



## Ανάπτυξη και Αρχιτεκτονικές Πληροφοριακών Συστημάτων

### 3<sup>η</sup> Ατομική Άσκηση

#### ΕΙΣΑΓΩΓΗ - ΠΡΟΕΤΟΙΜΑΣΙΑ

Η 3<sup>η</sup> ατομική άσκηση αποτελεί συνέχεια της 1<sup>ης</sup> και 2<sup>ης</sup> ατομικής άσκησης, οπότε θα πρέπει να έχετε ήδη υλοποιήσει τα παρακάτω:

1. **loginController\_ex2\_8XXXXXX.jsp** (Θέμα 1 - 2<sup>η</sup> ατομική)
2. **dashboard\_ex2\_8XXXXXX.jsp** (Θέμα 2 - 2<sup>η</sup> ατομική)
3. **index\_ex2\_8XXXXXX.jsp** (Θέμα 3 - 2<sup>η</sup> ατομική)
4. **about\_ex2\_8XXXXXX.jsp** (Θέμα 3 - 2<sup>η</sup> ατομική)
5. **login\_ex2\_8XXXXXX.jsp** (2<sup>η</sup> ατομική)
6. **logout\_ex2\_8XXXXXX.jsp** (2<sup>η</sup> ατομική)
7. **error\_ex2\_8XXXXXX.jsp** (2<sup>η</sup> ατομική)
8. **theme\_8XXXXXX.css** (2<sup>η</sup> ατομική)
9. **register\_8XXXXXX.html** (Θέμα 1 - 1<sup>η</sup> ατομική)
10. **RegisterController\_8XXXXXX.java** (Θέμα 2 - 1<sup>η</sup> ατομική)

Δημιουργήστε ένα αντίγραφο όλων των αρχείων (jsp pages, servlet, css, javascript, images κτλ) της 2<sup>ης</sup> ατομικής άσκησης και στην συνέχεια αντικαταστήστε όπου υπάρχει το **ex2** με **ex3** και το **8XXXXXX** με τον αριθμό μητρώου σας.

Για την υλοποίηση της ατομικής άσκησης σας δίνονται:

- A. Το αρχείο **exercise3\_2018\_2019\_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 **exercise3\_2018\_2019\_8XXXXXX**. Αλλάξτε το όνομα του φακέλου (java package) και στα περιεχόμενα αρχεία (3 κλάσεις) βάζοντας όπου **8XXXXXX** τον αριθμό μητρώου σας. Στη συνέχεια να τροποποιήσετε τις παρακάτω κλάσεις (μόνο) όπου επιτρέπονται-απαιτούνται αλλαγές:
1. **DB.java**  
(Δεν επιτρέπεται καμία αλλαγή πλην της αντικατάστασης του αριθμού μητρώου και την συμπλήρωση των ρυθμίσεων (*dbName*, *dbusername*, *dbpassword*) για πρόσβαση στην βάση σας)
  2. **User.java**  
(Δεν επιτρέπεται καμία αλλαγή πλην της αντικατάστασης του αριθμού μητρώου)
  3. **UserDAO.java**  
(«ημιτελής» κλάση, αντικατάσταση του αριθμού μητρώου και βλέπε **Θέμα 1**)
- B. Το αρχείο **database\_ex3\_8xxxxxx.sql** το οποίο περιέχει τις εντολές SQL που απαιτούνται για:
- a) τη δημιουργία του πίνακα **users\_2018\_ex3\_8xxxxxx** στη βάση σας (**ismgroupY**)
  - b) την εισαγωγή των απαραίτητων εγγγραφών (3) στον πίνακα **users\_2018\_ex3\_8xxxxxx**

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.

2. Πρόσβαση στην Βάση της εφαρμογής (MySQL Database)

Με τη βοήθεια του προγράμματος **MySQL Workbench** εισάγετε τα παρακάτω στοιχεία:

Hostname:	195.251.249.131
Port:	3306
Username:	ismgroup100
Password:	
Default Schema:	ismgroup100

Figure 1: Παράδειγμα: ρυθμίσεις για πρόσβαση στη βάση της ομάδας 100

## ΘΕΜΑ 1

Ανοίξτε την κλάση **UserDAO** και υλοποιήστε το σώμα των μεθόδων **getUsers**, **findUser**, **authenticate** και **registerUser** 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\_2018\_ex3\_8xxxxxx** 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» την σύνδεση με την 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\_2018\_ex3\_8xxxxxx** 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, θα παράγει εξαίρεση (**Exception**) με το παρακάτω μήνυμα:  
«**Could not find User with username:** » και δίπλα η τιμή του **username**.
- 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» την σύνδεση με την 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\_2018\_ex3\_8xxxxxx** για να βρει εάν υπάρχει καταχωρημένος χρήστης με **username** και **password** ίσα με την τιμές των ορισμάτων (**username** και **password**

αντίστοιχα).

Σε περίπτωση που δεν υπάρχει καταχωρημένος χρήστης με αυτά τα συνθηματικά, 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» την σύνδεση με την βάση δεδομένων (**close**).

#### D. Η μέθοδος **registerUser**

- 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 εισάγει (μέσω **PreparedStatement**) στον πίνακα **users\_2018\_ex3\_8xxxxxx** 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» την σύνδεση με την βάση δεδομένων (**close**).

## ΘΕΜΑ 2

Με βάση τον κώδικα της σελίδας: **register\_8XXXXXX.html** της 1<sup>ης</sup> ατομικής άσκησης να δημιουργήσετε τη σελίδα **register\_ex3\_8XXXXXX.jsp**.

Τροποποιήστε τον κώδικα έτσι ώστε η σελίδα:

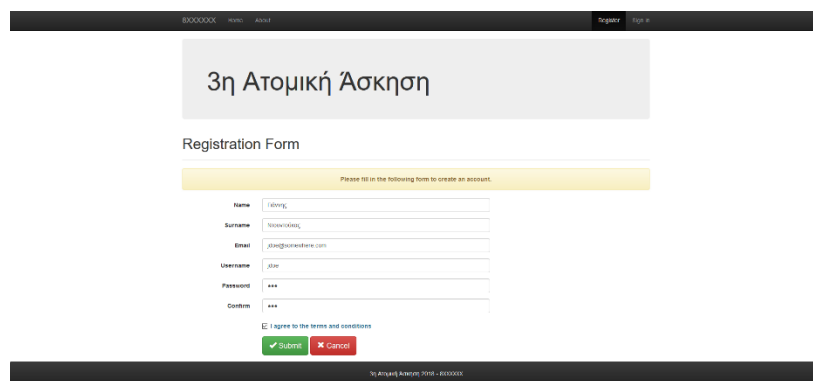
1. να δηλώνει ως error page την **error\_ex3\_8XXXXXX.jsp**
2. να ελέγχει εάν ο χρήστης έχει πραγματοποιήσει είσοδο (μέσω του attribute **ex3-user-object** στο **Session**) και σε περίπτωση που έχει πραγματοποιήσει είσοδο (Sign in) τότε θα τον ανακατευθύνει (redirect) στη σελίδα **dashboard\_ex3\_8XXXXXX.jsp**
3. να εμφανίζει στο μενού αριστερά οι επιλογές (κουμπιά): «**Home**» «**About**» και δεξιά οι: «**Register**» «**Sign in**» όπως στο **Figure 2**  
Όλα τα κουμπιά θα πρέπει να είναι λειτουργικά και θα πρέπει να οδηγούν στις αντίστοιχες σελίδες.  
Ενημερώστε και την επιλογή **Register** στο μενού των υπόλοιπων σελίδων της άσκησης.
4. να στέλνει τα δεδομένα (φόρμα) στο Servlet **RegisterController\_ex3\_8XXXXXX** του **Θέματος 3**

**Figure 2:** Η σελίδα **register\_ex3\_8XXXXXX.jsp**

## ΘΕΜΑ 3

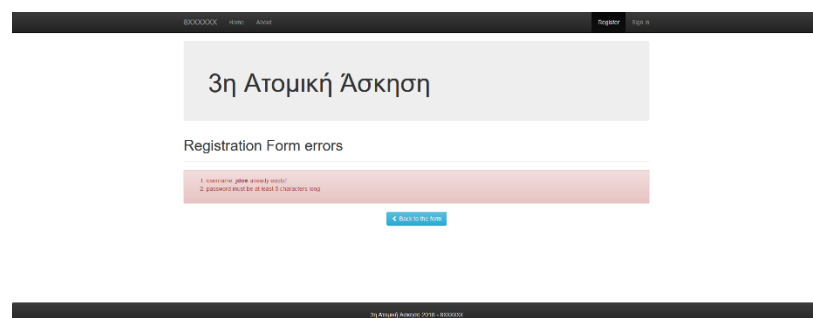
Με βάση το Servlet **RegisterController\_8XXXXXX** της 1<sup>ης</sup> ατομικής άσκησης να δημιουργήσετε το **RegisterController\_ex3\_8XXXXXX** το οποίο θα κάνει ότι και το προηγούμενο (Θέμα 2 – 1<sup>η</sup> ατομική) με την προσθήκη των παρακάτω λειτουργιών:

1. Στον κανόνα **b)** (Θέμα 2 2.β – 1<sup>η</sup> ατομική ) θα κάνει επιπλέον έλεγχο εάν το **username** που εισήγαγε ο χρήστης χρησιμοποιείται ήδη από κάποιο άλλο χρήστη.  
Πιο συγκεκριμένα, θα δημιουργεί ένα αντικείμενο της κλάσης **UserDAO** και μέσω της μεθόδου **findUser** θα ελέγχει εάν υπάρχει ήδη εγγεγραμμένος χρήστης με αυτό το username.  
Σε περίπτωση που το **username** χρησιμοποιείται ήδη τότε θα εμφανίζει το κατάλληλο μήνυμα στη διατεταγμένη λίστα με τα υπόλοιπα πεδία που δεν είναι έγκυρα (εάν υπάρχουν).  
Βλέπε **Figure 3 & 4**.
2. Σε περίπτωση που όλα τα πεδία είναι έγκυρα τότε επιπλέον (όσων κάνει από την 1<sup>η</sup> ατομική) θα εγγράφει τον χρήστη στο σύστημα. Πιο συγκεκριμένα, θα καλεί τη μέθοδο **registerUser** για να αποθηκεύσει τα στοιχεία του χρήστη στον πίνακα **users\_2018\_ex3\_8xxxxxx** της βάσης δεδομένων.
3. Θα εμφανίζει μενού όπως φαίνεται στο **Figure 4**.



The screenshot shows a web application titled "3η Ατομική Άσκηση" with a "Registration Form". The form contains fields for Name, Surname, Email, Username, Password, and Confirm. The Username field contains "john". Below the form, there is a checkbox for "I agree to the terms and conditions" and two buttons: "Submit" and "Cancel". The form is highlighted with a yellow background, indicating it is the focus of the figure.

**Figure 3:** Η σελίδα **register\_ex3\_8XXXXXX.jsp** με μη σωστά στοιχεία



The screenshot shows the same registration form as Figure 3, but with error messages displayed below the form. The errors are: "1. username: please already exist!" and "2. password must be at least 5 characters long". A "Back to the form" button is visible at the bottom of the error messages.

**Figure 4:** Το **RegisterController\_ex3\_8XXXXXX** εμφανίζει τα λάθη της φόρμας

## ΘΕΜΑ 4

Να δημιουργήσετε μια σελίδα **JSP** με όνομα **footer\_ex3\_8XXXXXX.jsp** η οποία θα περιέχει το footer που περιέχουν όλες οι σελίδες JSP και το Servlet. Αλλάξτε το footer έτσι ώστε να εμφανίζεται στο κέντρο το κείμενο: «3η Ατομική Άσκηση 2018 - 8XXXXXX», όπου **8XXXXXX** ο αριθμός μητρώου σας (**Figure 3 & 4**).

Στη συνέχεια τροποποιήστε τον κώδικα κάθε σελίδας JSP και του Servlet (Θέματος 3) έτσι ώστε το footer τους να παράγεται μέσω της σελίδας **footer\_ex3\_8XXXXXX.jsp**.

Πιο συγκεκριμένα:

1. Οι σελίδες JSP να κάνουν **include** μέσω directive τη σελίδα **footer\_ex3\_8XXXXXX.jsp**.
2. Το Servlet **RegisterController\_ex3\_8XXXXXX** θα κάνει **include** την **footer\_ex3\_8XXXXXX.jsp** μέσω του **RequestDispatcher**.

---

**Προθεσμία Παράδοσης:**

**Τετάρτη 5/12/2018**

**Τρόπος Παράδοσης:**

Θα ανακοινωθεί στο EDU