

Έκθεση ομαδικής εργασίας ομάδας 09 – Εφαρμογή Υπηρεσιών Φοιτητή

Βάσεις Δεδομένων, 2022-2023

Μπουμπουκιώτη Μαρία-Ελένη, Α.Μ.: 1072880 – up1072880@upnet.gr

Συμεωνίδης Ανδρέας, Α.Μ.: 1066573 – up1066573@upnet.gr

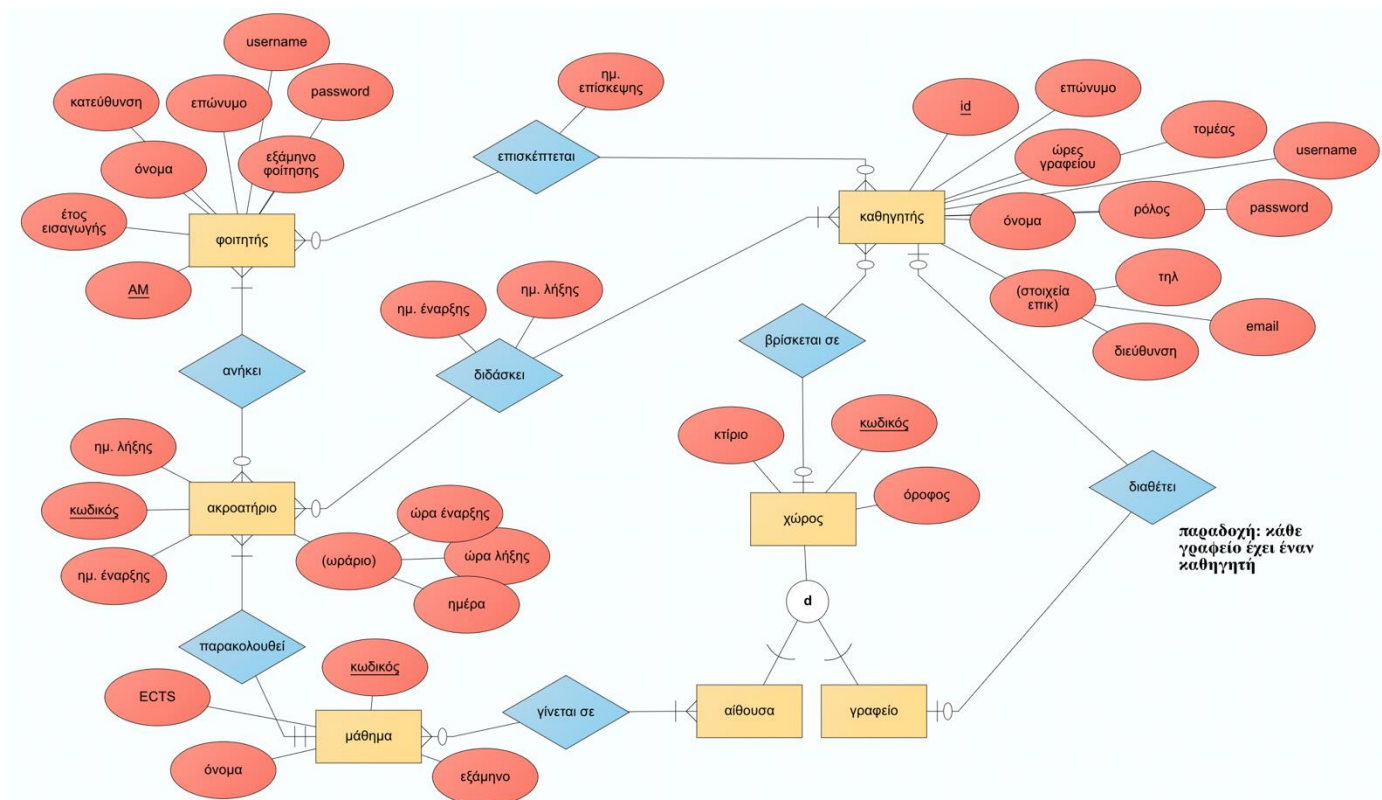
1 ΠΕΡΙΛΗΨΗ

Ζητούμενο της ομαδικής εργασίας που μας ανατέθηκε, είναι η ανάπτυξη εφαρμογής υπό την ομπρέλα του θέματος “Υπηρεσίες Φοιτητή”. Μετά από αρκετές ώρες brainstorming αναφορικά με τις ανάγκες μας ως φοιτητές, καταλήξαμε στην ιδέα του “UniCal”, μιας εφαρμογής που σκοπό έχει να βοηθήσει τους φοιτητές να οργανώνουν πιο αποτελεσματικά το πρόγραμμά τους. Τα ερωτήματα ως εκ τούτου που καλείται να λύσει η εφαρμογή μας, προέκυψαν έπειτα από συζήτηση τόσο μεταξύ μας, όσο και με άλλους φοιτητές και επεξεργασία των απαντήσεων που δόθηκαν στην ερώτηση “Τι θα ήθελες να βρίσκεις χωρίς κόπο, με το πάτημα ενός κουμπιού;” Το εννοιολογικό μοντέλο μας, χτίστηκε γύρω από τον φοιτητή και τα μαθήματα που αυτός επιλέγει να παρακολουθήσει. Στην εφαρμογή ενσωματώθηκε η δυνατότητα χρήσης και από καθηγητές, με σκοπό την πιο λειτουργική συνεργασία τους με τους φοιτητές, προς διευκόλυνση και των 2 πλευρών. Το κομμάτι που μας δυσκόλεψε περισσότερο ήταν στη φάση του σχεδιασμού της βάσης, καθώς δυσκολευτήκαμε αρκετά να συγκεκριμενοποιήσουμε το γενικότερο θέμα της «Εφαρμογής υπηρεσιών φοιτητή» σε ένα πιο συγκεκριμένο, που θα μας επέτρεπε την υλοποίησή του.

2 ΜΕΘΟΔΟΛΟΓΙΑ

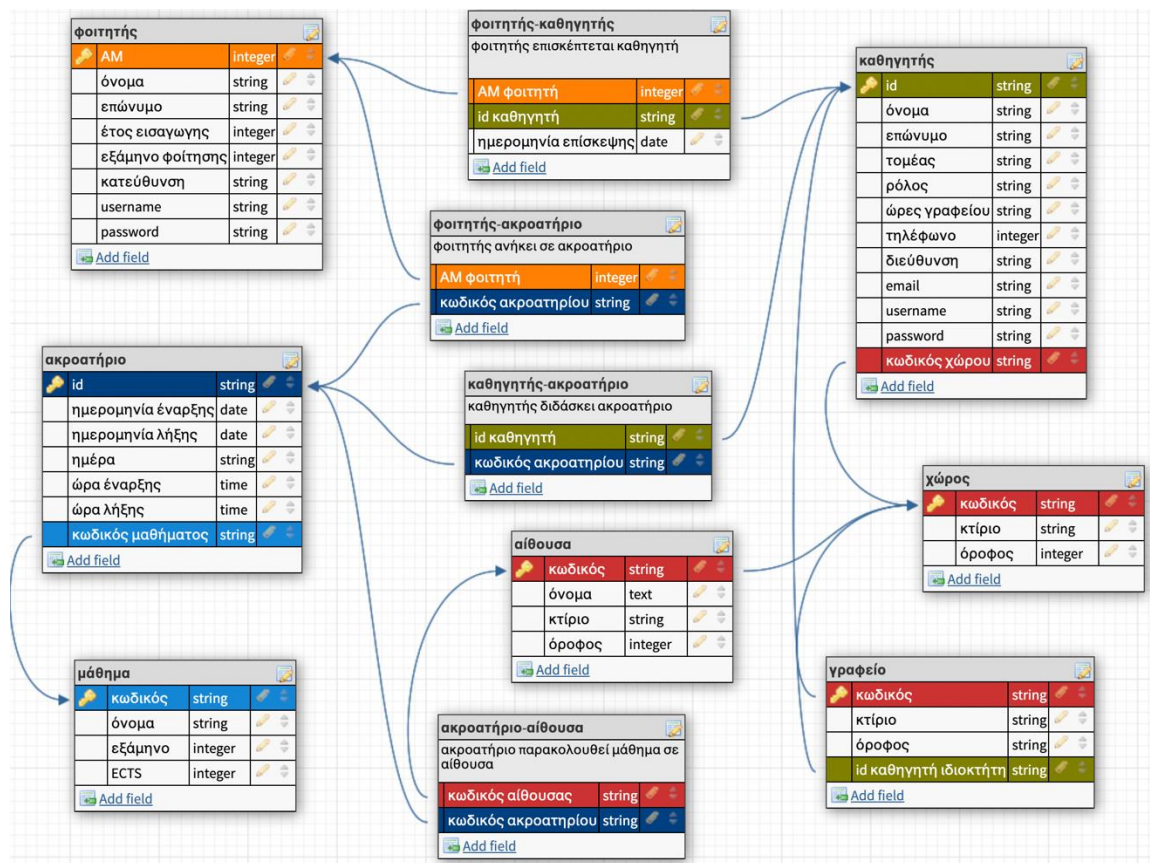
Όπως αναφέρθηκε και στην περίληψη, πριν ξεκινήσουμε την δημιουργία του ERD μοντέλου για την εφαρμογή μας, αφιερώσαμε πρωτίστως χρόνο στο να μετασχηματίσουμε το θέμα “Υπηρεσίες Φοιτητή” σε ένα πιο ειδικό θέμα, με γνώμονα την από κοινού επιθυμία μας να δώσουμε μια λειτουργική λύση σε μερικά από τα ήδη υπάρχοντα προβλήματα της φοιτητικής κοινότητας, για τα οποία δεν υπάρχει έως τώρα κάποιος tech τρόπος επίλυσης. Για να τα ορίσουμε καλύτερα, συμβουλευτήκαμε συμφοιτητές και φίλους και ανατρέξαμε στις δικές μας καθημερινές δυσκολίες στη σχολή. Παρ’ όλα αυτά, έπρεπε να απορρίψουμε πολλές ιδέες που θα θέλαμε να υλοποιήσουμε ως κομμάτι της εφαρμογής, διότι δεν ήταν δυνατό να τις συμπεριλάβουμε βάση του διαθέσιμου χρόνου που είχαμε για την υλοποίηση. Ερωτήματα όπως “Ποιά από τα μαθήματα επικαλύπτονται μεταξύ τους ώστε να μην τα διαλέξω στο ίδιο εξάμηνο;” ή “πού ακριβώς είναι η αίθουσα ΑΠΜ4 που λέει στο πρόγραμμα ότι έχω φυσική;”, καθώς και το να ρίχνεις μια γρήγορη ματιά στο πρόγραμμά σου για την Τετάρτη, χωρίς να χρειάζεται να “διασταυρώσεις” τα διάφορα μαθήματα που παρακολουθείς, από το PDF του προγράμματος εξαμήνου, αποτέλεσαν βασικά προβλήματα που θελήσαμε να λύσουμε.

Στην συνέχεια, προχωρήσαμε στην δημιουργία μιας πρώτης έκδοσης του εννοιολογικού μας μοντέλου, η οποία περιείχε μερικές βασικές οντότητες και συσχετίσεις. Έπειτα, αφού συμβουλευτήκαμε τον κ. Σιντόρη και λάβαμε κάποια πρώτα σχόλια για την ιδέα μας και το σκεπτικό υλοποίησης της, προχωρήσαμε στην προσθήκη περαιτέρω οντοτήτων, συσχετίσεων και γνωρισμάτων, με έμφαση στα βασικά στοιχεία που καθιστούν την εφαρμογή μας λειτουργική. Για την δημιουργία του εννοιολογικού μοντέλου, έγινε χρήση του εργαλείου ERD Maker (<https://erdmaker.com/>), που προτάθηκε στα πλαίσια των εργαστηριακών ασκήσεων. Το τελικό εννοιολογικό μοντέλο της βάσης μας είναι το εξής:



Εικόνα 1: Εννοιολογικό Μοντέλο ERD της βάσης μας

Μετά την ολοκλήρωση του εννοιολογικού διαγράμματος, προχωρήσαμε στον μετασχηματισμό του διαγράμματος αυτού στο αντίστοιχο σχεσιακό μοντέλο, με τη βοήθεια του εργαλείου DB Designer (<https://www.dbdesigner.net/>). Κατά τον μετασχηματισμό εφαρμόσαμε τα βήματα που παρουσιάστηκαν κατά την διάρκεια των διαλέξεων και εργαστηρίων, όσον αφορά τις διάφορες συσχετίσεις. Ιδιαίτερη προσοχή δώσαμε στην πολλαπλότητα των σχέσεων μεταξύ των οντοτήτων με έμφαση στην σχέση N-N, από την οποία προέκυψαν στο σχεσιακό μοντέλο μας οι ενδιάμεσοι βοηθητικοί πίνακες σύνδεσης. Η δομή του σχεσιακού μοντέλου, καθώς και τα κλειδιά μας, Primary και Foreign, φαίνονται παρακάτω:



Εικόνα 2: Λογικό σχεσιακό μοντέλο της βάσης μας

Επόμενο βήμα αποτέλεσε η δημιουργία των διάφορων πινάκων της βάσης δεδομένων μας. Καθώς αντιληφθήκαμε πως μέσω της εφαρμογής DB Browser for SQLite είναι εφικτό να δημιουργήσουμε πιο γρήγορα τους πίνακες μας, όπως και να εισάγουμε σε αυτούς εγγραφές, προτιμήσαμε να ακολουθήσουμε αυτή τη διαδικασία και να εξάγουμε από εκεί τις SQL εντολές, αντί να εισάγουμε αναλυτικά και “με το χέρι” τα δεδομένα μας. Όταν οι πίνακες μας είχαν έναν ικανοποιητικό αριθμό εγγραφών, προχωρήσαμε στην διασύνδεση των δεδομένων μεταξύ τους μέσω των βοηθητικών πινάκων, ώστε να επιτευχθεί η αλληλεπίδραση τους. Με την ολοκλήρωση αυτού του βήματος, ακολούθησε η εκτέλεση κάποιων πρώτων queries, με σκοπό να εξασφαλιστεί η σωστή λειτουργία της βάσης μας. Στην συνέχεια, τα διάφορα ερωτήματα μας, οδήγησαν στην ανάγκη εισαγωγής και επιπλέον εγγραφών. Με το πέρας και αυτής της διαδικασίας, περάσαμε στη φάση Γ, δηλαδή τη σχεδίαση μιας εφαρμογής σε γλώσσα Python, με σκοπό την αλληλεπίδραση του χρήστη με τη βάση μας. Κύριο μέλημα της σχεδιάσής μας ήταν να δίνεται στο χρήστη η δυνατότητα να εκτελεί πλήθος διαφορετικών λειτουργιών - άλλες πιο απλές και άλλες πιο σύνθετες – μέσα από μία ενιαία πλατφόρμα. Η εφαρμογή εν τέλει διαρθρώθηκε γύρω από δύο βασικούς άξονες: την αλληλεπίδραση του user ως φοιτητής, αλλά και την αλληλεπίδραση του στην περίπτωση που αποτελεί καθηγητής, με σκοπό την πιο λειτουργική συνεργασία του με τους φοιτητές, προς διευκόλυνση και των 2 πλευρών.

3 ΑΞΙΟΛΟΓΗΣΗ

Τα κριτήρια αξιολόγησης που θέσαμε, τόσο στην αρχή του project μας, όσο και κατά την ολοκλήρωση του, συνδέονται άμεσα με τον κύριο στόχο μας, τη λειτουργικότητα της βάσης μας. Ήταν εξαιρετικά σημαντικό για εμάς, να δημιουργήσουμε μια απλή, κατανοητή εφαρμογή, που θα έδινε στο χρήστη διάφορες δυνατότητες. Γνωρίζοντας πως μια τέτοια εφαρμογή θα μπορούσε να επεκταθεί σε πολλά και διαφορετικά μεταξύ τους ερωτήματα, επιλέξαμε να την δομήσουμε έτσι ώστε να μπορούν να υλοποιηθούν διάφορα queries, όπως η αναζήτηση και επιλογή μαθημάτων (εγγραφή σε ακροατήριο), η ενδεχόμενη αναζήτηση του ιστορικού επισκέψεων σε κάποιον διδάσκοντα, η προβολή των μαθημάτων που επικαλύπτονται, η εύρεση της τοποθεσίας γραφείων και αιθουσών στα βασικά κτίρια που εξυπηρετεί η έως τώρα εκδοχή μας (κτίρια CEID, HMTY, Πρυτανείας, Πολιτικών Μηχανικών, ΑΠ) και δυνητικά ,κατ'επέκταση, σε ολόκληρη την πανεπιστημιούπολη, κ.ο.κ

Σαν πρώτο στάδιο αξιολόγησης είχαμε αρχικά, όσον αφορά την δημιουργία του εννοιολογικού μοντέλου, την παρουσίαση που πραγματοποιήσαμε στα μέσα του εξαμήνου. Αφότου λάβαμε σχόλια σχετικά με το μοντέλο μας, βελτιώσαμε τα σημεία που μας επισημάνθηκαν και δουλέψαμε περαιτέρω πάνω σε αυτό, ώστε να απαιτούνται όσο το δυνατόν λιγότερες αλλαγές στην συνέχεια.

Έπειτα, όσον αφορά την μετατροπή του εννοιολογικού μοντέλου στο αντίστοιχο σχεσιακό διάγραμμα, δώσαμε πρωτίστως έμφαση στη σωστή αποτύπωση των συσχετίσεων, ώστε να αποτυπώνονται με τον κατάλληλο τρόπο – είτε με τη δημιουργία πινάκων συσχέτισης, είτε με την δημιουργία ξένων κλειδιών – οι σχέσεις πολλαπλότητας ανάμεσα στις οντότητες του εννοιολογικού διαγράμματός μας. Αναφορικά με την αξιολόγηση της βάσης, ως θεμελιώδη στόχο είχαμε τον αυστηρό καθορισμό όλων των κανόνων που είχαμε θεσπίσει κατά τη σχεδίαση του εννοιολογικού και του σχεσιακού διαγράμματος, και στους οποίους όφειλαν να υπόκεινται οι πίνακες των δεδομένων μας.

Τέλος, σχετικά με το τελευταίο κομμάτι του πρότζεκτ μας, την εφαρμογή που αναπτύξαμε σε γλώσσα Python, είχαμε ως στόχο την δημιουργία ενός «ολοκληρωμένου» αποτελέσματος, το οποίο θα έχει ως κύρια χαρακτηριστικά 1) μη προβληματική χρήση της εφαρμογής, ανεξάρτητα από τα inputs του χρήστη 2) αυτοματοποιημένες, γρήγορες διαδικασίες, με σκοπό να είναι κατανοητό από το χρήστη το πως μπορεί να χειριστεί την εφαρμογή.

Ως εκ τούτου, προς επίτευξη των παραπάνω στόχων, θεωρήσαμε συνετό να δοθεί προτεραιότητα στη λειτουργικότητα και την ολοκληρωμένη εικόνα της βάσης μας, παρά στην εισαγωγή γραφικού περιβάλλοντος.

4 ΔΕΛΟΜΕΝΑ

Η εισαγωγή των δεδομένων πραγματοποιήθηκε χειροκίνητα με τυχαίο τρόπο, προσομοιώνοντας το πως θα εισάγονταν από την πλατφόρμα του Eclass. Η έλλειψη κάποιου τρόπου απευθείας εισαγωγής των δεδομένων (λόγω και των κανόνων GDPR), περιόρισε σημαντικά το πλήθος των υπάρχουσών εγγραφών, ωστόσο η λειτουργικότητα μπορεί να επεκταθεί εύκολα και για πολύ μεγαλύτερο πλήθος δεδομένων.

5 ΚΑΤΑΜΕΡΙΣΜΟΣ ΕΡΓΑΣΙΩΝ

Ως επί το πλείστον ασχοληθήκαμε από κοινού σε όλες τις φάσεις δημιουργίας της βάσης μας, έχοντας θέση εξαρχής ως κανόνα της συνεργασίας μας την ταυτόχρονη εργασία σε κοινές ώρες, με σκοπό την πιο αποτελεσματική συνεννόηση

και τη λήψη κοινών αποφάσεων. Η δημιουργία του ERD, του λογικού σχήματος, της αναφοράς και της παρουσίασης πραγματοποιήθηκε με συνεισφορά και από τα δύο μέλη της ομάδας. Ένα σημείο που χωρίσαμε κάπως ρόλους ήταν στις φάσεις Β και Γ. Αφού πρωτίστως καταλήξαμε στο πώς επιθυμούμε να κινηθούμε και ολοκληρώσαμε την αρχική δημιουργία μιας πρώτης έκδοσης της βάσης δεδομένων, η Μαριλένα ανέλαβε να εμπλουτίσει τη βάση με στοιχεία, ενώ ο Ανδρέας να αναπτύσσει τον κορμό της εφαρμογής Python με κάποια βασικά χαρακτηριστικά και λειτουργικότητα. Οι δύο φάσεις, ούσες άμεσα συνδεδεμένες μεταξύ τους, κινούνταν παράλληλα, σε κοινό working session, όπου μπορούσαμε να συζητάμε, να δίνουμε inputs για την πορεία της δουλειάς μας, αλλά και να διακόπτουμε τα επιμέρους tasks μας, με σκοπό να λύσουμε οποιοδήποτε κώλυμα προέκυπτε από κοινού, για κάθε φάση. Κατά την ολοκλήρωση κάθε working session, συζητούσαμε την πρόοδο που έχουμε κάνει και θέταμε ερωτήματα και προτάσεις προς βελτίωση. Μετά την ολοκλήρωση της βάσης, και οι 2 συνεργαστήκαμε στην προσθήκη νέων λειτουργιών της εφαρμογής.

6 ΧΡΟΝΟΔΙΑΓΡΑΜΜΑ

Για την δημιουργία της εφαρμογής ακολουθήσαμε τις 3 φάσεις που προτείνονταν στο εγχειρίδιο του μαθήματος, δηλαδή την δημιουργία του εννοιολογικού μοντέλου, και τον μετασχηματισμό του σε αντίστοιχο λογικό μοντέλο, την δημιουργία της βάσης μας και τον εμπλουτισμό της με δεδομένα, ενώ σε τελευταίο στάδιο δημιουργήσαμε την εφαρμογή μας με χρήση Python ώστε ο χρήστης να μπορεί να αλληλοεπιδράσει με την βάση.

Αναλυτικά το χρονοδιάγραμμα που ακολουθήσαμε φαίνεται παρακάτω:

- Νοέμβριος-μέχρι την ενδιάμεση παρουσίαση του Project: Έρευνα για τις ανάγκες των φοιτητών – δόμηση της ιδέας – δημιουργία του εννοιολογικού διαγράμματος της βάσης δεδομένων
- Τέλη Νοεμβρίου - Μέσα Δεκεμβρίου: Βελτίωση του εννοιολογικού διαγράμματος μετά την ενδιάμεση παρουσίαση – Μετασχηματισμός του εκτεταμένου διαγράμματος οντοτήτων συσχετίσεων στο αντίστοιχο λογικό σχεσιακό μοντέλο
- Τέλη Δεκεμβρίου: Δημιουργία της αρχικής βάσης δεδομένων – Προσθήκη στοιχείων σε αυτήν – Εκκίνηση ανάπτυξης εφαρμογής στο περιβάλλον της Python
- Αρχές Ιανουαρίου: Ενσωμάτωση των queries (τόσο πιο απλών εκδοχών, όσο και των σύνθετων) – Προετοιμασία αναφοράς και παρουσίασης της ομαδικής εργασίας .

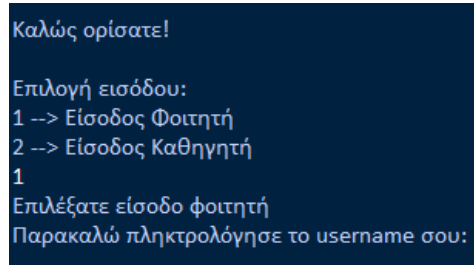
7 ΠΑΡΑΡΤΗΜΑ

7.1 Εκτέλεση της εφαρμογής:

- 1) Κατεβάζουμε το αρχείο με κατάληξη .db και ένα από τα εκτελέσιμα αρχεία με κατάληξη .exe, .py, από τον σύνδεσμο: https://github.com/symeona17/ECE_UPatras-Databases-UniCal
- 2) Τοποθετούμε το αρχείο .db στον ίδιο φάκελο με το εκτελέσιμο αρχείο που έχουμε κατεβάσει
- 3) Τρέχουμε το εκτελέσιμο αρχείο

7.2 Οδηγίες χρήσης της εφαρμογής

- 1) Κατά την είσοδό σας, θα ερωτηθείτε από την εφαρμογή αν θέλετε να συνδεθείτε ως φοιτητής ή ως καθηγητής.



Καλώς ορίσατε!

Επιλογή εισόδου:

1 --> Είσοδος Φοιτητή

2 --> Είσοδος Καθηγητή

1

Επιλέξατε είσοδο φοιτητή

Παρακαλώ πληκτρολογήσε το username σου:

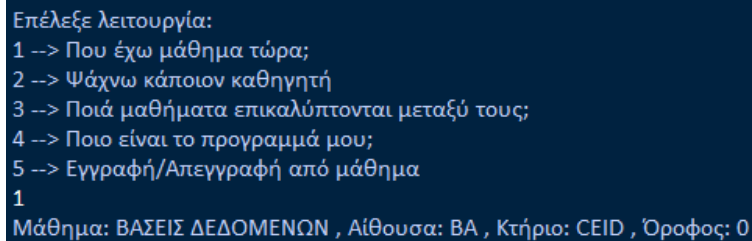
Πληκτρολογώντας τον αριθμό 1 ή 2, θα σας ζητηθεί να πληκτρολογήσετε το Username σας και κατόπιν το password σας. Αν συμπληρώσετε σωστά τα στοιχεία, θα μεταφερθείτε στο επόμενο βήμα της εφαρμογής, αλλιώς θα σας ζητηθεί η σύνδεσή σας ξανά, εμφανίζοντας το αντίστοιχο μήνυμα λάθους.

2) Features για φοιτητές

Για τους φοιτητές, η εφαρμογή έχει 5 διαθέσιμα features, τα οποία μπορεί να επιλέξει ο φοιτητής πληκτρολογώντας ένα αριθμό 1-5.

a. Εμφάνιση μαθήματος φοιτητή εκείνη τη χρονική στιγμή

Πατώντας τον αριθμό 1, η εφαρμογή θα σας εμφανίσει το μάθημα που έχετε εκείνη τη στιγμή (αν έχετε κάποιο μάθημα εκείνη την ώρα), μαζί με πληροφορίες για την αίθουσα στην οποία γίνεται το μάθημά σας. Αν κανένα από τα μαθήματα που είστε εγγεγραμμένος δεν είναι σε εξέλιξη, η εφαρμογή θα σας ενημερώσει αντίστοιχα.



Επέλεξε λειτουργία:

1 --> Που έχω μάθημα τώρα;

2 --> Ψάχνω κάποιον καθηγητή

3 --> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;

4 --> Ποιο είναι το προγραμμά μου;

5 --> Εγγραφή/Απεγγραφή από μάθημα

1

Μάθημα: ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ , Αίθουσα: ΒΑ , Κτήριο: CEID , Όροφος: 0

b. Αναζήτηση διδάσκοντα

Πατώντας τον αριθμό 2, η εφαρμογή σας δίνει την δυνατότητα να αναζητήσετε το γραφείο ενός διδάσκοντα, ή την αίθουσα στην οποία βρίσκεται αυτή τη στιγμή, αν ο διδάσκοντας είναι σε μάθημα. Η εφαρμογή θα σας ζητήσει να εισάγετε το ID του καθηγητή που αναζητείτε.

```
Επέλεξε λειτουργία:  
1 --> Που έχω μάθημα τώρα;  
2 --> Ψάχνω κάποιον καθηγητή  
3 --> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;  
4 --> Ποιο είναι το προγραμμά μου;  
5 --> Εγγραφή/Απεγγραφή από μάθημα  
2  
Εγγεγραμμένοι καθηγητές τμήματος:  
ID Καθηγητή: giannopoulos Ονοματεπώνυμο Καθηγητή: Ιωάννης Γιαννόπουλος  
ID Καθηγητή: melet.raz Ονοματεπώνυμο Καθηγητή: Μελέτιος Φωτεινός Ραζόφ  
ID Καθηγητή: syrou Ονοματεπώνυμο Καθηγητή: Στυλιανή Σύρου  
ID Καθηγητή: parageorgiou Ονοματεπώνυμο Καθηγητή: Ευάγγελος Παπαγεωργίου  
ID Καθηγητή: anastasiou Ονοματεπώνυμο Καθηγητή: Θανάσης Αναστασίου  
ID Καθηγητή: andreou Ονοματεπώνυμο Καθηγητή: Ιωάννης Ανδρέου  
ID Καθηγητή: apostolou Ονοματεπώνυμο Καθηγητή: Αριστοτέλης Αποστόλου  
ID Καθηγητή: alexoroulou Ονοματεπώνυμο Καθηγητή: Φωτεινή Αλεξοπούλου  
Πληκτρολόγησε το ID του καθηγητή που αναζητείς:
```

c. Εμφάνιση όλων των μαθημάτων με κοινές ώρες

Πατώντας τον αριθμό 3, η εφαρμογή θα σας εμφανίσει όλα τα μαθήματα που έχουν κοινές ώρες μεταξύ τους.

```
Επέλεξε λειτουργία:  
1 --> Που έχω μάθημα τώρα;  
2 --> Ψάχνω κάποιον καθηγητή  
3 --> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;  
4 --> Ποιο είναι το προγραμμά μου;  
5 --> Εγγραφή/Απεγγραφή από μάθημα  
3  
Το μάθημα ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ επικαλύπτεται με το μάθημα ΟΛΟΚΛΗΡΩΜΕΝΑ ΗΛΕΚΤΡΟΝΙΚΑ  
Το μάθημα ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ επικαλύπτεται με το μάθημα ΣΥΓΧΡΟΝΗ ΦΥΣΙΚΗ  
Το μάθημα ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ επικαλύπτεται με το μάθημα ΥΨΗΛΕΣ ΤΑΣΕΙΣ
```

d. Εμφάνιση προγράμματος φοιτητή

Πατώντας τον αριθμό 4, η εφαρμογή θα σας εμφανίσει το πρόγραμμά σας με βάση τα μαθήματα που είστε εγγεγραμμένος, είτε για μία συγκεκριμένη ημέρα είτε για όλες τις μέρες.

```
Παρακαλώ επέλεξε για ποια ημέρα θα ήθελες να δεις το προγραμμά σου:  
0 --> Κυριακή  
1 --> Δευτέρα  
2 --> Τρίτη  
3 --> Τετάρτη  
4 --> Πέμπτη  
5 --> Παρασκευή  
6 --> Σάββατο  
7 --> Όλες τις μέρες  
2  
Την Τρίτη έχεις:  
ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ από τις 09:00 μέχρι τις 11:00 στην αίθουσα ΚΥΠΕΣII η οποία βρίσκεται στο κτήριο HMTY στον όροφο 0
```

e. Εγγραφή/Απεγγραφή από μάθημα

Πατώντας τον αριθμό 5, η εφαρμογή θα σας ρωτήσει αν θέλετε να εγγραφείτε σε κάποιο μάθημα ή να απεγγραφείτε από κάποιο που είστε γραμμένος. Σε κάθε περίπτωση, η εφαρμογή θα σας εμφανίσει τα ακροατήρια διαθέσιμα για εγγραφή ή απεγγραφή αντίστοιχα, και εσείς θα πρέπει να εισάγετε τον κωδικό ακροατηρίου για να προχωρήσετε.

Επέλεξε:

1-->Εγγραφή σε μάθημα

2 --> Απεγγραφή από μάθημα

1

Τα διαθέσιμα μαθήματα προς εγγραφή είναι:

('BΔ1', 'ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ')

3) Features για καθηγητές

Για τους καθηγητές, η εφαρμογή έχει 3 διαθέσιμα features, τα οποία μπορεί να επιλέξει ο καθηγητής πληκτρολογώντας ένα αριθμό 1-3.

a. Εύρεση διαθεσιμότητας ακροατηρίου

Πατώντας τον αριθμό 1, μπορείτε να δείτε την διαθεσιμότητα των φοιτητών στα ακροατήρια που είστε διδάσκοντας για μία δοσμένη ημέρα και ώρα, σε περίπτωση για παράδειγμα που θέλετε να βρείτε μια κατάλληλη ώρα για αναπλήρωση διάλεξης. Η εφαρμογή θα σας ζητήσει να επιλέξετε για ποιο ακροατήριο θα θέλατε να κάνετε αναζήτηση, και για ποια ημέρα και ώρα.

```
Επιλέξτε λειτουργία:
1 --> Εύρεση διαθεσιμότητας φοιτητών
2 --> Εύρεση διαθέσιμης αίθουσας
3 --> Αλλαγή διδάσκοντα ακροατηρίου
1
Είστε διδάσκων στα ακροατήρια:
ΥΤ του μαθήματος ΨΗΛΕΣ ΤΑΣΕΙΣ
ΔΙ του μαθήματος ΑΝΑΛΥΣΗ ΔΙΚΤΥΩΝ ΙΣΧΥΟΣ
Πληκτρολογήστε τον κωδικό ακροατηρίου για το οποίο θα θέλατε να κάνετε αναζήτηση:
ΥΤ
Θα ήθελα να δω τη διαθεσιμότητα του ακροατηρίου την μέρα:
0 --> Κυριακή
1 --> Δευτέρα
```

b. Εύρεση διαθέσιμης αίθουσας

Πατώντας τον αριθμό 2, η εφαρμογή σας επιτρέπει να βρείτε ποιες αίθουσες θα είναι κενές βάσει προγράμματος, για την οργάνωση οποιασδήποτε δραστηριότητας. Η εφαρμογή θα σας ζητήσει να επιλέξετε για ποια μέρα και για ποιο διάστημα ωρών θα θέλατε να κάνετε αναζήτηση.

```
Παρακαλώ επιλέξτε ποια ημέρα θα θέλατε να αναζητήσετε διαθέσιμη αίθουσα:
0 --> Κυριακή
1 --> Δευτέρα
2 --> Τρίτη
3 --> Τετάρτη
4 --> Πέμπτη
5 --> Παρασκευή
6 --> Σάββατο
1
Θα ήθελα διαθέσιμη αίθουσα από τις: (Σε μορφή 0-24)
10
Μέχρι τις: (Σε μορφή 0-24)
13
Οι διαθέσιμες αίθουσες εκείνες τις ώρες είναι:
Αίθουσα ΒΑ στο κτήριο CEID στον όροφο 0
Αίθουσα ΑΠΜ4 στο κτήριο Πολιτικών Μηχανικών στον όροφο 1
```

c. Αλλαγή (update) του διδάσκοντα ενός ακροατηρίου

Πατώντας τον αριθμό 3, η εφαρμογή σας επιτρέπει να αλλάξετε τον διδάσκοντα που είναι assigned σε κάποιο ακροατήριο. Η εφαρμογή θα σας ζητήσει να πληκτρολογήσετε τον κωδικό του ακροατηρίου που θέλετε να γίνει η αλλαγή, καθώς και το ID του νέου διδάσκοντα αυτού του ακροατηρίου.

```
3
Αυτή τη στιγμή διδάσκοντες στα ακροατήρια είναι οι:
Ιωάννης Γιαννόπουλος με κωδικό ID gianopoulos στο ακροατήριο ΥΤ του μαθήματος ΨΗΛΕΣ ΤΑΣΕΙΣ
Στυλιανή Σύρου με κωδικό ID syrou στο ακροατήριο ΒΔ2 του μαθήματος ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ
Μελέτιος Φωτεινός Ραζόφ με κωδικό ID melet.raz στο ακροατήριο ΒΔ1 του μαθήματος ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ
Ιωάννης Ανδρέου με κωδικό ID andreou στο ακροατήριο ΔΤ1 του μαθήματος ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ
Μελέτιος Φωτεινός Ραζόφ με κωδικό ID melet.raz στο ακροατήριο ΒΔ3 του μαθήματος ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ
Αριστοτέλης Αποστόλου με κωδικό ID apostolou στο ακροατήριο ΣΦ1 του μαθήματος ΣΥΓΧΡΟΝΗ ΦΥΣΙΚΗ
Αριστοτέλης Αποστόλου με κωδικό ID apostolou στο ακροατήριο ΣΦ2 του μαθήματος ΣΥΓΧΡΟΝΗ ΦΥΣΙΚΗ
Θανάσης Αναστασίου με κωδικό ID anastasiou στο ακροατήριο ΨΑ του μαθήματος ΨΗΦΙΑΚΗ ΛΟΓΙΚΗ
Ευάγγελος Παπαγεωργίου με κωδικό ID papageorgiou στο ακροατήριο ΤΝ του μαθήματος ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ
Ιωάννης Γιαννόπουλος με κωδικό ID gianopoulos στο ακροατήριο ΔΙ του μαθήματος ΑΝΑΛΥΣΗ ΔΙΚΤΥΩΝ ΙΣΧΥΟΣ
Μελέτιος Φωτεινός Ραζόφ με κωδικό ID melet.raz στο ακροατήριο ΔΤ2 του μαθήματος ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ
Φωτεινή Αλεξοπούλου με κωδικό ID alexorouli στο ακροατήριο ΟΗ του μαθήματος ΟΛΟΚΛΗΡΩΜΕΝΑ ΗΛΕΚΤΡΟΝΙΚΑ

Επιλέξτε σε ποιο ακροατήριο θα θέλατε να αλλάξετε τον διδάσκοντα. Πληκτρολογήστε τον κωδικό του ακροατηρίου.
```

7.3 Παραδείγματα χρήσης της εφαρμογής

1. Είσοδος στην εφαρμογή

Καλώς ορίσατε!

Επιλογή εισόδου:

1 --> Είσοδος Φοιτητή

2 --> Είσοδος Καθηγητή

|

```
397
398 #Main
399
400 print("\nΚαλώς ορίσατε!")
401
402 def main(): #Επιλογή εισόδου, διαφορετικές συναρτήσεις για φοιτητή ή καθηγητή
403     x=0
404     while x!=" ":
405         x=input("\nΕπιλογή εισόδου: \n1 --> Είσοδος Φοιτητή \n2 --> Είσοδος Καθηγητή \n")
406         if x=="1":
407             login_foithths()
408         elif x=="2":
409             login_ka8hghths()
410         elif x==" ":
411             print("Εξοδος προγράμματος")
412             return
413         else:
414             print("Παρακαλώ δώσε κατάλληλο input\n")
415             main()
416         break
417     return
418
419 main()
420 data = c.fetchall();
421 conn.close();
```

2. Login & Main screen Φοιτητή

Καλώς ορίσατε!

Επιλογή εισόδου:

1 --> Είσοδος Φοιτητή

2 --> Είσοδος Καθηγητή

1

Επιλέξατε είσοδο φοιτητή

Παρακαλώ πληκτρολογήστε το username σου:

up1058216

Παρακαλώ πληκτρολογήστε τον κωδικό σου:

dscds

Επιτυχής σύνδεση!

Καλώς όρισε Αγγελική

Επέλεξε λειτουργία:

1 --> Που έχω μάθημα τώρα;

2 --> Ψάχνω κάποιον καθηγητή

3 --> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;

4 --> Ποιο είναι το προγραμματά μου;

5 --> Εγγραφή/Απεγγραφή από μάθημα

```
6 #Συναρτήσεις Login
7 def login_foithths(): #Login Φοιτητή
8     print("Επιλέξατε είσοδο φοιτητή")
9     username=input("Παρακαλώ πληκτρολογήστε το username σου: \n")
10    buf="SELECT * FROM Student WHERE username='"+username+"';"
11    try:
12        c.execute(buf)
13        result = c.fetchall()
14        if(len(result)!=1):
15            print("Το όνομα χρήστη που πληκτρολογήσατε δεν είναι σωστό. Παρακαλώ δοκιμάστε ξανά.")
16            main()
17            return
18        id=result[0][6]
19        AM=result[0][0]
20        swsto_password=result[0][7]
21        password=input("Παρακαλώ πληκτρολογήστε τον κωδικό σου: \n")
22        if password==swsto_password:
23            print("Επιτυχής σύνδεση! \nΚαλώς όρισε", result[0][1])
24            foithths(AM)
25        else:
26            print("Ο κωδικός που πληκτρολογήσατε είναι λανθασμένος. Παρακαλώ δοκιμάστε ξανά.\n")
27            main()
28        conn.commit()
29    except:
30        print("Failed to execute")
31    return
```

```
50
51 def foithths(AM):
52     #Επιλογή λειτουργίας
53     epilogh_foit=input("\nΕπέλεξε λειτουργία:\n1 --> Που έχω μάθημα τώρα;\n2 --> Ψάχνω κάποιον καθηγητή\n\n")
54     3 --> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;\n4 --> Ποιο είναι το προγραμματά μου; \n5 --> Εγγραφή/Απεγγραφή από μάθημα \n")
55     while True:
56         if epilogh_foit=="1":
57             pou_ma8hma(AM)
58         elif epilogh_foit=="2":
59             pou_ka8hghths(AM)
60         elif epilogh_foit=="3":
61             epikalisph(AM)
62         elif epilogh_foit=="4":
63             programma(AM)
64         elif epilogh_foit=="5":
65             eggrafh(AM)
66         elif epilogh_foit==" ":
67             print("Επιστροφή στην αρχική σελίδα.")
68             main()
69         else:
70             print("Παρακαλώ δώσε κατάλληλο input.\n")
71             foithths(AM)
72         break
```

3. Φοιτητής: «Που έχω μάθημα τώρα;»

```
85 def pou_ma8hma(AM): #ζυνάρτηση "Που έχω μάθημα τώρα"
86     buf="SELECT Course.name, Auditorium.name, Auditorium.building, Auditorium.floor FROM (Audience JOIN Course on course_id=Course.id)\
87 JOIN audience_auditorium JOIN Auditorium JOIN student_audience ON student_AM="+str(AM)+" WHERE start_time<=time('now','localtime')\
88 AND end_time>=time('now','localtime') AND start_date<=date('now','localtime') AND end_date>=date('now','localtime') AND strftime('%w','now')=Audience.day GROUP BY Course.id;"
89     try:
90         c.execute(buf)
91         result=c.fetchall()
92         print("Μάθημα:", result[0][0], ", Αίθουσα:", result[0][1], ", Κτήριο:", result[0][2], ", Όροφος:", result[0][3])
93     except:
94         print("Δεν έχεις κάποιο μάθημα τώρα.")
95     foithths(AM)
96     return
```

Καλώς ορίσατε!

Επιλογή εισόδου:

1 --> Είσοδος Φοιτητή

2 --> Είσοδος Καθηγητή

1

Επιλέξατε είσοδο φοιτητή

Παρακαλώ πληκτρολόγησε το username σου:

up1058216

Παρακαλώ πληκτρολόγησε τον κωδικό σου:

dscds

Επιτυχής σύνδεση!

Καλώς όρισες Αγγελική

Επέλεξε λειτουργία:

1 --> Που έχω μάθημα τώρα;

2 --> Ψάχνω κάποιον καθηγητή

3 --> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;

4 --> Ποιο είναι το προγραμμά μου;

5 --> Εγγραφή/Απεγγραφή από μάθημα

1

Μάθημα: ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ , Αίθουσα: BA , Κτήριο: CEID , Όροφος: 0

4. Φοιτητής: «Ψάχνω κάποιον καθηγητή»

```
99 def pou_kathghths(AM): #ζυνάρτηση "Ψάχνω κάποιον καθηγητή"
100     buf="SELECT id, fname, lname FROM Professor"
101     try:
102         c.execute(buf)
103         result=c.fetchall()
104         print("Εγγεγραμμένοι καθηγητές τμήματος:")
105         check_proff=[]
106         for i in range(len(result)):
107             print("ID Καθηγητή", result[i][0], "Όνοματεπώνυμο Καθηγητή:", result[i][1], result[i][2])
108             check_proff.append(result[i][0])
109             prof_id=input("Πληκτρολόγησε το ID του καθηγητή που αναζητείς:\n")
110             if prof_id not in check_proff: #Έλεγχος αν το input είναι κατάλληλο
111                 print("Παρακαλώ πληκτρολόγησε κατάλληλο κωδικό καθηγητή.")
112             else:
113                 #Έλεγχος εάν ο καθηγητής βρίσκεται αυτή τη στιγμή σε μάθημα
114                 buf="SELECT fname, lname, Auditorium.name, Auditorium.building, Auditorium.floor, Audience.end_time FROM professor_audience as pa \
115 JOIN Professor as p JOIN Audience JOIN audience_auditorium as aa JOIN Auditorium on Audience.id=aa.audience_id AND Auditorium.id=aa.auditorium_id \
116 AND pa.professor_id=p.id AND pa.audience_id=Audience.id WHERE p.id="+str(prof_id)+" AND start_time<=time('now','localtime') AND end_time>=time('now','localtime') and day=strftime('%w','now','localtime');"
117                 c.execute(buf)
118                 result=c.fetchall()
119                 if len(result)=0: #εμφάνιση της αίθουσας που βρίσκεται ο καθηγητής
120                     print("Ο Καθηγητής", result[0][0], result[0][1], "αυτή τη στιγμή βρίσκεται σε διάλεξη στην αίθουσα", result[0][2], "η οποία βρίσκεται στο κτήριο", result[0][3], "στον όροφο", result[0][4], "η/η διάλεξη που παραδίδει ολοκληρώνεται στις", result[0][5])
121                     #Ευαλλασκεα εμφάνιση γραφείου του καθηγητή
122                 else:
123                     buf="SELECT p.fname, p.lname, p.sector, p.visiting_hours, o.building, o.floor FROM Professor as p JOIN Office as o on p.id=o.owner_id WHERE p.id="+str(prof_id)+";"
124                     try:
125                         c.execute(buf)
126                         result=c.fetchall()
127                         print("Το γραφείο του καθηγητή βρίσκεται στο κτήριο", result[0][4], "στον όροφο", result[0][5])
128                     try:
129                         if len(result[0][3])=0: #Έλεγχος εάν υπάρχουν ώρες γραφείου περασμένες στη βάση
130                             print("Οι ώρες γραφείου είναι", result[0][3])
131                         except:
132                             print("Ο καθηγητής δεν έχει δηλωμένες ώρες γραφείου.")
133                     except:
134                         print("Error")
135             except:
136                 print("Error")
137     foithths(AM)
138     return
```

Επέλεξε λειτουργία:

1 --> Που έχω μάθημα τώρα;
2 --> Ψάχνω κάποιον καθηγητή
3 --> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;
4 --> Ποιο είναι το προγραμμα μου;
5 --> Εγγραφή/Απεγγραφή από μάθημα

2

Εγγεγραμμένοι καθηγητές τμήματος:

ID Καθηγητή: giannopoulos Όνοματεπώνυμο Καθηγητή: Ιωάννης Γιαννόπουλος
ID Καθηγητή: melet.raz Όνοματεπώνυμο Καθηγητή: Μελέτιος Φωτεινός Ραζόφ
ID Καθηγητή: syrou Όνοματεπώνυμο Καθηγητή: Στυλιανή Σύρου
ID Καθηγητή: rapageorgiou Όνοματεπώνυμο Καθηγητή: Ευάγγελος Παπαγεωργίου
ID Καθηγητή: anastasio Όνοματεπώνυμο Καθηγητή: Θανάσης Αναστασίου
ID Καθηγητή: andreou Όνοματεπώνυμο Καθηγητή: Ιωάννης Ανδρέου
ID Καθηγητή: apostolou Όνοματεπώνυμο Καθηγητή: Αριστοτέλης Αποστόλου
ID Καθηγητή: alexorou Όνοματεπώνυμο Καθηγητή: Φωτεινή Αλεξοπούλου
Πληκτρολόγησε το ID του καθηγητή που αναζητείς:

syrou

Ο Καθηγητής Στυλιανή Σύρου αυτή τη στιγμή βρίσκεται σε διάλεξη στην αίθουσα HΛ7 η οποία βρίσκεται στο κτήριο HMTY στον όροφο 0

Η διάλεξη που παραδίδει ολοκληρώνεται στις 23:59

Επέλεξε λειτουργία:

1 --> Που έχω μάθημα τώρα;
2 --> Ψάχνω κάποιον καθηγητή
3 --> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;
4 --> Ποιο είναι το προγραμμα μου;
5 --> Εγγραφή/Απεγγραφή από μάθημα

2

Εγγεγραμμένοι καθηγητές τμήματος:

ID Καθηγητή: giannopoulos Όνοματεπώνυμο Καθηγητή: Ιωάννης Γιαννόπουλος
ID Καθηγητή: melet.raz Όνοματεπώνυμο Καθηγητή: Μελέτιος Φωτεινός Ραζόφ
ID Καθηγητή: syrou Όνοματεπώνυμο Καθηγητή: Στυλιανή Σύρου
ID Καθηγητή: rapageorgiou Όνοματεπώνυμο Καθηγητή: Ευάγγελος Παπαγεωργίου
ID Καθηγητή: anastasio Όνοματεπώνυμο Καθηγητή: Θανάσης Αναστασίου
ID Καθηγητή: andreou Όνοματεπώνυμο Καθηγητή: Ιωάννης Ανδρέου
ID Καθηγητή: apostolou Όνοματεπώνυμο Καθηγητή: Αριστοτέλης Αποστόλου
ID Καθηγητή: alexorou Όνοματεπώνυμο Καθηγητή: Φωτεινή Αλεξοπούλου
Πληκτρολόγησε το ID του καθηγητή που αναζητείς:

giannopoulos

Το γραφείο του καθηγητή βρίσκεται στο κτήριο HMTY στον όροφο 2

Οι ώρες γραφείου είναι: Τρίτη:11:00-14:00

5. Φοιτητής: «Ποια μαθήματα επικαλύπτονται μεταξύ τους;»

```
141 def epikalisp(AM): #Συνάρτηση "Ποια μαθήματα επικαλύπτονται μεταξύ τους"
142     for k in range(0,6):
143         buf="SELECT C.name,D.name FROM (Audience A JOIN Course C on A.course_id=C.id) JOIN (Audience B JOIN Course D on B.course_id=D.id)\
144 WHERE A.start_time<=B.start_time AND A.end_time>B.start_time AND A.course_id!=B.course_id AND A.day='"+str(k)+"' AND B.day='"+str(k)+"' GROUP BY A.course_id AND B.course_id;"
145     try:
146         c.execute(buf)
147         result=c.fetchall()
148         for m in range(len(result)):
149             print("Το μάθημα", result[m][0], "επικαλύπτεται με το μάθημα", result[m][1])
150     except:
151         print("Error")
152     foithths(AM)
153     return
```

Επέλεξε λειτουργία:

1 --> Που έχω μάθημα τώρα;

2 --> Ψάχνω κάποιον καθηγητή

3 --> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;

4 --> Ποιο είναι το προγράμμα μου;

5 --> Εγγραφή/Απεγγραφή από μάθημα

3

Το μάθημα ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ επικαλύπτεται με το μάθημα ΟΛΟΚΛΗΡΩΜΕΝΑ ΗΛΕΚΤΡΟΝΙΚΑ

Το μάθημα ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ επικαλύπτεται με το μάθημα ΣΥΓΧΡΟΝΗ ΦΥΣΙΚΗ

Το μάθημα ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ επικαλύπτεται με το μάθημα ΥΨΗΛΕΣ ΤΑΞΕΙΣ

6. Φοιτητής: «Εμφάνιση Προγράμματος»

```

156 def programme(AM): #Ενάρτηση εμφάνισης προγράμματος φοιτητή
157     days=["Κυριακή", "Δευτέρα", "Τρίτη", "Τετάρτη", "Πέμπτη", "Παρασκευή", "Σάββατο"]
158     k=range(1)
159     if init(k) not in range(0,8):
160         print("Παρακαλώ επέλεξε μία από τις παραπάνω επιλογές. (0-7)")
161     else:
162         if k=="7": #Αν ο Φοιτητής θέλει να δει το πρόγραμμά του για όλες τις μέρες
163             for m in range(0,6):
164                 buf="select c.name, a.id, au.name, au.building, au.floor, start_time, end_time from Audience as a, Course as c, student_audience as sa,
165                 audience_auditorium as aa, Auditorium as au where ( (sa.audience_id = a.id and c.id = a.course_id and aa.audience_id=a.id and au.id=aa.auditorium_id) and day="+str(m)+" and sa.student_AM = '"+str(AM)+"');"
166                 try:
167                     c.execute(buf)
168                     result=c.fetchall()
169                     if len(result)==0:
170                         print("Την", days[m], "δεν έχεις μάθημα.")
171                     else:
172                         print("Την", days[m], "έχεις")
173                         for l in range(len(result)):
174                             print(result[l][0], "από τις", result[l][5], "μέχρι τις", result[l][6], "στην αίθουσα", result[l][2], "η οποία βρίσκεται στο κτήριο", result[l][3], "στον όροφο", result[l][4], "η")
175                 except:
176                     print("Error")
177             #Διαφορετική περίπτωση για το Σάββατο για την χρήση σωστού άρθρου
178             buf="select c.name, a.id, au.name, au.building, au.floor, start_time, end_time from Audience as a, Course as c, student_audience as sa, audience_auditorium as aa,
179             Auditorium as au where ( (sa.audience_id = a.id and c.id = a.course_id and aa.audience_id=a.id and au.id=aa.auditorium_id) and day="+str(m)+" and sa.student_AM = '"+str(AM)+"');"
180             try:
181                 c.execute(buf)
182                 result=c.fetchall()
183                 if len(result)==0:
184                     print("Το", days[6], "δεν έχεις μάθημα.")
185                 else:
186                     print("Το", days[6], "έχεις")
187                     for l in range(len(result)):
188                         print(result[l][0], "από τις", result[l][5], "μέχρι τις", result[l][6], "στην αίθουσα", result[l][2], "η οποία βρίσκεται στο κτήριο", result[l][3], "στον όροφο", result[l][4], "η")
189                 except:
190                     print("Error")
191
192     else: #Αν ο φοιτητής θέλει να δει το πρόγραμμα για μια συγκεκριμένη ημέρα
193         buf="select c.name, a.id, au.name, au.building, au.floor, start_time, end_time from Audience as a, Course as c, student_audience as sa, audience_auditorium as aa,
194         Auditorium as au where ( (sa.audience_id = a.id and c.id = a.course_id and aa.audience_id=a.id and au.id=aa.auditorium_id) and day="+str(k)+" and sa.student_AM = '"+str(AM)+"');"
195         try:
196             c.execute(buf)
197             result=c.fetchall()
198             if len(result)!=0:
199                 if k=="6":
200                     print("Το Σάββατο έχεις:")
201                 else:
202                     print("Την", days[init(k)], "έχεις:")
203                     for l in range(len(result)):
204                         print(result[l][0], "από τις", result[l][5], "μέχρι τις", result[l][6], "στην αίθουσα", result[l][2], "η οποία βρίσκεται στο κτήριο", result[l][3], "στον όροφο", result[l][4], "η")
205                 else:
206                     print("Δεν έχεις μάθημα αυτή την μέρα.")
207             except:
208                 print("Error")
209         foithths(AM)
210         return

```

Επέλεξε λειτουργία:
1 --> Που έχω μάθημα τώρα;
2 --> Ψάχνω κάποιον καθηγητή
3 --> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;
4 --> Ποιο είναι το πρόγραμμά μου;
5 --> Εγγραφή/Απεγγραφή από μάθημα
4

Παρακαλώ επέλεξε για ποια ημέρα θα ήθελες να δεις το προγράμμά σου:
0 --> Κυριακή
1 --> Δευτέρα
2 --> Τρίτη
3 --> Τετάρτη
4 --> Πέμπτη
5 --> Παρασκευή
6-->Σάββατο
7--> Όλες τις μέρες
7

Την Κυριακή δεν έχεις μάθημα.
Την Δευτέρα δεν έχεις μάθημα.
Την Τρίτη έχεις:
ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ από τις 09:00 μέχρι τις 11:00 στην αίθουσα ΚΥΠΕΣΙ η οποία βρίσκεται στο κτήριο HMTY στον όροφο 0

Την Τετάρτη έχεις:
ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ από τις 09:00 μέχρι τις 12:00 στην αίθουσα ΗΛ7 η οποία βρίσκεται στο κτήριο HMTY στον όροφο 0

Την Πέμπτη έχεις:
ΥΨΗΛΕΣ ΤΑΞΕΙΣ από τις 12:00 μέχρι τις 14:00 στην αίθουσα ΗΛ3 η οποία βρίσκεται στο κτήριο HMTY2 στον όροφο 0

ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ από τις 11:00 μέχρι τις 14:00 στην αίθουσα ΗΛ7 η οποία βρίσκεται στο κτήριο HMTY στον όροφο 0

ΨΗΦΙΑΚΗ ΛΟΓΙΚΗ από τις 09:00 μέχρι τις 11:00 στην αίθουσα ΒΑ η οποία βρίσκεται στο κτήριο CEID στον όροφο 0

Την Παρασκευή δεν έχεις μάθημα.
Το Σάββατο δεν έχεις μάθημα.

7. Φοιτητής: «Εγγραφή σε ακροατήριο»

```
211 def eggrafi(AM): Επισύσταση εγγραφής/απεγγραφής από ακροατήριο
212 epil_foit=input("\nΕπέλεξε: \n1->Εγγραφή σε μάθημα \n2 -> Απεγγραφή από μάθημα \n")
213 if epil_foit!="1": #Εγγραφή
214     buf="SELECT Audience.id, Course.name FROM Audience JOIN Course on Audience.course_id=Course.id;"
215     try:
216         c.execute(buf)
217         result = c.fetchall()
218         math_list=[]
219         print("\n διαθέσιμα μαθήματα προς εγγραφή είναι:")
220         for i in range(len(result)):
221             print(result[i])
222             math_list.append(result[i][0])
223         mathima=input(" Πληκτρολόγησε τον κωδικό ακροατηρίου που θα ήθελες να εγγραφείς \n")
224         if mathima not in math_list: #Έλεγχος αν το input είναι κατάλληλο
225             print("Παρακαλώ πληκτρολόγησε κατάλληλο κωδικό μαθήματος. ")
226         else:
227             buf="SELECT Audience.id, Course.name, student_audience.student_AM FROM Course JOIN Audience JOIN student_audience on Course.id=Audience.course_id \
228 AND Audience.id=student_audience.audience_id WHERE audience_id='"+str(mathima)+"' AND student_AM='"+str(AM)+"';"
229             c.execute(buf)
230             result = c.fetchall()
231             if len(result)!=1: #Έλεγχος αν ο φοιτητής είναι ήδη γραμμένος στο μάθημα
232                 buf="insert into student_audience (audience_id, student_AM) SELECT '"+str(mathima)+"', '"+str(AM)+"' WHERE not exists(SELECT * FROM student_audience WHERE audience_id='"+str(mathima)+"' AND student_AM='"+str(AM)+"');"
233                 c.execute(buf)
234             buf="SELECT Audience.id, Course.name FROM Audience JOIN Course on Audience.course_id=Course.id WHERE audience_id='"+str(mathima)+"';"
235             c.execute(buf)
236             conn.commit()
237             result = c.fetchall()
238             print("Μόλις γράφτηκες στο ακροατήριο", result[0][0], "του μαθήματος", result[0][1])
239         else:
240             print("Είσαι ήδη γραμμένος στο ακροατήριο", result[0][0], "του μαθήματος", result[0][1])
241     except:
242         print("Error")
243
```

Επέλεξε:

1->Εγγραφή σε μάθημα
2 -> Απεγγραφή από μάθημα

1

Τα διαθέσιμα μαθήματα προς εγγραφή είναι:

('ΒΔ1', 'ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ')
('ΒΔ2', 'ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ')
('ΒΔ3', 'ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ')
('ΣΦ1', 'ΣΥΓΧΡΟΝΗ ΦΥΣΙΚΗ')
('ΣΦ2', 'ΣΥΓΧΡΟΝΗ ΦΥΣΙΚΗ')
('ΨΛ', 'ΨΗΦΙΑΚΗ ΛΟΓΙΚΗ')
('ΥΤ', 'ΥΨΗΛΕΣ ΤΑΣΕΙΣ')
('ΤΝ', 'ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ')
('ΔΙ', 'ΑΝΑΛΥΣΗ ΔΙΚΤΥΩΝ ΙΣΧΥΟΣ')
('ΔΤ1', 'ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ')
('ΔΤ2', 'ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ')
('ΟΗ', 'ΟΛΟΚΛΗΡΩΜΕΝΑ ΗΛΕΚΤΡΟΝΙΚΑ')

Πληκτρολόγησε τον κωδικό ακροατηρίου που θα ήθελες να εγγραφείς:

ΟΗ

Μόλις γράφτηκες στο ακροατήριο ΟΗ του μαθήματος ΟΛΟΚΛΗΡΩΜΕΝΑ ΗΛΕΚΤΡΟΝΙΚΑ

Επέλεξε λειτουργία:

1 -> Που έχω μάθημα τώρα;
2 -> Ψάχνω κάποιον καθηγητή
3 -> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;
4 -> Ποιο είναι το προγραμματά μου;
5 -> Εγγραφή/Απεγγραφή από μάθημα
5

Επέλεξε:

1->Εγγραφή σε μάθημα
2 -> Απεγγραφή από μάθημα

1

Τα διαθέσιμα μαθήματα προς εγγραφή είναι:

('ΒΔ1', 'ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ')
('ΒΔ2', 'ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ')
('ΒΔ3', 'ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ')
('ΣΦ1', 'ΣΥΓΧΡΟΝΗ ΦΥΣΙΚΗ')
('ΣΦ2', 'ΣΥΓΧΡΟΝΗ ΦΥΣΙΚΗ')
('ΨΛ', 'ΨΗΦΙΑΚΗ ΛΟΓΙΚΗ')
('ΥΤ', 'ΥΨΗΛΕΣ ΤΑΣΕΙΣ')
('ΤΝ', 'ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ')
('ΔΙ', 'ΑΝΑΛΥΣΗ ΔΙΚΤΥΩΝ ΙΣΧΥΟΣ')
('ΔΤ1', 'ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ')
('ΔΤ2', 'ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ')
('ΟΗ', 'ΟΛΟΚΛΗΡΩΜΕΝΑ ΗΛΕΚΤΡΟΝΙΚΑ')

Πληκτρολόγησε τον κωδικό ακροατηρίου που θα ήθελες να εγγραφείς:

ΟΗ

Είσαι ήδη γραμμένος στο ακροατήριο ΟΗ του μαθήματος ΟΛΟΚΛΗΡΩΜΕΝΑ ΗΛΕΚΤΡΟΝΙΚΑ

8. Φοιτητής: «Απεγγραφή από ακροατήριο»

```
if epil_foith=="2": #Απεγγραφή
    try:
        buf="SELECT audience_id, name FROM student_audience JOIN Audience JOIN course ON student_audience.audience_id=Audience.id AND Audience.course_id=Course.id WHERE student_AM="+str(AM)+";"
        c.execute(buf)
        result = c.fetchall()
        math_list=[]
        print("Είσαι εγγεγραμμένος στα παρακάτω ακροατήρια: \n")
        for i in range(len((result))):
            print(result[i])
            math_list.append(result[i][0])
        mathima_id=input("Πληκτρολόγησε τον κωδικό ακροατηρίου από το οποίο θα ήθελες να απεγγραφείς: \n")
        if mathima_id not in math_list: #Έλεγχος αν το input είναι κατάλληλο
            print("Παρακαλώ πληκτρολόγησε κάποιον από τους παραπάνω κωδικούς μαθήματος.")
        else:
            buf="delete from student_audience where student_AM = '"+str(AM)+"' and audience_id='"+str(mathima_id)+"';"
            c.execute(buf)
            c.execute(buf)
            conn.commit()
            print("Η διαδικασία ολοκληρώθηκε. \n")
            buf="SELECT audience_id, name FROM student_audience JOIN Audience JOIN course ON student_audience.audience_id=Audience.id AND Audience.course_id=Course.id WHERE student_AM="+str(AM)+";"
            c.execute(buf)
            result = c.fetchall()
            print("Είσαι εγγεγραμμένος στα παρακάτω ακροατήρια: \n")
            for i in range(len((result))):
                print(result[i])
        except:
            print("Error")
    foithths(AM)
    return
```

Επέλεξε λειτουργία:

```
1 --> Που έχω μάθημα τώρα;
2 --> Ψάχνω κάποιον καθηγητή
3 --> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;
4 --> Ποιο είναι το προγραμμα μου;
5 --> Εγγραφή/Απεγγραφή από μάθημα
5
```

Επέλεξε:

```
1-->Εγγραφή σε μάθημα
2 --> Απεγγραφή από μάθημα
2
```

Είσαι εγγεγραμμένος στα παρακάτω ακροατήρια:

```
('BΔ1', 'ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ')
('ΔΤ1', 'ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ')
('ΨΛ', 'ΨΗΦΙΑΚΗ ΛΟΓΙΚΗ')
('BΔ3', 'ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ')
('ΟΗ', 'ΟΛΟΚΛΗΡΩΜΕΝΑ ΗΛΕΚΤΡΟΝΙΚΑ')
Πληκτρολόγησε τον κωδικό ακροατηρίου από το οποίο θα ήθελες να απεγγραφείς:
ΟΗ
Η διαδικασία ολοκληρώθηκε.
```

Είσαι εγγεγραμμένος στα παρακάτω ακροατήρια:

```
('BΔ1', 'ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ')
('ΔΤ1', 'ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ')
('ΨΛ', 'ΨΗΦΙΑΚΗ ΛΟΓΙΚΗ')
('BΔ3', 'ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ')
```

Επέλεξε λειτουργία:

```
1 --> Που έχω μάθημα τώρα;
2 --> Ψάχνω κάποιον καθηγητή
3 --> Ποιά μαθήματα επικαλύπτονται μεταξύ τους;
4 --> Ποιο είναι το προγραμμα μου;
5 --> Εγγραφή/Απεγγραφή από μάθημα
```

9. Login & Main screen Καθηγητή

```
2
3 def login_ka8ghghths(): #Login Καθηγητή
4     print("Επιλέξτε είσοδο Καθηγητή")
5
6     username=input("Παρακαλώ πληκτρολόγησε το username σου: \n")
7     buf="SELECT * FROM Professor WHERE username='"+username+"';"
8     try:
9         c.execute(buf)
10        result = c.fetchall()
11        if(len(result)!=1):
12            print("Το όνομα χρήστη που πληκτρολογήσατε δεν είναι σωστό. Παρακαλώ δοκίμασε ξανά.")
13            main()
14            return
15        id=result[0][9]
16        swsto_password=result[0][10]
17        password=input("Παρακαλώ πληκτρολόγησε τον κωδικό σου: \n")
18        if password==swsto_password:
19            print("Επιτυχής σύνδεση! \nΚαλώς ήρθες", result[0][1])
20            ka8ghghths(username)
21        else:
22            print("Ο κωδικός που πληκτρολογήσατε είναι λανθασμένος. Παρακαλώ δοκίμασε ξανά.\n")
23            main()
24            conn.commit()
25    except:
26        print("Error")
27        return
28
29 #Συναρτήσεις Καθηγητή
30
31 def ka8ghghths(username):
32     #Επιλογή λειτουργίας
33     epilogh_ka8ghghth=input("\nΕπιλέξτε λειτουργία: \n1 --> Εύρεση διαθεσιμότητας φοιτητών \n2 --> Εύρεση διαθέσιμης αίθουσας \n3 --> Αλλαγή διδάσκοντα ακροατηρίου \n")
34     while True:
35         if epilogh_ka8ghghth=="1":
36             dia8esimothta_foithwn(username)
37         elif epilogh_ka8ghghth=="2":
38             dia8esimothta_aithousas(username)
39         elif epilogh_ka8ghghth=="3":
40             allagh_didaskonta(username)
41         elif epilogh_ka8ghghth==" ":
42             print("Επιστροφή στην αρχική σελίδα.")
43             main()
44         else:
45             print("Παρακαλώ δώστε κατάλληλο input\n")
46             ka8ghghths(username)
47     break
```

Καλώς ήρθατε!

Επιλογή εισόδου:

1 --> Είσοδος Φοιτητή

2 --> Είσοδος Καθηγητή

2

Επιλέξατε είσοδο Καθηγητή

Παρακαλώ πληκτρολόγησε το username σου:

giannopoulos

Παρακαλώ πληκτρολόγησε τον κωδικό σου:

gian1234

Επιτυχής σύνδεση!

Καλώς ήρθες Ιωάννης

Επιλέξτε λειτουργία:

1 --> Εύρεση διαθεσιμότητας φοιτητών

2 --> Εύρεση διαθέσιμης αίθουσας

3 --> Αλλαγή διδάσκοντα ακροατηρίου

|

10. Καθηγητής: «Εύρεση διαθεσιμότητας φοιτητών»

```
293 def dialismotha_faithou(username):
294     buf="SELECT Audience.id, Course.name, Professor.id, Professor.name FROM Audience JOIN Course on Audience.course_id=Course.id \
295     JOIN professor_audience on professor_audience.audience_id=Audience.id JOIN Professor on Professor.id=professor_audience.professor_id WHERE Professor.id='"+str(username)+"'"
296     try:
297         c.execute(buf)
298         result = c.fetchall()
299         print("Είστε διδάσκων στα ακροατήρια:")
300         check_akroat=[]
301         for i in range(len(result)):
302             print(result[i][0], "του μαθήματος", result[i][1])
303             check_akroat.append(result[i][0])
304             akroat=input("Πληκτρολογείτε τον κωδικό ακροατηρίου για το οποίο θα θέλατε να κάνετε αναζήτηση: ")
305             if akroat not in check_akroat:
306                 print("Παρακαλώ επιλέξτε έναν κατάλληλο κωδικό ακροατηρίου.")
307             else:
308                 d=input("Θα ήθελα να δω τη διαθεσιμότητα του ακροατηρίου την μέρα: ")
309                 if int(d) not in range(0,7):
310                     print("Παρακαλώ επιλέξτε κατάλληλη ημέρα.")
311                 else:
312                     a=input("Από τις: (Σε μορφή 0-24) ")
313                     b=input("Μέχρι τις: (Σε μορφή 0-24) ")
314                     if (a not in range(0,25)) or (b not in range(0,25)):
315                         print("Παρακαλώ πληκτρολογείτε κατάλληλο αριθμό ωρών.")
316                     else:
317                         buf="select count(student_AM) from student_audience where audience_id='"+str(akroat)+"' and student_AM in (select student_AM from student_audience \
318                         where student_audience.audience_id = Audience.id except select student_AM from student_audience where student_audience.audience_id = Audience.id and day='"+str(d)+"' \
319                         and ((strftime('%H',start_time)<='"+str(a)+"' AND strftime('%H',start_time)<='"+str(b)+"') OR (strftime('%H',end_time)>='"+str(a)+"' AND strftime('%H',end_time)<='"+str(b)+"')) OR (strftime('%H',start_time)<='"+str(a)+"' AND strftime('%H',end_time)<='"+str(b)+"'))]"
320                         c.execute(buf)
321                         result = c.fetchall()
322                         print("Τις ώρες που επιλέξατε από το ακροατήριο", akroat, "είναι διαθέσιμοι", result[0][0], "φοιτητές.")
323             except:
324                 print("Error")
325             ka8hghthi(username)
326     return
327
```

Επιλέξτε λειτουργία:

1 --> Εύρεση διαθεσιμότητας φοιτητών

2 --> Εύρεση διαθέσιμης αίθουσας

3 --> Αλλαγή διδάσκοντα ακροατηρίου

1

Είστε διδάσκων στα ακροατήρια:

ΥΤ του μαθήματος ΥΨΗΛΕΣ ΤΑΞΕΙΣ

ΔΙ του μαθήματος ΑΝΑΛΥΣΗ ΔΙΚΤΥΩΝ ΙΣΧΥΟΣ

Πληκτρολογήστε τον κωδικό ακροατηρίου για το οποίο θα θέλατε να κάνετε αναζήτηση:

ΥΤ

Θα ήθελα να δω τη διαθεσιμότητα του ακροατηρίου την μέρα:

0 --> Κυριακή

1 --> Δευτέρα

2 --> Τρίτη

3 --> Τετάρτη

4 --> Πέμπτη

5 --> Παρασκευή

6-->Σάββατο

2

Από τις: (Σε μορφή 0-24)

10

Μέχρι τις: (Σε μορφή 0-24)

12

Τις ώρες που επιλέξατε από το ακροατήριο ΥΤ είναι διαθέσιμοι 2 φοιτητές.

11. Καθηγητής: «Εύρεση διαθέσιμης αίθουσας»

```
329
330 def diasimesmtha_aithousas(username): #Αναζήτηση διαθέσιμων αίθουσών για συγκεκριμένες ώρες τις βδομάδας
331     d=int(input("Παρακαλώ επιλέξτε ποια ημέρα θα θέλατε να αναζητήσετε διαθέσιμη αίθουσα: \n0 --> Κυριακή \n1 --> Δευτέρα \n2 --> Τρίτη \n3 --> Τετάρτη \n4 --> Πέμπτη \n5 --> Παρασκευή \n6-->Σάββατο \n"))
332     if d not in range(0,7):
333         print("Παρακαλώ πληκτρολογήστε κατάλληλο αριθμό ημέρας.")
334     a=int(input("Θα ήθελα διαθέσιμη αίθουσα από τις: (Σε μορφή 0-24) \n"))
335     b=int(input("Μέχρι τις: (Σε μορφή 0-24) \n"))
336     if (a not in range(0,25)) or (b not in range(0,25)):
337         print("Παρακαλώ πληκτρολογήστε κατάλληλο αριθμό ωρών.")
338     else:
339         buf="SELECT * FROM Auditorium WHERE id NOT IN(SELECT Auditorium.id FROM Auditorium JOIN audience_auditorium on auditorium_id=Auditorium.id \
340 JOIN Audience on audience_id=Audience.id WHERE Audience.day="+str(d)+" AND ((strftime('%H',start_time)<="+str(a)+" AND strftime('%H',start_time)<="+str(d)+" ) \
341 OR (strftime('%H',end_time)>="+str(a)+" AND strftime('%H',end_time)<="+str(b)+" )) OR (strftime('%H',start_time)<="+str(a)+" AND strftime('%H',end_time)>="+str(b)+" ));"
342         try:
343             c.execute(buf)
344             result = c.fetchall()
345             print("Οι διαθέσιμες αίθουσες εκείνες τις ώρες είναι:")
346             for i in range(len(result)):
347                 print("Αίθουσα", result[i][1], "στο κτήριο", result[i][2], "στον όροφο", result[i][3])
348         except:
349             print("Error")
350     ka8ghths(username)
351     return
```

Επιλέξτε λειτουργία:

1 --> Εύρεση διαθεσιμότητας φοιτητών

2 --> Εύρεση διαθέσιμης αίθουσας

3 --> Αλλαγή διδάσκοντα ακροατηρίου

2

Παρακαλώ επιλέξτε ποια ημέρα θα θέλατε να αναζητήσετε διαθέσιμη αίθουσα:

0 --> Κυριακή

1 --> Δευτέρα

2 --> Τρίτη

3 --> Τετάρτη

4 --> Πέμπτη

5 --> Παρασκευή

6-->Σάββατο

3

Θα ήθελα διαθέσιμη αίθουσα από τις: (Σε μορφή 0-24)

11

Μέχρι τις: (Σε μορφή 0-24)

13

Οι διαθέσιμες αίθουσες εκείνες τις ώρες είναι:

Αίθουσα ΒΑ στο κτήριο CEID στον όροφο 0

Αίθουσα ΑΠΜ4 στο κτήριο Πολιτικών Μηχανικών στον όροφο 1

Αίθουσα ΚΥΠΕΣII στο κτήριο HMTY στον όροφο 0

Αίθουσα ΑΠ2 στο κτήριο Αμφιθέατρα Πολυτεχνικών Σχολών στον όροφο 2

Αίθουσα ΑΒ στο κτήριο Α' Πρυτανείας στον όροφο 0

Αίθουσα ΗΛ3 στο κτήριο HMTY2 στον όροφο 0

Αίθουσα ΚΥΠΕΣI στο κτήριο HMTY στον όροφο 0

12. Καθηγητής: «Αλλαγή διδάσκοντα σε ακροατήριο»

```
355 def allagh_didaskonta(username): #Αλλαγή/Update του διδάσκοντα ενός ακροατηρίου
356     buf= 'SELECT Audience.id, Course.name, Professor.id, Professor.fname, Professor.lname FROM Audience JOIN Course on Audience.course_id=Course.id \
357 JOIN professor_audience on professor_audience.audience_id=Audience.id JOIN Professor on Professor.id=professor_audience.professor_id;"
358     try:
359         c.execute(buf)
360         result = c.fetchall()
361         print("Αυτή τη στιγμή διδάσκοντες στα ακροατήρια είναι οι:")
362         check_akroat=[]
363         check_proff=[]
364         for i in range(len(result)):
365             print(result[i][3], result[i][4], "με κωδικό ID", result[i][2], "στο ακροατήριο", result[i][0], "του μαθήματος", result[i][1])
366             check_akroat.append(result[i][0])
367             check_proff.append(result[i][2])
368         akroat=input("\nΕπιλέξτε σε ποιο ακροατήριο θα θέλατε να αλλάξετε τον διδάσκοντα. Πληκτρολογήστε τον κωδικό του ακροατηρίου. \n")
369         if akroat not in check_akroat: #Έλεγχος αν το input είναι κατάλληλο
370             print("Παρακαλώ πληκτρολογήστε κατάλληλο κωδικό ακροατηρίου.")
371         else:
372             new_proff=input("Πληκτρολογήστε τον κωδικό ID του διδάσκοντα που θα διδάσκει πλέον αυτό το ακροατήριο: \n")
373             if new_proff not in check_proff: #Έλεγχος αν το input είναι κατάλληλο
374                 print("Παρακαλώ πληκτρολογήστε κατάλληλο κωδικό διδάσκοντα.")
375             else:
376                 buf='SELECT professor_id FROM professor_audience WHERE audience_id="'+str(akroat)+'";'
377                 c.execute(buf)
378                 result = c.fetchall()
379                 if result[0][0]==str(new_proff):
380                     print("Ο καθηγητής αυτός είναι ήδη υπεύθυνος για το ακροατήριο", akroat)
381                 else:
382                     buf="UPDATE professor_audience SET professor_id='"+str(new_proff)+"' WHERE audience_id='"+str(akroat)+"';"
383                     c.execute(buf)
384                     conn.commit()
385                     print("Η διαδικασία ολοκληρώθηκε. \n")
386                     buf= 'SELECT Audience.id, Course.name, Professor.id, Professor.fname, Professor.lname FROM Audience JOIN Course on Audience.course_id=Course.id \
387 JOIN professor_audience on professor_audience.audience_id=Audience.id JOIN Professor on Professor.id=professor_audience.professor_id WHERE Audience.id="'+str(akroat)+'";'
388                     c.execute(buf)
389                     result = c.fetchall()
390                     print("Το ακροατήριο", result[0][0], "έχει νέο διδάσκοντα τον", result[0][3], result[0][4])
391     except:
392         print("Error")
393     ka8hghths(username)
394     return
395
```

3 --> Αλλαγή διδάσκοντα ακροατηρίου

3

Αυτή τη στιγμή διδάσκοντες στα ακροατήρια είναι οι:

Ιωάννης Γιαννόπουλος με κωδικό ID giannopoulos στο ακροατήριο YT του μαθήματος ΥΨΗΛΕΣ ΤΑΣΕΙΣ

Στυλιανή Σύρου με κωδικό ID syrou στο ακροατήριο BΔ2 του μαθήματος ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Μελέτιος Φωτεινός Ραζόφ με κωδικό ID melet.raz στο ακροατήριο BΔ1 του μαθήματος ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Ιωάννης Ανδρέου με κωδικό ID andreu στο ακροατήριο ΔΤ1 του μαθήματος ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ

Στυλιανή Σύρου με κωδικό ID syrou στο ακροατήριο BΔ3 του μαθήματος ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Αριστοτέλης Αποστόλου με κωδικό ID apostolou στο ακροατήριο ΣΦ1 του μαθήματος ΣΥΓΧΡΟΝΗ ΦΥΣΙΚΗ

Αριστοτέλης Αποστόλου με κωδικό ID apostolou στο ακροατήριο ΣΦ2 του μαθήματος ΣΥΓΧΡΟΝΗ ΦΥΣΙΚΗ

Θανάσης Αναστασίου με κωδικό ID anastasiou στο ακροατήριο ΨΛ του μαθήματος ΨΗΦΙΑΚΗ ΛΟΓΙΚΗ

Ευάγγελος Παπαγεωργίου με κωδικό ID papageorgiou στο ακροατήριο ΤΝ του μαθήματος ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ

Ιωάννης Γιαννόπουλος με κωδικό ID giannopoulos στο ακροατήριο ΔΙ του μαθήματος ΑΝΑΛΥΣΗ ΔΙΚΤΥΩΝ ΙΣΧΥΟΣ

Μελέτιος Φωτεινός Ραζόφ με κωδικό ID melet.raz στο ακροατήριο ΔΤ2 του μαθήματος ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ

Φωτεινή Αλεξοπούλου με κωδικό ID alexorouli στο ακροατήριο ΟΗ του μαθήματος ΟΛΟΚΛΗΡΩΜΕΝΑ ΗΛΕΚΤΡΟΝΙΚΑ

Επιλέξτε σε ποιο ακροατήριο θα θέλατε να αλλάξετε τον διδάσκοντα. Πληκτρολογήστε τον κωδικό του ακροατηρίου.

BΔ3

Πληκτρολογήστε τον κωδικό ID του διδάσκοντα που θα διδάσκει πλέον αυτό το ακροατήριο:

melet.raz

Η διαδικασία ολοκληρώθηκε.

Το ακροατήριο BΔ3 έχει νέο διδάσκοντα τον Μελέτιος Φωτεινός Ραζόφ

Επιλέξτε λειτουργία:

1 --> Εύρεση διαθεσιμότητας φοιτητών

2 --> Εύρεση διαθέσιμης αίθουσας

3 --> Αλλαγή διδάσκοντα ακροατηρίου

3

Αυτή τη στιγμή διδάσκοντες στα ακροατήρια είναι οι:

Ιωάννης Γιαννόπουλος με κωδικό ID giannopoulos στο ακροατήριο YT του μαθήματος ΥΨΗΛΕΣ ΤΑΣΕΙΣ

Στυλιανή Σύρου με κωδικό ID syrou στο ακροατήριο BΔ2 του μαθήματος ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Μελέτιος Φωτεινός Ραζόφ με κωδικό ID melet.raz στο ακροατήριο BΔ1 του μαθήματος ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Ιωάννης Ανδρέου με κωδικό ID andreu στο ακροατήριο ΔΤ1 του μαθήματος ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ

Μελέτιος Φωτεινός Ραζόφ με κωδικό ID melet.raz στο ακροατήριο BΔ3 του μαθήματος ΒΑΣΕΙΣ ΔΕΔΟΜΕΝΩΝ

Αριστοτέλης Αποστόλου με κωδικό ID apostolou στο ακροατήριο ΣΦ1 του μαθήματος ΣΥΓΧΡΟΝΗ ΦΥΣΙΚΗ

Αριστοτέλης Αποστόλου με κωδικό ID apostolou στο ακροατήριο ΣΦ2 του μαθήματος ΣΥΓΧΡΟΝΗ ΦΥΣΙΚΗ

Θανάσης Αναστασίου με κωδικό ID anastasiou στο ακροατήριο ΨΛ του μαθήματος ΨΗΦΙΑΚΗ ΛΟΓΙΚΗ

Ευάγγελος Παπαγεωργίου με κωδικό ID papageorgiou στο ακροατήριο ΤΝ του μαθήματος ΤΕΧΝΗΤΗ ΝΟΗΜΟΣΥΝΗ

Ιωάννης Γιαννόπουλος με κωδικό ID giannopoulos στο ακροατήριο ΔΙ του μαθήματος ΑΝΑΛΥΣΗ ΔΙΚΤΥΩΝ ΙΣΧΥΟΣ

Μελέτιος Φωτεινός Ραζόφ με κωδικό ID melet.raz στο ακροατήριο ΔΤ2 του μαθήματος ΔΙΑΔΡΑΣΤΙΚΕΣ ΤΕΧΝΟΛΟΓΙΕΣ

Φωτεινή Αλεξοπούλου με κωδικό ID alexorouli στο ακροατήριο ΟΗ του μαθήματος ΟΛΟΚΛΗΡΩΜΕΝΑ ΗΛΕΚΤΡΟΝΙΚΑ

Επιλέξτε σε ποιο ακροατήριο θα θέλατε να αλλάξετε τον διδάσκοντα. Πληκτρολογήστε τον κωδικό του ακροατηρίου.

BΔ3

Πληκτρολογήστε τον κωδικό ID του διδάσκοντα που θα διδάσκει πλέον αυτό το ακροατήριο:

melet.raz

Ο καθηγητής αυτός είναι ήδη υπεύθυνος για το ακροατήριο BΔ3