



Ανάπτυξη και Αρχιτεκτονικές Πληροφοριακών Συστημάτων

2η Ατομική Άσκηση

ΕΙΣΑΓΩΓΗ - ΠΡΟΕΤΟΙΜΑΣΙΑ

Η 2^η ατομική άσκηση αποτελεί συνέχεια της 1^{ης} ατομικής άσκησης (μέρος Α' και Β'), οπότε θα πρέπει να έχετε ήδη υλοποιήσει τα παρακάτω:

1. index_ex1_b_8XXXXXX.jsp
2. about_ex1_b_8XXXXXX.jsp
3. register_ex1_b_8XXXXXX.jsp
4. registerController_ex1_b_8XXXXXX.jsp
5. login_ex1_b_8XXXXXX.jsp
6. header_ex1_b_8XXXXXX.jsp
7. footer_ex1_b_8XXXXXX.jsp
8. loginController_ex1_b_8XXXXXX.jsp
9. dashboard_ex1_b_8XXXXXX.jsp
10. logout_ex1_b_8XXXXXX.jsp
11. error_ex1_b_8XXXXXX.jsp

Δημιουργήστε ένα αντίγραφο για κάθε μια από τις παραπάνω σελίδες και δώστε τα παρακάτω ονόματα αντίστοιχα:

1. index_ex2_8XXXXXX.jsp
2. about_ex2_8XXXXXX.jsp
3. register_ex2_8XXXXXX.jsp
4. registerController_ex2_8XXXXXX.jsp
5. login_ex2_8XXXXXX.jsp
6. header_ex2_8XXXXXX.jsp
7. footer_ex2_8XXXXXX.jsp
8. loginController_ex2_8XXXXXX.jsp
9. dashboard_ex2_8XXXXXX.jsp
10. logout_ex2_8XXXXXX.jsp
11. error_ex2_8XXXXXX.jsp

, όπου 8XXXXXX ο αριθμός μητρώου σας.

Στη συνέχεια ενημερώστε τους αντίστοιχους συνδέσμους (hyperlinks) στο μενού και όπου αλλού χρειάζεται όπως για παράδειγμα στο action της φόρμας στη **register_ex2_8XXXXXX.jsp**, στα forwards μέσω JSP Actions, στα include directives κτλ. έτσι ώστε να οδηγούν στις σελίδες που δημιουργήσατε παραπάνω (δηλ. *_ex2_8XXXXXX.jsp) και όχι στις παλιές (δηλ. *_ex1_b_8XXXXXX.jsp).

Για την υλοποίηση της ατομικής άσκησης σας δίνεται επιπλέον:

- A. Το αρχείο **exercise2_2020_2021_8XXXXXX.zip** το οποίο θα πρέπει να το κατεβάσετε και να κάνετε εξαγωγή (extract, unzip) στην επιφάνεια εργασίας του υπολογιστή σας. Με αυτόν τον τρόπο θα δημιουργηθεί ο φάκελος **exercise2_2020_2021_8XXXXXX** (πακέτο - package) ο οποίος περιέχει τα παρακάτω αρχεία (κλάσεις):

1. **DB.java**

(Δεν επιτρέπεται καμία αλλαγή πλην της αντικατάστασης του αριθμού μητρώου και τη συμπλήρωση των ρυθμίσεων (dbName, dbusername, dbpassword) για πρόσβαση στη βάση δεδομένων.)

2. **User.java**

(Δεν επιτρέπεται καμία αλλαγή πλην της αντικατάστασης του αριθμού μητρώου)

3. **UserDAO.java**

(«ημιτελής» κλάση, δεν επιτρέπεται καμία αλλαγή στις υπογραφές και τον τύπο επιστροφής των μεθόδων, αντικατάσταση του αριθμού μητρώου και βλέπε **Θέμα 1**)

Αλλάξτε το όνομα του φακέλου (java package) και στα παραπάνω αρχεία βάζοντας όπου **8XXXXXX** τον αριθμό μητρώου σας.

Στη συνέχεια, φροντίστε ότι οι νέες σελίδες (*_ex2_8XXXXXX.jsp) θα χρησιμοποιούν τις κλάσεις **User** και **UserDAO** του καινούργιου πακέτου και όχι της 1^{ης} ατομικής άσκησης – μέρος Β'.

Εάν ακολουθήσετε τις παραπάνω οδηγίες σωστά θα πρέπει να βλέπετε την σελίδα **login_ex2_8XXXXXX.jsp** (βλέπε **Figure 1**) στο παρακάτω URL:

http://ism.dmst.aueb.gr/ismgroupY/login_ex2_8XXXXXX.jsp, όπου **Y** και **8XXXXXX** ο αριθμός ομάδας και ο αριθμός μητρώου αντίστοιχα.

- B. Το αρχείο: **users_table_data.sql** το οποίο περιέχει τις εντολές SQL που απαιτούνται για:

1. τη δημιουργία του πίνακα **users_ex2_8xxxxxx_2020_2021** στη βάση σας (ismgroupY)
2. την εισαγωγή των απαραίτητων εγγραφών (3) στον πίνακα **users_ex2_8xxxxxx_2020_2021**.

Αντικαταστήστε όπου **8XXXXXX** τον αριθμό μητρώου σας και με την βοήθεια του **MySQL Workbench** εκτελέστε τα **SQL Statements** έτσι ώστε να δημιουργηθούν ο πίνακας με τις απαραίτητες εγγραφές.

Please sign in

username

password

Sign in

help:

- For John Doe: idno 1111
- For Mary Smith: idno11 1111

Figure 1: Η σελίδα login_ex2_8XXXXXX.jsp

ΘΕΜΑ 1

Να τροποποιήσετε την κλάση **UserDAO** υλοποιώντας το σώμα των μεθόδων **getUsers**, **findUser**, **authenticate** και **register** όπως αναφέρεται παρακάτω:

A. Η μέθοδος **getUsers** θα λειτουργεί ως εξής:

- Θα δημιουργεί ένα αντικείμενο της κλάσης **DB** και θα χρησιμοποιεί τη μέθοδο **getConnection** για να «ανοίξει» μια συνεδρία σύνδεσης με τη βάση δεδομένων και να λάβει το αντικείμενο *Connection*.
- Θα εκτελεί αναζήτηση (query μέσω **PreparedStatement**) στον πίνακα **users_ex2_8xxxxxx_2020_2021** για να «λάβει» όλους τους καταχωρημένους χρήστες. Για κάθε χρήστη που βρίσκει θα δημιουργεί το αντίστοιχο αντικείμενο **User** και θα το προσθέτει σε λίστα ([java.util.List](#)), και στο τέλος θα επιστρέφει τη λίστα.
- Σε περίπτωση που συμβεί κάποιο σφάλμα κατά την εκτέλεση της μεθόδου τότε θα παράγει εξαίρεση (**Exception**) με το κατάλληλο μήνυμα.
- Σε κάθε περίπτωση (είτε συμβεί κάποιο σφάλμα ή όχι) θα «κλείνει» τη σύνδεση με τη βάση δεδομένων (**close**).

B. Η μέθοδος **findUser**

- Θα δημιουργεί ένα αντικείμενο της κλάσης **DB** και θα χρησιμοποιεί τη μέθοδο **getConnection** για να «ανοίξει» μια συνεδρία σύνδεσης με τη βάση δεδομένων και να λάβει το αντικείμενο *Connection*.
- Θα εκτελεί αναζήτηση (query μέσω **PreparedStatement**) στον πίνακα **users_ex2_8xxxxxx_2020_2021** για να βρει εάν υπάρχει καταχωρημένος χρήστης με **username** ίσο με την τιμή του ορίσματος (**username**). Σε περίπτωση που υπάρχει καταχωρημένος χρήστης, τότε με βάση τα στοιχεία του θα δημιουργεί ένα αντικείμενο της κλάσης **User** και θα επιστρέφει το αντικείμενο. Διαφορετικά (δεν υπάρχει καταχωρημένος χρήστης), θα επιστρέφει την τιμή **null**.
- Σε περίπτωση που συμβεί κάποιο σφάλμα κατά την εκτέλεση της μεθόδου τότε θα παράγει εξαίρεση (**Exception**) με το κατάλληλο μήνυμα.
- Σε κάθε περίπτωση (είτε συμβεί κάποιο σφάλμα ή όχι) θα «κλείνει» τη σύνδεση με τη βάση δεδομένων (**close**).

C. Η μέθοδος **authenticate**

- Θα δημιουργεί ένα αντικείμενο της κλάσης **DB** και θα χρησιμοποιεί τη μέθοδο **getConnection** για να «ανοίξει» μια συνεδρία σύνδεσης με τη βάση δεδομένων και να λάβει το αντικείμενο *Connection*.
- Θα εκτελεί αναζήτηση (query μέσω **PreparedStatement**) στον πίνακα **users_ex2_8xxxxxx_2020_2021** για να βρει εάν υπάρχει καταχωρημένος χρήστης με **username** και **password** ίσα με την τιμές των ορισμάτων (**username** και **password** αντίστοιχα). Σε περίπτωση που βρεθεί-υπάρχει χρήστης, τότε επιστρέφει το αντικείμενο **User** που αντιστοιχεί στο χρήστη με αυτά τα συνθηματικά (**username** & **password**). Στην περίπτωση που δεν υπάρχει καταχωρημένος χρήστης, τότε θα παράγει εξαίρεση (**Exception**) με το παρακάτω μήνυμα:
«**Wrong username or password**».
- Σε περίπτωση που συμβεί κάποιο σφάλμα κατά την εκτέλεση της μεθόδου τότε θα παράγει εξαίρεση (**Exception**) με το κατάλληλο μήνυμα.
- Σε κάθε περίπτωση (είτε συμβεί κάποιο σφάλμα ή όχι) θα «κλείνει» τη σύνδεση με τη βάση δεδομένων (**close**).

D. Η μέθοδος **register**

- Θα δημιουργεί ένα αντικείμενο της κλάσης **DB** και θα χρησιμοποιεί τη μέθοδο **getConnection** για να «ανοίξει» μια συνεδρία σύνδεσης με τη βάση δεδομένων και να λάβει το αντικείμενο *Connection*.

- Θα χρησιμοποιεί το αντικείμενο **user** (όρισμα, τύπου **User**) για να ελέγξει (θα εκτελεί αναζήτηση - query μέσω PreparedStatement) στον πίνακα **users_ex2_8xxxxxx_2020_2021** εάν υπάρχει (ήδη) καταχωρημένος χρήστης με το συγκεκριμένο **username** ή το **email** (που περιέχονται στο αντικείμενο **user**) και:
 1. Σε περίπτωση που υπάρχει (ήδη) καταχωρημένος χρήστης τότε θα παράγει εξαίρεση με το μήνυμα: «**username or email already registered**».
 2. Διαφορετικά (δηλ. username και email δεν χρησιμοποιούνται από κάποιο χρήστη), θα εισάγει (μέσω **PreparedStatement**) στον πίνακα **users_ex2_8xxxxxx_2020_2021** τα στοιχεία του χρήστη.
 - Σε περίπτωση που συμβεί κάποιο σφάλμα κατά την εκτέλεση της μεθόδου τότε θα παράγει εξαίρεση (**Exception**) με το κατάλληλο μήνυμα.
 - Σε κάθε περίπτωση (είτε συμβεί κάποιο σφάλμα ή όχι) θα «κλείνει» τη σύνδεση με τη βάση δεδομένων (**close**).
-

ΘΕΜΑ 2

Να τροποποιήσετε την σελίδα **registerController_ex2_8XXXXXX.jsp** (όπου **8XXXXXX** ο αριθμός μητρώου σας) έτσι ώστε, σε περίπτωση που τα στοιχεία της φόρμας εγγραφής είναι έγκυρα (βλέπε Θέμα 2^ο κανόνες a,b,c,d – 1^η ατομική μέρος Α'), τότε μέσω της μεθόδου **register** (κλάση **UserDAO** – Θέμα 1^ο D) θα αποθηκεύει τα στοιχεία του χρήστη στον πίνακα **users_ex2_8xxxxxx_2020_2021** στη βάση δεδομένων.

ΘΕΜΑ 3

Να τροποποιήσετε την σελίδα **dashboard_ex2_8XXXXXX.jsp** (όπου **8XXXXXX** ο αριθμός μητρώου σας) έτσι ώστε (επιπλέον των όσων έκανε από την 1^η ατομική):

- A. Να εμφανίζει τίτλο: «**Available Users**» και δίπλα το σύνολο των καταχωρημένων χρηστών.
- B. Να εμφανίζει τα στοιχεία **username**, **name**, **surname** και **email** όλων των χρηστών με τη μορφή πίνακα όπως στο παρακάτω σχήμα **Figure 2**.
- C. Σε κάθε γραμμή το στοιχείο **username** θα είναι υπερσύνδεσμός (hyperlink-link) κατά τέτοιο τρόπο «ορισμένος» έτσι ώστε όταν ο χρήστης κάνει «κλικ» σε κάποιο username θα πηγαίνει στη σελίδα **profile_ex2_8XXXXXX.jsp** (Θέμα 4^ο) μαζί με μια παράμετρο με όνομα **uname** και τιμή το username (που έκανε κλικ).

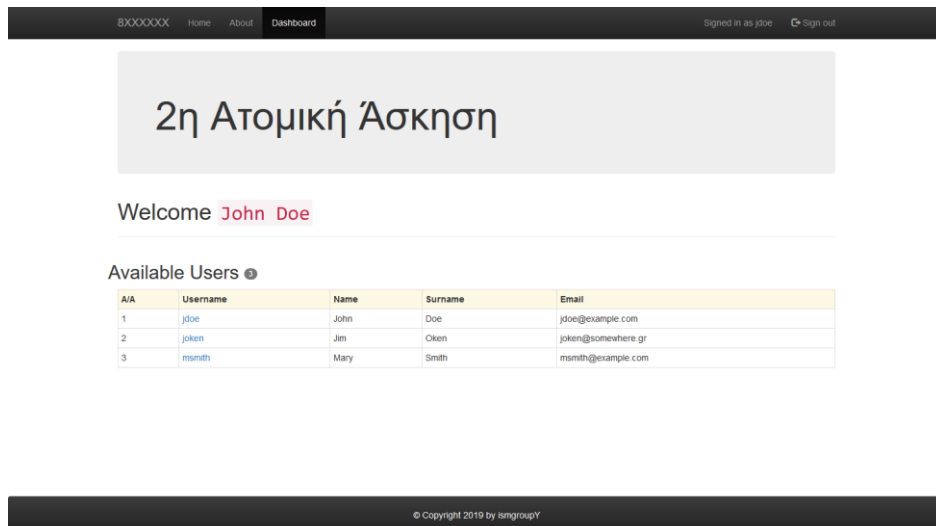


Figure 2: Η σελίδα dashboard_ex2_8XXXXXX.jsp

Σημείωση:

Χρησιμοποιήστε τη μέθοδο **getUsers** της κλάσης **UserDAO** για τον «υπολογισμό» του συνόλου αλλά και τη δημιουργία του πίνακα με τους χρήστες.

ΘΕΜΑ 4

Να δημιουργήσετε μια σελίδα με όνομα **profile_ex2_8XXXXXX.jsp** (όπου **8XXXXXX** ο αριθμός μητρώου σας) με τις εξής λειτουργίες:

- Όμοια με τη **dashboard_ex2_8XXXXXX.jsp** θα ελέγχει (μέσω του attribute **userObj2020** στο **Session**) εάν ο χρήστης έχει κάνει **Sign in** και:
 - Σε περίπτωση που ο χρήστης **δεν** έχει κάνει **Sign in** τότε θα προσθέτει στο **request** ένα attribute με όνομα **message** και τιμή (τύπου String):
«You are not authorized to access this resource. Please login.»
και θα προωθεί (μέσω JSP Action) στη σελίδα **login_ex2_8XXXXXX.jsp**
 - Διαφορετικά (έχει κάνει **Sign in**), θα ανακτά το αντικείμενο **User** από το **Session** και θα εμφανίζει στο μενού δεξιά το κείμενο:
«Signed in as U» όπου **U** το **username** που χρησιμοποιήθηκε για **Sign in**
- Θα διαβάζει από το request την παράμετρο **uname** (βλέπε παραπάνω θέμα 3) και μέσω της μεθόδου **findUser** της κλάσης **UserDAO** θα εκτελεί αναζήτηση στη βάση για να βρει τα στοιχεία του χρήστη που έχει username ίσο με την τιμή της παραμέτρου **uname**.
- Θα εμφανίζει τίτλο **«Profile of:»** και δίπλα το ονοματεπώνυμο του χρήστη που βρήκε στη βάση και από κάτω θα εμφανίζει σε μορφή μη διατεταγμένης λίστας (**list-unstyled**, Bootstrap) τις τιμές των **name**, **surname**, **email** και **username**, όπως στο σχήμα **Figure 3**.
- Σε περίπτωση που **δεν βρεθεί** καταχωρημένος χρήστης με username ίσο με την τιμή της παραμέτρου **uname** θα εμφανίζει το μήνυμα: **«User not found.»** όπως στο σχήμα **Figure 4**

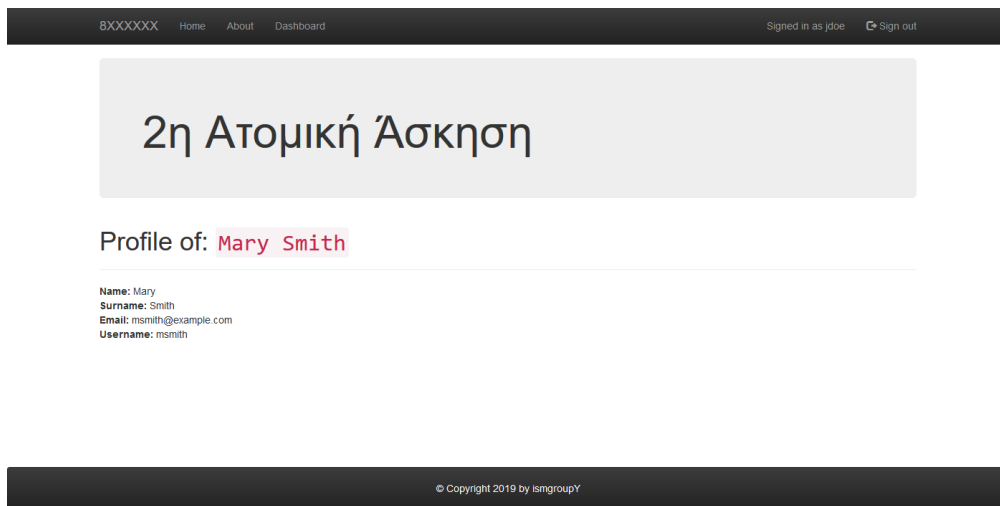


Figure 3: Η σελίδα profile_ex2_8XXXXXX.jsp για τον χρήστη με username **msmith**

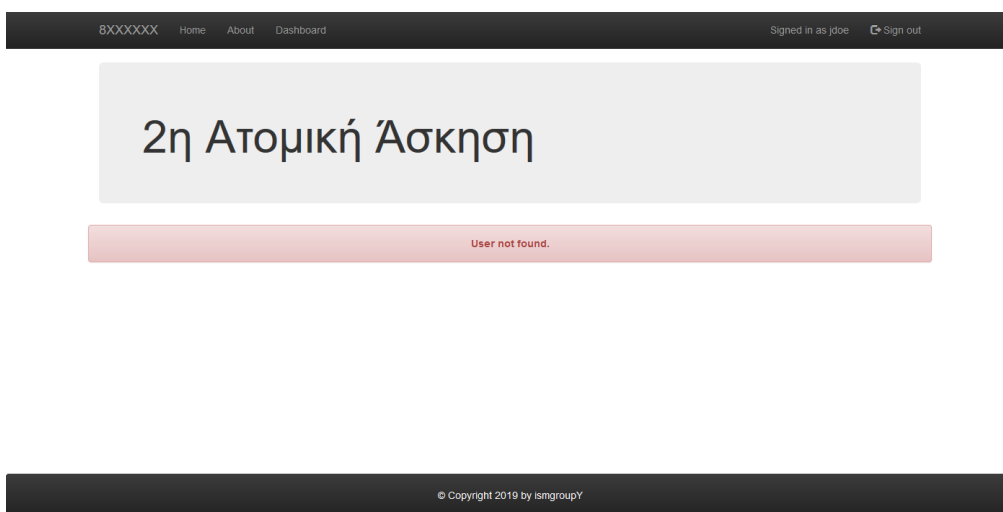


Figure 4: Η σελίδα profile_ex2_8XXXXXX.jsp όταν δεν βρεθεί χρήστης με username ίσο με την παράμετρο **uname**

Προθεσμία Παράδοσης:
Παρασκευή 4/12/2020
Τρόπος Παράδοσης:
Θα ανακοινωθεί στο EDU