

ΕΡΓΑΣΙΑ Σ.Δ.Β.Δ. 5^{ου} ΕΞΑΜΗΝΟΥ

Ονοματεπώνυμο/Α.Μ.:

Σταύρος Τσουκαλάς – Π18235

Πέτρος Στέργιος – Π18146

Βασίλης Βελλιανίτης – Π18018

Παρατήρηση: Τα αρχεία “database.py” και “vsmdb.py” είναι επεξεργασμένα, όχι τα πρωτότυπα.

ΠΕΡΙΓΡΑΦΗ ΛΕΙΤΟΥΡΓΙΚΟΤΗΤΑΣ ΤΟΥ ΚΩΔΙΚΑ:

Η κεντρική ιδέα για την υλοποίηση του persistent log για τις εντολές που καλεί για την βάση ή table, αλλά και για τα rollback που μπορεί να καλέσει ήταν η εξής:

- Για το **log**: προσθέσαμε στις διάφορες εντολές που μπορεί να καλέσει ο χρήστη στο αρχείο “database.py”, μία εντολή όπου καλεί την μέθοδο «`analyt_log(record)`» όπου προσθέτει στα αρχεία των log (`log_data.txt`, `user_log.txt`) τις εντολές ή το rollback που κάλεσε ο χρήστης. Πιο συγκεκριμένα, η μέθοδος αυτή καλείται

στις εξής μεθόδους του “database.py” για το “user_log.txt”:

- `__init__`
- `drop_db`
- `create_table`
- `drop_table`
- `table_from_csv`
- `table_to_csv`
- `table_from_object`
- `cast_column`
- `insert`
- `update`
- `delete`
- `select`
- `show_table`
- `sort`
- `inner_join`
- `create_index`
- `_has_index`

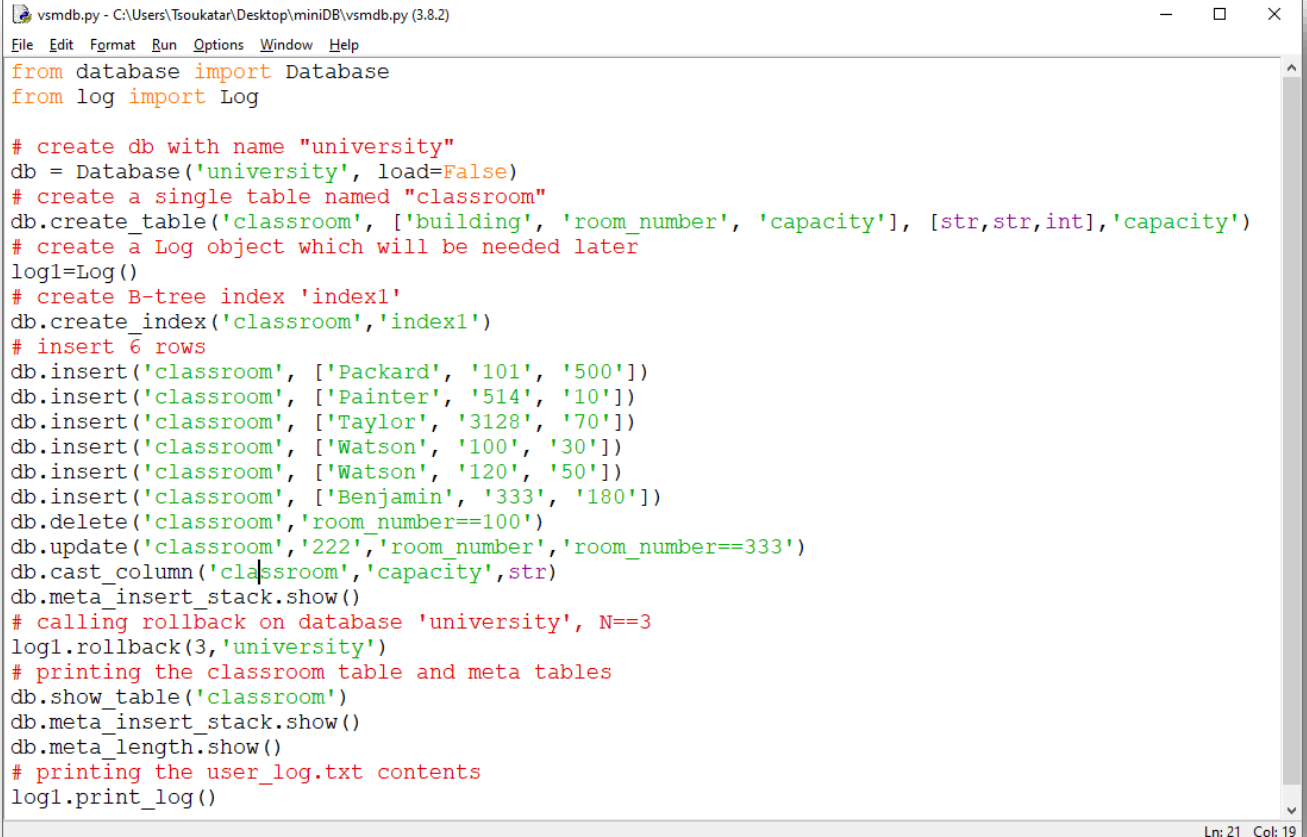
- Για το **rollback**: κρατάμε την κατάσταση της βάσης καθώς και τα `insert_stacks` στα αρχεία «old_tables.data» και «old_stack.data», αντίστοιχα. Όταν καλείται το **rollback** διαβάζουμε από αυτά τα δεδομένα και βρίσκουμε τις προηγούμενες καταστάσεις της βάσης/του πίνακα και τις αντικαθιστούμε στην υπάρχουσα βάση/πίνακα αντίστροφα (καθώς το `rollback` είναι της λογικής LIFO). Ο χρήστης μπορεί να καλέσει το `rollback` για όσες εντολές N επιθυμεί. Δεν

έχει υλοποιηθεί default τιμή N, άρα ο χρήστης δηλώνει στα ορίσματα του rollback και το N. Αν ο χρήστης δώσει N μεγαλύτερο του πλήθους των εντολών που έχουν εκτελεστεί, το rollback θα εκτελεστεί για N=πλήθους εντολών του χρήστη για τη βάση/τον πίνακα και θα διακοπεί. Υπάρχει λειτουργικότητα για rollback σε βάση και σε πίνακα βάσης.

Τέλος, έχουμε υλοποιήσει και μία μέθοδο (print_log()) όπου γίνεται print στην οθόνη το περιεχόμενο του "user_log.txt" αρχείου.

ΕΝΔΕΙΚΤΙΚΑ SCREENSHOTS ΜΕ ΠΑΡΑΔΕΙΓΜΑΤΑ

ΕΚΤΕΛΕΣΗΣ:

A screenshot of a Python script execution window titled 'vsmdb.py - C:\Users\Tsoukatar\Desktop\miniDB\vsmdb.py (3.8.2)'. The window has a menu bar with 'File', 'Edit', 'Format', 'Run', 'Options', 'Window', and 'Help'. The script content is as follows:

```
from database import Database
from log import Log

# create db with name "university"
db = Database('university', load=False)
# create a single table named "classroom"
db.create_table('classroom', ['building', 'room_number', 'capacity'], [str,str,int], 'capacity')
# create a Log object which will be needed later
log1=Log()
# create B-tree index 'index1'
db.create_index('classroom','index1')
# insert 6 rows
db.insert('classroom', ['Packard', '101', '500'])
db.insert('classroom', ['Painter', '514', '10'])
db.insert('classroom', ['Taylor', '3128', '70'])
db.insert('classroom', ['Watson', '100', '30'])
db.insert('classroom', ['Watson', '120', '50'])
db.insert('classroom', ['Benjamin', '333', '180'])
db.delete('classroom', 'room_number==100')
db.update('classroom', '222', 'room_number', 'room_number==333')
db.cast_column('classroom', 'capacity', str)
db.meta_insert_stack.show()
# calling rollback on database 'university', N==3
log1.rollback(3, 'university')
# printing the classroom table and meta tables
db.show_table('classroom')
db.meta_insert_stack.show()
db.meta_length.show()
# printing the user_log.txt contents
log1.print_log()
```

The status bar at the bottom right indicates 'Ln: 21 Col: 19'.

```

New table "meta_length"
New table "meta_locks"
New table "meta_insert_stack"
New table "meta_indexes"
New table "classroom"
Creating Btree index.
Deleted 1 rows

```

```

## meta_insert_stack ##
table_name (str)      indexes (list)
-----
classroom             [3]

```

```

Initiating rollback.
Loaded "university".
Deleted 1 rows
Deleted 1 rows
Deleted 1 rows
Loaded "university".
Deleted 1 rows
Deleted 1 rows
Deleted 1 rows
Loaded "university".
Deleted 1 rows
Deleted 1 rows
Deleted 1 rows
Terminating rollback.

```

```

## classroom ##
building (str)      room_number (str)      capacity (int) #PK#
-----
Packard              101              500
Painter              514              10
Taylor               3128              70
Watson               100              30
Watson               120              50
Benjamin             333             180

```

```

## meta_insert_stack ##
table_name (str)      indexes (list)
-----
classroom             []

```

```

## meta_length ##
table_name (str)      no_of_rows (int)
-----
classroom              6

```

```
TIMESTAMP: 2021-02-27 18:02:32.423116 FUNCTION: database_creation DATABASE: university
TIMESTAMP: 2021-02-27 18:02:32.458571 FUNCTION: table_creation DATABASE: university TABLE: classroom
TIMESTAMP: 2021-02-27 18:02:32.470509 FUNCTION: create_index DATABASE: university TABLE: classroom INDEX NAME: index1
TIMESTAMP: 2021-02-27 18:02:32.494474 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:32.480514 FUNCTION: insert DATABASE: university TABLE: classroom INSERT ROW: ['Packard', '101', 500]
TIMESTAMP: 2021-02-27 18:02:32.525505 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:32.509514 FUNCTION: insert DATABASE: university TABLE: classroom INSERT ROW: ['Painter', '514', 10]
TIMESTAMP: 2021-02-27 18:02:32.557417 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:32.541430 FUNCTION: insert DATABASE: university TABLE: classroom INSERT ROW: ['Taylor', '3128', 70]
TIMESTAMP: 2021-02-27 18:02:32.590106 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:32.573150 FUNCTION: insert DATABASE: university TABLE: classroom INSERT ROW: ['Watson', '100', 30]
TIMESTAMP: 2021-02-27 18:02:32.623867 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:32.604941 FUNCTION: insert DATABASE: university TABLE: classroom INSERT ROW: ['Watson', '120', 50]
TIMESTAMP: 2021-02-27 18:02:32.662791 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:32.642839 FUNCTION: insert DATABASE: university TABLE: classroom INSERT ROW: ['Benjamin', '333', 180]
TIMESTAMP: 2021-02-27 18:02:32.677717 FUNCTION: delete DATABASE: university TABLE: classroom
TIMESTAMP: 2021-02-27 18:02:32.698661 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:32.758535 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:32.739582 FUNCTION: update DATABASE: university TABLE: classroom SET COLUMN room_number
TIMESTAMP: 2021-02-27 18:02:32.793438 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:32.777480 FUNCTION: cast_column DATABASE: university TABLE: classroom COLUMN: capacity CAST TYPE: <class 'str'>
TIMESTAMP: 2021-02-27 18:02:32.856815 FUNCTION: ROLLBACK DATABASE: university
TIMESTAMP: 2021-02-27 18:02:32.887761 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:32.885735 FUNCTION: table_drop DATABASE: university TABLE: classroom
TIMESTAMP: 2021-02-27 18:02:32.931822 FUNCTION: table_creation DATABASE: university TABLE: classroom
TIMESTAMP: 2021-02-27 18:02:32.958181 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:32.957181 FUNCTION: table_drop DATABASE: university TABLE: classroom
TIMESTAMP: 2021-02-27 18:02:33.005627 FUNCTION: table_creation DATABASE: university TABLE: classroom
TIMESTAMP: 2021-02-27 18:02:33.030566 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:33.028537 FUNCTION: table_drop DATABASE: university TABLE: classroom
TIMESTAMP: 2021-02-27 18:02:33.074216 FUNCTION: table_creation DATABASE: university TABLE: classroom
TIMESTAMP: 2021-02-27 18:02:32.856815 FUNCTION: END OF ROLLBACK
TIMESTAMP: 2021-02-27 18:02:33.094134 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:33.101115 FUNCTION: has_index DATABASE: university TABLE: meta_locks
TIMESTAMP: 2021-02-27 18:02:33.093137 FUNCTION: show_table DATABASE: university TABLE: classroom
```