

ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ
ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ
ΤΜΗΜΑ ΠΛΗΡΟΦΟΡΙΚΗΣ



ΕΚΠΑΙΔΕΥΤΙΚΟ ΛΟΓΙΣΜΙΚΟ

Παραδοτέο από:

Υφαντής Γιώργος,
Τουλούπης Γιάννης

A.M.:

Π18163,
Π18217

Επιβλέποντες Καθηγητές:

Μαρία Βίρβου, Ευάγγελος Σακκόπουλος

ΕΓΧΕΙΡΙΔΙΟ ΑΝΑΛΥΣΗΣ ΚΑΙ

ΣΧΕΔΙΑΣΜΟΥ ΤΗΣ ΕΦΑΡΜΟΓΗΣ

Περιεχόμενα

- Εκφώνηση
- Εισαγωγή
- Ανάλυση και επεξήγηση των σελίδων της εφαρμογής
- Ανάλυση και επεξήγηση του κώδικα για την κάθε φόρμα
- Βάση Δεδομένων
- Παρουσίαση Εκτέλεσης
- Βιβλιογραφία

Εκφώνηση

Ζητείται να γίνει ένα αλληλεπιδραστικό λογισμικό εκπαίδευσης μαθητών για ένα από τα παρακάτω πεδία διδασκαλίας:

A. Γλώσσα Προγραμματισμού C#, ή Java ή Python.

B. Την Ελληνική Γλώσσα.

Η εργασία αυτή θα περιλαμβάνει τρόπους παρουσίασης του διδακτικού υλικού με στόχο να γίνει το θέμα κατανοητό και να μπορεί να απομνημονευθεί από τους μαθητές καθώς και να εμπεδωθεί η ύλη μέσω ασκήσεων. Ο κύριος σκοπός της εργασίας είναι ο καλός σχεδιασμός και υλοποίηση του εκπαιδευτικού λογισμικού (διδασκαλία – αξιολόγηση του μαθητή) και όχι η εισαγωγή μεγάλου μέρους κεφαλαίων. Συγκεκριμένα ζητούνται τα παρακάτω:

ΒΑΣΙΚΕΣ ΛΕΙΤΟΥΡΓΙΕΣ

1. ΠΑΡΟΥΣΙΑΣΗ ΔΙΔΑΣΚΑΛΙΑΣ

2. ΕΡΩΤΗΣΕΙΣ / ΤΕΣΤ ΑΥΤΟΑΞΙΟΛΟΓΗΣΗΣ

2.1 Κατασκευή των τεστ.

- Θα πρέπει να κατασκευάζονται τεστ για την αυτοαξιολόγηση των μαθητών, για κάθε ενότητα διδασκαλίας.
- Στα επαναληπτικά τεστ, θα πρέπει να παρουσιάζονται ασκήσεις απ' όλες τις ενότητες.
- Η μορφή των ασκήσεων μπορεί να είναι πολλαπλών επιλογών ή άλλης μορφής ανάλογα με τη δική σας ανάλυση απαιτήσεων και σχεδιασμό.

2.2 Αποθήκευση στατιστικών στοιχείων προόδου του μαθητή. Θα πρέπει να υπάρχει μια βάση δεδομένων όπου να αποθηκεύονται στοιχεία για κάθε μαθητή σχετικά με την πρόοδό του. Τα στοιχεία θα βασίζονται στην απόδοση των μαθητών στα τεστ αυτοαξιολόγησης αλλά και στα στατιστικά επισκεψιμότητας της παρουσίασης του θέματος.

2.3 Διάγνωση λαθών του μαθητή και αλληλεπίδραση. Στη διάγνωση λαθών ζητείται να μπορεί το σύστημα να εντοπίσει αν ο μαθητής έχει πρόβλημα σε κάποια κατηγορία λαθών (π.χ. συντακτικά λάθη/ λάθη λογικής/ τυπογραφικά λάθη). Αν εντοπιστεί κάτι τέτοιο θα πρέπει το σύστημα να παρουσιάζει πάλι τη θεωρία και περισσότερες ερωτήσεις στο συγκεκριμένο θέμα, όπου υπάρχει πρόβλημα. Επίσης θα πρέπει να καταγράφεται η συγκεκριμένη αδυναμία στα στατιστικά προόδου του μαθητή και να σβήνεται όταν ο μαθητής φαίνεται ότι έχει πια μάθει το συγκεκριμένο θέμα.

Εισαγωγή

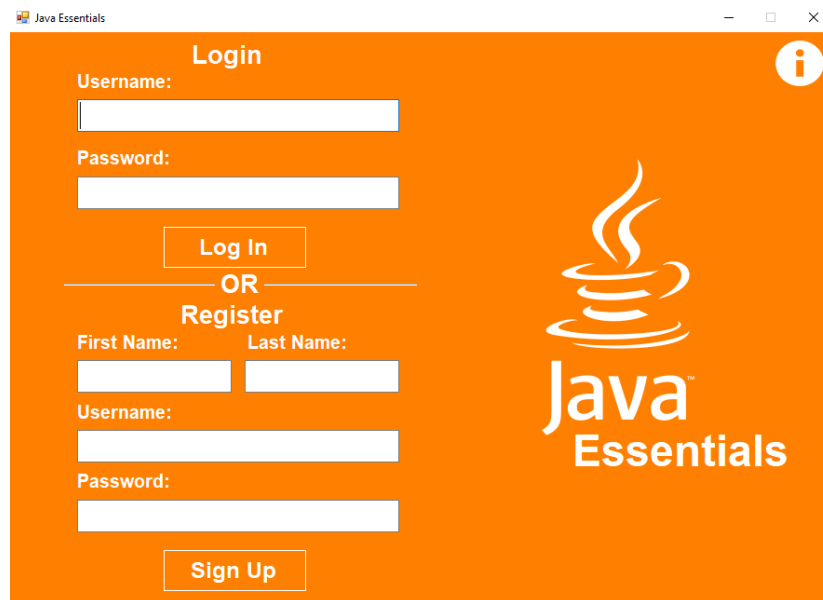
Για την υλοποίηση αυτής της εφαρμογής χρησιμοποιήσαμε το Windows Forms του Visual Studio 2019. Πιο συγκεκριμένα χρησιμοποιήσαμε την γλώσσα προγραμματισμού C#.

Για το θέμα της εργασίας αποφασίσαμε η εφαρμογή να είναι ένα αλληλεπιδραστικό λογισμικό εκπαίδευσης για την γλώσσα προγραμματισμού Java. Για να ολοκληρώσουμε αυτό τον σκοπό επρέπε να δημιουργήσουμε μια εφαρμογή μέσω της οποίας ο κάθε χρήστης (μαθητής) θα μπορεί να δημιουργήσει έναν λογαριασμό, ή να συνδεθεί σε ήδη υπάρχων, να μπορεί να μάθει διαφορές γνώσεις σχετικά με την γλώσσα Java, χωρισμένη ανα κεφάλαια καθώς και να μπορεί να ελέγξει τις γνώσεις του πάνω σε κάθε κεφάλαιο με κάποιου ήδη τεστ. Επιπλέον θα μπορεί να τεσταρει τις γενικές του γνώσεις από όλα τα κεφάλαια και να αναγνωρίσει ευκολα τα λάθη του.

Τέλος θα πρέπει να μπορεί να δει την προοδο του καθώς και ποια κεφάλαια ή διαγωνισματα του έχουν μείνει να ολοκληρωσει η να επαναλαβει.

Ανάλυση και επεξήγηση των σελίδων της εφαρμογής

Login – Register Form

The image shows a web application window titled "Java Essentials". The background is orange. On the left, there are two forms. The top form is for "Login" and contains fields for "Username:" and "Password:", followed by a "Log In" button. Below this is a horizontal line with "OR" in the center. The bottom form is for "Register" and contains fields for "First Name:", "Last Name:", "Username:", and "Password:", followed by a "Sign Up" button. On the right side of the window, there is a large white Java logo with the text "Java Essentials" below it. In the top right corner of the orange area, there is a small white circle with an "i" inside, representing an information icon.

Σε αυτή την φόρμα ο χρήστης - μαθητής μπορεί να δημιουργήσει έναν νέο λογαριασμό ή να συνδεθεί σε έναν υπάρχοντα.

Για να δημιουργηθεί ένας νέος λογαριασμός ο χρήστης θα πρέπει να συμπληρώσει όλα τα πεδία, καθώς και να επιλέξει ένα όνομα χρήστη (Username) το οποίο δεν είναι ήδη σε χρήση από κάποιον άλλον.

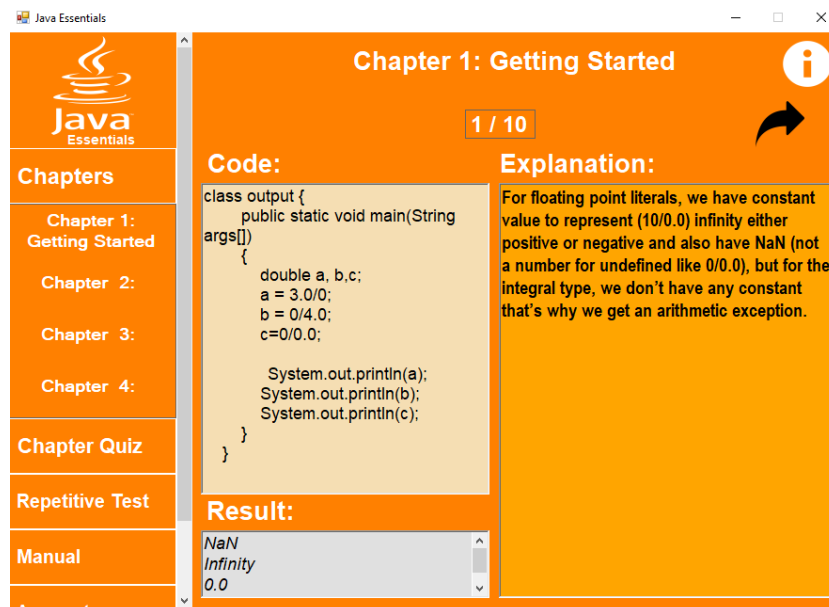
Home Form



Αυτή είναι η αρχική φόρμα της εφαρμογής, η οποία εμφανίζεται όταν συνδεεται ο χρήστης.

Από εδώ ο μαθητής μπορεί να επιλέξει τι θέλει να κάνει. Πιο συγκεκριμένα μπορεί να επιλέξει ποιο κεφάλαιο θέλει να διαβάσει, ποιανου κεφαλαίου τις ερωτήσεις να απαντήσει κλπ.

Chapter Forms



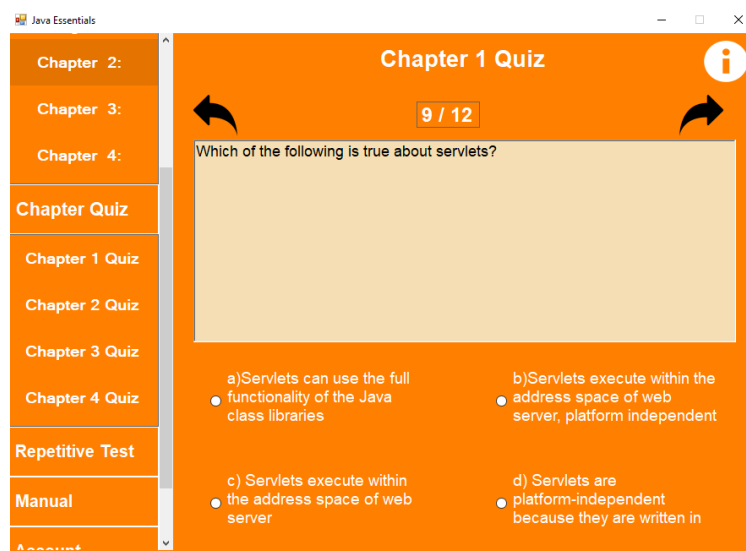
Στην παραπάνω φόρμα βλέπουμε ένα παραδειγμα από μια φόρμα θεωρίας ενός κεφαλαίου. Πιο συγκεκριμένα βλέπουμε ότι αποτελείται 3 μέρη, το πρώτο το κομμάτι στο οποίο θα υπάρχει ο κώδικας, 2 το κομμάτι που θα

φαινεται το αποτελεσμα και τελος 3 το μερος στο οποιο θα υπαρχει η επεξηγηση.

Σε αυτό το σημείο να τονιστεί ότι θα υπάρχουν και περιπτώσεις στις οποίες τα πεδία του κωδικα και του αποτελεσματος δεν θα περιεχουν κατι καθως δεν θα χρειαζονται.

Πατώντας το βελακι πανω δεξια (και αντιστοιχα αριστερα σε επομενες σελιδες) μπορούμε να μεταφερθουμε στην επομενη (η προηγουμενη) σελιδα. Κάθε φορά που προχωραμε σε μια καινουργια σελιδα αποθηκευεται και την επομενη φορά που θα ξαναανοιξουμε αυτή την φορμα μας φορτωσει απευθειας σε αυτή την σελιδα.

Quiz Forms



Εδώ βλέπουμε την φορμα που χρησιμοποιειτε για την πραγματοποιησει των διαγωνισματων. Συγκεκριμενα βλέπουμε ένα μερος στο οποιο θα υπαρχει η ερωτηση και 4 επιλογες από τις οποίες μονο 1 είναι η σωστη.

Αυτή η φορμα χρησιμοποιειτε τοσο για τα διαγωνισματα του κάθε κεφαλαιου καθως και για το επαναληπτικο, με την διαφορα ότι για το επαναληπτικο θα επιλεχθουν τυχαια 20 ερωτησεις από όλα τα κεφαλαια.

Result Forms

Java Essentials

Chapter 1 Quiz

Exit

Results:

Correct: 5 / 12

Question:

Which of these statements is incorrect about Thread?

Explanation:

run() method is used to define the code that constitutes the new thread, it contains the code to be executed. start() method is used to begin execution of the thread that is execution of run(). run() itself is never used for starting execution of the thread.

Αυτή η φόρμα εμφανίζεται αφού ολοκληρώσουμε ένα διαγωνισμό και σε αυτή βλέπουμε τα αποτελέσματα. Επιπλέον για όλες τις λανθασμένες απαντήσεις βλέπουμε την ερώτηση καθώς και την επεξήγηση για την σωστή απάντηση χωρίς να φαίνεται όμως ακριβώς ποια είναι η σωστή απάντηση ώστε ο μαθητής να μπορεί να το καταλάβει από την εξήγηση.

Εδώ να σημειωθεί ότι ο μαθητής περάσε το διαγωνισμό με επιτυχία αν έχει απαντήσει σωστά πάνω από το 60 της εκατο των ερωτήσεων.

Account Forms

Java Essentials

Account

First name: tasos

Last name: tasopoulos

Username: tasos

Progress

Chapters Completed: 4 / 4

Quizzes Completed Successfully: 2 / 4

To be completed:

Chapters	Quizzes
	Chapter 3 Quiz Chapter 4 Quiz

You have to repeat:

Chapters	Quizzes

Τέλος σε αυτή την φόρμα ο χρήστης μπορεί να δει την προοδο του. Συγκεκριμένα μπορεί να δει ποια κεφάλαια και διαγωνισματα πρέπει να

ολοκληρωσει καθως και ποια από αυτά πρεπει να επαναλαβει γιατι δεν τα πηγε αρκετα καλα στα διαγωνισματα.

Ανάλυση και επεξήγηση του κώδικα για την κάθε φόρμα

SignUpForm.cs

- **LoginBTN Click():**

Αυτή η μεθοδο καλειτε όταν ο χρηστης πατησει το κουμπι για να συνδεθει στον λογαριασμο του (login Button). Ξεκιναι με το να ελεγχει αν τα πεδια για το ονομα χρηστη και του κωδικου είναι κενα (γραμμη 50), αν δεν είναι ότι ανοιγει την συνδεση με την βαση δεδομενων και δημιουργουμε μια νεα SQL εντολη. Πιο συγκεκριμενα ζηταμε να μας επιστραφουν απο την βαση ολοι οι λογαριασμοι οπου το username είναι το ιδιο με αυτό που εχει βαλει ο χρηστης. (γραμμες 51 εως 62) Εκτελουμε την εντολη και ελεγχουμε εχουμε λαβει κατι από την βαση, αν ναι τοτε ελεγχουμε να δουμε αν ο κωδικος που εχει βαλει ο χρηστης ταιριαζει με τον αποθηκευμενο κωδικο. (γραμμες 65 με 83). Ένα όλα τα στοιχεια ταιριαζουν τοτε γινεται η συνδεση με την βοηθεια της μεθοδου userLoggedIn(), αλλιως βγαζει αντιστοιχο μηνυμα λαθους.

- **signUpBTN Click():**

Παρομοια με την προηγουμενη μεθοδο ελεγχουμε αν όλα τα απαιριτητα πεδια είναι συμπληρωμενα. Αν είναι τοτε ψαχνουμε στην βαση για να δουμε υπαρχει ηδη λογαριασμος με αυτό το ονομα χρηστη και αν υπαρχει βγαζουμε αντιστοιχο μηνυμα (γραμμες 119 – 152,190-201). Αν δεν υπαρχει ηδη λογαριασμος με αυτό το ονομα χρηστη τοτε εισαγουμε στην βαση δεδομενων τα στοιχεια του νεου λογαριασμου και συνδεουμε τον χρηστη. (γραμμες 154 με 190).


HomeForm.cs

- **load SubForm():**

Αυτή η μεθοδο χρησιμοποιειτε για να φορτωσουμε ολες τις φορμες της εφαρμογης στο πλαισιο μεσα στην αρχικη φορμα. Ξεκιναι με το να κλεινει την τρεχον ενεργη φορμα και να θετει ως νεα ενεργη την φορμα που θελουμε

να ανοιξουμε. Επειτε βεβαιωνόμαστε ότι η νεα φορμα δεν είναι στο υψηλοτερο επιπεδο καθως και ότι δεν εχει καποιο πλαισιο γυρω του. Τελος προσθετουμε στο panel την νεα φορμα και την εμφανιζουμε. (γραμμες 71 με 87)

- **helpBTN_Click()**

Αυτή καλειτε όταν ο χρηστης παταει το κουμπι για την online βοηθεια που υπαρχει σε ολες τις φορμες της εφαρμογης (). Πατωντας σε αυτό το κουμπι θα εμφανιστει ένα συντομο μηνυμα το οποιο θα δειχνει τι μπορει να κανει ο χρηστης σε αυτή την φορμα.

ChapterForm.cs

- **ChapterForm_Load()**

Ξεκινουμε παιρνοντας ολες τις σελιδες του κεφαλαιου αυτου από την βαση δεδομενων και τις βαζουμε σε μια λιστα (γραμμες 33 εως 56). Επειτα ελεγχουμε να δουμε αν υπαρχει αποθηκευμενη προοδο από τον χρηστη στο συγκεκριμενο κεφαλαιο και αν υπαρχει βαζουμε ως την σελιδα που θα αρχισει αυτή στην οποια ειχε μεινει προηγουμενως.(γραμμες 61 με 78) Τελος φορτωνουμε την σελιδα.

- **LoadLesson()**

Ελεχουμε ότι ο αριθμος των σελιδων είναι πανω από 0, αν είναι και δεν ειμαστε στην τελευταια σελιδα τοτε φορτωνει τα κειμενα για τον κωδικα, το αποτελεσμα και την επεξηγηση καθως και δειχνει και σε ποια σελιδα ειμαστε.(γραμμες 84 με 92) Επιπλεον αναλογα σε ποια σελιδα ειμαστε εμφανιζεται, ή όχι, το κουμπι που μας επιτρεπει να παμε στην προηγουμενη καθως και αλλαζει και το κουμπι για το κουμπι να παμε στην επομενη σελιδα. (Γραμμες 95 με 108) Αν βρισκομαστε στην τελευταια σελιδα τοτε απλα αποθηκευουμε σε μια παραμετρο ότι ολοκληρωθηκε το κεφαλαιο και επιστρεφουμε στο αρχικο μενου.

- **SaveProgress()**

Ξεκινουμε ελεγχοντας αν υπαρχει προηγουμενη προοδος από τον τρεχον χρηστη για αυτό το κεφαλαιο αποθηκευμενη στην βαση δεδομενων. Αν υπαρχει ελεγχουμε αν είναι μεγαλυτερη από τρεχουσα (γραμμες 133 με 169).

Ένα η προηγούμενη προοδος είναι μικροτερη από την τρεχουμε τότε ανανεωνουμε τα δεδομενα στην βάση δεδομενων (γραμμες 171 με 182) αλλιως αν δεν υπαρχει καν προηγουμενη προοδος προσθετουμε τα νεα δεδομενα στην βάση. (Γραμμες 184 με 196)

quizForm.cs

- **quizForm Load ()**

Αρχικα ελεγχουμε αν το chapterId (ο αριθμος του κεφαλαιου) ισουται με -1, το οποιο δηλωνει πως είναι επαναληπτικο τεστ, ή όχι. Αν είναι ισο με -1 τότε τραβαμε ολες τις ερωτησεις από την βάση δεδομενων και προσθετουμε την κάθε μια σε μια λιστα. (γραμμες 96 με 116). Επειτα αφαιρουμε τυχαια ερωτησεις από την λιστα μεχρι να φτασουμε να εχουμε το πολύ 20 ερωτησεις. (γραμμες 118 με 122) Τελος εμφανιζουμε την επομενη ερωτηση.

Αν δεν ισουται με -1 τότε σημαινει ότι ειμαστε στις ερωτησεις για καποιο συγκεκριμενο κεφαλαιο, επομενως τραβαμε από την βάση δεδομενων ολες τις ερωτησεις από το συγκεκριμενο κεφαλαιο και τις βαζουμε σε μια λιστα (γραμμες 131 με 157). Τελος εμφανιζουμε την επομενη ερωτηση.

- **loadQuiz ()**

Ένα ο αριθμος των ερωτησεων είναι πανω από 0 και αν δεν ειμαστε στην τελευταια ερωτηση τότε, εμφανιζουμε την ερωτηση καθως και ολες τις απαντησεις (γραμμες 173 με 179) καθως και τον αριθμο της ερωτησης. Ελεγχουμε αν ο αριθμος των απαντησεων που εχει δωσει ο χρηστης είναι μεγαλυτερο η ισως με την σελιδα στην οποια είναι ο χρηστης και αν είναι τότε φορτωνουμε την απαντηση που εχει δωσει προηγουμενως ο χρηστης (γραμμες 185 – 202). Τελος αναλογα σε ποια σελιδα ειμαστε εμφανιζουμε το κουμπι για να παμε στην προηγουμενη ερωτηση και αλλαζουμε το κουμπι για την επομενη στο κουμπι για να ολοκληρωθει το τεστ. (γραμμες 204 με 217)

ResultForm.cs

- **saveProgress ()**

Ελεγχουμε αν το διαγωνισμα ηταν επαναληπτικο η από καποιο συγκεκριμενο κεφαλαιο. Αν ηταν επαναληπτικο τότε δημιουργουμε ένα δυσδιαστατο array

με 4 γραμμες που αντιπροσωπευουν τα 4 κεφαλαια. Στην πρωτη στηλη θα αποθηκευτουν το πληθος των ερωτησεων από το κάθε κεφαλαιο και στην δευτερο το πληθος των ερωτησεων που απαντηθηκαν λαθος. (γραμμες 95 με 104) Επειτα για κάθε από τα κεφαλαια αν το πληθος των ερωτησεων από αυτό το κεφαλαιο ηταν πανω από 0, ελεγχουμε αν το ποσοστο επιτυχειας του μαθητη είναι κατω από 60 της εκατο. (γραμμες 108 με 110) Αν είναι μικροτερο από 60 τοτε ελεγχουμε αν ο χρηστης εχει δωσει στο παρελθον το διαγωνισμα για το αντιστοιχο κεφαλαιο (γραμμες 111 με 137). Στην περιπτωση που το εχει δωσει τοτε ανανεωνουμε την βαθμολογια και βαζουμε την αντιστοιχη από το επαναληπτικο κεφαλαιο ετσι ώστε ο μαθητης να πρεπει να επαναλαβει το κεφαλαιο που δεν τα πηγε καλα (γραμμες 139 με 149), αλλιως αν δεν το πραγματοποιησει τοτε προσθετουμε απλα νεα εγγραφη με τον ιδιο στοχο.

Στην περιπτωση που το διαγωνισμα δεν ηταν επαναληπτικο αλλα ηταν ενός κεφαλαιου τοτε δουλευουμε με παρομοιο τροπο με πριν απλα για το συγκεκριμενο κεφαλαιο μονο.

accountForm.cs

- **accountForm_Load ()**

Ξεκινουμε καθαριζοντας ολους του πινακες κλπ και εμφανιζοντας τα στοιχεια του μαθητη. (γραμμες 29 με 36). Επειτα επιλεγουμε από την βαση τα μαθηματα που εχει παρακολουθησει ο χρηστης και κραταμε αυτά που εχει ολοκληρωσει (γραμμες 48 με 60). Επειτα κανουμε το ιδιο και για τα διαγωνισματα των κεφαλαιων αλλα αναλογα αν η βαθμολογια του ηταν κατω από 60 της εκατο η όχι βαζουμε το κεφαλαιο και το τεστ στην λιστα αυτων που πρεπει να επαναλαβει. (γραμμες 67 με 90). Τελος φορτωνουμε όλα τα στοιχεια στις αντιστοιχες θεσεις τους (γραμμες 94 με 116).

Program.cs

- **GetConnection ()**

Επιστρεφει την συνδεση με την βαση δεδομενων. (εδώ πρεπει να αλλαξουμε το database string)

Βάση δεδομένων

Για την υλοποίηση της εργασίας χρησιμοποιήσαμε PostgreSQL. Πιο συγκεκριμένα δημιουργήσαμε 5 πίνακες οι οποίοι χρησιμοποιούνται από την εφαρμογή για διαφορετικές λειτουργίες.

Πίνακας 1: accounts

	id [PK] integer	firstname character varying (50)	lastname character varying (50)	username character varying (50)	password character varying (50)
1	1	dimitra	dimitropoulou	dimi	12345
2	2	kostas	kostopoulos	kostis	z
3	3	poulos	paulopoulos	lospav	123456
4	4	tasos	tasopoulos	tasos	a

Ο παραπάνω πίνακας είναι υπεύθυνος για την αποθήκευση των στοιχείων του χρήστη, έτσι ώστε να κρατάμε δεδομένα για τον κάθε χρήστη ξεχωριστά και την πρόοδο του.

Πίνακας 2: lessons

	id [PK] integer	chapter integer	code text	result text	explanation text	page integer
1	16	1	class Mo...	5.640000...	Modulus op...	3
2	17	1	class incr...	32	Operator ++...	4
3	18	1	class Out...	3 4 4	None.	5
4	19	1	class isin...	true	isInfinite() ...	6
5	20	1	class incr...	32	Operator ++...	7
6	21	1	class out...	NaN Infini...	For floating...	8
7	22	1	class vari...	CExceptio...	Second prin...	9
8	23	1	byte b = 5...	* operator...	While evalu...	10
9	24	1	class left...	200	None.	11
10	25	1	class Out...	1 2 3	arr.length() ...	12
11	26	2	At line nu...	final, stati...	Every interf...	1
12	27	2	class Alli...	Both x,an...	Both x,and ...	2
13	28	2	final clas...	Exception...	class A has...	3
14	29	2	class Abc...	The varia...	The value a...	4
15	30	2	class A { i...	2	r is referenc...	5
16	31	2	class box ...	6	None.	6
17	32	2	In the foll...	all of the ...	sum is a va...	7
18	33	2	class are...	error: can...	Variable hei...	8
19	34	2	class box ...	25	None.	9
20	35	2	class Out...	error: can...	Variable hei...	10
21	36	3	int arr[] = ...	Class na...	If we trying ...	1

Στον συγκεκριμένο πίνακα είναι αποθηκευμένη η θεωρία και τα παραδείγματα κώδικα για το κάθε κεφάλαιο. Η στήλη chapter δηλώνει το κεφάλαιο, η code περιέχει το παράδειγμα κώδικα που δίνεται στον χρήστη, η result το αποτέλεσμα του κώδικα, explanation η εξήγηση του κώδικα και η στήλη page αριθμεί τις σελίδες του κάθε κεφαλαίου.

Πίνακας 3: account_lesson

	accountid integer	chapterid integer	lessonid integer	finished boolean
1	1	1	10	true
2	1	2	10	true
3	1	4	10	true
4	1	3	10	true
5	4	1	10	true
6	4	3	10	true
7	4	4	10	true
8	4	2	10	true

Στον πίνακα account_lesson αποθηκεύουμε το κάθε chapter που ολοκληρώνει ο χρήστης από το κομμάτι της θεωρίας .

Πίνακας 4: quiz

	id [PK] integer	chapter integer	question_number integer	question text	choice1 text	choice2 text	choice3 text	choice4 text	answer integer	explanation text
1	2	1	1	Which sta...	a) Java is...	b) Java is...	c) Java is...	d) Java is...	4	Java is call...
2	3	1	2	Which co...	a) JRE	b) JIT	c) JDK	d) JVM	3	JDK is a cor...
3	4	1	3	Which on...	a) Object...	b) Use of ...	c) Portable	d) Dynam...	2	Pointers is ...
4	5	2	1	Which of ...	a) identifi...	b) identifier	c) keyword	d) none o...	3	Keywords a...
5	6	2	2	What is th...	a) .js	b) .txt	c) .class	d) .java	4	Java files h...
6	7	2	3	Which en...	a) MAVEN...	b) JavaPA...	c) JAVA	d) JAVA_...	4	JAVA_HOM...
7	8	3	1	Which of ...	a) Polym...	b) Inherit...	c) Compil...	d) Encaps...	3	There are 4 ...
8	9	3	2	Which of ...	a) Multipl...	b) Compil...	c) Multile...	d) Executi...	2	There are t...
9	10	3	3	Which ex...	a) Memor...	b) OutOf...	c) Memor...	d) Memor...	4	The Xms fla...
10	11	1	4	Which of ...	a) start() ...	b) run() m...	c) A threa...	d) A threa...	2	run() metho...
11	12	1	5	Which of ...	a) check	b) throw	c) catch	d) try	4	try is used f...
12	13	1	6	Which on...	a) Protect...	b) Void	c) Public	d) Private	2	Public, priv...
13	14	1	7	What is th...	a) 0 to 256	b) -128 to ...	c) 0 to 65...	d) 0 to 32...	3	Char occupi...
14	15	1	8	Which cla...	a) Server	b) ServerR...	c) Socket	d) Server...	3	ServerSock...
15	16	1	9	Which of ...	a) Servlet...	b) Servlet...	c) Servlet...	d) Servlet...	2	Servlets ex...
16	17	1	10	An expres...	a) int	b) long	c) byte	d) float	1	An expressi...
17	18	1	11	Which of ...	a) -1.7e+...	b) -3.4e+...	c) +1.7e+...	d) 3.4e+0...	2	Range of fl...
18	19	1	12	What is th...	a) -128 to...	b) -32768 ...	c) -21474...	d) None o...	1	Byte occupi...
19	20	2	4	What wou...	a) Runtim...	b) Throws...	c) compil...	d) Runs s...	2	this() and s...
20	21	2	5	Which of ...	a) malloc	b) alloc	c) new	d) give	3	Operator ne...
21	22	2	6	Which of ...	a) Every c...	b) Applet...	c) There c...	d) main() ...	2	Every class ...

Ο πίνακας quiz περιέχει ερωτήσεις από 4 κεφάλαια και τις 4 απαντήσεις για την κάθε απάντηση που καλείται να διαλέξει ο χρήστης. Επιπλέον, υπάρχει η σωστή απάντηση (answer) για τον έλεγχο της επιλογής του χρήστη και η στήλη explanation για την εξήγηση της σωστής απάντησης.

Πίνακας 5: account_quiz

	id [PK] integer	account_id integer	quiz_chapter_id integer	success_per integer
1	1	1	2	0
2	2	1	3	100
3	3	4	1	100
4	4	4	2	100
5	5	1	1	0
6	6	1	4	0

Τέλος στον παραπάνω πίνακα κρατάμε ποσοστό επιτυχίας των χρηστών για το κάθε τεστ πολλαπλών επιλογών που ολοκληρώνουν. Το quiz_chapter_id δηλώνει ποιο απο τα τέσσερα chapter είναι ολοκληρωμένο και το account_id είναι μοναδικός αριθμός χρήστη από τον πίνακα accounts που έχει δοθεί στον αντίστοιχο χρήστη.

Παρουσιαση Εκτελεσης

Java Essentials

— □ ×

Login

Username:

Password:

Log In

OR


Register


First Name: Last Name:

Username:

Password:

Sign Up





Java Essentials

— □ ×

Home



Chapters

Chapter Quiz

Repetitive Test

Manual

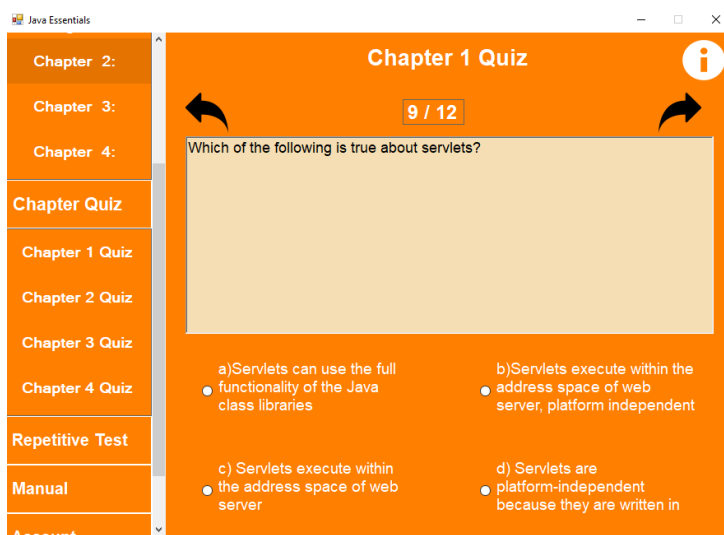
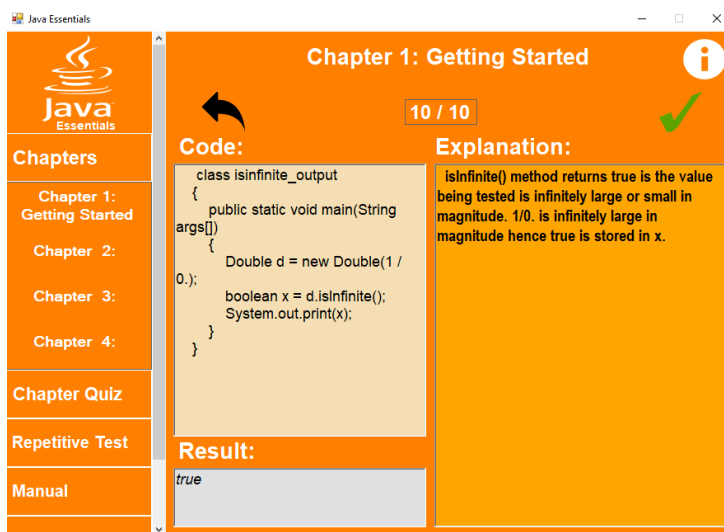
Account

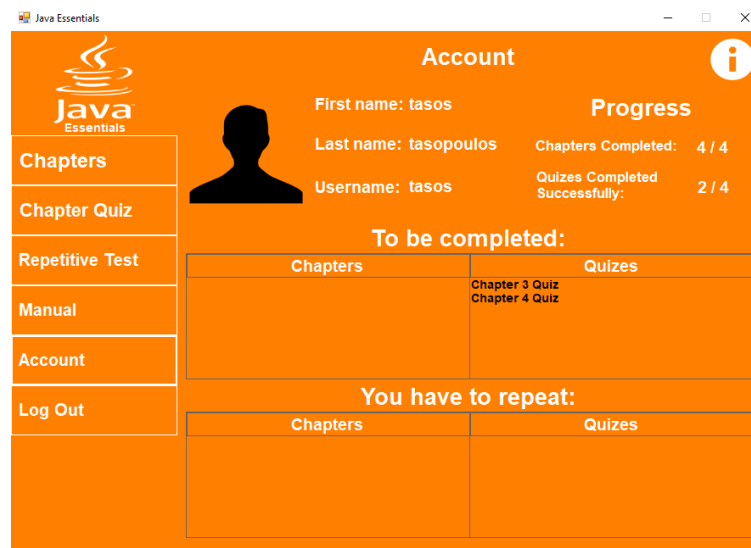
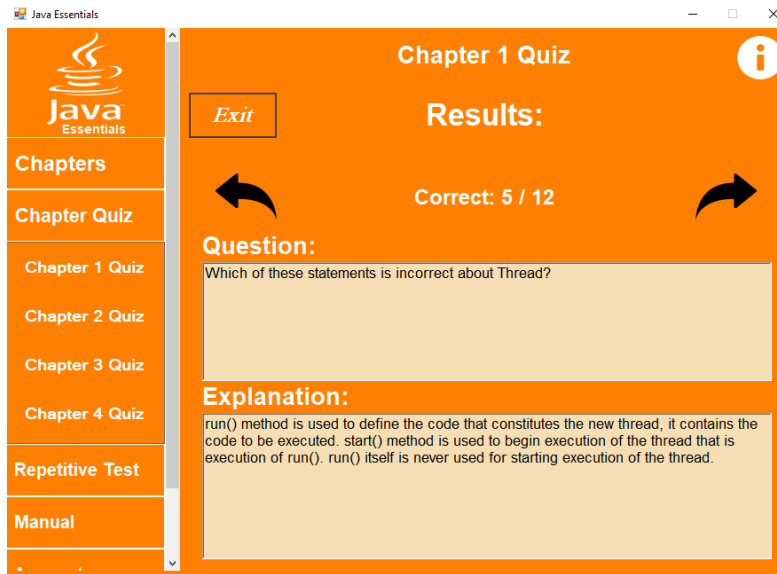
Log Out

Welcome to









Βιβλιογραφία

- Stackoverflow.com
- <https://www.geeksforgeeks.org>
- <https://www.postgresql.org/docs/>
- <https://www.sanfoundry.com/java-questions-answers-freshers-experienced/>