

# Ανάπτυξη εφαρμογής εξυπηρετητή και πελάτη chat με χρήση της python.



## ΟΜΑΔΙΚΗ ΕΡΓΑΣΙΑ ΟΜΑΔΑΣ 19

### ΜΕΛΗ ΟΜΑΔΑΣ

ΓΙΑΝΝΑΚΑΚΗΣ ΒΑΣΙΛΕΙΟΣ	1046070 up1046070@upnet.gr
ΜΑΚΗΣ ΣΠΥΡΙΔΩΝ	1047000 up1047000@upnet.gr
ΜΙΛΤΙΑΔΟΥΣ ΜΑΡΙΟΣ	1021041 up1021041@upnet.gr
ΠΑΥΛΟΠΟΥΛΟΣ ΛΑΜΠΡΟΣ	1046960 up1046960@upnet.gr
ΣΚΑΛΙΩΝΤΑΣ ΑΝΔΡΕΑΣ	1050653 up1050653@upnet.gr
ΤΣΟΜΩΚΟΣ ΚΩΝΣΤΑΝΤΙΝΟΣ	1047010 up1047010@upnet.gr

Πέμπτη, 26 Μαΐου 2016

## Εισαγωγή

### Σκοπός εργασίας

Η εργασία μας έχει ως σκοπό την δημιουργία ενός chat room με την γλώσσα Python. Μέσα από αυτή την διαδικασία θα κατανοήσουμε ποια είναι η δομή ενός δικτύου υπολογιστών και πως λειτουργεί

### Computer network programming

Ο διαδικτυακός προγραμματισμός είναι άμεσα συνδεδεμένος με την καθημερινότητα πολλών ατόμων, το ηλεκτρονικό ταχυδρομείο οι ιστοσελίδες πληροφόρησης και τα chat room είναι μερικές από τις εφαρμογές του.

### Τι είναι όμως ο διαδικτυακός προγραμματισμός(network programming);

Διαδικτυακός προγραμματισμός είναι η διαδικασία συγγραφής προγραμμάτων που διευκολύνουν την επικοινωνία σε ένα δίκτυο υπολογιστών (σύμφωνα με το Wikipedia.org).

Όπως αναφέραμε ένα από τα πιο συχνά παραδείγματα χρήσης του διαδικτυακού προγραμματισμού είναι ένα chat room.

### Τι είναι η διαδικτυακή συνομιλία (chat);

Η διαδικτυακή συνομιλία (chat) είναι ένας τρόπος επικοινωνίας με αποστολή γραπτών μηνυμάτων σ' ανθρώπους στο ίδιο chatroom («εικονικό σημείο συζήτησης») την ίδια στιγμή (Πηγή: Wikipedia). Το πρώτο διαδικτυακό chat ονομαζόταν Talkomatic και κατασκευάστηκε από τους Doug Brown and David R. Woolle μη την χρήση του συστήματος PLATO στο πανεπιστήμιο του ilinois.

### Γενικά χαρακτηριστικά ενός chatroom

Τα κύρια χαρακτηριστικά ενός chatroom είναι η αμεσότητα και η ταχύτητα της επικοινωνίας. Οι χρήστες των chatroom συνδέονται σε αυτό της επιλογής τους χρησιμοποιώντας ένα όνομα της αρεσκείας τους. Μέσα σε αυτό μπορούν να δουν πια άτομα είναι συνδεδεμένα. Η πληκτρολόγηση των μηνυμάτων γίνεται σε ένα textbox και είναι προσβάσιμα για ανάγνωση από όλους τα άτομα που είναι συνδεδεμένα στο δωμάτιο ακόμα και αν δεν συμμετέχουν ενεργά στην συζήτηση. Επειδή η επικοινωνία είναι άμεση ως προς τον χρόνο πολλές φορές παρατηρείται αρνητική συμπεριφορά για αυτό το λόγο έχουν δημιουργηθεί έλεγχοι στον κώδικα του προγράμματος, οι οποίοι εντοπίζουν προσβλητικές λέξεις και αποτρέπουν την εμφάνισή τους. Τέλος μπορούν και οι ίδιοι οι χρήστες να εμποδίσουν την είσοδο στα άτομα με αρνητική συμπεριφορά.

### Τρόπος λειτουργίας του ως προς τον προγραμματιστή

Ένα δωμάτιο συνομιλίας δεν έχει τόσο απλή δομή όπως φαίνεται από τον χρήστη. Αυτό λειτουργεί με το μοντέλο server και client. Το μοντέλο αυτό λειτουργεί με τον εξής τρόπο. Αρχικά ο διακοσμητής(server)διαχειρίζεται κάποιο πόρο, και παρέχει κάποια εξυπηρέτηση των πελατών με το χειρισμό αυτού του πόρου. Όταν ένας πελάτης χρειάζεται να εκτελέσει μια διεργασία, δημιουργεί μια συναλλαγή με την αποστολή ενός αιτήματος στο διακομιστή.

Για παράδειγμα όταν ένας χρήστης σε ένα δωμάτιο άμεσων μηνυμάτων θέλει να επικοινωνήσει με τους υπόλοιπους χρήστες στέλνει το μήνυμα στον διακοσμητή του δωματίου. Αυτός αφού το επεξεργαστεί με την σειρά του το στέλνει στους υπόλοιπους χρήστες .

## Μεθοδολογία και μέσα που χρησιμοποιήθηκαν για την κατασκευή του προγράμματος.

Για την δημιουργία του κώδικα χρησιμοποιήθηκε η γλώσσα προγραμματισμού Python. Για να μπορέσουμε να εξελίξουμε τον κώδικα χρησιμοποιήθηκαν οι βιβλιοθήκες της python socket, threading και time.

### Socket

Η socket ήταν η βασική βιβλιοθήκη που χρησιμοποιήσαμε. Τι είναι όμως η socket;

Socket στα ελληνικά σημαίνει υποδοχή και είναι τα άκρα ενός αμφίδρομου καναλιού επικοινωνίας. Αυτές οι υποδοχές μπορούν να επικοινωνούν στο πλαίσιο μιας διαδικασίας, ή πολλών διεργασιών στον ίδιο υπολογιστή, ή σε διαφορετικούς υπολογιστές. Η socket μπορεί να εφαρμοστεί σε μια σειρά από διαφορετικούς τύπους καναλιών όπως Unix, TCP, UDP, και ούτω καθεξής. Η βιβλιοθήκη αυτή παρέχει συγκεκριμένες τάξεις για το χειρισμό των κοινών μεταφορών, καθώς και μια γενική διεπαφή για το χειρισμό των υπόλοιπων.

Η Socket στην γλώσσα python εισάγεται ως βιβλιοθήκη και έχει διάφορες μεθόδους. Κατά την δημιουργία του κώδικα με χρήση αυτής ο προγραμματιστής καθορίζει ποια οικογένεια πρωτοκόλλων θα εισάγει αλλά και των τύπο των πρωτοκόλλων με την δημιουργία του αντικειμένου `s = socket.socket(socket_family, socket_type)`. Για τους δύο τύπους πρωτοκόλλων UDP και TCP υπάρχουν διαφορετικοί μέθοδοι. Μερικές από αυτές τις μεθόδους φαίνονται στο παρακάτω πίνακα.

Μέθοδος	Περιγραφή
<code>s.connect()</code>	Ενεργοποιεί την σύνδεση σε TCP διακοσμητή
<code>s.recv()</code>	Δέχεται δεδομένα σε TCP πρωτόκολλο
<code>s.send()</code>	Μεταδίδει δεδομένα σε TCP πρωτόκολλο
<code>s.recvfrom()</code>	Δέχεται δεδομένα σε UDP πρωτόκολλο
<code>s.sendto()</code>	Μεταδίδει δεδομένα σε UDP πρωτόκολλο
<code>s.close()</code>	Τερματίζει την λειτουργία της socket
<code>s.bind()</code>	Ενώνει την διεύθυνση του υπολογιστή με την socket

Πίνακας 1 Βασικές εντολές της βιβλιοθήκης socket

### Βιβλιοθήκη *threading*.

Στα ελληνικά *threading* σημαίνει νήμα. Τα νήματα είναι ελαφριές διεργασίες που τρέχουν παράλληλα μέσα στο χώρο μνήμης της διεργασίας. Τα νήματα μπορούν εύκολα να επικοινωνήσουμε μεταξύ τους και να δούνε κοινές μεταβλητές σε αντίθεση με τις διεργασίες που τρέχουν ανεξάρτητα. Εμείς χρησιμοποιήσαμε την *threading* για τους ήξεις λόγους. Αρχικά πολλές διεργασίες μεταξύ των διαφόρων χρηστών θα δημιουργούσαν προβλήματα συγχρονισμού τα οποία όμως αντιμετωπίζονται από αυτή τη βιβλιοθήκη. Επίσης κάποιες διεργασίες δεν πρέπει να τερματίζονται αλλά να συνεχίσουν να τρέχουν στο παρασκήνιο όπως η λήψη των μηνυμάτων από τους υπόλοιπους χρήστες.

## Δημιουργία του κώδικα

Για την δημιουργία του chat room χρειάζονται δύο κώδικες, ένας για το πρόγραμμα του server και ένας για τους client.

### Κώδικας του server

Για τον server χρησιμοποιήθηκαν οι βιβλιοθήκες *socket* και *time*. Αρχικά εισάγαμε ένα αντικείμενο τύπου *socket* το οποίο καθορίζει την οικογένεια και τον τύπο των πρωτοκόλλων που θα εισάγει. Στον διακοσμητή χρησιμοποιείται *UDP Protocol* επειδή θέλαμε να στέλνει την ίδια πληροφορία σε όλους τους συνδεδεμένους χρήστες. Για το *hostname* έχει οριστεί μια μεταβλητή η οποία αναφέρεται στον τοπικού δικτύου. Για τη μεταβλητή που καθορίζει την πύλη χρησιμοποιείται ένα *input* έτσι ώστε ο χρήστης να καθορίσει ο ίδιος αυτό που θέλει.

Στην συνέχεια δηλώνουμε, με την χρήση της εντολής *s.recvfrom(1024)* τον μέγιστο αριθμό των bytes που μπορεί να δεχτεί από ένα χρήστη. Με την χρήση ενός *while loop* γίνεται έλεγχος στα μηνύματα για την ύπαρξη της λέξης *Quit* εάν εντοπιστεί αυτό σηματοδοτεί και την αποσύνδεση του χρήστη από τον sever. Αν απουσιάζει το μήνυμα μαζί με τα δεδομένα που το αφορούν (όνομα-διεύθυνση χρήστη ,ημερομηνία και ώρα αποστολής) τυπώνονται στον server αυτό επιτυγχάνεται με την χρήση της εντολής «*print time.ctime(time.time()) + str(addr) + ": " + str(data)*».

Όπως εξηγήσαμε προηγουμένως σε ένα chat room ο διακοσμητής δέχεται δεδομένα από έναν χρήστη και τα στέλνει με την σειρά του στους υπόλοιπους που είναι συνδεδεμένοι. Στο πρόγραμμα μας αυτό το επιτυγχάνουμε με τον εξής τρόπο. Η λίστα *clients* που έχει δημιουργηθεί περιέχει τις διευθύνσεις των συνδεδεμένων υπολογιστών στο chat. Με την χρήση ενός «*for loop*» ως προς την λίστα στέλνουμε τα μηνύματα σε όλους τους χρήστες μέσω της εντολής της *socket* «*s.sendto(data, client)*».

### Κώδικας του προγράμματος client.

Οι βιβλιοθήκες της python που χρησιμοποιήθηκαν είναι η *socket*, η *time* και η *threading*

Αρχικά δημιουργήσαμε ένα αντικείμενο τύπου *Lock*. Έπειτα κατασκευάσαμε μια συνάρτηση η οποία ρυθμίζει την λήψη δεδομένων από τον διακοσμητή η οποία ελέγχει και την σειρά με την οποία λαμβάνονται. Όπως και στον server έτσι και εδώ αντικείμενο τύπου *socket* το οποίο καθορίζει την οικογένεια και τον τύπο των πρωτοκόλλων. με την χρήση εντολών εισόδου πληροφοριών από το πληκτρολόγιο ο χρήστης καθορίζει την διεύθυνση του server

που θέλει να συνδεθεί. Στην συνέχεια καλείται το νήμα που αφορά την λήψη μηνυμάτων από τον server.

Ένα από τα προβλήματα που αντιμετωπίσαμε ήταν αυτό της ανανέωσης της σελίδας. Αυτό επιλύθηκε με τον εξής τρόπο. Αρχικά με την χρήση ενός βρόγχου επανάληψης «while» ελέγχεται αν ο χρήστης επιθυμεί να αποσυνδεθεί. Έπειτα ένας άλλος έλεγχος αναγνωρίζει αν το μήνυμα είναι κενό, αν δεν είναι το στέλνει στον διακοσμητή. Κατά την διάρκεια αυτή στο παρασκήνιο υπάρχει το νήμα της λήψης δεδομένων, αυτό έρχεται ξανά στο προσκήνιο με αποτέλεσμα να μπορεί να εμφανίσει τα μηνύματα των υπόλοιπων χρηστών.

## Λειτουργία προγράμματος

Αρχικά ο διαχειριστής του server ανοίγει το πρόγραμμα. Αμέσως του ζητείται να πληκτρολογήσει τον αριθμό της πύλης που θέλει να χρησιμοποιήσει. Αμέσως ξεκινά να τρέχει ο διακοσμητής του chat room και πια μπορούν να συνδεθούν οι χρήστες.

Όταν κάποιος ξεκινά να τρέχει το πρόγραμμα των χρηστών αρχικά του ζητείται το hostname και ο αριθμός της πύλης του δωματίου που θέλει να συνδεθεί. Στην συνέχεια δίνει το όνομα που επιθυμεί να χρησιμοποιήσει. Ποια μπορεί να συμμετάσχει στην συνομιλία. Πληκτρολογεί το μήνυμα που θέλει και το στέλνει πατώντας enter. Η ανανέωση του chat room πραγματοποιείται όποτε ο χρήστης στέλνει ένα μήνυμα ή όποτε στέλνει κενό μήνυμα. Τα κενά μηνύματα δεν εμφανίζονται στους χρήστες όποτε πληκτρολογεί καποιος «quit» αποσυνδέεται από την συνομιλία.

## Βιβλιογραφία

Core Python Applications Programming Wesley Chun Prentice Hall

[http://www.tutorialspoint.com/python/python\\_multithreading.htm](http://www.tutorialspoint.com/python/python_multithreading.htm)

[http://www.tutorialspoint.com/python/python\\_networking.htm](http://www.tutorialspoint.com/python/python_networking.htm)

<http://csapp.cs.cmu.edu/2e/ch11-preview.pdf>

<http://www.python-course.eu/threads.php>

[https://en.wikipedia.org/wiki/Computer\\_network\\_programming](https://en.wikipedia.org/wiki/Computer_network_programming)

<http://www.binarytides.com/code-chat-application-server-client-sockets-python>

[https://en.wikipedia.org/wiki/Network\\_socket](https://en.wikipedia.org/wiki/Network_socket)