## ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



# Εργασία Μαθήματος «συστηματα διαχείρισης βασέων δεδομένων»

Όνομα φοιτητή – Αριθμός Μητρώου	Πετρίδης Αχιλλέας Π18211
	Φύτρος Ευάγγελος Π18220
Ημερομηνία παράδοσης	23/02/2021

### ΣΥΣΤΗΜΑΤΑ ΔΙΑΧΕΙΡΙΣΗΣ ΒΑΣΕΩΝ ΔΕΔΟΜΕΝΩΝ



## ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

Server	. 3
Client	4
	•
Execution example	4
	Client  Execution example



### 1 Server

```
from database import Database
import socket
import sys
from io import StringIO
def trexa(receiver):
    good = False
        if receiver:
            usr_input = conn.recv(1024)
            usr_input = usr_input.decode("utf-8")
            db = Database('vsmdb', load=True)
splitted = usr_input.split(" ", 3)
                 if splitted[0] == "SELECT":
                     if splitted[1] == "*":
                         good = True
                         table = splitted[3]
                         output = sys.stdout
                         dbout = StringIO()
                         sys.stdout = dbout
                         db.select(table,'*')
                         sys.stdout = output
                         dbout_string = dbout.getvalue()
                conn.sendall(str.encode("Lathos kati"))
       conn.sendall(str.encode("Lathos kati"))
        conn.close()
    if good:
        conn.sendall(str.encode(dbout string))
        conn.sendall(str.encode("Lathos kati"))
    print("The info you requested has been sent")
    conn.close()
    exit()
    with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as s:
       s.setsockopt(socket.SOL_SOCKET, socket.SO_REUSEADDR, 1)
        s.bind(('127.0.0.1',54321))
        s.listen()
        conn, addr = s.accept()
        with conn:
            trexa(True)
            s.close()
```

Στο κομμάτι του server δημιουργείται ένα socket το οποίου θέτουμε σε listen mode και περιμένει το client side.

Όταν συνδεθεί ο client τότε μπαίνει στην trexa παίρνει το Input που έχει σταλεί μέσω του client ,φορτώνει την βάση vsmdb και ελέγχει αν είναι σωστά δομημένο χωρίζοντας το σε μέλη ανά κενό. Έπειτα τρέχει την db.select με το table name που δόθηκε από το input και την στέλνει έτσι ώστε να μπορεί να παραληφθεί από τον client.



## 2 Client

```
import socket

input_query = input(
    "Insert the query you want to send to the server\n example: SELECT * FROM classroom\nInput: ")

with socket.socket(socket.AF_INET, socket.SOCK_STREAM) as sok:

while True:
    sok.connect(('127.0.0.1',54321))
    sok.sendall(input_query.encode())
    data = sok.recv(1000)
    print('Received response from server\n', data.decode('utf8', 'strict'))
    sok.close()
    print("Goodbye")

exit()
```

Στο κομμάτι του client ζητάμε από τον χρήστη το query που θέλει να στείλει στον server.

Μόλις δοθεί το input συνδεόμαστε στην ιδιά διεύθυνση με τον server και στέλνουμε το query και περιμένουμε απάντηση. Μόλις πάρουμε την απάντηση από τον server την εκτυπώνουμε.

## 3 Execution example

Ανοίγουμε πρώτα τον Server



#### Έπειτα ανοίγουμε το client

```
□ X

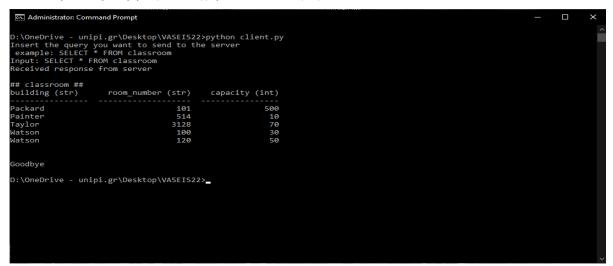
Administrator.Command Prompt - python client.py

□:\OneDrive - unipi.gr\Desktop\VASEIS22>python client.py

Insert the query you want to send to the server example: SELECT * FROM classroom

Input:
```

#### Στέλνουμε το query μας και δεχόμαστε απάντηση.



#### Οπού την ιδιά στιγμή στην μεριά του server έχουμε αυτό

```
Administrator: Command Prompt

D:\OneDrive - unipi.gr\Desktop\VASEIS22>python the_server.py
Loaded "vsmdb".

The info you requested has been sent

D:\OneDrive - unipi.gr\Desktop\VASEIS22>
```