

Σύστημα μηχανογράφησης Βιβλιοθήκης

Εργασία Τεχνολογίας Βάσεων Δεδομένων

1 Ιανουαρίου 2022

ΔΙΠΑΕ Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων

185192 Γιώργος Καραχρήστος, 185296 Βασίλης Τσαντίλης Ρέις, 185307 Νικολέττα Φωτοπούλου

Περιεχόμενα

Πρόλογος	3
Λειτουργία Εφαρμογής	3
Διεπαφή.....	3
Βάση Δεδομένων.....	5
Κώδικας Java.....	9
Log file	12
Συμπεράσματα.....	16

Πρόλογος

Στα πλαίσια του μαθήματος Τεχνολογία Βάσεων Δεδομένων επιλέξαμε να υλοποιήσουμε για εργασία μία εφαρμογή για σύστημα μηχανογράφησης βιβλιοθήκης. Η εφαρμογή απευθύνεται στο προσωπικό της βιβλιοθήκης για να υλοποιούν την οργάνωση των μελών, των βιβλίων, των υλικών υποδομών, των υπαλλήλων και των δανεισμών.

Γίνεται χρήση της βάσης δεδομένων Oracle μέσω του SQL Developer, όπως επίσης χρήση της Java Swing μέσω του Apache Netbeans IDE.

Θα ακολουθήσει παρουσίαση των λειτουργιών της εφαρμογής, της βάσης, του διαγράμματος ER και των log files.

Λειτουργία Εφαρμογής

Η εφαρμογή έχει δημιουργηθεί για να υλοποιεί κάποιες συγκεκριμένες λειτουργίες. Γίνεται διαχείριση των βιβλίων, των μελών, του προσωπικού, των ραφιών και των δανεισμών. Για καθένα από αυτά τα στοιχεία, δίνεται η δυνατότητα εισαγωγής, διαγραφής, τροποποίησης και εμφάνισης των δεδομένων στην διεπαφή. Επίσης, έχουν δημιουργηθεί 10 δυναμικά ερωτήματα, που ο χρήστης επιλέγοντας από ένα combo box με ήδη υπάρχοντα στοιχεία μπορεί να πάρει μία απάντηση στο ερώτημα που υλοποιείται. Τέλος, με την επιλογή εμφάνισης των log files ο χρήστης μπορεί να δει τις ενέργειες εισαγωγή, διαγραφή και τροποποίηση για καθένα από τα πέντε πίνακες που προαναφέρθηκαν.

Διεπαφή

Με το άνοιγμα της εφαρμογής εμφανίζονται όλες οι επιλογές που έχει ο χρήστης ως προς τις ενέργειές του. Με το κουμπί Έξοδος, μετά από ένα μήνυμα επιβεβαίωσης ο χρήστης μπορεί να τερματίσει την λειτουργία της εφαρμογής.



Για την εισαγωγή, την επεξεργασία, την εμφάνιση, την διαγραφή και το log file το παράθυρο που ανοίγεται είναι παρόμοιο, καθώς σε όλα δίνεται η επιλογή για ποιον από τους πέντε πίνακες θέλει ο χρήστης να εκτελέσει την ενέργεια που επέλεξε. Υπάρχει πάντα το κουμπί εξόδου αλλά πλέον και το κουμπί πίσω, που θα επιτρέψει την επιστροφή στο αρχικό μενού.



Ανάλογα με το τι επιλέχθηκε θα εμφανιστεί η ανάλογη φόρμα. Για την εισαγωγή θα εμφανίζονται πλαίσια για συμπλήρωση των στοιχείων, στην διαγραφή ένα combo box για επιλογή ID του στοιχείου που επιθυμεί ο χρήστης να διαγράψει, στην επεξεργασία η ίδια φόρμα που ανοίγει και στην εισαγωγή, αλλά επιλέγοντας ένα υπάρχον ID θα συμπληρωθούν αυτόματα όλα τα υπόλοιπα στοιχεία στα πλαίσια για να γίνει η επεξεργασία τους. Στην εμφάνιση θα παρουσιάζεται κάθε φορά ένας πίνακας με όλα τα περιεχόμενα του πίνακα που επιλέχθηκε, ενώ στο log file ένας πίνακας με τις ενέργειες εισαγωγής, διαγραφής και επεξεργασίας που πραγματοποιήθηκαν για κάθε πίνακα από τους πέντε.

Log file Μέλος										
OPERATI...	STAMP	USERID	ID	Όνομα	Επώνυμο	Email	Τηλέφωνο	Διεύθυνση	Ημερομην...	Ημερομην...
D	2022-01-...	IT185192	2	μελος2	2μελος	μειλ2	54657674	διευθυνση2	1997-10-...	2018-05-...
D	2021-12-...	IT185192	2	aa	ss	ww	1234567...	g	1960-07-...	2011-09-...
D	2021-12-...	IT185192	1	οπο	ερο	ema	123	kap	1969-03-...	2013-06-...
I	2021-12-...	IT185192	3	μελος3	3μελος	μειλ3	4362547	διευθυνση3	2002-07-...	2017-12-...
I	2021-12-...	IT185192	2	Νικολέττα	Φωτοπού...	ni@gmail...	69123	abc	2000-10-...	2020-12-...
I	2021-12-...	IT185192	4	μελος4	4μελος	μειλ4	594326576	διευθυνση4	2001-08-...	2021-06-...
I	2021-12-...	IT185192	1	Γιώργος	Καραχρή...	gi@gmail...	69876	bca	2009-05-...	2018-03-...
D	2021-12-...	IT185192	2	Νικολέττα	Φωτοπού...	ni@gmail...	69123	abc	2000-10-...	2020-12-...
D	2021-12-...	IT185192	1	Γιώργος	Καραχρή...	gi@gmail...	69876	bca	2009-05-...	2018-03-...
I	2021-12-...	IT185192	1	μελος1	1μελος	μειλ1	341325143	διευθυνση1	1995-06-...	2020-04-...
I	2021-12-...	IT185192	2	μελος2	2μελος	μειλ2	54657674	διευθυνση2	1997-10-...	2018-05-...
I	2021-12-...	IT185192	5	μελος5	5μελος	μειλ5	93426892	διευθυνση5	1997-06-...	2016-11-...

ΕΜΦΑΝΙΣΗ ΜΕΛΟΣ							
ID	Όνομα	Επώνυμο	Email	Τηλέφωνο	Διεύθυνση	Ημερομηνία γέν...	Ημερομηνία εγ...
3	μελος3	3μελος	μειλ3	4362547	διευθυνση3	2002-07-26	2017-12-21
4	μελος4	4μελος	μειλ4	594326576	διευθυνση4	2001-08-27	2021-06-23
1	μελος1	1μελος	μειλ1	341325143	διευθυνση1	1995-06-05	2020-04-04
5	μελος5	5μελος	μειλ5	93426892	διευθυνση5	1997-06-28	2016-11-20

Πατώντας τα ερωτήματα θα εμφανιστεί ένα παράθυρο με δέκα κουμπιά, το καθένα από τα οποία υλοποιεί ένα ερώτημα σε ένα καινούριο παράθυρο. Το ερώτημα είναι δυναμικό και ανάλογα με την επιλογή ενός στοιχείου από το combo box θα αλλάξουν τα δεδομένα που παρουσιάζονται στον πίνακα.

Αναζήτηση βιβλίων με βάση τις εκδόσεις

Βιβλία από τις εκδόσεις

ISBN

ΟΞΥ

Μίνωας

Κλειδάριθμος

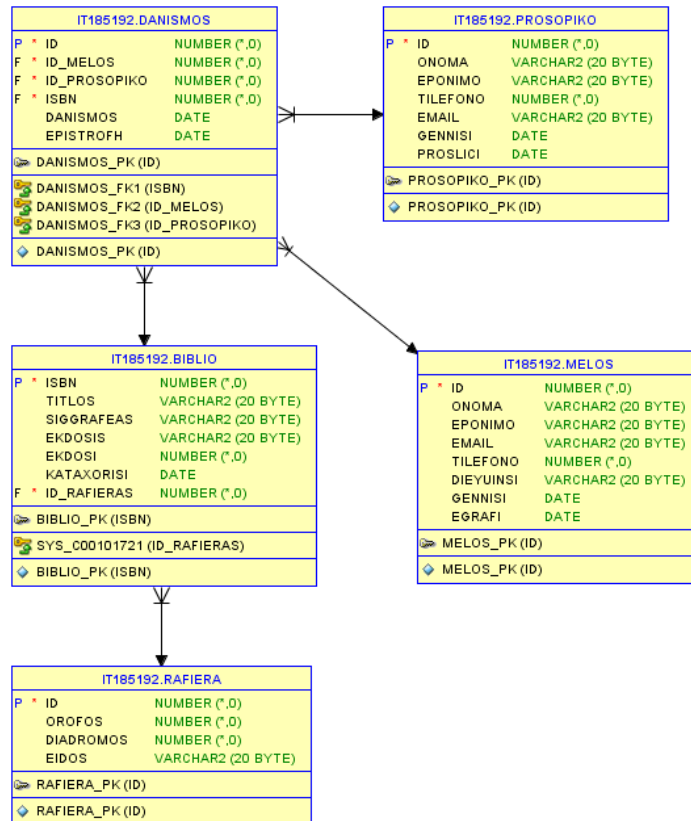
Πατάκη

Ψυχογιός

Πίσω

Βάση Δεδομένων

Είναι σημαντικό να παρουσιαστεί η δομή της βάσης δεδομένων και οι σχέσεις μεταξύ των πινάκων. Παρουσιάζεται το διάγραμμα ER που δημιουργήθηκε μέσα από το SQL Developer της Oracle.



Όπως βλέπουμε και στο διάγραμμα ER, η εφαρμογή κάνει χρήση αυτών των πέντε πινάκων, ο καθένας από τους οποίους έχει το δικό του μοναδικό κύριο κλειδί. Όμως για την σωστή λειτουργία της είναι απαραίτητη η μεταξύ τους σύνδεση μέσω των ξένων κλειδιών.

Το βιβλίο έχει ως κύριο κλειδί το ISBN, που είναι η μοναδική ταυτότητα του κάθε βιβλίου. Πέρα από τα απαραίτητα στοιχεία που έχει ως πεδία, όπως τον συγγραφέα, τον τίτλο κλπ. υπάρχει και ένα ξένο κλειδί ID_RAFIERAS που το συνδέει με μία μοναδική ραφίερα στον χώρο της βιβλιοθήκης.

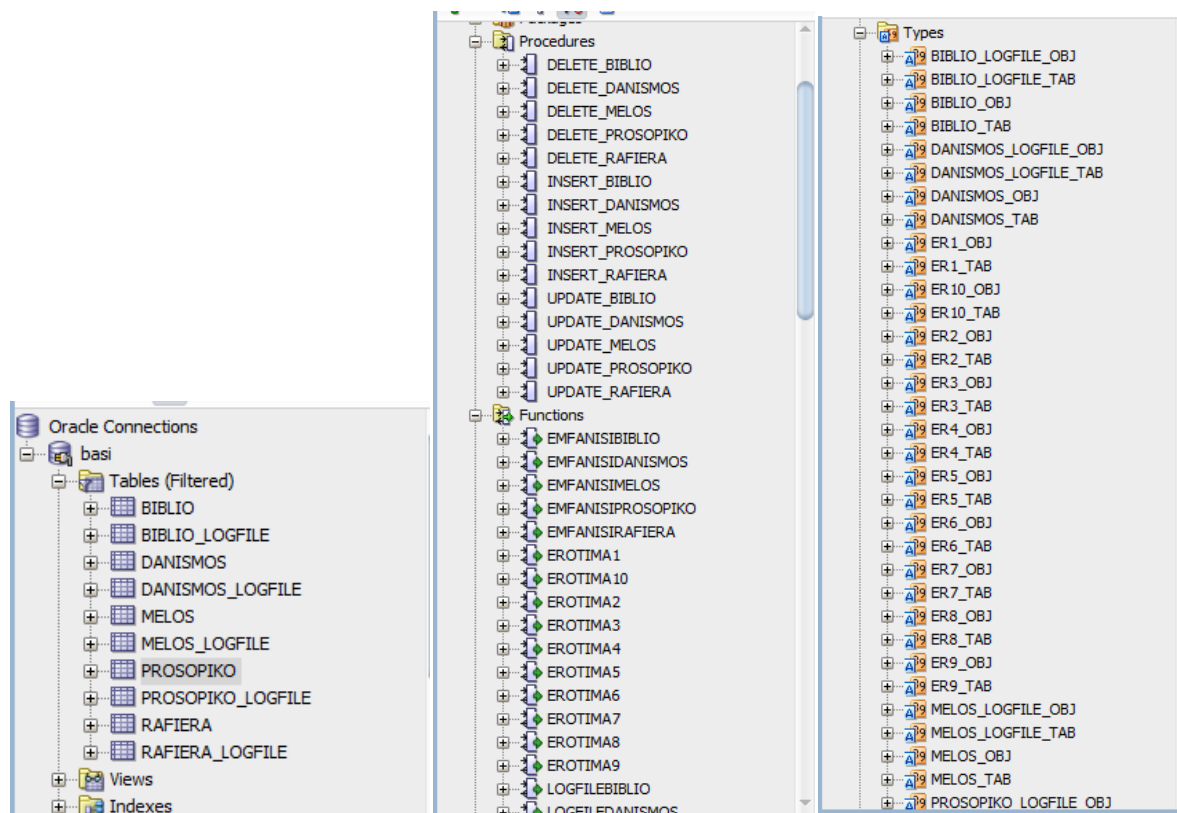
Η ραφίερα με την σειρά της έχει ως κύριο κλειδί ένα μοναδικό ID αλλά και άλλα στοιχεία όπως τον όροφο, τον διάδρομο αλλά κυρίως το είδος των βιβλίων που δέχεται να τοποθετηθούν σε αυτή.

Ο πίνακας μέλος με την σειρά του αποθηκεύει όλα τα δεδομένα από τα μέλη που εγγράφονται στην βιβλιοθήκη για να δανείζονται βιβλία, όπως τηλέφωνο, διεύθυνση κλπ. Έχει ως κύριο κλειδί ένα μοναδικό ID.

Το προσωπικό έχει τα στοιχεία των υπαλλήλων όπως τηλέφωνο, διεύθυνση κλπ. Κύριο κλειδί είναι επίσης ένα μοναδικό ID.

Τελευταίος πίνακας είναι αυτός των δανεισμών. Ο κάθε δανεισμός έχει ως κύριο κλειδί ένα μοναδικό ID, όμως για να πραγματοποιηθεί ο δανεισμός χρειάζεται στοιχεία από όλους τους παραπάνω πίνακες. Για αυτό περιέχει το ISBN που είναι ξένο κλειδί για να ταυτοποιηθεί το βιβλίο για το οποίο γίνεται ο

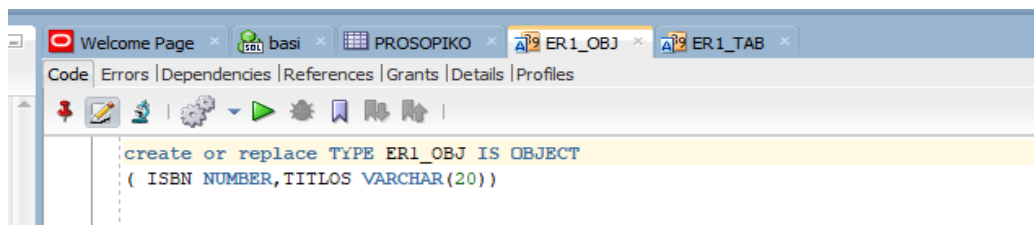
δανεισμός, το ID του μέλους το οποίο δανείζεται το βιβλίο, το ID του προσωπικού που εξυπηρετεί, καθώς επίσης και τις ημερομηνίες που ξεκίνησε ο δανεισμός και πότε τελειώνει η προθεσμία του.

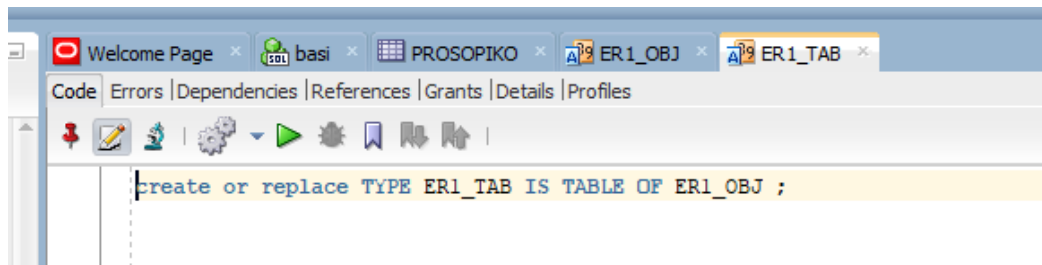


Για την υλοποίηση των ενεργειών έπρεπε να χρησιμοποιήσουμε αποθηκευμένες διαδικασίες. Για την εισαγωγή, την διαγραφή και την επεξεργασία των δεδομένων έγινε χρήση procedure, ενώ για την εμφάνιση, το log file και τα δέκα ερωτήματα έγινε χρήση function.

Για όσα από τα παραπάνω επέστρεφαν κάποιο αποτέλεσμα, χρειάστηκαν να υλοποιηθούν από δύο types, το πρώτο από τα οποία θα ήταν ένα object που θα είχε ως πεδία όλα αυτά που θέλουμε να διαχειριστούμε, και το δεύτερο ένας πίνακας με τύπο το ανάλογο Object που δημιουργήθηκε.

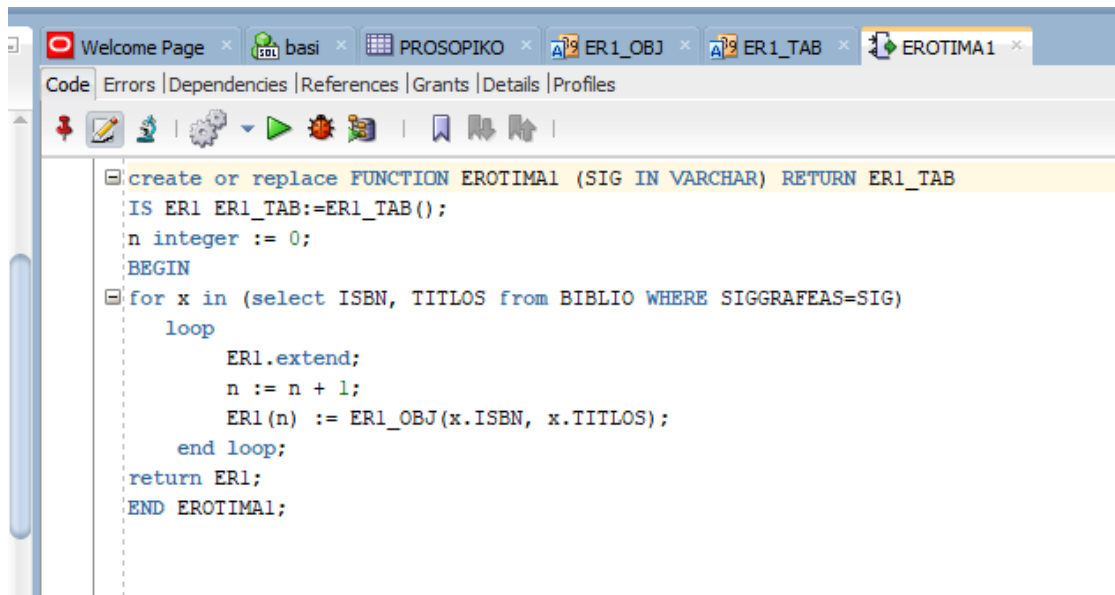
Για παράδειγμα, για το ερώτημα 1 θέλαμε να εμφανίζουμε βιβλία ανάλογα με τον συγγραφέα που επιλέγεται. Επιλέξαμε να εμφανίζουμε μόνο το ISBN και τον τίτλο του κάθε βιβλίου, για αυτό και στο object έχουμε μόνο αυτά τα δύο ως πεδία. Έπειτα δημιουργούμε ένα table από το προηγούμενο object. Και τα δύο θα χρησιμοποιηθούν από την function που διαχειρίζεται το ερώτημα 1.





SQL Developer interface showing the 'Code' tab for the 'ER1_TAB' object. The code editor contains the following SQL statement:

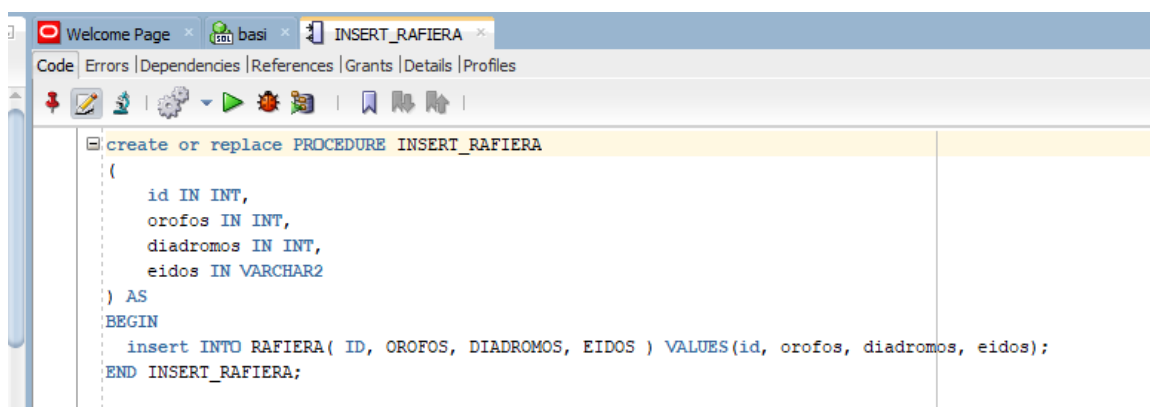
```
create or replace TYPE ER1_TAB IS TABLE OF ER1_OBJ ;
```



SQL Developer interface showing the 'Code' tab for the 'EROTIMA1' object. The code editor contains the following PL/SQL function:

```
create or replace FUNCTION EROTIMA1 (SIG IN VARCHAR) RETURN ER1_TAB
IS ER1 ER1_TAB:=ER1_TAB();
n integer := 0;
BEGIN
for x in (select ISBN, TITLOS from BIBLIO WHERE SIGGRAFEAS=SIG)
loop
    ER1.extend;
    n := n + 1;
    ER1(n) := ER1_OBJ(x.ISBN, x.TITLOS);
end loop;
return ER1;
END EROTIMA1;
```

Η συνάρτηση δέχεται ως παράμετρο εισόδου τον συγγραφέα και επιστρέφει έναν πίνακα, τον οποίο δημιουργήσαμε πριν στα types. Δηλώνουμε μέσα στην συνάρτηση μία μεταβλητή τύπου πίνακα και έναν ακέραιο μετρητή n. Δίνουμε τις εντολές για κάθε στοιχείο του πίνακα των βιβλίων να παίρνει όλα τα στοιχεία που έχουν τον ίδιο συγγραφέα με αυτόν της μεταβλητής εισόδου και να επεκτείνει τον πίνακα που μόλις δημιουργήσαμε με τις ανάλογες γραμμές, για τις στήλες ISBN και TITLOS που δηλώσαμε στο object προηγουμένως. Τέλος, επιστρέφεται ο πίνακας που πλέον περιέχει όλα τα στοιχεία που χρειαζόμαστε για να απαντήσουμε στο ερώτημα. Αυτά θα παρουσιαστούν στον πίνακα της Java Swing.



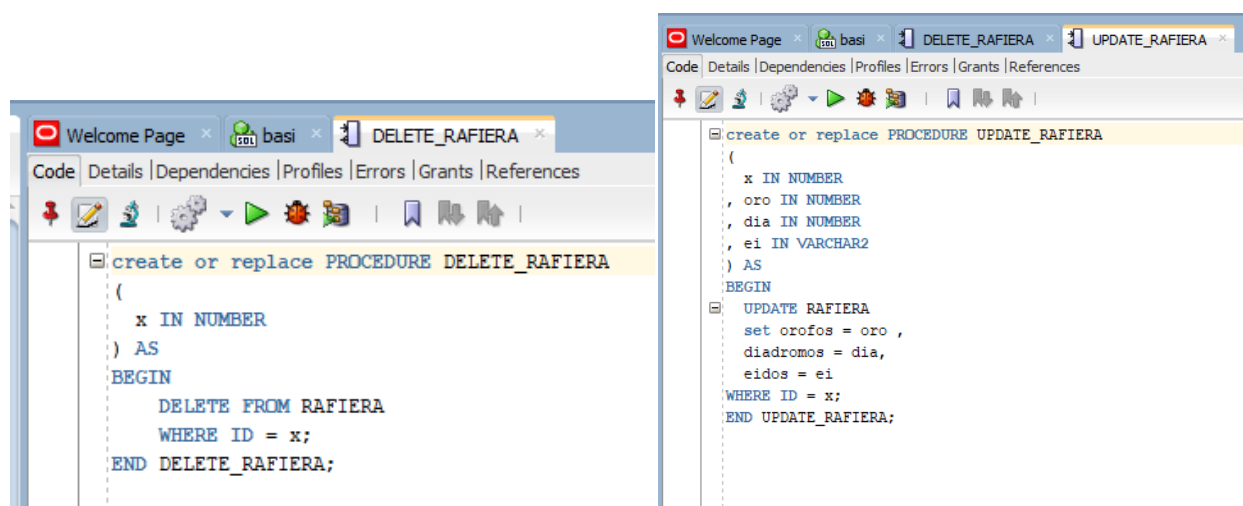
SQL Developer interface showing the 'Code' tab for the 'INSERT_RAFIERA' object. The code editor contains the following PL/SQL procedure:

```
create or replace PROCEDURE INSERT_RAFIERA
(
    id IN INT,
    orofos IN INT,
    diadromos IN INT,
    eidos IN VARCHAR2
) AS
BEGIN
insert INTO RAFIERA( ID, OROFOS, DIADROMOS, EIDOS ) VALUES(id, orofos, diadromos, eidos);
END INSERT_RAFIERA;
```


Μία από τις διαδικασίες που χρησιμοποιήθηκαν είναι για την εισαγωγή της ραφιέρας. Η διαδικασία παίρνει παραμετρικά ως μεταβλητές εισόδου όλα τα δεδομένα που εισήγαγε ο χρήστης στα πεδία, το ID, τον όροφο, τον διάδρομο και το είδος. Μέσω της insert αυτά εισάγονται στον πίνακα της ραφιέρας, δίνοντας σε καθένα από τα πεδία του πίνακα τιμή αντίστοιχη με αυτές που περάστηκαν παραμετρικά.

Με παρόμοιο τρόπο, για την διαγραφή περνάμε παραμετρικά το ID που επιλέγουμε από ένα combo box. Η διαδικασία με την χρήση της Delete διαγράφει την εγγραφή η οποία έχει ίδιο ID με αυτό που εισαγάγαμε.

Το update λειτουργεί με τον ίδιο τρόπο, απλά γίνεται η χρήση στις εντολής Update.



Κώδικας Java

Αρχικά, σε κάθε JFrame που απαιτεί σύνδεση με την βάση δεδομένων προσθέτουμε τον απαραίτητο κώδικα με τα στοιχεία που απαιτούνται για να επιτευχθεί.

```
28
29     static String driverClassName = "oracle.jdbc.OracleDriver";
30     static String url = "jdbc:oracle:thin:@192.168.6.21:1521:dblabs";
31     static Connection dbConnection = null;
32     static String username = "it185192";
33     static String passwd = "giorgosgtav";
34     static PreparedStatement pstatement = null;
35     static Statement statement = null;
36     static ResultSet rs = null;
37
```

Για να δούμε πώς διαχειριζόμαστε την εισαγωγή στοιχείων στην βάση από την Java, θα δούμε την κλάση για την εισαγωγή βιβλίου. Παρουσιάστηκε πιο πάνω η φόρμα συμπλήρωσης των απαραίτητων στοιχείων, όπως το ISBN, ο τίτλος, ο συγγραφέας κλπ. Πατώντας το κουμπί υποβολής, ενεργοποιείται η μέθοδος Action Performed, κατά την οποία ελέγχεται εάν το κύριο και το ξένο κλειδί έχουν συμπληρωθεί. Αν όχι θα εκτελεστεί το exception στο catch. Αν είναι συμπληρωμένα, μπαίνει μέσα στο

if και μετατρέπει τις τιμές των combo box σε μία date μεταβλητή. Έπειτα καλεί την μέθοδο insertBiblio όπου περνάει παραμετρικά όλες τις τιμές των Text Fields και του combo box για την ραφίερα.

Αφού γίνει αυτή η διαδικασία εμφανίζεται ένα μήνυμα επιβεβαίωσης, τα πεδία αδειάζουν και τα combo box επιστρέφουν στην αρχική, κενή τους μορφή.

```
private void jButtonIpboliActionPerformed(java.awt.event.ActionEvent evt) {  
    // TODO add your handling code here:  
    try {  
        if (!jTextFieldISBN.getText().isBlank() && jComboBoxRafiera.getSelectedIndex() != -1) {  
            LocalDate kataxoris = LocalDate.of(jComboBoxXroniaKataxoris.getSelectedIndex() + 1900,  
                Month.of(jComboBoxMinesKataxoris.getSelectedIndex() + 1),  
                jComboBoxHmeresKataxoris.getSelectedIndex() + 1);  
            Date datekataxoris = Date.valueOf(kataxoris);  
            insertBiblio(Integer.parseInt(jTextFieldISBN.getText()),  
                jTextFieldTitlos.getText(),  
                jTextFieldSiggrafeas.getText(),  
                jTextFieldEkdosis.getText(),  
                Integer.parseInt(jTextFieldEkdosi.getText()),  
                datekataxoris,  
                Integer.parseInt((String) jComboBoxRafiera.getSelectedItem()));  
            JOptionPane.showMessageDialog(null, "Επιτυχής εισαγωγή");  
            jTextFieldISBN.setText("");  
            jTextFieldTitlos.setText("");  
            jTextFieldSiggrafeas.setText("");  
            jTextFieldEkdosis.setText("");  
            jTextFieldEkdosi.setText("");  
            jComboBoxHmeresKataxoris.setSelectedIndex(-1);  
            jComboBoxMinesKataxoris.setSelectedIndex(-1);  
            jComboBoxRafiera.setSelectedIndex(-1);  
            jComboBoxXroniaKataxoris.setSelectedIndex(-1);  
        } else {  
            JOptionPane.showMessageDialog(null, "Τα πεδία ISBN και Ραφίερα είναι υποχρεωτικά");  
        }  
    }  
}
```

Η μέθοδος insertBiblio πραγματοποιεί την σύνδεση με την βάση με την χρήση του url, του username και του κωδικού. Έπειτα καλεί με την preparedStatement την αποθηκευμένη διαδικασία INSERT_BIBLIO περνώντας παραμετρικά όλες τις τιμές που πέρασαν μέσα στα πεδία και στάλθηκαν στην μέθοδο όταν κλήθηκε από την action performed. Χωρίς να δώσουμε τα ονόματα των στηλών και των πινάκων της βάσης στην Oracle, κάνουμε αναφορά στις στήλες της με αριθμούς και για κάθε ερωτηματικό της διαδικασίας, δίνουμε και μια από τις τιμές. Η διαδικασία είναι που θα τις αποθηκεύσει στον πίνακα των βιβλίων στην βάση. Αφού τελειώσει η διαδικασία, κλείνει η σύνδεση με την Oracle.

```

238
239 public void insertBiblio(int isbn, String titlos, String siggrafeas, String ekdosis, int ekdosi, Date kataxoris, int rafiera) throws Exce
240     Class.forName(driverClassName);
241     dbConnection = DriverManager.getConnection(url, username, passwd);
242
243     pstatement = dbConnection.prepareStatement("BEGIN INSERT_BIBLIO(?,?,?,?,?,?): END;");
244     pstatement.setInt(1, isbn);
245     pstatement.setString(2, titlos);
246     pstatement.setString(3, siggrafeas);
247     pstatement.setString(4, ekdosis);
248     pstatement.setInt(5, ekdosi);
249     pstatement.setDate(6, kataxoris);
250     pstatement.setInt(7, rafiera);
251
252     pstatement.executeUpdate();
253
254     pstatement.close();
255     dbConnection.close();
256
257 }

```

Τον παρακάτω τρόπο χρησιμοποιούμε κάθε φορά που θέλουμε να γεμίσουμε με στοιχεία ένα combo box. Στην εισαγωγή του βιβλίου έχουμε το combo box που περιέχει τα ID από τις ήδη υπάρχουσες ραφιέρες, για να αντιστοιχιστεί σε μία από αυτές. Στην μέθοδο για να προσθέτει στοιχεία σε αυτό, αρχικά ορίζουμε το combo box με το όνομα της μεταβλητής του από το JFrame, και ορίζουμε επίσης ένα προσωρινό αντικείμενο rowData που μπορεί να παίρνει μέχρι 50 στοιχεία σαν όριο. Αρχικοποιούμε έναν μετρητή i=0, και σε try εκτελούμε ξανά την σύνδεση με την βάση καθώς καλούμε την συνάρτηση από την Oracle η οποία επιστρέφει τον πίνακα της ραφιέρας. Όμως, μέσα στο while δίνουμε τις εντολές να πάρει το rowData μόνο τις τιμές από την πρώτη στήλη, που είναι τα ID από τις ραφιέρες, το πρωτεύον κλειδί τους. Στο τέλος το combo box παίρνει όλα τα στοιχεία του rowData.

```

256
257
258 void addDataTojComboBoxRafiera() {
259     DefaultComboBoxModel ComboBox = (DefaultComboBoxModel) jComboBoxRafiera.getModel();
260
261     Object rowData[] = new Object[50];
262
263     int i = 0;
264     try {
265         dbConnection = DriverManager.getConnection(url, username, passwd);
266         statement = dbConnection.createStatement();
267         rs = statement.executeQuery("SELECT * FROM TABLE (EMFANISIRAFIERA)");
268         while (rs.next()) {
269             rowData[i] = rs.getString(1);
270             i++;
271         }
272     } catch (SQLException ex) {
273         System.out.println("\n -- SQL Exception --- \n");
274         while (ex != null) {
275             System.out.println("Message: " + ex.getMessage());
276             ex = ex.getNextException();
277         }
278     }
279     jComboBoxRafiera.addItem("");
280     int j;
281     for (j = 0; j < i; j++) {
282         ComboBox.addElement(rowData[j]);
283     }
284 }

```

Για τα δυναμικά ερωτήματα χρησιμοποιήσαμε σε όλα τον παραπάνω τρόπο για να δίνουμε την επιλογή στον χρήστη να διαλέξει αυτό που θέλει. Όταν επιλέγεται κάποια τιμή από το combo box (στο συγκεκριμένο ερώτημα έχει επιλεγεί κάποιος συγγραφέας) καλούμε την ανάλογη συνάρτηση με το απαραίτητο ερωτηματικό για να περάσουμε παραμετρικά την τιμή του και να γίνει η σύγκριση μέσα στην συνάρτηση για να εντοπιστούν με SQL τα βιβλία που έχουν για συγγραφέα το όνομα που

επιλέχθηκε. Έπειτα τις δύο στήλες του πίνακα που έχουμε βάλει τις γεμίζουμε με τις δύο επιστρεφόμενες τιμές ISBN και τίτλο που έχουμε ορίσει.

```
43  L      */
44      @SuppressWarnings("unchecked")
45      Generated Code
121
122  private void jComboBoxSiggrafeasActionPerformed(java.awt.event.ActionEvent evt) {
123      // TODO add your handling code here:
124      DefaultTableModel table = (DefaultTableModel) jTable1.getModel();
125      Object row[] = new Object[2];
126      try {
127          dbConnection = DriverManager.getConnection(url, username, passwd);
128          pstatement = dbConnection.prepareStatement("SELECT * FROM table (EROTIMAL(?))");
129          pstatement.setString(1, (String) jComboBoxSiggrafeas.getSelectedItem());
130          rs = pstatement.executeQuery();
131          while (rs.next()) {
132              row[0] = rs.getString(1);
133              row[1] = rs.getString(2);
134              table.addRow(row);
135          }
136          pstatement.close();
137
138      } catch (SQLException ex) {
139          System.out.println("\n -- SQL Exception --- \n");
140          while (ex != null) {
141              System.out.println("Message: " + ex.getMessage());
142              ex = ex.getNextException();
143          }
144      }
145  }
```

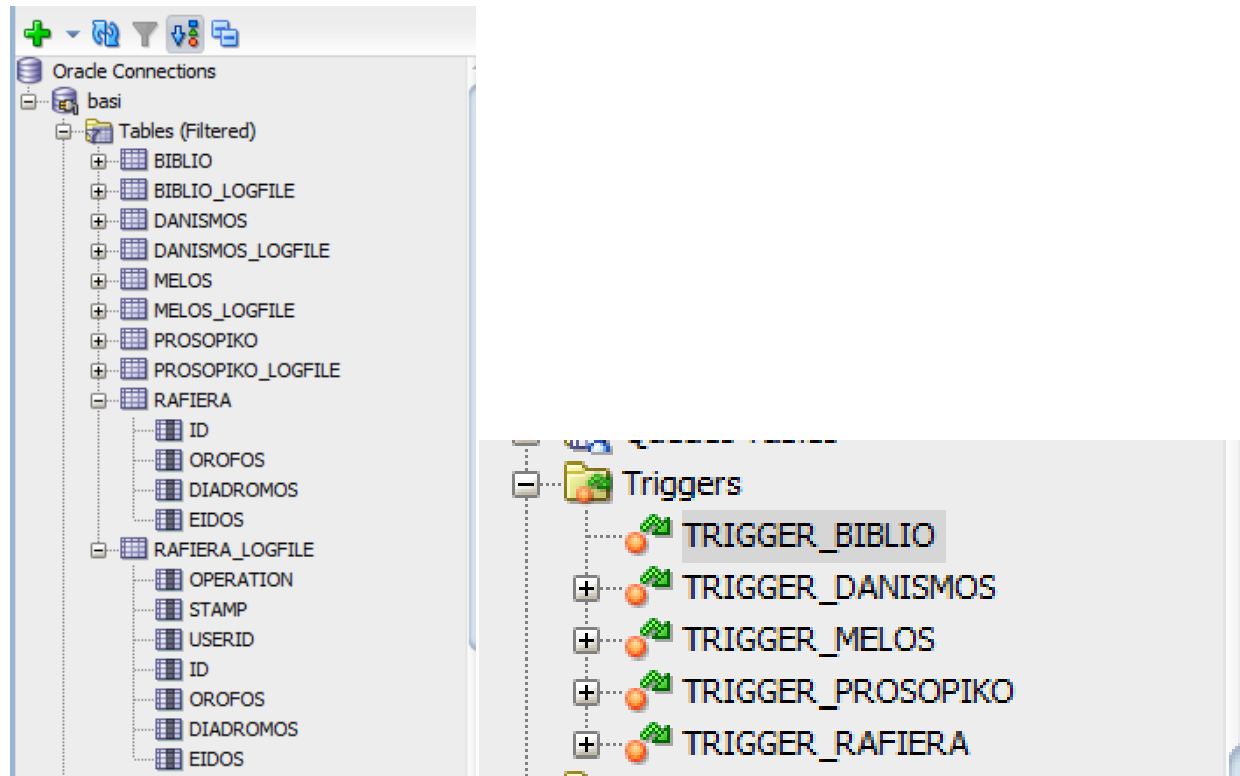
Χρειάστηκε να χρησιμοποιήσουμε μία ακόμα μέθοδο για όταν αλλάζει η επιλογή του combo box να καθαρίζει ο πίνακας πριν μπουν τα καινούρια στοιχεία, μηδενίζοντας τις σειρές του. Επιπλέον, σε περίπτωση που πατήσουμε το κουμπί πίσω, κλείνει το παράθυρο.

```
147  private void jComboBoxSiggrafeasItemStateChanged(java.awt.event.ItemEvent evt) {
148      // TODO add your handling code here:
149      DefaultTableModel table = (DefaultTableModel) jTable1.getModel();
150      if (evt.getStateChange() == ItemEvent.SELECTED)
151          table.setRowCount(0);
152  }
153
154  private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {
155      // TODO add your handling code here:
156      close();
157  }
158
159  public void close() {
160      WindowEvent closeWindow = new WindowEvent(this, WindowEvent.WINDOW_CLOSING);
161      Toolkit.getDefaultToolkit().getSystemEventQueue().postEvent(closeWindow);
162  }
```

Log file

Το log file εμφανίζει όλες τις ενέργειες που έγιναν για κάποιον πίνακα, καταχωρώντας τα ως εισαγωγή, διαγραφή ή επεξεργασία. Για να γίνει αυτό αρχικά δημιουργήθηκαν για καθένα από τους πίνακες (τα βιβλία, τα μέλη, το προσωπικό, τις ραφιέρους και τους δανεισμούς) από ένας δεύτερος πίνακας που

συνοδεύεται από το όνομα log file. Έχουν τα ίδια πεδία με τους αρχικούς πίνακες, και επιπλέον προστίθενται τρία ακόμα: το operation που δηλώνει τι τύπου από τις τρεις ήταν η ενέργεια που εκτελέστηκε, το stamp που είναι η ημερομηνία και ώρα που έγινε η ενέργεια και το userid που είναι το όνομα του χρήστη που κατέχει την βάση στην oracle.



Για να υλοποιηθούν τα log files χρειάστηκε να γίνει η χρήση trigger για την ανίχνευση των αλλαγών στους πίνακες, ένα για τον καθένα.

Για παράδειγμα, για το log file του βιβλίου δηλώνουμε να ενεργοποιείται σε κάθε delete, update, insert. Σε περίπτωση διαγραφής κάνει εισαγωγή στον πίνακα log file του βιβλίου αυτές τις τιμές, μαζί με το D στο operation, την ημερομηνία και ώρα του συστήματος και το όνομα του χρήστη. Σε περίπτωση εισαγωγής ή επεξεργασίας των στοιχείων γίνεται το ίδιο πράγμα, εισάγοντας στον πίνακα τις νέες πλέον τιμές.

```
create or replace TRIGGER TRIGGER_BIBLIO
BEFORE DELETE OR INSERT OR UPDATE ON BIBLIO
FOR EACH ROW
BEGIN
    IF DELETING THEN
        insert INTO BIBLIO_LOGFILE
        (OPERATION, STAMP, USERID, ISBN, TITLOS, SIGGRAFEAS, EKDOSIS, EKDOSI, KATAXORISI, ID_RAFIERAS)
        VALUES
        ('D', SYSDATE, user, :old.ISBN, :old.TITLOS, :old.SIGGRAFEAS, :old.EKDOSIS, :old.EKDOSI, :old.KATAXORISI, :old.ID_RAFIERAS);
    ELSIF INSERTING THEN
        insert INTO BIBLIO_LOGFILE
        (OPERATION, STAMP, USERID, ISBN, TITLOS, SIGGRAFEAS, EKDOSIS, EKDOSI, KATAXORISI, ID_RAFIERAS)
        VALUES
        ('I', SYSDATE, user, :new.ISBN, :new.TITLOS, :new.SIGGRAFEAS, :new.EKDOSIS, :new.EKDOSI, :new.KATAXORISI, :new.ID_RAFIERAS);
    ELSIF UPDATING THEN
        insert INTO BIBLIO_LOGFILE
        (OPERATION, STAMP, USERID, ISBN, TITLOS, SIGGRAFEAS, EKDOSIS, EKDOSI, KATAXORISI, ID_RAFIERAS)
        VALUES
        ('U', SYSDATE, user, :new.ISBN, :new.TITLOS, :new.SIGGRAFEAS, :new.EKDOSIS, :new.EKDOSI, :new.KATAXORISI, :new.ID_RAFIERAS);
    END IF;
END;
```

Για τα log files χρειάστηκε επίσης να δημιουργήσουμε τα δύο types που προαναφέρθηκαν, καθώς επίσης και από μία συνάρτηση για το καθένα για να μπορούμε αργότερα να επιστρέψουμε τα περιεχόμενα των πινάκων στην διεπαφή της Java Swing.

```
create or replace FUNCTION LOGFILEBIBLIO RETURN BIBLIO_LOGFILE_TAB IS
allbiblio BIBLIO_LOGFILE_TAB := BIBLIO_LOGFILE_TAB();
n integer :=0;
BEGIN
    for r in (SELECT * FROM BIBLIO_LOGFILE)
    loop
        allbiblio.extend;
        n:=n+1;
        allbiblio(n) := BIBLIO_LOGFILE_OBJ(r.OPERATION,r.STAMP,r.USERID,r.ISBN,r.TITLOS,r.SIGGRAFEAS,r.EKDOSIS,r.EKDOSI,r.KATAXORISI,r.ID_RAFIERAS);
    end loop;
    RETURN allbiblio;
END LOGFILEBIBLIO;
```

Στον κώδικα της Java αφού βάλουμε στην αρχή τα στοιχεία για την σύνδεση της βάσης όπως και πριν, εκτελούμε την σύνδεση, καλούμε την συνάρτηση για το log file του βιβλίου και περνάμε στις 10 στήλες του πίνακα τα δεδομένα για όσες γραμμές υπάρχουν στον αντίστοιχο πίνακα στην oracle. Το αποτέλεσμα θα είναι όπως το θέλουμε, με όλες τις ενέργειες να παρουσιάζονται όταν επιλέξουμε να δούμε κάποιο από τα πέντε log files.

```

82
83     Object rowData[] = new Object[10];
84
85     void addDataToJTable1() {
86         DefaultTableModel table = (DefaultTableModel) jTable1.getModel();
87
88         try {
89             dbConnection = DriverManager.getConnection(url, username, passwd);
90             statement = dbConnection.createStatement();
91             rs = statement.executeQuery("SELECT * FROM table (LOGFILEBIBLIO)");
92             while (rs.next()) {
93                 rowData[0] = rs.getString(1);
94                 rowData[1] = rs.getString(2);
95                 rowData[2] = rs.getString(3);
96                 rowData[3] = rs.getString(4);
97                 rowData[4] = rs.getString(5);
98                 rowData[5] = rs.getString(6);
99                 rowData[6] = rs.getString(7);
100                rowData[7] = rs.getString(8);
101                rowData[8] = rs.getString(9).substring(0,10);
102                rowData[9] = rs.getString(10);
103                table.addRow(rowData);
104            }
105        } catch (SQLException ex) {
106            System.out.println("\n -- SQL Exception --- \n");
107            while (ex != null) {
108                System.out.println("Message: " + ex.getMessage());
109                ex = ex.getNextException();
110            }
111        }
112    }

```

Log file βιβλίο									
OPERATION	STAMP	USERID	ISBN	Τίτλος	Συγγραφέας	Εκδόσεις	Έκδοση	Καταχώριση	Ραφίδα
D	2021-12-2...	IT185192	123456	αα	αα	rr	123456	1970-08-20	3
D	2021-12-3...	IT185192	123	Δικτύωση ...	James F	α	456	2016-06-05	1
I	2021-12-3...	IT185192	44	Άρχοντας τ...	Graham M...	ΟΞΥ	2001	2008-03-24	2
I	2022-01-0...	IT185192	123456789	Το ημερολ...	Kinney Jeff	Ψυχογιός	2021	2021-01-02	2
D	2022-01-0...	IT185192	123456789	Το ημερολ...	Kinney Jeff	Ψυχογιός	2021	2021-01-02	2
U	2022-01-0...	IT185192	657	τιπλοσ	φηη	φγηφ	4757	1903-03-29	
I	2021-12-3...	IT185192	1	Σταχτοπού...	Αδελφοί Γκ...	Μίνωας	1950	2010-10-05	1
I	2021-12-3...	IT185192	6	Misery	Stephen Ki...	Κλειδάριθμ...	1987	2012-05-10	2
I	2021-12-2...	IT185192	1	Κώδικας D...	Dan Brown	διοπτρα	2000	2003-02-04	1
I	2021-12-3...	IT185192	657	3ηφ	φηη	φγηφ	4757	1903-03-29	7
U	2022-01-0...	IT185192	657	τιπλοσ	φηη	φγηφ	4757	1903-03-29	7
I	2021-12-2...	IT185192	123456	αα	αα	rr	123456	1970-08-20	3
U	2021-12-2...	IT185192	123	α	James F	α	456	2016-06-05	1
U	2021-12-2...	IT185192	123	Δικτύωση ...	James F	α	456	2016-06-05	1
D	2021-12-3...	IT185192	1	Κώδικας D...	Dan Brown	διοπτρα	2000	2003-02-04	1
D	2022-01-0...	IT185192	657	τιπλοσ	φηη	φγηφ	4757	1903-03-29	
I	2021-12-2...	IT185192	123	α	α	α	456	2016-06-05	1
I	2021-12-3...	IT185192	2	Χιονάτη	Αδελφοί Γκ...	Πατάκη	1938	2003-06-03	1
I	2021-12-3...	IT185192	3	Κώδικας D...	Dan Brown	Ψυχογιός	2003	2018-05-06	2
U	2022-01-0...	IT185192	6	Misery	Stephen Ki...	Κλειδάριθμ...	1987	2012-05-10	2
U	2022-01-0...	IT185192	6	Misery	Stephen Ki...	Κλειδάριθμ...	1987	2012-05-10	2

Συμπεράσματα

Η εργασία στο μάθημα της Τεχνολογίας Βάσεων Δεδομένων μας έδωσε την ευκαιρία να χρησιμοποιήσουμε με πρακτικό τρόπο την SQL και την Java Swing, δημιουργώντας ένα πρόγραμμα που θα μπορούσε να χρησιμοποιηθεί για να δώσει λύση σε προβλήματα στην καθημερινή ζωή. Χρησιμοποιήσαμε για πρώτη φορά την βάση της Oracle, τις αποθηκευμένες διαδικασίες και τα εναύσματα και μάθαμε πως είναι χρήσιμα εργαλεία για την δημιουργία ευέλικτων συστημάτων.