CourseID	Μάθημα	Βαθμολογία
En_1	Μυρτώ	Γεωργίου	En_GR_A1	Grammar_A1	70
En_1	Μυρτώ	Γεωργίου	En_COM_A1	Composition_A1	78
En_1	Μυρτώ	Γεωργίου	En_LS_A1	Listening_A1	76
En_1	Μυρτώ	Γεωργίου	En_SP_A1	Speaking_A1	88

Εικόνα 29:Εμφάνιση των αποτελεσμάτων-απαντήσεων στο ερώτημα «Βαθμολογία τριμήνου»

Παράδειγμα Αναφοράς

Οι αναφορές, όπως έχουμε ήδη αναφέρει, είναι εργαλεία που μας βοηθούν να τυπώσουμε επιλεγμένα στοιχεία της βάσης δεδομένων μας με καλαίσθητο και λειτουργικό τρόπο. Στο σημείο αυτό, θα παρουσιάσουμε ένα παράδειγμα δημιουργίας αναφοράς, με τη χρήση οδηγού. Πιο συγκεκριμένα, θα δημιουργήσουμε μια αναφορά, στηριζόμενοι στο τελευταίο ερώτημα με τίτλο «Βαθμολογία Τριμήνου», για να τυπώσουμε τους αντίστοιχους ελέγχους τριμήνου.

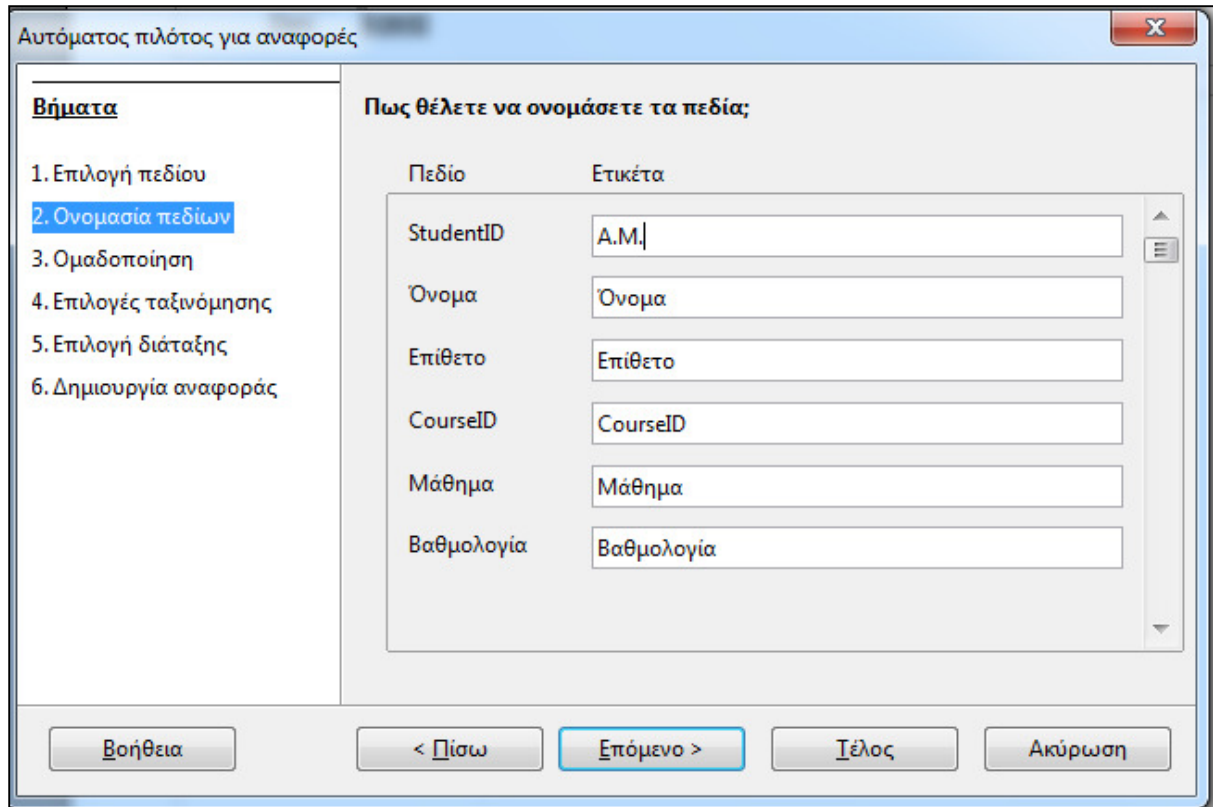
Στις εικόνες που ακολουθούν, φαίνονται τα βήματα που ακολουθήθηκαν για να δημιουργηθεί η αναφορά.



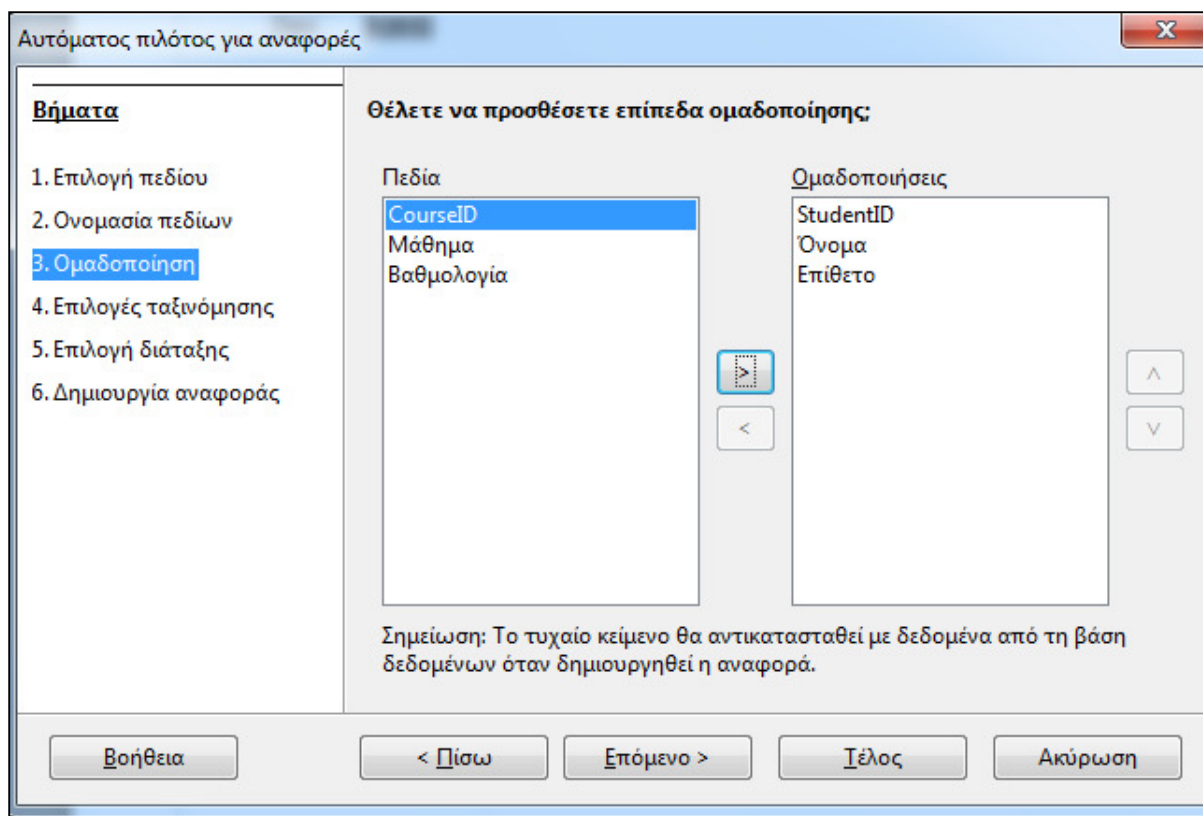
Εικόνα 30: Προβολή του αρχικού πλαισίου όταν επιλεγεί η χρήση πιλότου για τη δημιουργία αναφοράς

Όπως φαίνεται στην Εικόνα 30, επιλέξαμε όλα τα πεδία που συμπεριλαμβάνονται στο συγκεκριμένο ερώτημα και θέλουμε να εμφανίζονται στην αναφορά.

Στη συνέχεια, μας δόθηκε η δυνατότητα να αλλάξουμε το όνομα κάποιου πεδίου και αξιοποιήσαμε αυτή τη δυνατότητα μετονομάζοντας (για τον τρόπο προβολής στην αναφορά) το πεδίο StudentID σε A.M. (βλ. Εικόνα 31).

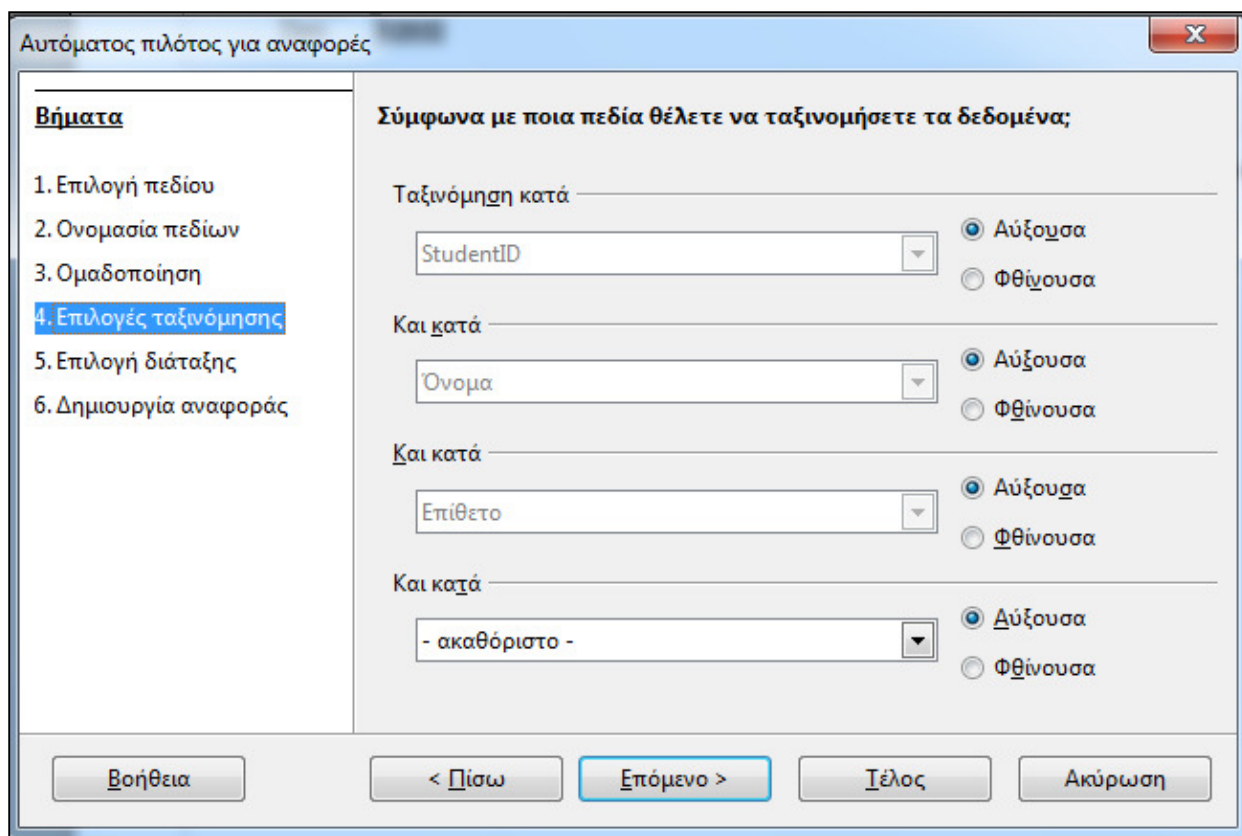


Εικόνα 31: Προβολή του πλαισίου που δίνει τη δυνατότητα μεταβολής των αρχικών ονομάτων των πεδίων



Εικόνα 32: Επιλογή των πεδίων που θα ομαδοποιηθούν

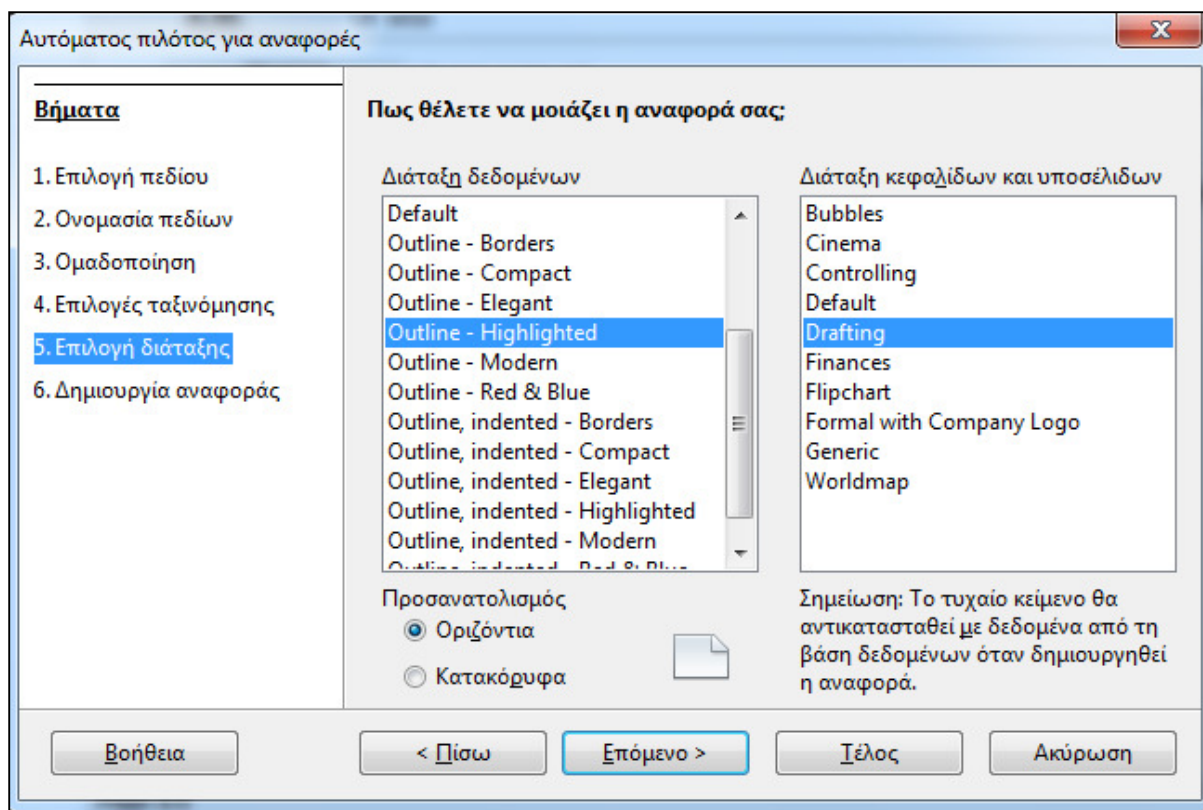
Όπως φαίνεται στην Εικόνα 32 επιλέξαμε τα πεδία StudentID (που θα εμφανίζεται ως Α.Μ.), Όνομα και Επίθετο για να ομαδοποιηθούν. Αυτό θα έχει ως αποτέλεσμα, όπως θα δούμε παρακάτω (Εικόνα 36) τα στοιχεία αυτά να ξεχωρίζουν από τα υπόλοιπα, στην κορυφή του εγγράφου.



Εικόνα 33: Προβολή του πλαισίου που δίνει τη δυνατότητα επιλογής του τρόπου ταξινόμησης των δεδομένων με βάση τα διαφορετικά πεδία

Στο τέταρτο πλαίσιο διαλόγου του Αυτόματου πιλότου για αναφορές (Εικόνα 33) μας επιτρέπει να καθορίσουμε τη σειρά ταξινόμησης των εγγραφών που θα εμφανίζονται στην έκθεση.

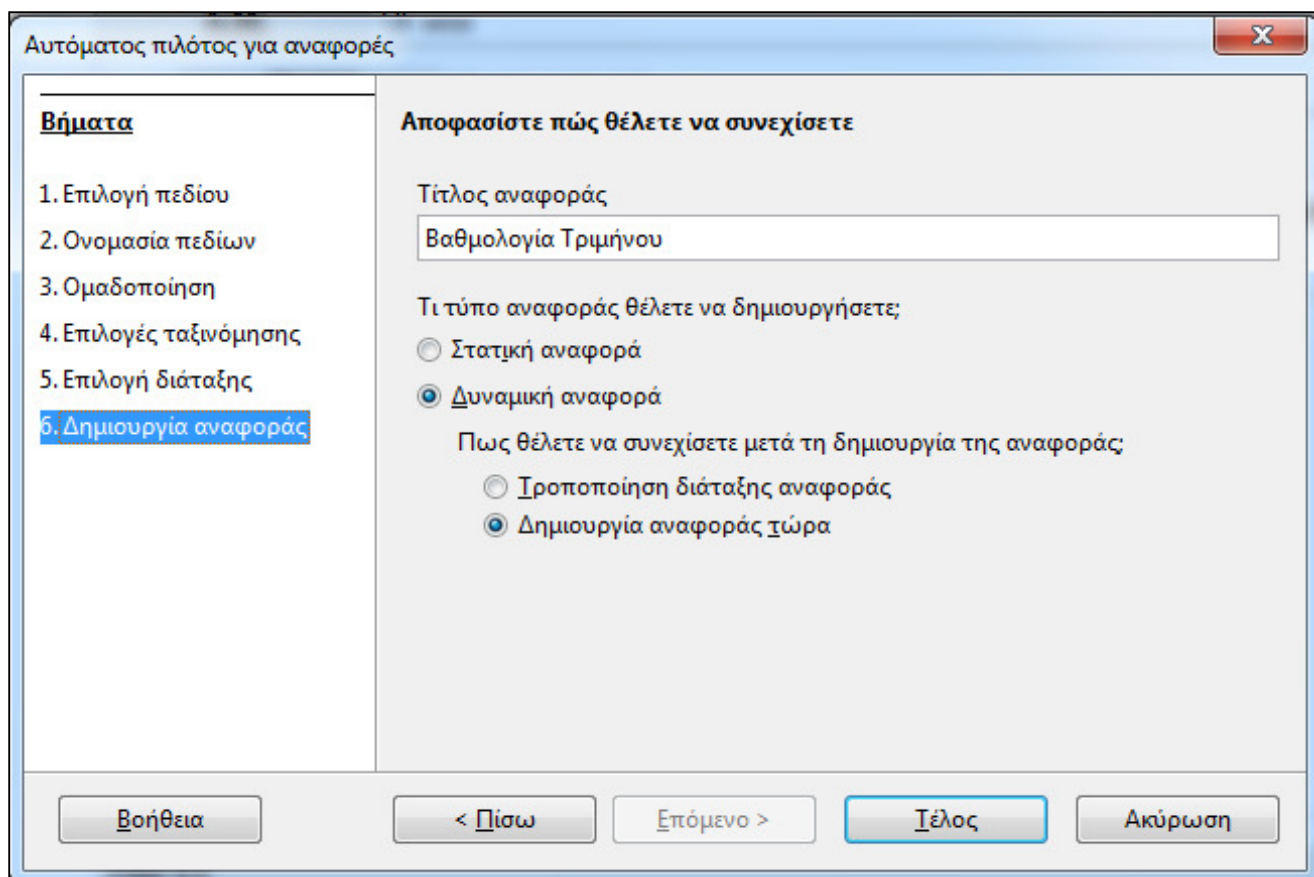
Σημαντικό είναι να τονίσουμε ότι οι διάφορες ομάδες που καθορίσαμε για την αναφορά μας θα ταξινομηθούν αυτόματα ως προς τα πεδία ομαδοποίησης-έτσι σε αυτό το πλαίσιο διαλόγου καθορίσουμε ουσιαστικά τον τρόπο ταξινόμησης των εσωτερικών στοιχείων (των λεπτομερειών) κάθε ομάδας.



Εικόνα 34: Προβολή του πλαισίου που δίνει τη δυνατότητα επιλογής της διάταξης των δεδομένων

Όπως έχει ήδη αναφερθεί, και φαίνεται στην εικόνα 34, στο πέμπτο πλαίσιο διαλόγου επιλέγουμε τη γενική διάταξη της έκθεσής μας και τον προσανατολισμό της (οριζόντιο ή κατακόρυφο). Στη συγκεκριμένη περίπτωση επιλέξαμε οριζόντιο προσανατολισμό, καθώς και τις επιλογές Outline-Highlighted για τη διάταξη των δεδομένων και Drafting για τη διάταξη των κεφαλίδων και υποσέλιδων.

Η επιλογή έγινε μετά από πολλές δοκιμές και έλεγχο με προεπισκόπηση.



Εικόνα 35: Προβολή του έκτου πλαισίου διαλόγου

Στο έκτο και τελευταίο πλαίσιο διαλόγου που φαίνεται στην εικόνα 35 καθορίζουμε αρχικά το όνομα της αναφοράς. Στη συγκεκριμένη περίπτωση ονομάσαμε την αναφορά «Βαθμολογία Τριμήνου» και στη συνέχεια επιλέξαμε τον τύπο της αναφοράς – πρόκειται για μια δυναμική αναφορά, αφού θέλουμε να υπάρχει η δυνατότητα τροποποίησης. Τέλος, επιλέξαμε να δημιουργήσουμε την αναφορά άμεσα.

Έτσι, ακολουθώντας τα βήματα αυτά, καταλήξαμε στην παρακάτω αναφορά, που αποτελεί υπόδειγμα ελέγχου βαθμολογίας τριμήνου (Εικόνα 36).



September 29, 2015

A.M. En_1

Όνομα Μυρτώ

Επίθετο Γεωργίου

<i>CourseID</i>	<i>Μάθημα</i>	<i>Βαθμολογία</i>
En_GR_A1	Grammar_A1	70
En_COM_A1	Composition_A1	78
En_LS_A1	Listening_A1	76
En_SP_A1	Speaking_A1	88

Εικόνα 36: Η αναφορά που δημιούργησε η Base με βάση τις επιλογές μας

Βιβλιογραφία

1. <http://dide.flo.sch.gr/Plinet/Tutorials/Tutorials-DataBasesTheory.html>
2. Εισαγωγή στη Σύγχρονη Επιστήμη των Υπολογιστών, Les Goldschlager & Andrew Lister, Εκδόσεις Δίαυλος
3. Εφαρμογή Πληροφορικής Υπολογιστών Α',Β',Γ' Ενιαίου Λυκείου, ΟΕΔΒ (Ε.ΓΙΑΚΟΥΜΑΚΗΣ, Κ. ΓΚΥΡΤΗΣ, Β.Σ. ΜΠΕΛΕΣΙΩΤΗΣ, Π.ΞΥΝΟΣ, Ν.ΣΤΕΡΓΙΟΠΟΥΛΟΥ-ΚΑΛΑΝΤΖΗ)
4. http://www.ct.aegean.gr/people/vkavakli/databases_winter_2011/slides/lecture2.pdf
5. <https://wiki.documentfoundation.org/images/e/e8/BH40-BaseHandbook.pdf>
6. <https://euclid.ee.duth.gr/courses/old/2006-07/Databases/DraftSlides/LecDB03-ER2Relational-Large.pdf>