



**ΑΛΕΞΑΝΔΡΕΙΟ Τ.Ε.Ι. ΘΕΣΣΑΛΟΝΙΚΗΣ ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΚΩΝ
ΕΦΑΡΜΟΓΩΝ ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ**

Εκπαιδευτικά Περιβάλλοντα Διδασκαλίας Σχεσιακής Άλγεβρας και Σχεσιακού Λογισμού

Του φοιτητή Χρήστου Τσαμαδού με Α.Μ. 04/2641 Επιβλέπων καθηγητής Δέρβος Δημήτριος

ΠΡΟΛΟΓΟΣ

Η παρούσα πτυχιακή αποτελεί ένα μικρό εκπαιδευτικό εγχειρίδιο για προγράμματα εκμάθησης σχεσιακής άλγεβρας όπως επίσης και ένα συγκριτικό τεστ αυτών .

ΠΕΡΙΛΗΨΗ-SUMMARY

Η παρούσα πτυχιακή εργασία εξετάζει τρία περιβάλλοντα εκμάθησης σχεσιακής άλγεβρας, τις δυνατότητες και τις χρηστικές τους διεπαφές. Στόχο αποτελεί η συγκριτική τους αξιολόγηση όσον αφορά στο εύρος των δυνατοτήτων τους και την ευκολία στη χρήση τους. Για το σκοπό αυτό, χρησιμοποιούνται μία δεδομένη σχεσιακή βάση και ένα κοινό σύνολο αιτημάτων σχεσιακής άλγεβρας. Από τη σύγκριση προκύπτει το περιβάλλον εκείνο το οποίο θα μπορούσε να αποτελέσει την καλύτερη επιλογή ώστε να χρησιμοποιηθεί για τη διδασκαλία του θέματος της σχεσιακής άλγεβρας σε ένα τυπικό εισαγωγικό μάθημα Βάσεων Δεδομένων.

The present final year thesis considers three environments supporting relational algebra operations on a given relational database schema. Emphasis is given on the spectrum of operations supported, as well as the corresponding user interfaces. The three environments are compared and they are rated on the basis of their potential and ease of use. In this respect, one common relational database schema and a common set of user queries (formulated in relational algebra syntax) are used. As a result, a clear winner emerges, comprising the best candidate for adoption in teaching relational algebra as a topic in a typical introductory database technology course.

Ένα μεγάλο ευχαριστώ στην οικογένεια μου, τους φίλους μου, το κ. Δέρβο και τους καθηγητές που τόσα χρόνια με βοήθησαν σε αυτό το στάδιο της ζωής μου.

ΕΥΡΕΤΗΡΙΟ ΠΕΡΙΕΧΟΜΕΝΩΝ

| | |
|---|--|
| Εισαγωγή.....σελ. 8 | |
| 1. Κεφάλαιο 1.....σελ. 10 | |
| i. Εισαγωγή.....σελ. 10 | |
| ii. Το σχεσιακό μοντέλο.....σελ. 10 | |
| iii. Σχεσιακή άλγεβρα.....σελ. 13 | |
| iv. Σχεσιακός λογισμός πλειάδων.....σελ. 14 | |
| v. Σχεσιακός λογισμός πεδίου.....σελ. 16 | |
| vi. Σχέσεις παραδειγμάτων.....σελ. 17 | |
| vii. Ερωτήματα.....σελ. 21 | |
| viii. Προγράμματα.....σελ. 23 | |
| 2. Κεφάλαιο 2 | |
| i. Εισαγωγή-winLear.....σελ. 24 | |
| ii. Λειτουργικότητα και Βασικές Εντολές.....σελ. 25 | |
| iii. Δημιουργία του σχεσιακού σχήματος και της βάσης.....σελ. 27 | |
| iv. Εκτέλεση ερωτημάτων.....σελ. 30 | |
| 3. Κεφάλαιο 3σελ. 34 | |
| i. Εισαγωγή-winRDBI.....σελ. 34 | |
| ii. Λειτουργικότητα και Βασικές Εντολές.....σελ. 35 | |
| iii. Εκτέλεση ερωτημάτωνσελ. 41 | |
| i. Δημιουργία ερωτημάτων σχεσιακής άλγεβρας.....σελ. 41 | |
| ii. Εισαγωγικά ερωτήματα στη σχεσιακή άλγεβρα.....σελ. 42 | |
| iii. Εκτέλεση ερωτημάτων στη σχεσιακή άλγεβρα.....σελ. 44 | |
| iv. Σχεσιακός λογισμός πλειάδων και εκτέλεση ερωτημάτων.....σελ. 51 | |
| v. Σχεσιακός λογισμός πεδίου και εκτέλεση ερωτημάτων.....σελ. 54 | |
| vi. Δημιουργία και εκτέλεση ερωτημάτων SQL.....σελ. 56 | |
| 4. Κεφάλαιο 4.....σελ.58 | |
| i. Εισαγωγή-Relational.....σελ.58 | |
| ii. Λειτουργικότητα και Βασικές Εντολέςσελ. 59 | |
| iii. Εκτέλεση ερωτημάτων σχεσιακής άλγεβρας.....σελ. 62 | |

| | |
|------------------------------|---------|
| 5. Κεφάλαιο 5.....σελ.66 | σελ.66 |
| i. Εισαγωγή.....σελ. 66 | σελ. 66 |
| ii. Συμπεράσματα.....σελ. 68 | σελ. 68 |
| i. winRDBI.....σελ. 68 | σελ. 68 |
| ii. winLeap.....σελ.68 | σελ.68 |
| iii. Relationalσελ. 69 | σελ. 69 |
| 6. Βιβλιογραφία.....σελ. 71 | σελ. 71 |
| 7. URLs.....σελ. 72 | σελ. 72 |

ΕΥΡΕΤΗΡΙΟ ΕΙΚΟΝΩΝ-ΠΙΝΑΚΩΝ

| | |
|---|----|
| Εικόνα 1, αναπαράσταση σχέσης.....σελ. | 12 |
| Εικόνα 2, αρχική οθόνη winLeap.....σελ. | 24 |
| Εικόνα 3, αρχική οθόνη winRDBI.....σελ. | 34 |
| Εικόνα 4, επιλογή βάσης.....σελ. | 35 |
| Εικόνα 5, απεικόνιση βάσης σε rdb.....σελ. | 36 |
| Εικόνα 6, απεικόνιση βάσης σε xml.....σελ. | 36 |
| Εικόνα 7, παράθυρο επιλογής τύπου ερωτήματος.....σελ. | 38 |
| Εικόνα 8, query pane τύπου σχεσιακής άλγεβρας.....σελ. | 38 |
| Εικόνα 9, screen shot του winRDBI με όλα τα query panes.....σελ. | 39 |
| Εικόνα 10, screen shot με όλα τα εισαγωγικά ερωτήματα στη σχεσιακή άλγεβρασελ. | 43 |
| Εικόνα 11, Εμφάνιση ενός αποτελέσματος διαίρεσης.....σελ. | 44 |
| Εικόνα 12, αποτελέσματα Q12.....σελ. | 53 |
| Εικόνα 13, ερώτημα Q11.....σελ. | 55 |
| Εικόνα 14, ερώτημα Q17.....σελ. | 57 |
| Εικόνα 15, αρχική οθόνη Relational.....σελ. | 59 |
| Εικόνα 16, δημιουργία νέας σχέσης.....σελ. | 59 |
| Εικόνα 17, περιγραφή χαρακτηριστικών στοιχείων οθόνης.....σελ. | 61 |
| Εικόνα 18, εκτέλεση ερωτήματος.....σελ. | 62 |
| Εικόνα 19, χρήση πακέτου relational σε κώδικα.....σελ. | 68 |
| | |
| Πίνακας 1, σχέση sailors.....σελ. | 17 |
| Πίνακας 2, σχέση boats.....σελ. | 18 |
| Πίνακας 3, σχέση reserves.....σελ. | 19 |
| Πίνακας 4, στιγμιότυπο s1.....σελ. | 20 |
| Πίνακας 5, στιγμιότυπο s2.....σελ. | 20 |
| Πίνακας 6, στιγμιότυπο r1.....σελ. | 20 |
| Πίνακας 7, συγκριτικός προγραμμάτων.....σελ. | 69 |

ΕΙΣΑΓΩΓΗ

Στόχος της συγκεκριμένης πτυχιακής εργασίας συνιστά η διερεύνηση των δυνατοτήτων που υφίστανται σήμερα να υιοθετηθούν ένα ή περισσότερα περιβάλλοντα υποβοηθητικά της διδασκαλίας της θεωρίας και της εφαρμογής στην πράξη του Σχεσιακού Λογισμού και της Σχεσιακής Άλγεβρας. Πρόκειται για τη θεωρητική/μαθηματική υποδομή των δομημένων γλωσσών διατύπωσης αιτημάτων (structural query languages), συνεπώς και του πρότυπου SQL. Ακόμα, η κατανόηση και η γνώση της εν λόγω θεωρίας συνιστούν προϋπόθεση για την κατανόηση των σχημάτων υπολογισμού/διεκπεραίωσης και της βελτιστοποίησης των αιτημάτων από το σύγχρονο σύστημα διαχείρισης βάσεων δεδομένων (DBMS). Στα κεφάλαια που θα ακολουθήσουν θα εκτελεστούν ερωτήματα πάνω στα τρία προγράμματα εκμάθησης σχεσιακής άλγεβρας το winLeap, winRDBI και Relational και στο τέλος θα παρουσιαστούν κάποια συμπεράσματα για αυτές τις εφαρμογές.

ΚΕΦΑΛΑΙΟ 1

I. Εισαγωγή

Στο κεφάλαιο αυτό θα μιλήσουμε για το σχεσιακό μοντέλο αλλά και για τα εργαλεία που το υποστηρίζουν, τη σχεσιακή άλγεβρα και το σχεσιακό λογισμό. Στη συνέχεια θα ακολουθήσουν οι πίνακες, τα ερωτήματα και μια μικρή προλόγηση των προγραμμάτων πάνω στα οποία θα εκτελεστούν τα ερωτήματα σχεσιακής άλγεβρας.

II. Το σχεσιακό μοντέλο

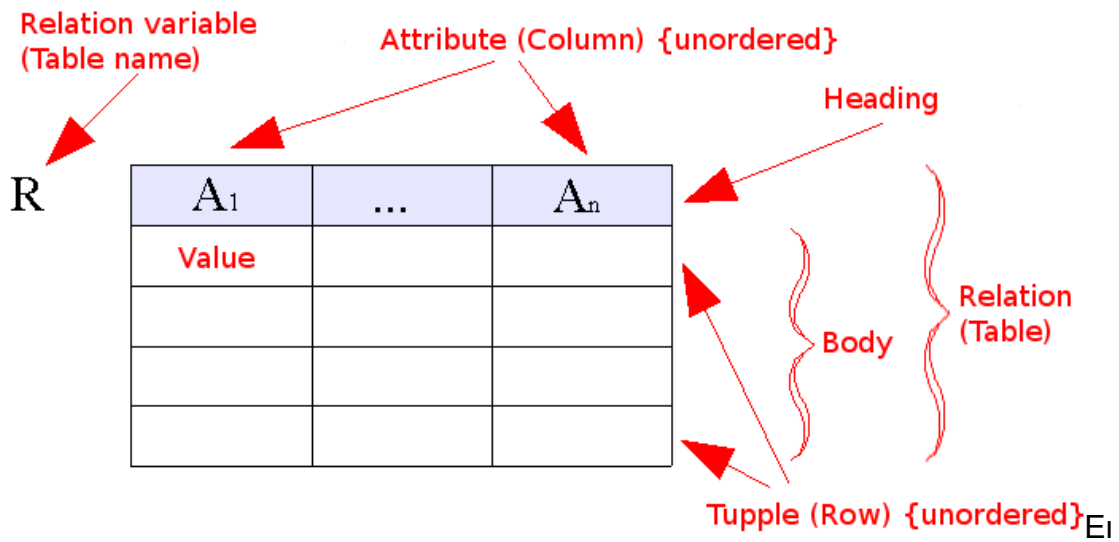
Προτάθηκε για πρώτη φορά από τον E.F. Codd στην εργασία του με τίτλο «A relational model of data for large data banks» (1970). Αυτή η εργασία αποτέλεσε ένα γενικά αποδεκτό ορόσημο πάνω στα συστήματα βάσεων δεδομένων, αν και πρέπει να αναφέρουμε ότι ένα παρόμοιο μοντέλο αναφέρθηκε από τον Childs (1968). Οι στόχοι του μοντέλου καθορίστηκαν όπως αναφέρονται παρακάτω:

- Να επιτρέψει σε μεγάλο βαθμό την ανεξαρτησία των δεδομένων. Τα προγράμματα-εφαρμογές δεν πρέπει να επηρεάζονται από αλλαγές στην εσωτερική απεικόνιση-περιγραφή των δεδομένων, ειδικά από αλλαγές στην οργάνωση των αρχείων και του συστήματος αρχείων, την ταξινόμηση των εγγραφών και των μονοπατιών πρόσβασης (access paths).
- Να δώσει έδαφος για την ενασχόληση με τη σημασιολογία των δεδομένων (data semantics), τα προβλήματα συνέπειας (consistency) και πλεονασμών (redundancy). Συγκεκριμένα, στην εργασία του Codd εισάγεται η έννοια των κανονικοποιημένων σχέσεων, δηλαδή σχέσεις που δεν επαναλαμβάνονται.
- Να επιτρέψει τη δημιουργία συνόλου εντολών για το χειρισμό δεδομένων.

- Να βοηθήσει στην ανάπτυξη μιας δομημένης γλώσσας ερωτημάτων τη γνωστή SQL(Structured Query Language), η οποία μέσω του διεθνούς οργανισμού τυποποίησης έγινε η de facto γλώσσα χειρισμού των σχεσιακών συστημάτων διαχείρισης βάσεων δεδομένων RDBMSs.
- Να βοηθήσει στο χώρο του εμπορίου τη δημιουργία και παραγωγή πολλών προϊόντων DBMS όπως η DB2 της IBM και το ORACLE από την ORACLE corporation.

Μια δεύτερη εργασία που αποτέλεσε κι αυτή σημαντική προσφορά στην ανάπτυξη του σχεσιακού μοντέλου ήταν το INGRES (INteractive GRaphics REtrieval SYstem) από το πανεπιστήμιο Μπέρκλεϋ της Καλιφόρνια. Το INGRES περιέλαβε την ανάπτυξη ενός πρωτότυπου RDBMS το οποίο είχε ίδιους στόχους με το project R. Το αποτέλεσμα ήταν η παραγωγή μιας ακαδημαϊκής έκδοσης του INGRES η οποία συνέβαλε σημαντικά στην ανάπτυξη του σχεσιακού μοντέλου. Επίσης έχουμε και την εργασία «Peterlee Relational Test Vehicle» από την IBM UK η οποία ήταν μια πιο θεωρητική προσέγγιση σε σχέση με τις δύο προηγούμενες. Αυτή έδινε έμφαση σε θέματα επεξεργασίας ερωτημάτων (query processing) και βελτιστοποίησης (optimization) αυτών. Το σχεσιακό μοντέλο βασίζεται στη μαθηματική έννοια των σχέσεων όπου απεικονίζεται από έναν πίνακα. Ο Codd χρησιμοποίησε από τα μαθηματικά την ορολογία από τη θεωρία των συνόλων και τη λογική κατηγορήματος.

Τα στοιχεία μιας σχέσης(**R**elation) είναι οι γραμμές(rows) ή πλειάδες(tuples) του πίνακα(table). Στο σχεσιακό μοντέλο, οι σχέσεις χρησιμοποιούνται για να κρατήσουν πληροφορίες για τα αντικείμενα που εκπροσωπούν στη βάση. Μια σχέση παρουσιάζεται σαν ένας δυσδιάστατος πίνακας του οποίου οι γραμμές-πλειάδες ανταποκρίνονται σε ξεχωριστή εγγραφή η καθεμιά. Οι στήλες(column) αναφέρονται σε ξεχωριστά γνωρίσματα(attributes). Τα γνωρίσματα μπορούν να εμφανιστούν σε οποιαδήποτε σειρά και αυτό δεν αλλάζει τη σχέση. Το πεδίο ορισμού της κάθε στήλης, δηλ. τι πληροφορία θα μπει και τι τύπου θα είναι προσδιορίζεται από την επικεφαλίδα(heading) της στήλης. (Εικόνα 1)



κόνα 1, Παρουσιάζεται ένας δομημένος πίνακας μαζί με τα γνωρίσματα και τις τιμές γνωρισμάτων.

III. Σχεσιακή άλγεβρα

Η σχεσιακή άλγεβρα αποτελεί ένα από τα δύο εργαλεία τα οποία δημιουργήθηκαν για την υποστήριξη του σχεσιακού μοντέλου. Αποτελεί τη μαθηματική θεμελίωση του σχεσιακού μοντέλου αλλά και των πράξεών του. Είναι η βάση για τη δημιουργία της γλώσσας επερωτήσεων SQL και αποτελεί μια μη δηλωτική γλώσσα .

Οι πράξεις στη σχεσιακή άλγεβρα είναι προερχόμενες από τη θεωρία των συνόλων(συνολοθεωρητικές) και σχεσιακές. Από τη θεωρία συνόλων είναι η ένωση (union) ' \cup ', τομή (intersect) ' \cap ', διαφορά (difference) ' $-$ ' και καρτεσιανό γινόμενο (Cartesian product) ' \times '. Οι σχεσιακές βασικές πράξεις είναι επιλογή (selection) ' σ ', προβολή (projection) ' π ', μετονομασία (rename) ' ρ ' και σύζευξη (join). Υπάρχουν και άλλες σχεσιακές πράξεις δευτερεύουσες οι οποίες μπορούν να εκφραστούν μέσα από συνδυασμό των βασικών σχεσιακών και θεωρίας συνόλων. Μερικές από αυτές είναι η φυσική σύζευξη, η σύζευξη ισότητας, η σύζευξη συνθήκης και η διαίρεση. Τέλος θα αναφερθούμε σε μια βασική ιδιότητα της σχεσιακής άλγεβρας που μας ενδιαφέρει σε αυτή την εργασία και είναι η **σχεσιακή κλειστότητα (relational closure)**, περιλαμβάνει τέσσερις υπό-ιδιότητες:

- Η έξοδος μιας σχεσιακής πράξης είναι μια άλλη σχέση.
- Η έξοδος οποιασδήποτε πράξης είναι ένα αντικείμενο του ίδιου τύπου με την είσοδό της (είναι και τα δύο σχέσεις).
- Η έξοδος μιας πράξης μπορεί να γίνει είσοδος για μια άλλη πράξη.
- Παραστάσεις με ένθετες υπό-παραστάσεις.

Οι παραπάνω πράξεις επιτρέπουν στο χρήστη να προσδιορίσει βασικά αιτήματα ανάκτησης δεδομένων(πλειάδων). Έτσι παράγονται νέες σχέσεις, στις οποίες μπορεί να γίνει περαιτέρω επεξεργασία χρησιμοποιώντας τους ίδιους τελεστές. Μια ακολουθία πράξεων μεταξύ των γνωρισμάτων κάθε στιγμιοτύπου της σχεσιακής άλγεβρας σχηματίζει μια έκφραση της σχεσιακής άλγεβρας, της οποίας το αποτέλεσμα είναι επίσης σχέση.

IV. Σχεσιακός λογισμός πλειάδων (TRC)

Ο σχεσιακός λογισμός πλειάδων εισήχθη από τον Codd ως μια δηλωτική γλώσσα που να βρίσκει πλειάδες, δοθέντων μεταβλητών που αντιπροσωπεύουν πλειάδες μιας σχέσης και μιας φόρμουλας όπου πρέπει να την επαληθεύουν.

Στο σχεσιακό λογισμό γράφουμε μια δηλωτική έκφραση για να προσδιορίσουμε ένα αίτημα ανάκτησης και επομένως δεν υπάρχει περιγραφή για τον τρόπο εκτέλεσης μιας επερώτησης. Μια έκφραση του σχεσιακού λογισμού προσδιορίζει *τί θα ανακτηθεί και όχι το πώς θα ανακτηθεί*. Επομένως ο σχεσιακός λογισμός θεωρείται μη διαδικαστική γλώσσα.

Ένα απλό παράδειγμα επερώτησης στο σχεσιακό λογισμό πλειάδων είναι της μορφής:

{a | CONDITION(a) }

Όπου **a** είναι μια μεταβλητή πλειάδων και **CONDITION(a)** είναι μια έκφραση συνθήκης που περιλαμβάνει το **a**. Το αποτέλεσμα μιας τέτοιας επερώτησης είναι ένα σύνολο πλειάδων **a** που ικανοποιούν την **CONDITION(a)**.

Σε μια έκφραση του σχεσιακού λογισμού πρέπει να προσδιορίσουμε:

- Για κάθε μεταβλητή πλειάδων **a**, τη σχέση τιμών **R** της **a**. Η σχέση αυτή προσδιορίζεται με μια συνθήκη της μορφής **R(a)**.
- Μια συνθήκη για την επιλογή ενός συγκεκριμένου συνδυασμού πλειάδων. Εφ' όσον οι μεταβλητές πλειάδων παίρνουν τιμές από τις αντίστοιχες σχέσεις τιμών, η συνθήκη αποτιμάται για κάθε πιθανό συνδυασμό πλειάδων ώστε να αναγνωριστούν οι επιλεγόμενοι συνδυασμοί για τους οποίους η συνθήκη αποτιμάται σε TRUE.
- Ένα σύνολο από γνωρίσματα που πρέπει να ανακτηθούν, τα ζητούμενα γνωρίσματα. Οι τιμές των γνωρισμάτων αυτών ανακτώνται για κάθε επιλεγόμενο συνδυασμό πλειάδων.

Έστω το στιγμιότυπο **S1**

s1(sid/numeric,sname/char,rating/numeric,age/numeric):sid

22, 'Dustin', 7, 45

31, 'Lubber', 8, 55

58, 'Rusty', 10, 35

Θέλουμε να ανακτήσουμε το όνομα και την ηλικία του ναύτη με **sid=22**.

Το αποτέλεσμα θα μας δινόταν εκτελώντας το παρακάτω ερώτημα:

Q: {a.sname, a.age | s1(a) a.sid='22'}

Μια γενική έκφραση του σχεσιακού λογισμού πλειάδων είναι της μορφής:

{a₁.T₁, a₂.T₂, ..., a_n.T_n | CONDITION(a₁, a₂, ..., a_n, a_{n+1}, a_{n+2}, ..., a_{n+m})}

V. Σχεσιακός λογισμός πεδίου (DRC)

Ο σχεσιακός λογισμός πεδίου σε αντίθεση με το σχεσιακό λογισμό πλειάδων βρίσκεται στο ότι οι μεταβλητές του παίρνουν τιμές από «domain» δηλαδή από τις στήλες της εκάστοτε σχέσης και όχι από τις πλειάδες.

Έστω η παράσταση $\{d_1, d_2, \dots, d_n | P(d_1, d_2, \dots, d_n)\}$ οι μεταβλητές d_1, d_2, \dots αντιπροσωπεύουν τα πεδία μιας σχέσης όπου αληθεύουν τη $P(d_1, d_2, \dots, d_n)$.

Έστω το στιγμιότυπο **S1**

s1(sid/numeric,sname/char,rating/numeric,age/numeric):sid

22, 'Dustin', 7, 45

31, 'Lubber', 8, 55

58, 'Rusty', 10, 35

Θέλουμε να βρούμε τους ναύτες με ηλικία >45;

Το αποτέλεσμα θα μας δινόταν από το παρακάτω ερώτημα:

Q: {I, N, T, A | sailors(I, N, T, A) and A > 45};

Μια γενική έκφραση του σχεσιακού λογισμού πεδίων είναι της μορφής:

{ d1, d2, ..., dn | CONDITION(d1, d2, ..., dn)}

VI. Σχέσεις παραδειγμάτων

Στην παρούσα πτυχιακή οι πίνακες που θα χρησιμοποιηθούν είναι από το βιβλίο «Συστήματα Διαχείρισης Βάσεων Δεδομένων, Ramakrishnan-Gehrke, Α' τόμος, 2^η έκδοση, εκδόσεις Τζιόλας».

Ο πρώτος πίνακας είναι ο «sailors» ο οποίος αποτελείται από τα πεδία:

- sid = ταυτότητα ναύτη (sailor id) | integer
- sname = όνομα ναύτη (sailor name) | char
- rating = βαθμός | integer
- age = ηλικία | integer

| <i>sid</i> | <i>sname</i> | <i>rating</i> | <i>age</i> |
|------------|--------------|---------------|------------|
| 22 | Dustin | 7 | 45 |
| 29 | Brutus | 1 | 33 |
| 31 | Lubber | 8 | 55 |
| 32 | Andy | 8 | 25 |
| 58 | Rusty | 10 | 35 |
| 64 | Horatio | 7 | 35 |
| 71 | Zorba | 10 | 16 |
| 74 | Horatio | 9 | 35 |
| 85 | Art | 3 | 25 |
| 95 | Bob | 3 | 63 |

Πίνακας 1

Δεύτερος πίνακας «boats»:

- bid = ταυτότητα βάρκας (boat id) | integer
- bname = όνομα βάρκας (boat name) | char
- color = χρώμα βάρκας | char

| <i>bid</i> | <i>bname</i> | <i>color</i> |
|------------|------------------|--------------|
| 101 | Interlake | Blue |
| 102 | Interlake | Red |
| 103 | Clipper | Green |
| 104 | Marine | Red |

Πίνακας 2

Τρίτος πίνακας «reservations»:

- sid = ταυτότητα ναύτη (sailor id) | integer
- bid = ταυτότητα βάρκας (boat id) | integer
- date = ημερομηνία κράτησης | char

| <i>sid</i> | <i>bid</i> | <i>date</i> |
|------------|------------|-------------|
| 22 | 101 | 10/10/98 |
| 22 | 102 | 10/10/98 |
| 22 | 103 | 10/8/98 |
| 22 | 104 | 10/7/98 |
| 31 | 102 | 11/10/98 |
| 31 | 103 | 11/6/98 |
| 31 | 104 | 11/12/98 |
| 64 | 101 | 9/5/98 |
| 64 | 102 | 9/8/98 |
| 74 | 103 | 9/8/98 |

Πίνακας 3

Επίσης θα χρησιμοποιήσουμε και τα παρακάτω στιγμιότυπα των παραπάνω πινάκων.

s1:

| <i>sid</i> | <i>sname</i> | <i>rating</i> | <i>age</i> |
|------------|--------------|---------------|------------|
| 22 | Dustin | 7 | 45 |
| 31 | Lubber | 8 | 55 |
| 58 | Rusty | 10 | 35 |

Πίνακας 4

s2(με την προσθήκη δύο ναυτών που δεν υπάρχουν στον sailors):

| <i>sid</i> | <i>sname</i> | <i>rating</i> | <i>age</i> |
|------------|--------------|---------------|------------|
| 28 | Yuppy | 9 | 35 |
| 31 | Lubber | 8 | 55 |
| 44 | Guppy | 5 | 35 |
| 58 | Rusty | 10 | 35 |

Πίνακας 5

r1:

| <i>sid</i> | <i>bid</i> | <i>date</i> |
|------------|------------|-------------|
| 22 | 101 | 10/10/96 |
| 58 | 103 | 11/12/96 |

Πίνακας 6

VII. Ερωτήματα που θα εκτελεστούν

Τα ερωτήματα που ακολουθούν είναι παρμένα από το βιβλίο «Συστήματα Διαχείρισης Βάσεων Δεδομένων, Ramakrishnan-Gehrke, Α' τόμος, 2^η έκδοση, εκδόσεις Τζιόλας», επιπλέον υπάρχουν και κάποια σκοπίμως δημιουργημένα. Θα χωριστούν σε τέσσερις κατηγορίες, η πρώτη ασχολείται με τη σχεσιακή άλγεβρα, η δεύτερη με το σχεσιακό λογισμό πλειάδων, η τρίτη με το σχεσιακό λογισμό πεδίων και η τέταρτη με SQL.

i. Σχεσιακή άλγεβρα (ALG)

- 1) Να βρεθούν τα ονόματα των ναυτικών οι οποίοι έχουν κάνει κράτηση στη βάρκα με κωδικό 103;
- 2) Να βρεθούν τα ονόματα των ναυτικών οι οποίοι έχουν κάνει κράτηση σε κόκκινη βάρκα ;
- 3) Να βρεθούν τα χρώματα των βαρκών για τις οποίες έχει κάνει κράτηση ο ναυτικός που ονομάζεται «Lubber»;
- 4) Να βρεθούν τα ονόματα όλων των ναυτικών που καθένας από τους οποίους έχει κάνει κράτηση τουλάχιστον σε μια βάρκα;
- 5) Να βρεθούν τα ονόματα των ναυτικών οι οποίοι έχουν κάνει κράτηση είτε σε κόκκινη είτε σε πράσινη βάρκα;
- 6) Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κράτηση και σε κόκκινες και σε πράσινες βάρκες;
- 7) Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κρατήσεις σε τουλάχιστον δύο (διαφορετικές) βάρκες;
- 8) Να βρεθούν οι κωδικοί sid των ναυτικών που έχουν ηλικία μεγαλύτερη των 20 ετών και δεν έχουν κάνει κράτηση σε κόκκινη βάρκα;
- 9) Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κρατήσεις σε όλες τις βάρκες;
- 10) Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κρατήσεις σε όλες τις βάρκες που έχουν όνομα «Interlake»;

ii. Σχεσιακός λογισμός πλειάδων (TRC)

- 11) Να βρεθούν όλοι οι ναυτικοί που έχουν βαθμό (rating) μεγαλύτερο του 7;

- 12) Να βρεθούν τα ονόματα και οι ηλικίες των ναυτικών που έχουν τιμή διατίμησης μεγαλύτερη του 7;
- 13) Να βρεθούν το όνομα του ναυτικού, ο κωδικός της βάρκας και η ημερομηνία κράτησης για την κάθε μια κράτηση που καταχωρείται στη βάση;
- 14) Να βρεθούν οι ναυτικοί που έχουν κάνει κρατήσεις σε όλες τις κόκκινες βάρκες;

iii. Σχεσιακός λογισμός πεδίων (DRC)

- 15) Να βρεθούν οι ναυτικοί που έχουν τιμή διατίμησης > 7 ;

iv. SQL

- 16) Να βρεθεί ο ναύτης με τη μεγαλύτερη ηλικία;

VIII. Προγράμματα που θα χρησιμοποιηθούν

WinRDBI

Το winRDBI δημιουργήθηκε σε πλατφόρμα java, με γραφικό περιβάλλον και δυνατότητα εκτέλεσης ερωτημάτων όχι μόνο σχεσιακής άλγεβρας αλλά και TRC, DRC, SQL, DRC by name. Επίσης υποστηρίζει την πολλαπλή εκτέλεση ερωτημάτων του ίδιου είδους μέσα στο ίδιο παράθυρο.

WinLEAP

Το winleap είναι η έκδοση για λειτουργικό ms windows του εργαλείου εκμάθησης σχεσιακής άλγεβρας Leap για Linux. Το εργαλείο Leap είναι σχεδιασμένο ως ένας ερμηνευτής ερωτημάτων σχεσιακής άλγεβρας.

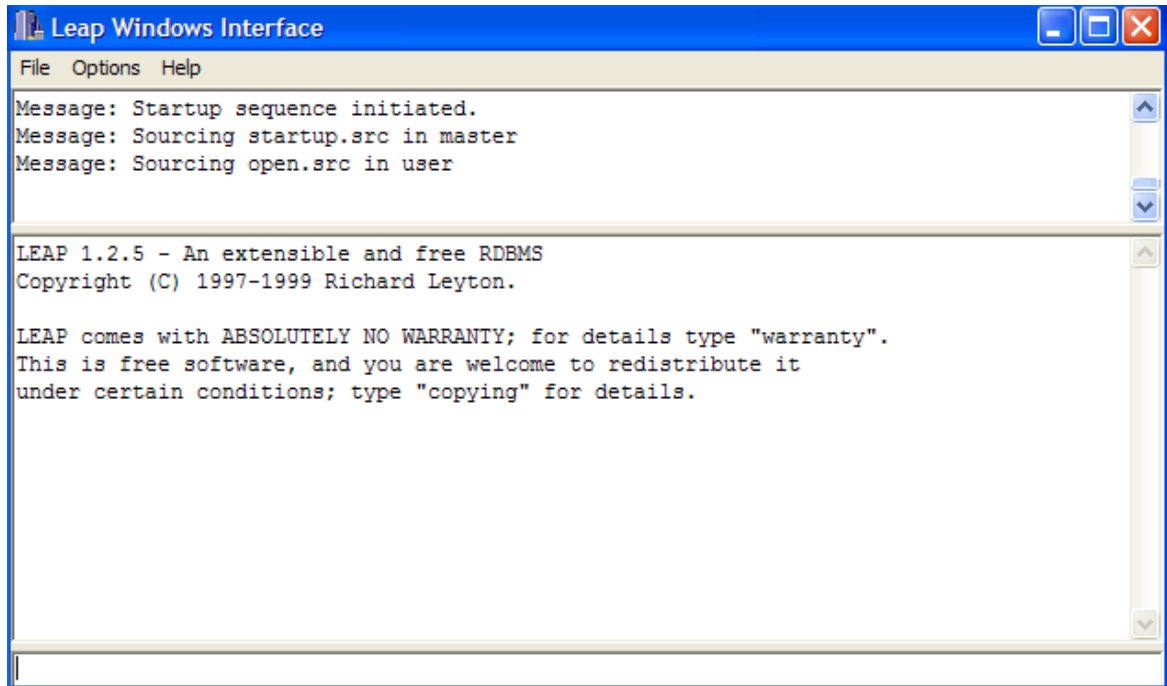
Relational

Το Relational είναι ένα εργαλείο φτιαγμένο αποκλειστικά και μόνο για τον έλεγχο ερωτημάτων σχεσιακής άλγεβρας όπως και το winLEAP είναι ένας ερμηνευτής ερωτημάτων σχεσιακής άλγεβρας. Διαφέρει όμως στο περιβάλλον όπου είναι γραφικό αλλά και στο ότι δουλεύεται καλύτερα λόγω καλύτερης διασύνδεσης χρήστη.

ΚΕΦΑΛΑΙΟ 2

I. ΕΙΣΑΓΩΓΗ-WinLeap

Το WinLeap είναι η έκδοση για λειτουργικό ms windows του εργαλείου εκμάθησης σχεσιακής άλγεβρας Leap για Linux. Το εργαλείο Leap είναι σχεδιασμένο ως ένας ερμηνευτής ερωτημάτων σχεσιακής άλγεβρας. Τα παραδείγματα τα τρέχει σε γραμμή εντολών, δε χρειάζεται εγκατάσταση (τρέχει σε κέλυφος nutshell)(εικόνα 2).



```
Leap Windows Interface
File Options Help
Message: Startup sequence initiated.
Message: Sourcing startup.src in master
Message: Sourcing open.src in user

LEAP 1.2.5 - An extensible and free RDBMS
Copyright (C) 1997-1999 Richard Leyton.

LEAP comes with ABSOLUTELY NO WARRANTY; for details type "warranty".
This is free software, and you are welcome to redistribute it
under certain conditions; type "copying" for details.
```

Εικόνα 2, Η διασύνδεση χρήστη του winLeap-αρχική οθόνη.

II. Λειτουργικότητα και Βασικές Εντολές

Πριν δούμε τα παραδείγματα θα πούμε μερικά πράγματα που πρέπει να γίνουν, όπως επίσης και κάποιους περιορισμούς του προγράμματος.

- Κάθε σχέση-πίνακας ή έκφραση σχεσιακής άλγεβρας πρέπει να εσωκλείεται σε παρενθέσεις.
- Οι λίστες με τα χαρακτηριστικά πρέπει να διαχωρίζονται με κόμμα.
- Υπάρχουν δύο τρόποι να γίνουν οι υπολογισμοί
 - Να αποθηκεύουμε κάθε πράξη σε ξεχωριστή σχέση π.χ.
subjcl=project (subject) (class)
indcla=project (index) (class)
result=difference (subjcl) (indcla)
print result
 - Να χρησιμοποιήσουμε την αρχή «relational closure»,
π.χ.(project(subject) (class)) difference (project (index) (class)
)
print @last

Το πρόγραμμα υποστηρίζει τους παρακάτω σχεσιακούς τελεστές

Προβολή: project (expression) (attrib1, attrib2, ..., attribN)

Ένωση: (expression) union (expression)

Τομή: (expression) intersect (expression)

Διαφορά: (expression) difference (expression)

Καρτεσιανό γινόμενο: (expression) product (expression)

Επιλογή: select (expression) ((condition) {[and|or] (condition)}))

Σύζευξη: join (relation) (relation) ((condition) {[and|or] (condition)})

Φυσική σύζευξη: natjoin (relation) (relation)

Επίσης η διαφορά μπορεί να αντικατασταθεί με ένα μείον «-», ακόμα οι τελεστές ένωση, τομή, διαφορά και καρτεσιανό γινόμενο μπορούν να τοποθετηθούν σε οποιαδήποτε θέση της έκφρασης π.χ. «**union (expression) (expression) is valid, as is (expression) (expression) operator**».

Οι υποστηριζόμενες συνθήκες είναι οι παρακάτω

[value|attribute] <, <=, =, >=, >, <>, ~ [value|attribute]

Δημιουργώντας μια βάση και μια σχέση-πίνακα

Για να δημιουργήσουμε μια βάση πάνω στην οποία θα δημιουργήσουμε τις σχέσεις πρέπει στη γραμμή εντολών να δώσουμε τις παρακάτω εντολές:

create <όνομα_βάσης> → για δημιουργία βάσης.

use <όνομα_βάσης> → για να μπορούμε να χρησιμοποιήσουμε τη βάση.

list <όνομα_βάσης> → για να δούμε αν όντως έχει δημιουργηθεί η βάση στο σύστημα.

Το συντακτικό πρέπει να ακολουθεί τον κανόνα:

relation (name) ((attribute1, type1, length1), (a2,t2,l2), ..., (aN,tN,lN))

σχέση (όνομα σχέσης) ((χαρακτηριστικό1, τύπος1(integer|string), μήκος1),)

Π.χ. relation (Reserves) ((sid, integer, 2), (bid, integer, 3), (day, string, 10))

Εισαγωγή και Διαγραφή στοιχείων σε μια σχέση

Για να κάνουμε την εισαγωγή στοιχείων στη βάση χρησιμοποιούμε την εντολή **add**, ενώ για διαγραφή την εντολή **delrel <όνομα_σχέσης>**.

Π.χ. add (Reserves) (22, 101, 10/10/98), delrel Reserves

Σε περίπτωση που θέλουμε να δούμε τα στοιχεία που έχει μια σχέση χρησιμοποιούμε την εντολή **print <όνομα_σχέσης>**.

Π.χ. print Reserves

III. Δημιουργία του σχεσιακού σχήματος και της βάσης

Πριν προχωρήσουμε στην εκτέλεση των ερωτημάτων θα κάνουμε τη δημιουργία της βάσης πάνω στην οποία θα δουλέψουμε, τη δημιουργία των σχέσεων και την εισαγωγή των στοιχείων σ' αυτές.

ΠΡΟΣΟΧΗ!!!

Αν δημιουργήσατε κάποια βάση έως τώρα θα χαθεί διότι στο πρόγραμμα ως default οι βάσεις είναι temporary. Για να αποφύγετε να χάσετε δεδομένα θα ανοίξετε το φάκελο configs που βρίσκεται μέσα στο φάκελο του WinLeap, και με ένα text editor θα ανοίξετε το αρχείο vars, εκεί θα βρείτε τις παραμέτρους *temporary* και *tempdb*. Από *on* θα τις κάνετε *off*.

Το λογισμικό δεν υποστηρίζει τους τύπους date και double, οπότε όπου υπήρχε πεδίο date μετατράπηκε σε string και όπου υπήρχε πεδίο double μετατράπηκε σε integer.

Δημιουργία βάσης και πινάκων

Πληκτρολογούμε τις παρακάτω εντολές:

```
create rama
```

```
use rama
```

```
relation (Sailors) ((sid, integer, 2), (sname, string, 10), (rating, integer, 2),  
(age,integer, 2))
```

```
relation (Reserves) ((sid, integer, 2), (bid, integer, 3), (day, string, 10))
```

```
relation (Boats) ((bid, integer, 3), (bname, string, 10), (color, string, 5))
```

Εισαγωγή στοιχείων

Για να εισάγουμε τα στοιχεία στους πίνακες πληκτρολογούμε ένα ένα όλα τα παρακάτω:

Για τον πίνακα Sailors

add (Sailors) (22, Dustin, 7, 45)

add (Sailors) (29, Brutus, 1, 33)

add (Sailors) (31, Lubber, 8, 55)

add (Sailors) (32, Andy, 8, 25)

add (Sailors) (58, Rusty, 10, 35)

add (Sailors) (64, Horatio, 7, 35)

add (Sailors) (71, Zorba, 10, 16)

add (Sailors) (74, Horatio, 9, 35)

add (Sailors) (85, Art, 3, 25)

add (Sailors) (95, Bob, 3, 63)

Για τον πίνακα Reserves

add (Reserves) (22, 101, 10/10/98)

add (Reserves) (22, 102, 10/10/98)

add (Reserves) (22, 103, 10/8/98)

add (Reserves) (22, 104, 10/7/98)

add (Reserves) (31, 102, 11/10/98)

add (Reserves) (31, 103, 11/6/98)

add (Reserves) (31, 104, 11/12/98)

add (Reserves) (64, 101, 9/5/98)

add (Reserves) (64, 102, 9/8/98)

add (Reserves) (74, 103, 9/8/98)

Για τον πίνακα Boats

add (Boats) (101, Interlake, blue)

add (Boats) (102, Interlake, red)

add (Boats) (103, Clipper, green)

add (Boats) (104, Marine, red)

Αλλιώς δημιουργούμε ένα script σε έναν επεξεργαστή κειμένου και το αποθηκεύουμε με κατάληξη <όνομα αρχείου>.src. Εκεί γράφουμε όλα τα παραπάνω και το εκτελούμε γράφοντας στη γραμμή εντολών του προγράμματος **@ <scriptname>**, για να δούμε μόνο τα περιεχόμενά του γράφουμε **I(L) <scriptname>**. Το αποθηκεύουμε μέσα στο φάκελο *source* που βρίσκεται στο φάκελο της εκάστοτε βάσης. Αν θέλουμε να δημιουργήσουμε τη βάση του βιβλίου, το αποθηκεύουμε στη διαδρομή `.\winleap\databases\user\source` .

IV. Εκτέλεση Ερωτημάτων

Σε αυτό το κομμάτι θα ξεκινήσουμε την εκτέλεση των ερωτημάτων σχεσιακής άλγεβρας.

Q1: Να βρεθούν τα ονόματα των ναυτικών οι οποίοι έχουν κάνει κράτηση στη βάρκα με κωδικό 103;

```
temp1q1 = select (reserves) (reserves.bid='103')
```

```
temp2q1 = (temp1q1) natjoin (sailors)
```

```
q1 = project (temp2q1) (sname)
```

```
print q1
```

Q2: Να βρεθούν τα ονόματα των ναυτικών οι οποίοι έχουν κάνει κράτηση σε κόκκινη βάρκα;

```
temp1q2 = select (boats) (boats.color = 'red')
```

```
temp2q2 = project (temp1q2) (bid)
```

```
temp3q2 = (temp2q2) product (reserves)
```

```
temp4q2 = select (temp3q2) (temp3q2.bid = temp3q2.reserves.bid)
```

```
temp5q2 = (temp4q2) product (sailors)
```

```
temp6q2 = select (temp5q2) (temp6q2.sid = temp6q2.sailors.sid)
```

```
q2 = project (temp6q2) (sname)
```

```
print q2
```

Q3: Να βρεθούν τα χρώματα των βαρκών για τις οποίες έχει κάνει κράτηση ο ναυτικός που ονομάζεται Lubber;

```
temp1q3 = select (sailors) (sailors.sname = 'Lubber')
```

```
temp2q3 = (temp1q3) product (reserves)
```

```
temp3q3 = select (temp2q3) (temp2q3.sid = temp2q3.reserves.sid)
```

```
temp4q3 = (temp3q3) product (boats)
```

```
temp5q3 = select (temp4q3) (temp4q3.bid = temp4q3.boats.bid)
```

```
q3 = project (temp5q3) (color)
```

```
print q3
```

Q4: Να βρεθούν τα ονόματα όλων των ναυτικών κάθε ένας από τους οποίους έχει κάνει κράτηση τουλάχιστον σε μια βάρκα;

```
temp1q4 = (sailors) product (reserves)
```

```
temp2q4 = select (temp1q4) (temp1q4.sid = temp1q4.reserves.sid)
```

```
q4 = project (temp2q4) (sname)
```

```
print q4
```

Q5: Να βρεθούν τα ονόματα των ναυτικών οι οποίοι έχουν κάνει κράτηση είτε σε κόκκινη είτε σε πράσινη βάρκα;

```
temp1q5 = select (boats) (boats.color = 'red')
```

```
temp2 q5= select (boats) (boats.color = 'green')
```

```
temp3q5 = (temp1q5) union (temp2q5)
```

```
temp4q5 = (temp3q5) product (reserves)
```

```
temp5q5 = select (temp4q5) (temp4q5.bid = temp4q5.reserves.bid)
```

```
temp6q5 = (temp5q5) product (sailors)
```

```
temp7 q5= select (temp6q5) (temp6q5.sid = temp6q5.sailors.sid)
```

```
q5 = project (temp7q5) (sname)
```

Q6: Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κράτηση και σε κόκκινες και σε πράσινες βάρκες;

```
temp1q6 = select (boats) (color = 'red')
```

```
temp2q6 = (temp1q6) natjoin (reserves)
```

```
temp3q6 = project (temp2q6) (sid)
```

```
temp4q6 = select (boats) (color = 'green')
```

```
temp5q6 = (temp4q6) natjoin (reserves)
```

```
temp6q6 = project (temp5q6) (sid)
```

```
temp7q6 = (temp3q6) intersect (temp6q6)
```

```
temp8q6 = (temp7q6) natjoin (sailors)
```

```
q6 = project (temp8q6) (sname)
```

```
print q6
```

Q7: Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κρατήσεις σε τουλάχιστον δύο (διαφορετικές) βάρκες; Το εργαλείο δε μπορεί να επεξεργαστεί δύο διαφορετικούς πίνακες με τα ίδια πεδία.

```
temp1q7 = (sailors) natjoin (reserves);
```

```
temp2q7 = project (temp1q7) (sid, sname, bid)
```

```
temp3q7 = temp2q7
```

```
q7= (temp2q7) product (temp3q7)
```

```
print q7
```

Q8: Να βρεθούν οι κωδικοί sid των ναυτικών που έχουν ηλικία μεγαλύτερη των 20 ετών και δεν έχουν κάνει κράτηση σε κόκκινη βάρκα;

```
temp1q8 = select (sailors) (sailors.age > '20')
```

```
temp2q8 = project (temp1q8) (sid)
```

```
temp3q8 = select (boats) (boats.color = 'red')
```

```
temp4q8 = (temp3q8) product (reserves)
```

```
temp5q8 = select (temp4q8) (temp4q8.bid = temp4q8.reserves.bid)
```

```
temp6q8 = (temp5q8) product (sailors)
```

```
temp7q8= select (temp6q8) (temp6q8.sid = temp6q8.sailors.sid)
```

```
temp8q8 = project (temp7q8) (sid)
```

```
q8 = project (temp8q8) (sid)
```

```
print q8
```

Q9: Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κρατήσεις σε όλες τις βάρκες;

```
temp1q9 = project (reserves) (sid,bid)
```

```
temp2q9 = project (boats) (bid)
```

```
temp3q9 = (temp1q9) intersect (temp2q9)
```

```
temp4q9 = (temp3q9) product (sailors)
```

```
temp5q9 = select (temp4q9) (temp4q9.sid = temp4q9.sailors.sid)
```

```
q9 = project (temp5q9) (sname)
```

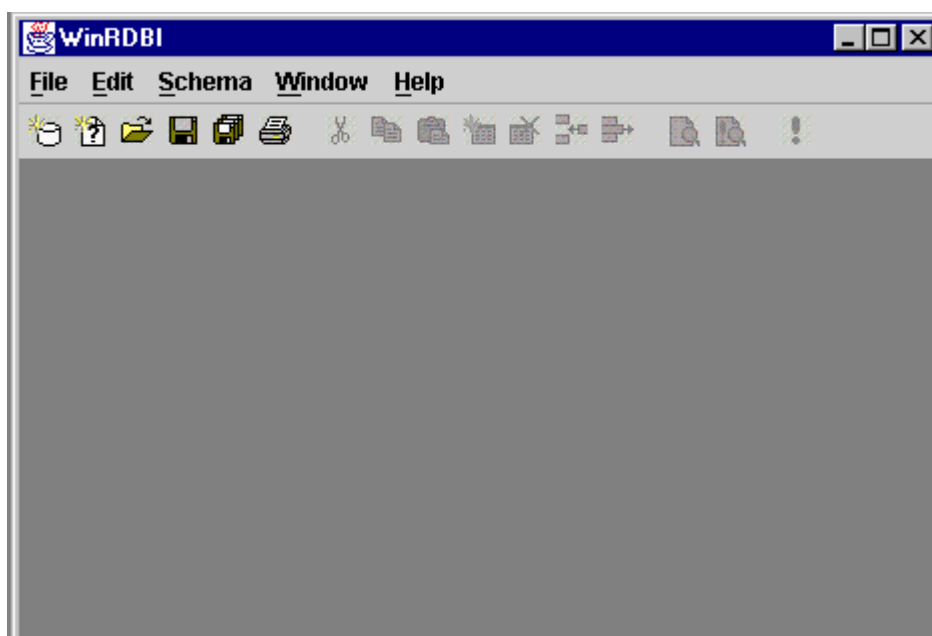

print q9

Μερικά ερωτήματα όπως διαπιστώσατε δεν έγιναν επειδή το πρόγραμμα δεν υποστηρίζει τη μετονομασία.

ΚΕΦΑΛΑΙΟ 3

I. ΕΙΣΑΓΩΓΗ-WinRDBI

Το WinRDBI εξελίχθηκε και σχεδιάστηκε ώστε να επιτρέψει την ομαλότερη εξοικείωση και εξάσκηση των χρηστών μέσω ενός ευχάριστου γραφικού περιβάλλοντος (εικόνα 3) MDI (Multiple Document Interface = δυνατότητα επεξεργασίας πολλών εγγράφων ταυτόχρονα), στις έννοιες της σχεσιακής άλγεβρας, σχεσιακού λογισμού ορισμάτων (DRC), σχεσιακού λογισμού ορισμάτων κατ' όνομα (DRC by name), σχεσιακού λογισμού πλειάδων (TRC) και γλώσσας διατύπωσης δομημένων αιτημάτων (SQL).



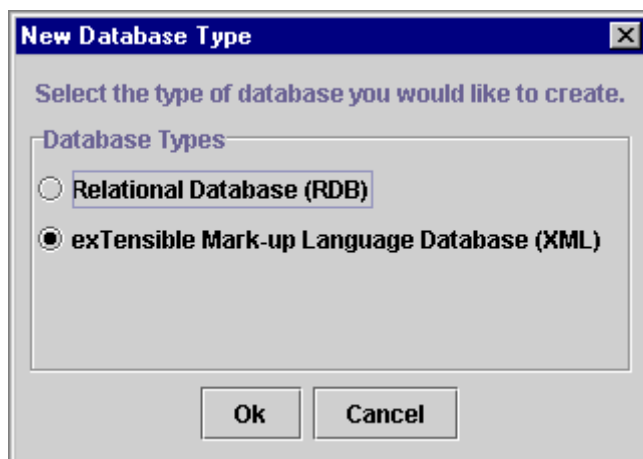
Εικόνα 3, Αρχική Οθόνη.

Η επίλυση των «ερωτημάτων» κάθε είδους γίνεται πάνω σε μια βάση είτε σχεσιακή είτε XML την οποία και δημιουργούμε ή την παίρνουμε έτοιμη (εικόνα 3 και 4).

II. Λειτουργικότητα και Βασικές Εντολές

Δημιουργία βάσης

Στο πρόγραμμα αυτό μπορούμε να επιλέξουμε δημιουργία βάσης είτε σε μορφή <όνομα>.rdb είτε σε <όνομα>.xml. Η απλούστερη μορφή και η καλύτερη, από την άποψη της εκμάθησης, είναι η σχεσιακή rdb. Για να τη δημιουργήσουμε πάμε στο μενού File και επιλέγουμε New Database. Ύστερα μας βγάζει το παράθυρο επιλογής όπως το δείχνει η εικόνα 4.



Εικόνα 4, Επιλογή βάσης.

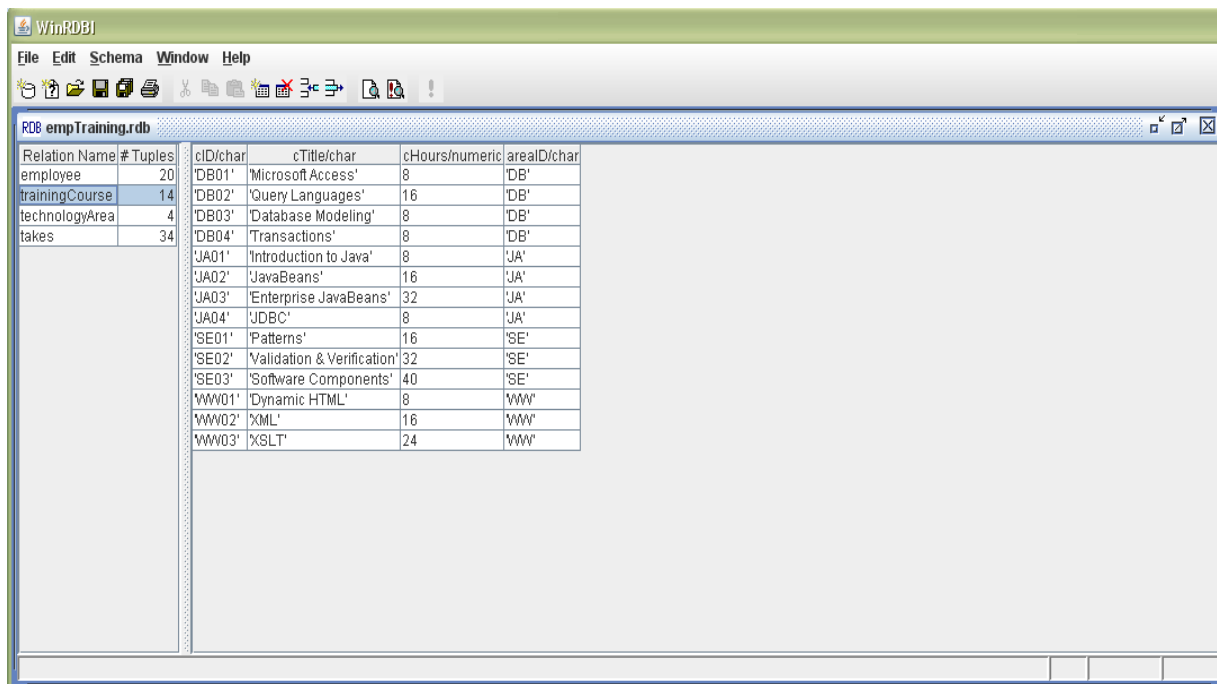
Επιλέγουμε ποια από τις δύο μορφές θα χρησιμοποιήσουμε και πατάμε Ok.

Τέλος μπορούμε να δημιουργήσουμε τη βάση με έναν text editor και να την αποθηκεύσουμε ανάλογα με το τι θέλουμε, rdb ή xml, ανοίγοντας απλά ένα κενό αρχείο κειμένου (txt) και αποθηκευοντάς το <όνομα_βάσης>.rdb ή xml. Έτσι έχουμε μια κενή βάση δεδομένων με το όνομα που επιθυμούμε.

Δημιουργία πινάκων και πεδίων, εισαγωγή στοιχείων

Πρώτα απ' όλα θα δειχθεί η δημιουργία πινάκων-σχέσεων.

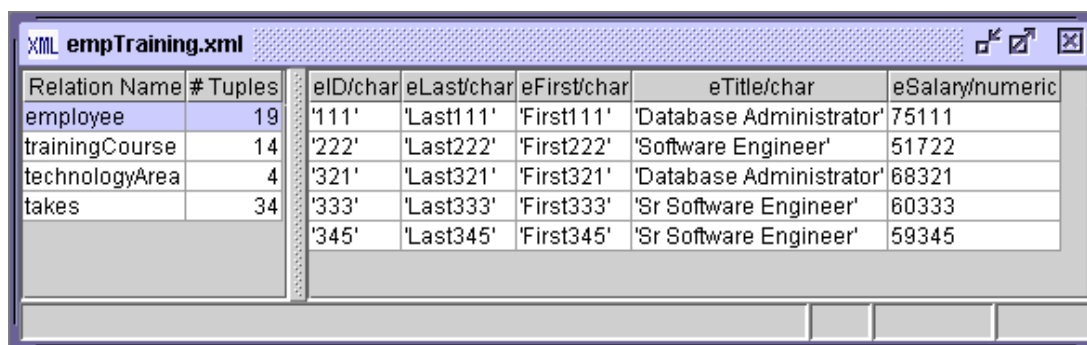
Για να δημιουργήσουμε τον πίνακα-σχέση, κάνουμε κλικ στο **schema** ύστερα επιλέγουμε **add relation**, δίνουμε το όνομα, τα πεδία και τον τύπο τους. Θα τον δούμε να δημιουργείται στα αριστερά της οθόνης.(εικόνα 5 κ' 6)



The screenshot shows the WinRDBJ application window. The title bar reads 'WinRDBJ'. The menu bar includes 'File', 'Edit', 'Schema', 'Window', and 'Help'. The toolbar contains various icons for file operations and navigation. The main window displays a table titled 'RDB empTraining.rdb'. The table has four columns: 'Relation Name', '# Tuples', 'cID/char', 'cTitle/char', 'cHours/numeric', and 'areaID/char'. The data rows are as follows:

| Relation Name | # Tuples | cID/char | cTitle/char | cHours/numeric | areaID/char |
|----------------|----------|----------|-----------------------------|----------------|-------------|
| employee | 20 | 'DB01' | 'Microsoft Access' | 8 | 'DB' |
| trainingCourse | 14 | 'DB02' | 'Query Languages' | 16 | 'DB' |
| technologyArea | 4 | 'DB03' | 'Database Modeling' | 8 | 'DB' |
| takes | 34 | 'DB04' | 'Transactions' | 8 | 'DB' |
| | | 'JA01' | 'Introduction to Java' | 8 | 'JA' |
| | | 'JA02' | 'JavaBeans' | 16 | 'JA' |
| | | 'JA03' | 'Enterprise JavaBeans' | 32 | 'JA' |
| | | 'JA04' | 'JDBC' | 8 | 'JA' |
| | | 'SE01' | 'Patterns' | 16 | 'SE' |
| | | 'SE02' | 'Validation & Verification' | 32 | 'SE' |
| | | 'SE03' | 'Software Components' | 40 | 'SE' |
| | | 'WWW01' | 'Dynamic HTML' | 8 | 'WWW' |
| | | 'WWW02' | 'XML' | 16 | 'WWW' |
| | | 'WWW03' | 'XSLT' | 24 | 'WWW' |

Εικόνα 5, στα αριστερά απεικονίζεται το όνομα του πίνακα και το πόσες πλειάδες έχει. Στα δεξιά τα πεδία και τα στοιχεία.



The screenshot shows the WinRDBJ application window. The title bar reads 'XML empTraining.xml'. The menu bar includes 'File', 'Edit', 'Schema', 'Window', and 'Help'. The toolbar contains various icons for file operations and navigation. The main window displays a table titled 'XML empTraining.xml'. The table has six columns: 'Relation Name', '# Tuples', 'eID/char', 'eLast/char', 'eFirst/char', 'eTitle/char', and 'eSalary/numeric'. The data rows are as follows:

| Relation Name | # Tuples | eID/char | eLast/char | eFirst/char | eTitle/char | eSalary/numeric |
|----------------|----------|----------|------------|-------------|--------------------------|-----------------|
| employee | 19 | '111' | 'Last111' | 'First111' | 'Database Administrator' | 75111 |
| trainingCourse | 14 | '222' | 'Last222' | 'First222' | 'Software Engineer' | 51722 |
| technologyArea | 4 | '321' | 'Last321' | 'First321' | 'Database Administrator' | 68321 |
| takes | 34 | '333' | 'Last333' | 'First333' | 'Sr Software Engineer' | 60333 |
| | | '345' | 'Last345' | 'First345' | 'Sr Software Engineer' | 59345 |

Εικόνα 6, μια βάση με τους πίνακες και τα πεδία της σε xml.

Στο menu Schema έχει κι άλλες επιλογές όπως είναι η επεξεργασία ενός πίνακα, η εισαγωγή-τροποποίηση-διαγραφή ενός πεδίου ή μιας πλειάδας κ.α. .

Σε περίπτωση που θέλουμε την εισαγωγή στοιχείων να την κάνουμε με ένα σκριπτάκι. Ανοίγουμε με τον επεξεργαστή κειμένου το αρχείο .rdb(ή xml) που είναι η βάση μας και γράφουμε όπως το παράδειγμα παρακάτω.

Π.χ.

```
@sailors(sid/numeric,sname/char,rating/numeric,age/numeric):sid
```

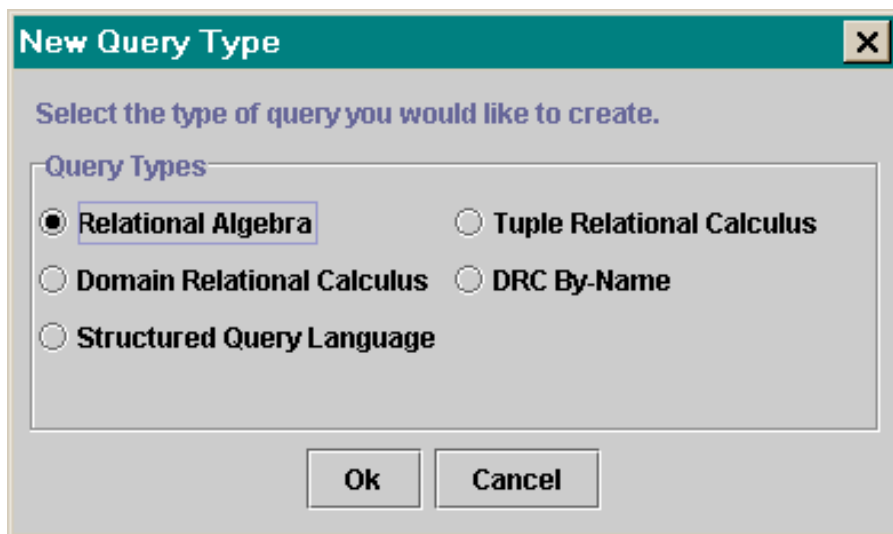
```
22,'Dustin',7,45
```

@όνομα_πίνακα(πεδίο1/τύπος_πεδίου,...,πεδίοN/τύπος_πεδίου):κλειδί

Η παραπάνω γραμμή δημιουργεί τον πίνακα και τη βάση ταυτόχρονα. Η βάση θα ονομάζεται ανάλογα με ποιο όνομα θα αποθηκεύσουμε το αρχείο. Όπως διακρίνουμε παραπάνω η σύνταξη και στη συνέχεια η δημιουργία της βάσης είναι πολύ απλή. Το πρόγραμμα υποστηρίζει δύο τύπους δεδομένων char ή numeric έτσι μειώνετε πολύ η πολυπλοκότητα. Στην ουσία του λέμε ότι «στην (AT) σχέση <όνομα> με χαρακτηριστικά (χαρακτηριστικό /τύπος(numeric ή char),...) και :<κύριο κλειδί> εισήγαγε τα παρακάτω δεδομένα διαχωρισμένα με κόμμα».

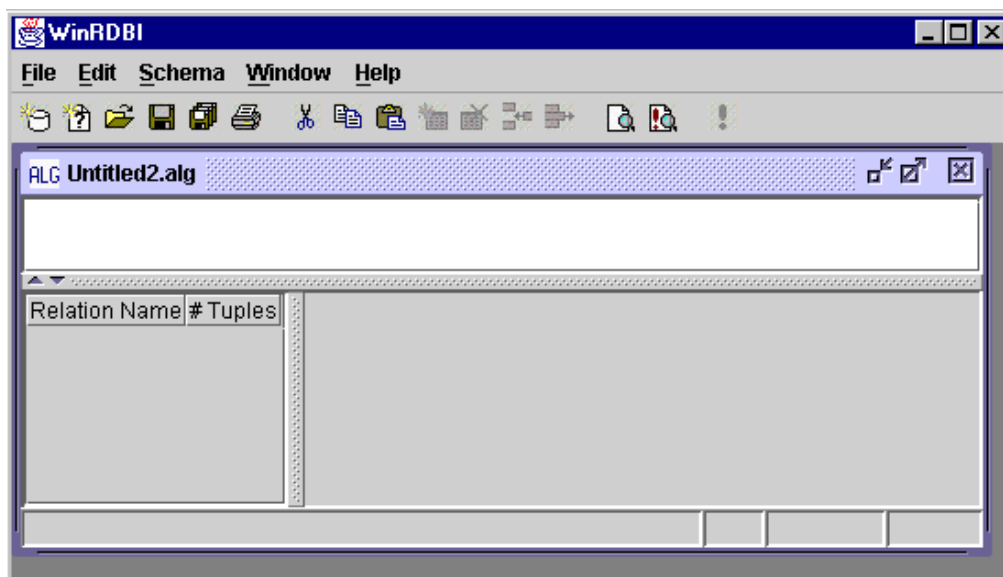
Εκτέλεση ερωτημάτων με το winRDBI

Για να εκτελέσουμε ένα ερώτημα οποιουδήποτε τύπου με το εργαλείο πηγαίνουμε στο μενού File και επιλέγουμε «new query».(εικόνα 7)

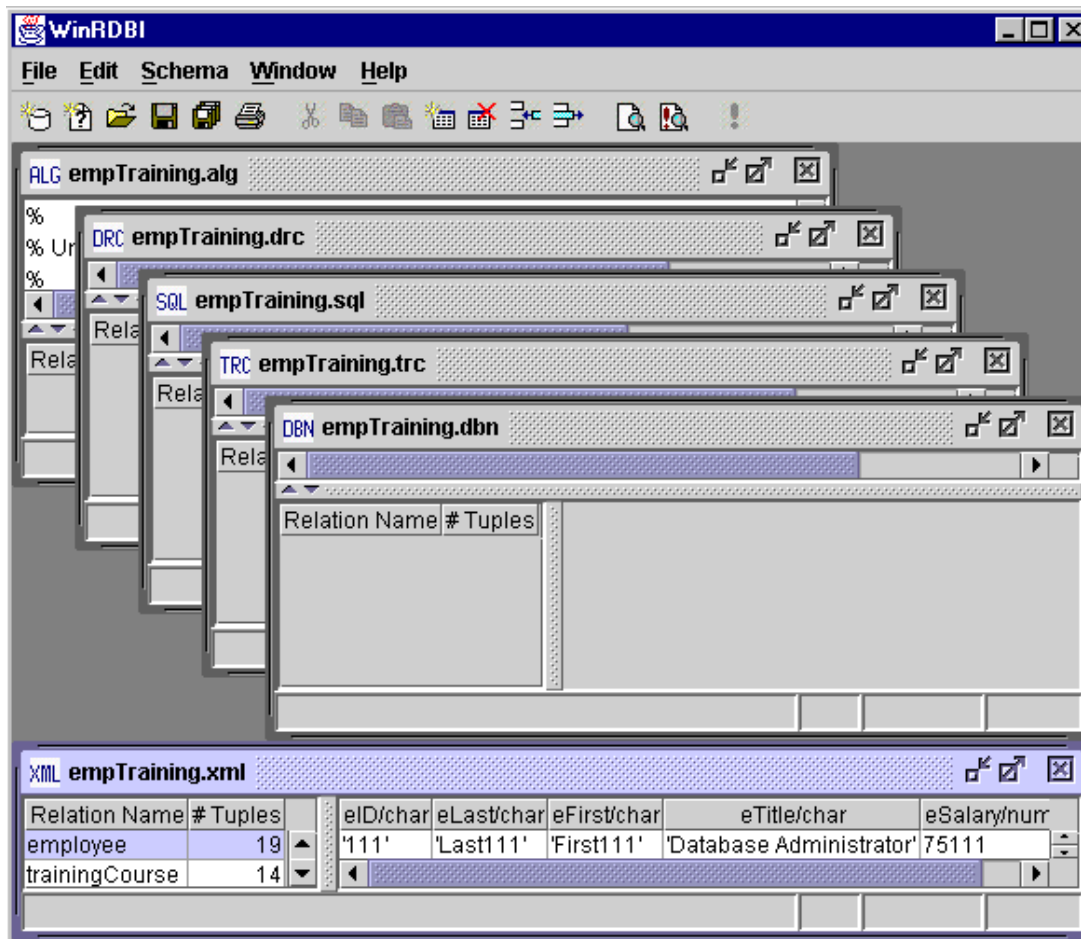


Εικόνα 7, παράθυρο επιλογής τύπου ερωτήματος.

Αφού επιλέξουμε τύπο, μας εμφανίζεται το query pane όπου και γράφουμε ένα ή περισσότερα ερωτήματα προς εκτέλεση. Στο ίδιο παράθυρο εμφανίζονται και τα αποτελέσματα (εικόνα 8 κ' 9).



Εικόνα 8, query pane τύπου σχεσιακής άλγεβρας.



Εικόνα 9, γενικό πλάνο του WinRDBI με όλα τα query panes.

Τέλος να αναφέρουμε ότι το WinRDBI έχει τη δυνατότητα να αποθηκεύσει τα ερωτήματα όπως και τις βάσεις σε κατάλληλα αρχεία ανάλογα με τον τύπο <name>.alg(σχεσιακή άλγεβρα), κ.α. και να τα εκτελέσει όταν του ζητηθεί δείχνοντας όλα τα ερωτήματα που περιλαμβάνει το «script». Επίσης να πούμε ότι και ο χρήστης μπορεί να φτιάξει τα δικά του σκριπτάκια με τη χρήση ενός απλού text editor αποθηκευόντάς τα με την κατάλληλη επέκταση (για καλύτερη κατανόηση ρίξτε μια ματιά στα επισυναπτόμενα σκριπτάκια).

ΠΡΟΣΟΧΗ!!!

Με την εξαίρεση της SQL οι υπόλοιπες γλώσσες είναι case sensitive. Οι σχέσεις και τα χαρακτηριστικά ονόματα πρέπει να αρχίζουν με ένα πεζό γράμμα π.χ. Relational Algebra (ALG)

qSelection := select eSalary > 100000 (employee); .

Οι σταθερές, όπως σε όλες τις γλώσσες, έτσι και στις βάσεις δεδομένων, είναι είτε single quoted string είτε αριθμητική σταθερά numeric. Οι μεταβλητές σε DRC, TRC and DBN επειδή είναι αναγνωριστικά θα αρχίζουν με κεφαλαίο.

Π.χ. TRC

qSelection := { **E** | employee(**E**) and **E**.eSalary > 100000 };

III. Εκτέλεση Ερωτημάτων

i. Δημιουργία ερωτήματος σχεσιακής άλγεβρας

Για να δημιουργήσουμε ένα ερώτημα σχεσιακής άλγεβρας επιλέγουμε New Query από το μενού File ή Open για να ανοίξουμε κάποιο script τύπου <όνομα>.alg. Θα παραθέσουμε παραδείγματα κώδικα παρμένα από το βιβλίο. Επίσης υπάρχει και το manual για τη δημιουργία των δικών σας ερωτημάτων.

ΠΡΟΣΟΧΗ!!

Οι πράξεις, *ένωση*, *τομή* και *διαφορά* απαιτούν οι τελεστές των σχέσεων να είναι συμβατοί, το καρτεσιανό γινόμενο απαιτεί τα ονόματα των μεταβλητών να είναι διαφορετικά και οι πράξεις της διαίρεσης και η α-join παραλείπονται ούτως ώστε οι μαθητευόμενοι να διερευνήσουν εναλλακτικές επιλογές με βάση τους βασικούς τελεστές.

```
<Query> → <Expression>
<Expression> → Identifier
                | ( <Expression> )
                | <Select_Expr>
                | <Project_Expr>
                | <Binary_Expr>
<Select_Expr> → select <Condition> ( <Expression> )
<Project_Expr> → project <Attributes> ( <Expression> )
<Binary_Expr> → ( <Expression> ) <Binary_Op> ( <Expression> )
<Condition> → <And_Condition>
                | <And_Condition> or <Condition>
<And_Condition> → <Rel_Formula>
                | <Rel_Formula> and <And_Condition>
<Rel_Formula> → <Operand> <Relational_Op> <Operand>
                | ( <Condition> )
<Binary_Op> → union
                | njoin
                | product
                | difference
                | intersect
<Attributes> → Identifier
                | Identifier, <Attributes>
<Operand> → Identifier
                | Constant
<Relational_Op> → =
```

| >
| <
| <>
| >=
| <=

ii. Εισαγωγικά ερωτήματα στη σχεσιακή άλγεβρα

Πράξη επιλογής:

$q_{\text{Selection}} := \text{select rating} > 8 (s_2);$

Πράξη προβολής:

$q_{\text{Projection}} := \text{project sname, rating} (s_2);$

Συνδυασμός επιλογής προβολής:

$q_{\text{SelProj}} := \text{project sname, rating} (\text{select rating} > 8 (s_2));$

Ένωση:

$q_{\text{Union}} := (\text{project sid, sname, rating, age} (s_1)) \text{ union } (\text{project sid, sname, rating, age} (s_2));$

Διαφορά:

$q_{\text{Difference}} := (\text{project sid, sname, rating, age} (s_1)) \text{ difference } (\text{project sid, sname, rating, age} (s_2));$

Τομή:

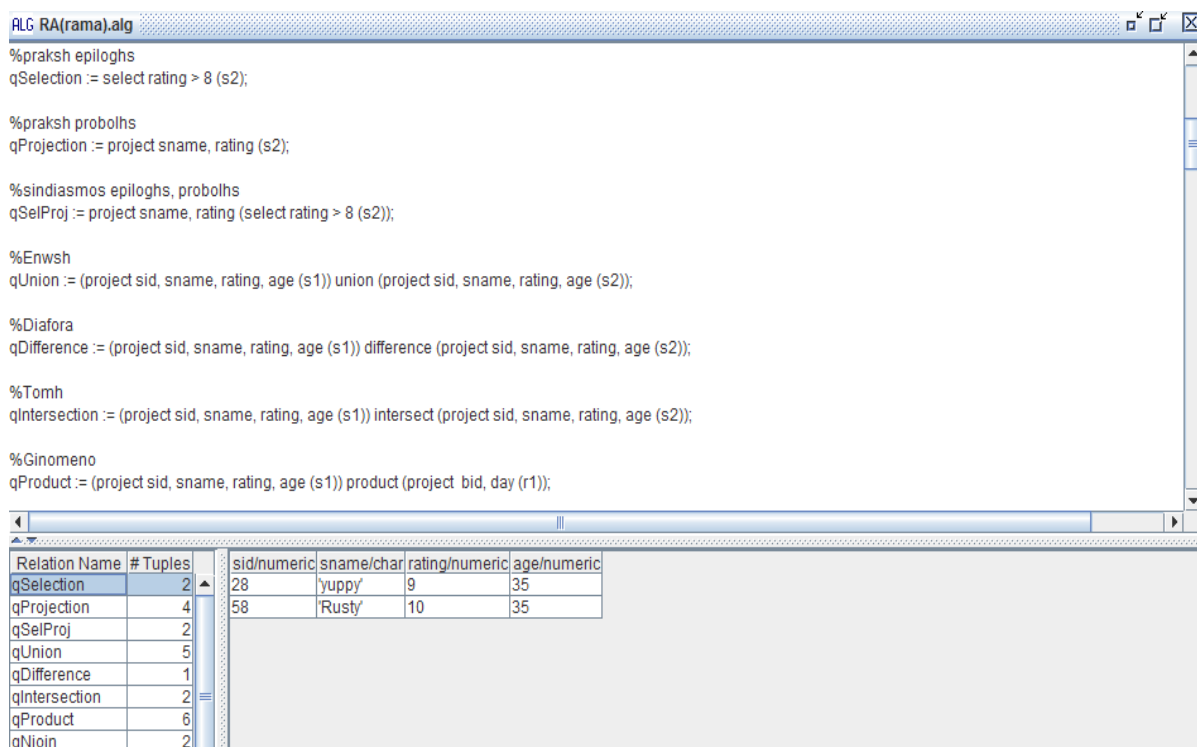
$qIntersection := (project\ sid, sname, rating, age\ (s1))\ intersect\ (project\ sid, sname, rating, age\ (s2));$

Καρτεσιανό Γινόμενο:

$qProduct := (project\ sid, sname, rating, age\ (s1))\ product\ (project\ bid, day\ (r1));$

Φυσική Σύζευξη:

$qNjoin := (s1)\ njoin\ (s2);$



```
%praksh epiloghhs
qSelection := select rating > 8 (s2);

%praksh probolhs
qProjection := project sname, rating (s2);

%sindiasmos epiloghhs, probolhs
qSelProj := project sname, rating (select rating > 8 (s2));

%Enwsh
qUnion := (project sid, sname, rating, age (s1)) union (project sid, sname, rating, age (s2));

%Diafora
qDifference := (project sid, sname, rating, age (s1)) difference (project sid, sname, rating, age (s2));

%Tomh
qIntersection := (project sid, sname, rating, age (s1)) intersect (project sid, sname, rating, age (s2));

%Ginomeno
qProduct := (project sid, sname, rating, age (s1)) product (project bid, day (r1));
```

| Relation Name | # Tuples | sid/numeric | sname/char | rating/numeric | age/numeric |
|---------------|----------|-------------|------------|----------------|-------------|
| qSelection | 2 | 28 | 'yuppy' | 9 | 35 |
| qProjection | 4 | 58 | 'Rusty' | 10 | 35 |
| qSelProj | 2 | | | | |
| qUnion | 5 | | | | |
| qDifference | 1 | | | | |
| qIntersection | 2 | | | | |
| qProduct | 6 | | | | |
| qNjoin | 2 | | | | |

Εικόνα 10, query pane στο οποίο εκτελέστηκαν τα παραπάνω ερωτήματα, επιλέχθηκαν και εμφανίζονται τα αποτελέσματα της επιλογής.

iii. Εκτέλεση ερωτημάτων σχεσιακής άλγεβρας

Q1: Να βρεθούν τα ονόματα των ναυτικών οι οποίοι έχουν κάνει κράτηση στη βάρκα με κωδικό 103;

```
temp1q1 := select bid=103 (reserves);
```

```
temp2q1 :=temp1q1 njoin (sailors);
```

```
q1 := project sname (temp2q1);
```

Εναλλακτικά:

```
q1alternative := project sname ((select bid=103 (reserves)) njoin (sailors));
```

Q2: Να βρεθούν τα ονόματα των ναυτικών οι οποίοι έχουν κάνει κράτηση σε κόκκινη βάρκα;

```
temp1q2 := select color = 'red' (boats);
```

```
temp2q2 := temp1q2 njoin (reserves);
```

```
temp3q2 := temp2q2 njoin (sailors);
```

```
q2 := project sname (temp3q2);
```

Εναλλακτικά:

```
q2alternative := project sname (((select color='red' (boats)) njoin (reserves)) njoin (sailors));
```

Q3: Να βρεθούν τα χρώματα των βαρκών για τις οποίες έχει κάνει κράτηση ο ναυτικός που ονομάζεται Lubber;

```
temp1q3 := select sname = 'Lubber' (sailors);
```

```
temp2q3 := temp1q3 njoin (reserves);
```

```
temp3q3 := temp2q3 njoin (boats);
```

```
q3 := project color (temp3q3);
```

Εναλλακτικά:

```
q3alternative := project color (((select sname='Lubber' (sailors)) njoin (reserves)) njoin (boats));
```

Q4: Να βρεθούν τα ονόματα όλων των ναυτικών κάθε ένας από τους οποίους έχει κάνει κράτηση τουλάχιστον σε μια βάρκα;

```
q4 := project sname ((sailors) njoin (reserves));
```

Q5: Να βρεθούν τα ονόματα των ναυτικών οι οποίοι έχουν κάνει κράτηση είτε σε κόκκινη είτε σε πράσινη βάρκα;

```
temp1q5 := select color = 'red' (boats);
```

```
temp2q5 := select color = 'green' (boats);
```

```
tempboats := (temp1q5) union (temp2q5);
```

```
temp3q5 := (tempboats) njoin (reserves);
```

```
temp4q5 := (temp3q5) njoin (sailors);
```

```
q5 := project sname (temp4q5);
```

Εναλλακτικά:

```
q5alternative := project sname (((select color='red' (boats)) union (select color='green' (boats))) njoin (reserves)) njoin (sailors));
```

Q6: Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κράτηση και σε κόκκινες και σε πράσινες βάρκες;

```
tempred:= project sid (((select color = 'red' (boats)) njoin (reserves)));  
tempgreen:= project sid (((select color = 'green' (boats)) njoin (reserves)));  
temp4q6 := (tempred) intersect (tempgreen);  
q6 := project sname ((temp4q6) njoin (sailors));
```

Q7: Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κρατήσεις σε τουλάχιστον δύο (διαφορετικές) βάρκες;

```
reservations := project sid, sname, bid ((sailors) njoin (reserves));  
reservationcopy(sid1, sname1, bid1) := (reservations);  
reservationpairs := (reservations) product (reservationcopy);  
q7 := project sname (select sid=sid1 and bid<>bid1 (reservationpairs));
```

ΠΡΟΣΟΧΗ:

Στο πρόγραμμα υπάρχει ένας περιορισμός που αφορά στην πράξη του γινομένου. Τα πεδία του πίνακα δεν πρέπει να έχουν το ίδιο όνομα με αυτά του άλλου, γι' αυτό κάνω τη μετονομασία όπως φαίνεται παραπάνω.

Q8: Να βρεθούν οι κωδικοί sid των ναυτικών που έχουν ηλικία μεγαλύτερη των 20 ετών και δεν έχουν κάνει κράτηση σε κόκκινη βάρκα;

```
temp1q8 := project sid (select age>20 (sailors));  
temp2q8 := reserves njoin boats;  
temp3q8 := project sid (select color = 'red' (temp2q8));  
temp4q8 := (temp1q8) difference (temp3q8);  
q8 := project sname ((sailors) njoin (temp4q8));
```

Q9: Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κρατήσεις σε όλες τις βάρκες;

```
temp1q9 := project sid (reserves);
```

```
temp1aq9 := project sid, bid (reserves);
```

```
temp2q9 := project bid (boats);
```

```
temp3q9:= temp1q9 product temp2q9;
```

```
temp4q9 := project sid ((temp3q9) difference (temp1aq9));
```

```
temp5q9 := project sid ((temp1q9) difference (temp4q9));
```

```
q9 := project sname ((temp5q9) njoin (sailors));
```

Στο ερώτημα 9 (έως και temp5q9) βλέπουμε πώς γίνεται η διαίρεση στο WinRDBI διότι για εκπαιδευτικούς λόγους οι δημιουργοί του δεν υλοποίησαν την πράξη της διαίρεσης αφήνοντας τον ασκούμενο να τη μάθει-υλοποιήσει μέσα από τις βασικές πράξεις σχεσιακής άλγεβρας.

Στο παρακάτω απόσπασμα κώδικα θα εξηγήσουμε πώς μπορεί να γίνει διαίρεση στο WinRDBI. Ο κώδικας βασίζεται σε sample code που παρέχεται μαζί με το πρόγραμμα. Με βάση το παρακάτω έγιναν τα δύο τελευταία queries. Πρώτα κατανοήσατε θεωρητικά τί είναι η διαίρεση και πώς μπορεί να εκφραστεί με άλλες πράξεις.

Έστω δύο πίνακες abTable και bTable, όπου για κάθε a υπάρχει ένα b στο πρώτο και ένας πίνακας όπου έχει μόνο τα b (ζητούμενα), εμείς ζητάμε στην ουσία εκείνα τα a όπου στον πίνακα abTable τους αντιστοιχεί και το b1 και το b2:

```
abTable(a/char, b/char):a, b
```

```
'a1','b1'
```

```
'a1','b2'
```

```
'a2','b2'
```

```
'a3','b1'
```

'a3','b2'

'a3','b3'

bTable(b/char):b

'b1'

'b2'

Ας προβάλουμε τώρα το a σε ένα ξεχωριστό πίνακα:

aTable := project a (abTable);

Το aTable τώρα περιέχει τα στοιχεία: a1, a2, a3.

Τώρα πρέπει να αντιστοιχίσουμε όλα τα a με όλα τα b(ζητούμενα), αυτό είναι επιτεύξιμο μέσω του γινομένου:

allaWithAllb := aTable product bTable;

'a1' 'b1'

Το allaWithAllb περιέχει τα παρακάτω στοιχεία:

'a1' 'b2'

'a2' 'b1'

'a2' 'b2'

'a3' 'b1'

'a3' 'b2'

Επόμενο βήμα είναι να βρούμε ποια a σχετίζονται με μερικά από τα b που ζητάμε, ΟΧΙ με όλα, για να τα αφαιρέσουμε αργότερα, γι' αυτό αφαιρούμε από το γινόμενο τον πίνακα abTable:

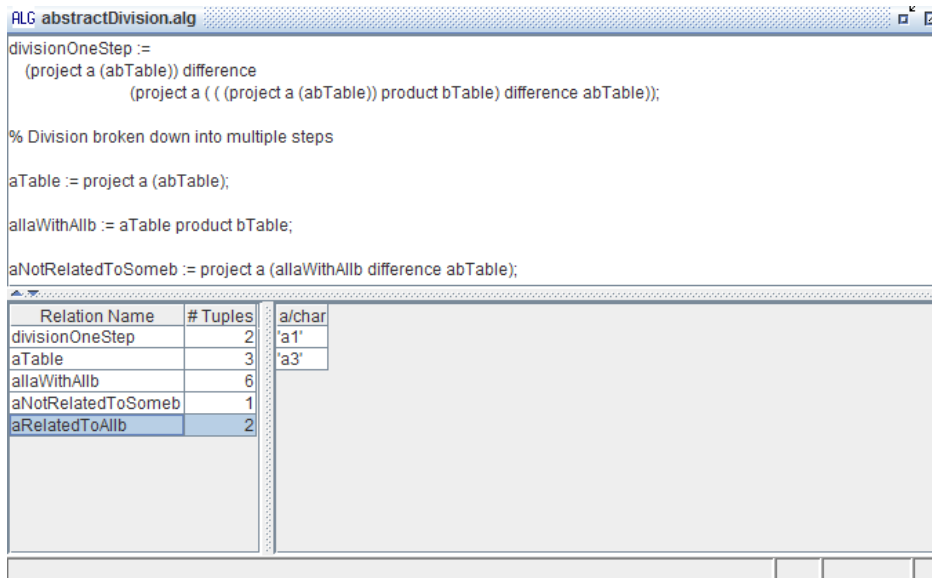
aNotRelatedToSomeb := project a (allaWithAllb difference abTable);

Το περιεχόμενο είναι το a2 και όντως αν κοιτάξουμε τον πίνακα abTable θα παρατηρήσουμε ότι στο a2 αντιστοιχεί μόνο το b2.

Τελευταίο βήμα είναι να βρούμε ποια a από τον aTable έχουν αντίστοιχα και b1 και b2. Αυτό είναι επιτεύξιμο αν αφαιρέσουμε από τον πίνακα μόνο με τα a, τον πίνακα που περιέχει τα a με μερική αντιστοιχία b.(εικόνα 11)

aRelatedToAllb := aTable difference aNotRelatedToSomeb;

Τελικό αποτέλεσμα:



Εικόνα 11

Εναλλακτικά:

divisionOneStep :=

(project a (abTable)) difference

(project a ((project a (abTable)) product bTable) difference

bTable));

Q10: Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κρατήσεις σε όλες τις βάρκες που έχουν όνομα Interlake;

temp1q10 := project sid (reserves);

temp1aq10 := project sid, bid (reserves);

temp2q10 := project bid (select bname = 'Interlake' (boats));

temp3q10 := (temp1q10) product (temp2q10);

temp4q10 := project sid ((temp3q10) difference (temp1aq10));

temp5q10 := project sid ((temp1q10) difference (temp4q10));

q10 := project sname ((temp5q10) njoin (sailors));

iv. Σχεσιακός λογισμός πλειάδων και εκτέλεση ερωτημάτων

Για να δημιουργήσουμε ένα ερώτημα «TRC» επιλέγουμε «New Query->Tuple Relational Calculus» από το μενού File ή Open για να ανοίξουμε κάποιο script « <όνομα>.trc ». Θα παραθέσουμε ερωτήματα παρμένα από το βιβλίο. Επίσης υπάρχει και το manual για τη δημιουργία των δικών σας ερωτημάτων.

$\langle \text{Query} \rangle \rightarrow \{ \langle \text{Vars_Attrs} \rangle \mid \langle \text{Formula} \rangle \}$

$\langle \text{Formula} \rangle \rightarrow \langle \text{And_Formula} \rangle$

$\mid \langle \text{And_Formula} \rangle \text{ or } \langle \text{Formula} \rangle$

$\langle \text{And_Formula} \rangle \rightarrow \langle \text{Sub_Formula} \rangle$

$\mid \langle \text{Sub_Formula} \rangle \text{ and } \langle \text{And_Formula} \rangle$

$\langle \text{Sub_Formula} \rangle \rightarrow \text{Identifier (Variable)}$

$\mid (\langle \text{Formula} \rangle)$

$\mid \text{not } \langle \text{Sub_Formula} \rangle$

$\mid \langle \text{Condition} \rangle$

$\mid \langle \text{Quantified_Formula} \rangle$

$\langle \text{Quantified_Formula} \rangle \rightarrow \langle \text{Quantifier} \rangle (\langle \text{Formula} \rangle)$

$\mid \langle \text{Quantifier} \rangle \langle \text{Quantified_Formula} \rangle$

$\langle \text{Quantifier} \rangle \rightarrow (\text{exists } \langle \text{Variables} \rangle)$

$\mid (\text{forall } \langle \text{Variables} \rangle)$

$\langle \text{Condition} \rangle \rightarrow \langle \text{Operand} \rangle \langle \text{Relational_Op} \rangle \langle \text{Operand} \rangle$

$\langle \text{Operand} \rangle \rightarrow \text{Identifier. Identifier}$

$\mid \text{Constant}$

$\langle \text{Vars_Attrs} \rangle \rightarrow \text{Identifier}$

$\mid \text{Identifier, } \langle \text{Vars_Attrs} \rangle$

| *Identifier.Identifier*

| *Identifier.Identifier, <Vars_Attrs>*

<Variables> → *Identifier*

| *Identifier, <Variables>*

<Relational_Op> → =

| >

| <

| <>

| >=

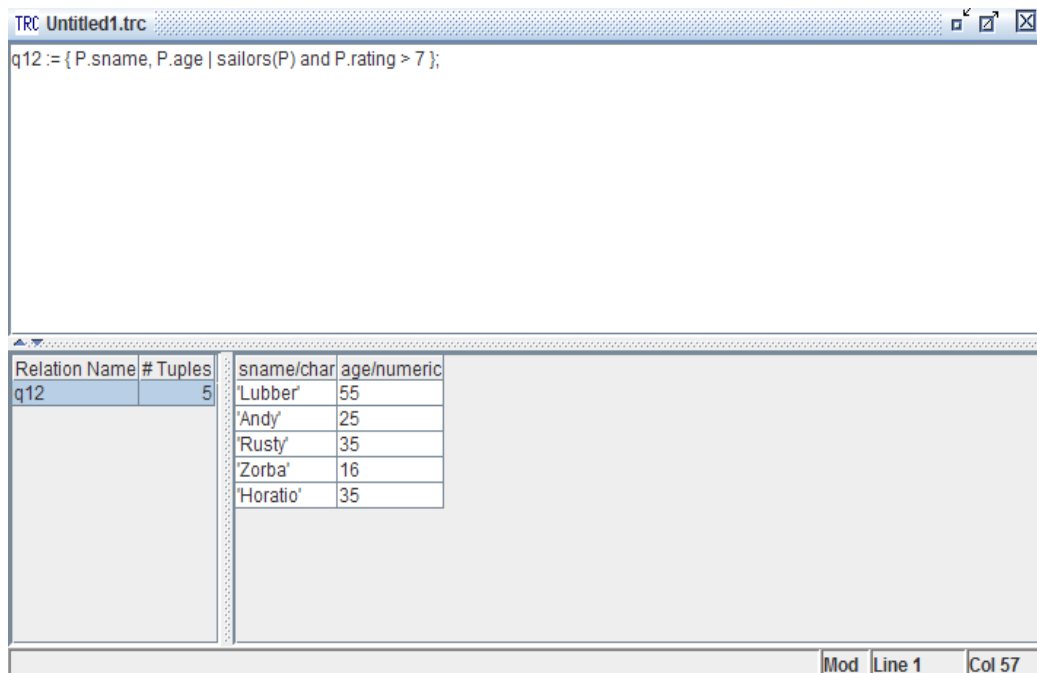
| <=

Q11: Να βρεθούν όλοι οι ναυτικοί που έχουν βαθμό (rating) μεγαλύτερο του 7;

q11 := { S | sailors(S) and S.rating > 7 };

Q12: Να βρεθούν τα ονόματα και οι ηλικίες των ναυτικών που έχουν τιμή διατίμησης μεγαλύτερη του 7;(εικόνα 12)

q12 := { P.sname, P.age | sailors(P) and P.rating > 7 };



The screenshot shows a window titled "TRC Untitled1.trc" containing the query: `q12 := { P.sname, P.age | sailors(P) and P.rating > 7 };`. Below the query, a table displays the results. The table has two columns: "Relation Name" and "# Tuples". The first row shows "q12" with 5 tuples. To the right, a detailed view of the tuples is shown with two columns: "sname/char" and "age/numeric".

| Relation Name | # Tuples | sname/char | age/numeric |
|---------------|----------|------------|-------------|
| q12 | 5 | 'Lubber' | 55 |
| | | 'Andy' | 25 |
| | | 'Rusty' | 35 |
| | | 'Zorba' | 16 |
| | | 'Horatio' | 35 |

Εικόνα 12, αποτελέσματα ερωτήματος Q12

Q13: Να βρεθούν το όνομα του ναυτικού, ο κωδικός της βάρκας και η ημερομηνία κράτησης για την κάθε μια κράτηση που καταχωρείται στη βάση;

q13 := { S.sname, R.bid, R.day | sailors(S) and reserves(R) and S.sid = R.sid };

Q14: Να βρεθούν οι ναυτικοί που έχουν κάνει κρατήσεις σε όλες τις κόκκινες βάρκες;

q14 := {S | sailors(S) and (exists R) (reserves(R) and S.sid=R.sid and (exists B) (boats(B) and R.bid=B.bid and B.color='red'))};

v. Σχεσιακός λογισμός πεδίων και εκτέλεση ερωτημάτων

Για να δημιουργήσουμε ένα ερώτημα «DRC» επιλέγουμε «New Query->Domain Relational Calculus» από το μενού File ή Open για να ανοίξουμε κάποιο script « <όνομα>.drc » . Θα παραθέσουμε ερωτήματα παρμένα από το βιβλίο. Επίσης υπάρχει και το manual για τη δημιουργία των δικών σας ερωτημάτων. Να σημειώσουμε ότι οι ανώνυμες μεταβλητές «_» και σταθερές μπορούν να χρησιμοποιηθούν σε μια DRC έκφραση.

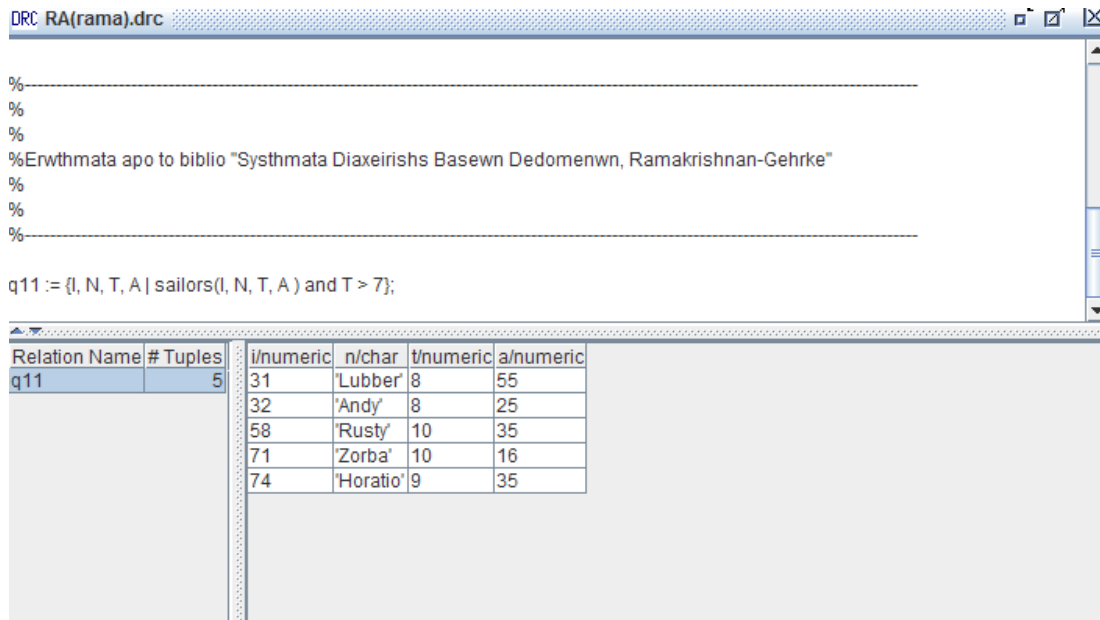
| | |
|----------------------|---|
| <Query> | →{ <Variables> <Formula> } |
| <Formula> | →<And_Formula> <And_Formula> or <Formula> |
| <And_Formula> | →<Sub_Formula> <Sub_Formula> and <And_Formula> |
| <Sub_Formula> | → <i>Identifier</i> (<Vars_Cons>) (<Formula>) not <Sub_Formula> <Condition> <Quantified_Formula> |
| <Quantified_Formula> | →<Quantifier> (<Formula>) <Quantifier> <Quantified_Formula> |
| <Quantifier> | →(exists <Variables>) (forall <Variables>) |
| <Condition> | →<Operand> <Relational_Op> <Operand> |
| <Operand> | → <i>Identifier</i> <i>Constant</i> |
| <Variables> | → <i>Identifier</i> <i>Identifier</i> , <Variables> |

<Vars_Cons> → Identifier
 | Identifier, <Vars_Cons>
 | Constant
 | Constant, <Vars_Cons>

<Relational_Op> → =
 | >
 | <
 | <>
 | >=
 | <=

Q11: Να βρεθούν όλοι οι ναυτικοί που έχουν βαθμό (rating) μεγαλύτερο του 7;
 (εικόνα 13)

q11 := {I, N, T, A | sailors(I, N, T, A) and T > 7};



Εικόνα 13, Αποτελέσματα ερωτήματος Q11

vi. Δημιουργία και εκτέλεση ερωτημάτων SQL

Για την SQL το μόνο που έχουμε να παραθέσουμε είναι κάποιιο κανόνες για την εκτέλεση των ερωτημάτων στο winRDBI. Κάποια παραδείγματα επισυνάπτονται στο σκριπτάκι «RA(rama).sql».

- Αν και η γλώσσα επιτρέπει φωλιασμένα ερωτήματα και συσσωματώσεις, ένα φωλιασμένο υποερώτημα δεν μπορεί να υποστεί συσσωμάτωση, γι' αυτό είναι καλό να σπάτε τέτοια ερωτήματα σε μικρότερα. Για παράδειγμα δείτε το παρακάτω ερώτημα.

Θεωρήστε το ερώτημα για την εύρεση του ναυτικού ο οποίος έχει τη μεγαλύτερη ηλικία. Αν και το πρώτο ερώτημα είναι έγκυρο δεν επιτρέπει το εργαλείο να εκτελεστεί το ερώτημα, αλλά μπορεί να εκτελεστεί «σπάζοντάς το» σε απλούστερα.

```
1. select sname, age
   from sailors
   where age = (select max(age)
                from sailors);
2. maxage(maxAge) := select max(age)
                    from sailors;
   select name, age
   from employee, maxage
   where age=maxAge;
```

- Το εργαλείο επιτρέπει τη χρήση φωλιασμένων εκφράσεων σύζευξης πινάκων στη FROM, αλλά η αναφορά στα χαρακτηριστικά πρέπει να γίνεται μέσω ψευδωνύμων. Παρακάτω παρατίθεται παράδειγμα που το πρώτο ερώτημα δεν εκτελείται από το εργαλείο.

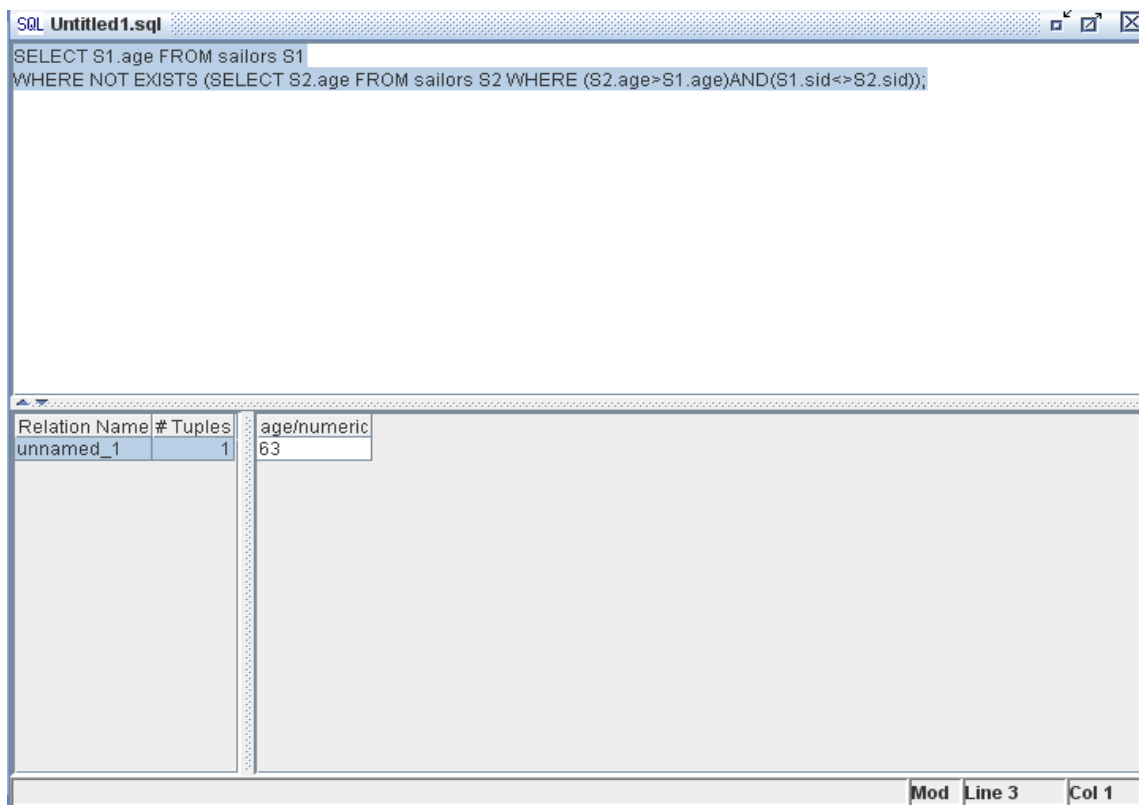
```
1. select distinct EmpDB.eID, EmpDB.eLast, EmpDB.eFirst, EmpDB.eTitle
   from (employee natural join (takes natural join dbCourse)) as EmpDB;
2. select distinct E.eID, E.eLast, E.eFirst, E.eTitle
   from (employee E natural join (takes T natural join dbCourse d));
```


Τέλος η πρόταση ORDER BY έχει ένα περιορισμό, αυτός είναι ότι γίνεται ταξινόμηση μόνο κατά αύξουσα ή φθίνουσα σειρά.

<Order_Clause> □ **order by** <Order_Varlist> [**asc** | **desc**]

Q16: Να βρεθεί ο ναύτης με τη μεγαλύτερη ηλικία;

```
SELECT S1.age FROM sailors S1
WHERE NOT EXISTS (SELECT S2.age FROM sailors S2 WHERE
(S2.age>S1.age)AND(S1.sid<>S2.sid));
```



The screenshot shows a window titled "SQL Untitled1.sql" containing the following SQL query:

```
SELECT S1.age FROM sailors S1
WHERE NOT EXISTS (SELECT S2.age FROM sailors S2 WHERE (S2.age>S1.age)AND(S1.sid<>S2.sid));
```

Below the query editor, a table displays the result of the query:

| Relation Name | # Tuples | age/numeric |
|---------------|----------|-------------|
| unnamed_1 | 1 | 63 |

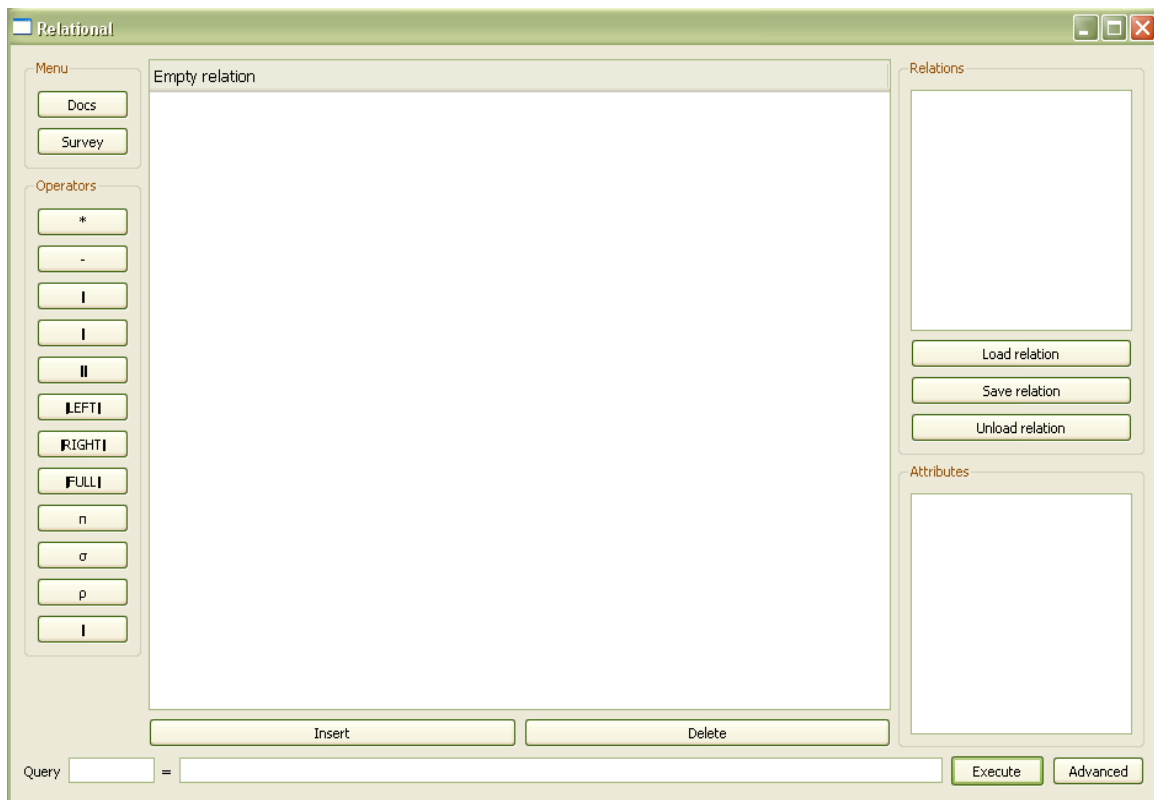
The status bar at the bottom of the window indicates "Mod Line 3 Col 1".

Εικόνα 14, Αποτελέσματα ερωτήματος Q17

ΚΕΦΑΛΑΙΟ 4

I. ΕΙΣΑΓΩΓΗ-Relational

Το Relational δημιουργήθηκε με σκοπό την εξάσκηση πάνω στα ερωτήματα σχεσιακής άλγεβρας. Συγκεκριμένα αποτελεί ένα εργαλείο για να διαπιστώσουμε μόνο αν ένα ερώτημα είναι σωστό ή όχι. Σε καμία περίπτωση δεν μπορεί να αντικαταστήσει τη γνώση στη σχεσιακή άλγεβρα. Απλά, ίσως μας κάνει τη ζωή πιο εύκολη.(εικόνα 15)

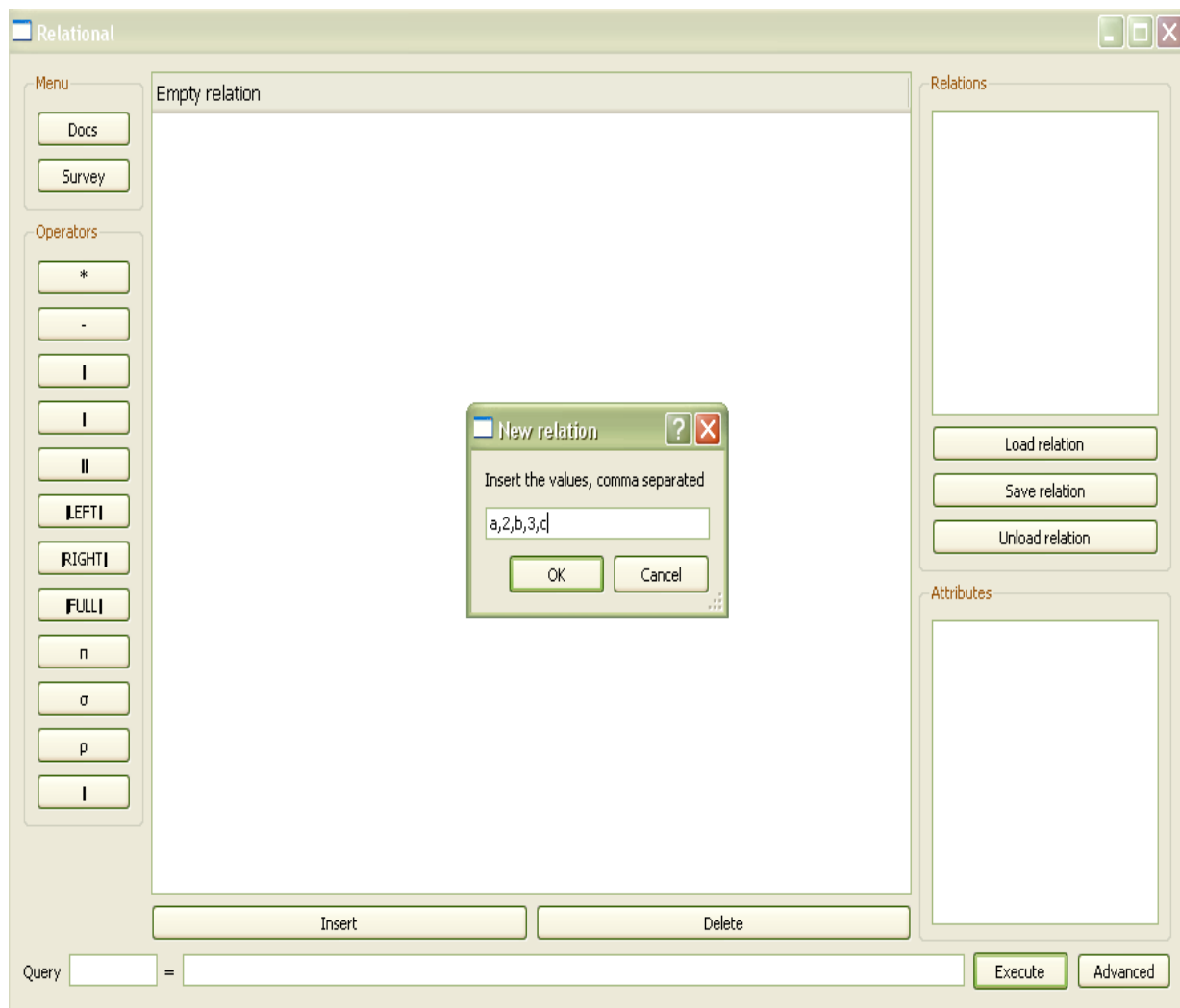


Εικόνα 15, αρχική οθόνη

II. Λειτουργικότητα και Βασικές Εντολές

Δημιουργία-Φόρτωμα-Σώσιμο πινάκων-σχέσεων

Η δημιουργία των πινάκων μπορεί να γίνει με δύο τρόπους. Πρώτος τρόπος είναι να κάνουμε κλικ πάνω στο κουμπί «insert» και να βάλουμε τις τιμές διαχωρισμένες με κώμα(εικόνα 16).



Εικόνα 16, δημιουργία νέας σχέσης-πίνακα.

Δεύτερος τρόπος είναι με έναν text editor, να δημιουργήσουμε scripts, όπως φαίνεται στο παράδειγμα παρακάτω.

sid,sname,rating,age

22,Dustin,7,45

31,Lubber,8,55

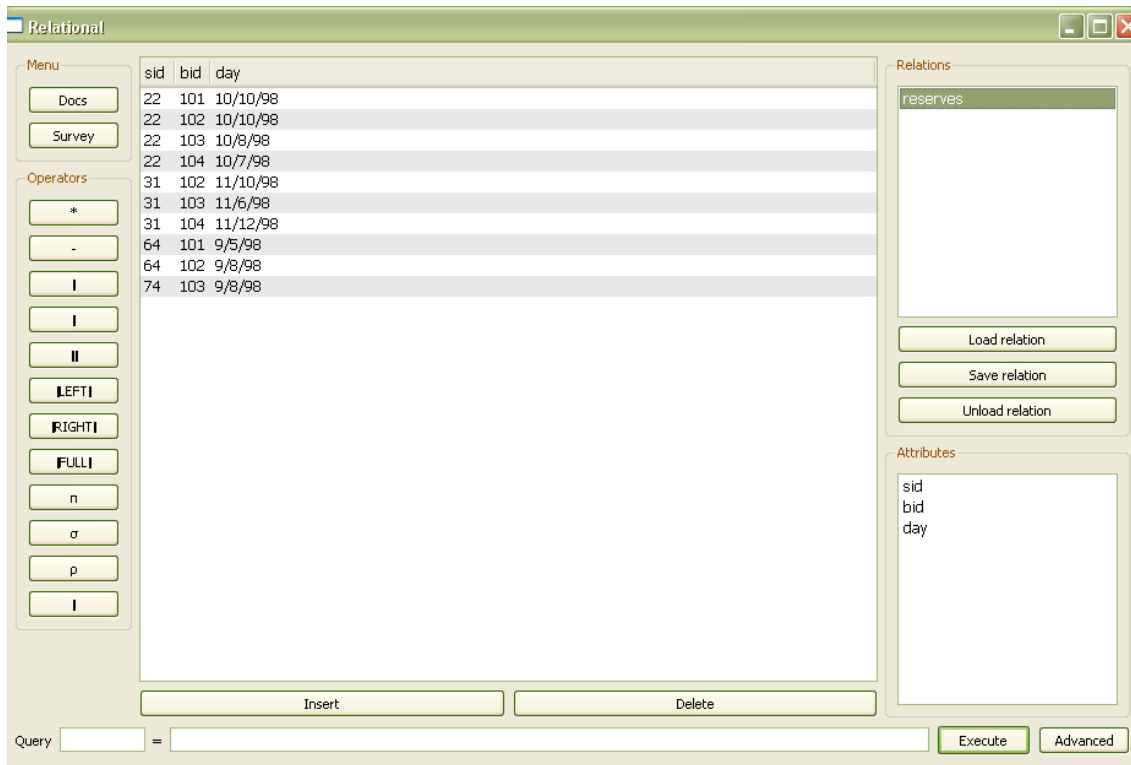
58,Rusty,10,35

Στη πρώτη γραμμή γράφουμε τα ονόματα των πεδίων και από κάτω τις τιμές. Ο διαχωρισμός πρέπει να γίνεται με κόματα. Όταν τελειώσουμε με την εισαγωγή των στοιχείων, το αρχείο πρέπει να το αποθηκεύσουμε με κατάληξη «**όνομα_αρχείου.csv**». Γενικά το πρόγραμμα δουλεύει τα αρχεία τύπου csv.

ΠΡΟΣΟΧΗ!!! Ανάμεσα από τις τιμές και τα κόματα δε βάζετε κενά «spaces».

Για να φορτώσουμε έναν πίνακα-σχέση κάνουμε κλικ στο κουμπί «Load relation». Αφού δώσουμε ένα όνομα στον πίνακα-σχέση που θα επεξεργαστούμε, πατάμε «ok», ύστερα για να εμφανίσουμε τα χαρακτηριστικά που απαρτίζεται αλλά και τα στοιχεία, κάνουμε κλικ στο όνομά του πάνω δεξιά(εικόνα 13). Για να σώσουμε οποιαδήποτε αλλαγή στον πίνακα-σχέση κάνουμε κλικ στο «save relation».

Ο συμβολισμός των τελεστών θα γίνεται με τη μορφή που υπάρχει στο βιβλίο με εξαίρεση μερικών. Μπορείτε να τους δείτε στην εικόνα 15 κ' 17 αριστερά που λέει operators.



Εικόνα 17, Πάνω δεξιά το όνομα που δώσαμε στο φόρτωμα φωτίζεται αφού έχει επιλεγθεί και μας εμφανίζονται στο κέντρο τα στοιχεία και κάτω αριστερά τα χαρακτηριστικά.

Πράξεις που υποστηρίζει

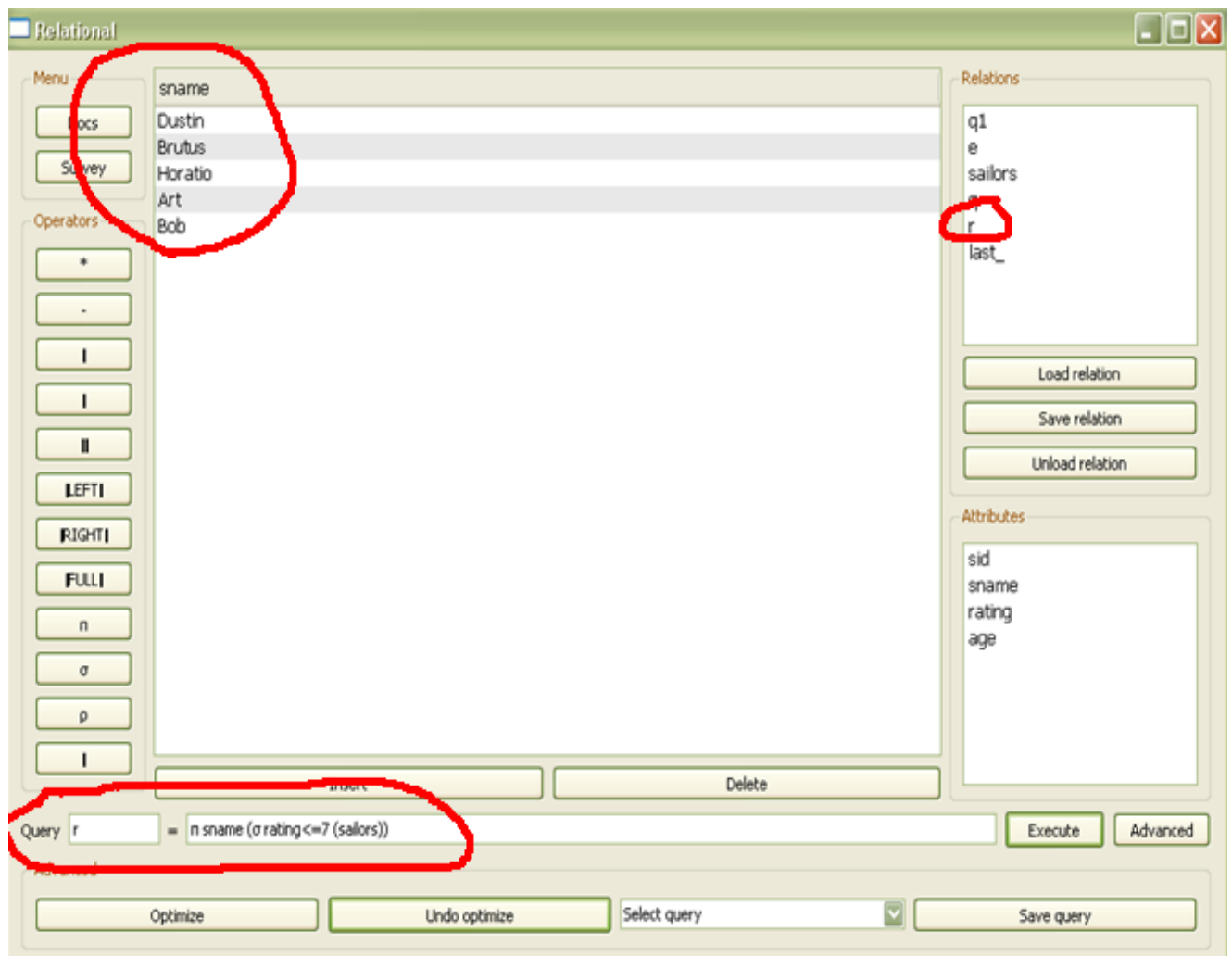
Οι πράξεις που υποστηρίζει είναι οι εξής: *γινόμενο, ένωση, διαφορά, τομή, φυσική σύζευξη, left outer join, right outer join, full outer join, προβολή, επιλογή, και μετονομασία.*

Υποστηριζόμενες συνθήκες

>, <, ==, != ή <>, >=, <=

III. Εκτέλεση ερωτημάτων

Ένα παράδειγμα εκτέλεσης ερωτήματος παρουσιάζεται στις εικόνες 18.



Εικόνα 18, εκτέλεση ερωτήματος σχεσιακής άλγεβρας.

Q1: Να βρεθούν τα ονόματα των ναυτικών οι οποίοι έχουν κάνει κράτηση στη βάρκα με κωδικό 103;

temp1q1 = σ sname==103 (reserves)

temp2q1 = (temp1q1) || (sailors)

q1 = π sname (temp2q1)

Q2: Να βρεθούν τα ονόματα των ναυτικών οι οποίοι έχουν κάνει κράτηση σε κόκκινη βάρκα;

temp1q2 = σ color=='red' (boats)

temp2q2 = π bid (temp1q2)

temp3q2 = (temp2q2) * (reserves)

temp4q2 = σ bid (temp3q2) == bid (reserves)

temp5q2 = (temp4q2) * (sailors)

temp6q2 = σ sid (temp5q2) == sid (sailors)

q2 = π sname (temp6q2)

Q3: Να βρεθούν τα χρώματα των βαρκών για τις οποίες έχει κάνει κράτηση ο ναυτικός που ονομάζεται Lubber;

temp1q3 = σ sname = 'Lubber' (sailors)

temp2q3 = (temp1q3) * (reserves)

temp3q3 = σ sid (temp2q3) == sid (reserves)

temp4q3 = (temp3q3) * (boats)

temp5q3 = σ bid (temp4q3) == bid (boats)

q3 = π color (temp5q3)

Q4: Να βρεθούν τα ονόματα όλων των ναυτικών κάθε ένας από τους οποίους έχει κάνει κράτηση τουλάχιστον σε μια βάρκα;

```
temp1q4 = (sailors) * (reserves)
temp2q4 = σ sid (temp1q4) == sid (reserves)
q4 = π sname (temp2q4)
```

Q5: Να βρεθούν τα ονόματα των ναυτικών οι οποίοι έχουν κάνει κράτηση είτε σε κόκκινη είτε σε πράσινη βάρκα;

```
temp1q5 = σ color == 'red' (boats)
temp2 q5= σ color == 'green' (boats)
temp3q5 = (temp1q5) | (temp2q5)
temp4q5 = (temp3q5) * (reserves)
temp5q5 = σ bid (temp4q5) == bid (reserves)
temp6q5 = (temp5q5) * (sailors)
temp7 q5= σ sid (temp6q5) == sid (sailors)
q5 = π sname (temp7q5)
```

Q6: Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κράτηση και σε κόκκινες και σε πράσινες βάρκες;

```
temp1q6 = σ color = 'red' (boats)
temp2q6 = (temp1q6) || (reserves)
temp3q6 = π sid (temp2q6)
temp4q6 = σ color = 'green' (boats)
temp5q6 = (temp4q6) || (reserves)
temp6q6 := π sid (temp5q6)
temp7q6 = (temp3q6) | (temp6q6)
temp8q6 = (temp7q6) || (sailors)
q6 = π sname (temp8q6)
```


Q7: Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κρατήσεις σε τουλάχιστον δύο (διαφορετικές) βάρκες;

temp1q7 = (sailors) || (reserves);

temp2q7 = π sid, sname, bid (temp1q7)

temp3q7 = temp2q7

q7 = (temp2q7) * (temp3q7)

Q8: Να βρεθούν οι κωδικοί sid των ναυτικών που έχουν ηλικία μεγαλύτερη των 20 ετών και δεν έχουν κάνει κράτηση σε κόκκινη βάρκα;

temp1q8 = σ sailors.age > '20' (sailors)

temp2q8 = π sid (temp1q8)

temp3q8 = σ boats.color = 'red' (boats)

temp4q8 = (temp3q8) * (reserves)

temp5q8 = σ bid (temp4q8) == bid (reserves)

temp6q8 = (temp5q8) * (sailors)

temp7q8 = σ sid (temp6q8) == sid (sailors)

temp8q8 = π sid (temp7q8)

q8 = π sid (temp8q8)

Q9: Να βρεθούν τα ονόματα των ναυτικών που έχουν κάνει κρατήσεις σε όλες τις βάρκες;

temp1q9 = π sid, bid (reserves)

temp2q9 = π bid (boats)

temp3q9 = (temp1q9) | (temp2q9)

temp4q9 = (temp3q9) * (sailors)

temp5q9 = σ sid (temp4q9) == sid (sailors)

q9 = π sname (temp5q9)

ΚΕΦΑΛΑΙΟ 5

I. ΕΙΣΑΓΩΓΗ

Στο τελευταίο κεφάλαιο της πτυχιακής αυτής θα σχολιάσουμε τα αποτελέσματα της εκτέλεσης των ερωτημάτων πάνω στα τρία προγράμματα, μέσα από τα οποία θα γίνει η σύγκριση. Θα παρουσιαστούν τα συν και τα πλην του καθενός.

II. ΣΥΜΠΕΡΑΣΜΑΤΑ

i. winRDBI

Το winRDBI ένα πρόγραμμα με ευχάριστο γραφικό περιβάλλον ανεπτυγμένο σε java και με πάρα πολλές δυνατότητες. Πρώτα θα αναφερθούμε στις πολλές δυνατότητες που έχει, κυρίως στη δημιουργία των βάσεων δεδομένων αλλά και στην εκτέλεση ερωτημάτων όχι μόνο της σχεσιακής άλγεβρας αλλά και άλλων όπως είναι η SQL.

Στα παραδείγματα που είδατε παρατηρήσατε όχι μόνο εκτέλεση ερωτημάτων σχεσιακής άλγεβρας αλλά και άλλων συναφών του σχεσιακού μοντέλου. Να υπενθυμίσουμε ότι είναι MDI (Multiple Document Interface = δυνατότητα επεξεργασίας πολλών εγγράφων ταυτόχρονα) περιβάλλον και αυτό το είδατε από τα screen shots.

Όσον αφορά τη σχεσιακή άλγεβρα να πούμε ότι οι πράξεις περιορίζονται στις βασικές(για καθαρά μαθησιακούς λόγους) επειδή οι υπόλοιπες δευτερεύουσες βασίζονται σε αυτές και μπορούν γραφούν «περιφραστικά» με βάση αυτές. Οι βάσεις δεδομένων μπορούν να δομηθούν είτε ως σχεσιακές είτε ως xml. Στην xml έχουμε ταυτόχρονα και μια περιγραφή του σχήματος της βάσης αλλά και των δεδομένων που θα υπάρξουν. Επίσης μπορεί να γίνει μετονομασία πεδίων ενός πίνακα όπως είδατε στο ερώτημα «Q7». Η ταυτόχρονη εκτέλεση πολλών ερωτημάτων είναι ένα ακόμα ατού το οποίο το κάνει ευέλικτο και εύχρηστο όταν έχουμε μεγάλο αριθμό ερωτημάτων και με πολλά δεδομένα. Το manual του επίσης είναι υπερπλήρες σε σχέση με τα άλλα δύο.

ii. winLeap

Το winLeap αποτελεί την έκδοση του εργαλείου Leap για windows. Ίσως αυτός είναι ένας από τους λόγους που δε δίνει η κοινότητα ελεύθερου λογισμικού και τόσο σημασία στην περαιτέρω εξέλιξη της έκδοσης αυτής. Η εκτέλεση ερωτημάτων περιορίζεται μόνο σε σχεσιακής άλγεβρας ερωτήματα τα οποία πρέπει να γράφονται ένα-ένα στη γραμμή εντολών του προγράμματος και με διαφορετικό όνομα, εκτός αν διαγράφονται στο τέλος κάθε ερωτήματος, το οποίο και διαπιστώσατε στην εκτέλεση. Όπως και το winRDBI έτσι και αυτό δεν υποστηρίζει όλες τις πράξεις. Ακόμα δεν υποστηρίζει τη μετονομασία κάτι που στο winRDBI υποστηρίζετε. Τέλος να πούμε ότι δεν υποστηρίζει ούτε σχεσιακό λογισμό πλειάδων, ούτε πεδίων αντίστοιχα, ούτε SQL.

iii. Relational

Το πρόγραμμα αυτό αποτελεί τη δημιουργία δύο φοιτητών από το «**Università di Catania-Department of Mathematics and Computer Science**». Αποτελείται από ένα ευχάριστο γραφικό περιβάλλον, οι σχεσιακοί τελεστές βρίσκονται οι περισσότεροι υπό μορφή button στα αριστερά, ώστε να μπορούν να επιλεγθούν. Στα δεξιά βρίσκουμε τους πίνακες αλλά και τα πεδία τους. Το εργαλείο αυτό περιλαμβάνει και κάποιους τελεστές πέρα των βασικών πράξεων, όπως είναι ο «outer join right operator». Ακόμα περιλαμβάνει και τον τελεστή για τη μετονομασία, κάτι που υπάρχει μόνο στο winRDBI. Από τη μια περιλαμβάνει query optimizer από την άλλη και αυτό όπως το winLeap εκτελεί μόνο ερωτήματα σχεσιακής άλγεβρας. Τέλος έχει τη δυνατότητα να το χρησιμοποιήσουν και οι προγραμματιστές δηλαδή να χρησιμοποιήσουν στοιχεία από τη βιβλιοθήκη-πακέτο Relational ώστε να τα ενσωματώσουν στον κώδικά τους εικόνα 19.

```
#Importing module  
import relation  
  
#Loading relations  
t=relation.relation("samples/skills.tlb")  
q=relation.relation("samples/people.tlb")  
  
#join them  
k=t.join(q)  
  
#Projection of names and skills  
k=k.projection("name","skill")  
print k
```

Εικόνα 19, εισαγωγή πακέτου relational και χρήση του με python.

| | winLeap | winRDBI | Relational |
|---|---------|---------|------------|
| Σχεσιακή άλγεβρα | X | X | X |
| Σχεσιακό λογισμό πλειάδων | - | X | - |
| Σχεσιακό λογισμό πεδίων | - | X | - |
| SQL | - | X | - |
| Τελεστής «ρ» | - | X | X |
| Εκτέλεση πολλαπλών ερωτημάτων | - | X | - |
| Ανάγκη εγκατάστασης «install» | - | X | X |
| Query optimizer | - | - | X |
| Παραθυρικό περιβάλλον | - | X | X |
| Δείκτης ευχρηστίας(1πολύ μικρή-5 πολύ μεγάλη) | 2 | 5 | 4 |
| Ενσωμάτωση ως βιβλιοθήκη σε κώδικα | - | - | X |

Πίνακας 7, Συγκριτικός πίνακας προγραμμάτων.

ΒΙΒΛΙΟΓΡΑΦΙΑ

1. Συστήματα Διαχείρισης Βάσεων Δεδομένων, 2^η έκδοση, Raghu Ramakrishnan, Johannes Gehrke: Εκδόσεις Τζιόλα 2002.
2. Database systems, A practical approach to Design, Implementation and Management, 4th edition, Thomas Connolly, Carolyn Begg: Addison Wesley 2004.
3. Understanding Relational Database Query Languages., Dietrich, S. W. Upper Saddle River: Prentice Hall, 2001.
4. Principles of Database and Knowledge-Base systems volume 1, Jeffrey D. Ullman Stanford university: Computer science press.
5. Data models, database languages and database management systems, Gottfried Vossen Rheinisch-Westfälische Technische Hochschule Aachen : Addison Wesley publishing company.
6. An Educational Tool for Relational Query Languages, Dietrich, S. W. ,Computer Science Education, Vol. 4, 1993, pp. 157-184.
7. WinRDBI: A Windows based Relational Database Educational Tool, Dietrich S. W., Eckert E., and Piscator K., SIGCSE, 1997, pp.126-130.

URLs

1. <http://www.eas.asu.edu/~winrdbi/>
2. <http://leap.sourceforge.net/>
3. <http://galileo.dmi.unict.it/wiki/relational/doku.php>
4. <http://www.cs.uoi.gr/~pitoura/courses/db/db09/index.html>
5. http://en.wikipedia.org/wiki/Relational_algebra