

Εργασία στο μάθημα «Συστήματα Διαχείρισης Βάσεων Δεδομένων»

Γιώργος Κωστάντος, Θοδωρής Γιαννόπουλος, Σοφία Γόμπου.

Γενικά

Και στα 3 parts της εργασίας ακολουθήσαμε μία κοινή στρατηγική:

1. Παίρνουμε το query από τον χρήστη.
2. Το στέλνουμε στον σέρβερ.
3. Το κάνουμε parse βρίσκοντας το table, τα columns που θέλει να επεξεργαστεί ο χρήστης και τη συνθήκη where(αν υπάρχει).
4. Εκτελούμε την αντίστοιχη εντολή χρησιμοποιώντας το αντικείμενο db που δημιουργούμε στην αρχή του κάθε server file.
5. Και τέλος, επιστρέφουμε το αποτέλεσμα, είτε αυτό είναι ο νέος πίνακας, μήνυμα επιτυχίας ή κάποιο σφάλμα που προέκυψε.

Το μόνο που διαφοροποιείται είναι ο τρόπος επιστροφής του αποτελέσματος και προβολής στον χρήστη.

SELECT

Ο χρήστης εισάγει το query: **"SELECT * FROM classroom WHERE capacity == 500"**. Αρχικά, δημιουργείτε η μεταβλητή ResultSet καλώντας τη συνάρτηση `Select_From_Table`. Στη `Select_From_Table(QUERY)`, χωρίζεται το query στα μέρη `table`, `table_columns` και `where_condition` καλώντας την αντίστοιχη συνάρτηση και τέλος γίνεται `return` το αποτέλεσμα της συνάρτησης `Execute_Select_Query`. Στη συνάρτηση `Execute_Select_Query` απαιτείται ο έλεγχος περιπτώσεων λόγω υλοποίησης της συνάρτησης `db.select()`. Για παράδειγμα αν ο χρήστης επιθυμεί να επιλέξει όλα τα columns ενός table, πρέπει να περαστεί ως όρισμα η μεταβλητή `"*"` και όχι η μεταβλητή `["*"]`. Σε κάθε περίπτωση το τελευταίο όρισμα είναι ίσο με `True` καθώς θέλουμε να πάρουμε τον πίνακα και να τον εμφανίσουμε στον χρήστη, και όχι στη μεριά του server.

Αφού εκτελεστεί η εντολή `db.select()`, η τιμή της μεταβλητής ResultSet τροποποιείται καλώντας τη συνάρτηση `Convert_ResultSetTable_toString`, όπου απλά μορφοποιείται με το `tabulate` το αποτέλεσμα της συνάρτησης `db.select()`. Τέλος, ο server στέλνει το αποτέλεσμα στον χρήστη.

UPDATE

Ο χρήστης εισάγει το query: **“UPDATE classroom SET building == Watson WHERE room_number == 514”**. Το πρόγραμμα δημιουργεί τη μεταβλητή table καλώντας τη συνάρτηση update_table στην οποία βρίσκει τις ζητούμενες μεταβλητές και εκτελεί την εντολή update στο αντικείμενο db. Το όνομα του table επιστρέφεται στη μεταβλητή table προκειμένου να δημιουργηθεί ένα μήνυμα επιτυχίας για τον χρήστη.

DELETE

Ο χρήστης εισάγει το query **“DELETE FROM classroom WHERE room_number == 101”**. Στη συνέχεια, δημιουργείται η μεταβλητή table καλώντας τη συνάρτηση delete_from_table. Αν η συνθήκη εκτελεστεί με επιτυχία, το όνομα του table θα εμφανιστεί στον χρήστη με ανάλογο μήνυμα επιτυχίας, αλλιώς θα εμφανιστεί στον χρήστη το σφάλμα το οποίο προέκυψε.

Παρατηρήσεις

Ένα θέμα που προέκυψε είναι ότι η βάση μετά από κάθε εκτέλεση query κλειδώνει και όλα τα επόμενα queries που εισάγει ο χρήστης αποτυγχάνουν. Για αυτόν τον λόγο, μετά από την εκτέλεση κάθε query, σε όλες τις περιπτώσεις, χρησιμοποιούμε τη συνάρτηση unlock_table.

Αρχεία

Select: Client_Side_Select.py, Server_Side_Select.py.

Update: ClientUpdate.py, ServerUpdate.py.

Delete: ClientDelete.py, ServerDelete.py.