

Coding Bootcamp

1^η Ατομική Άσκηση

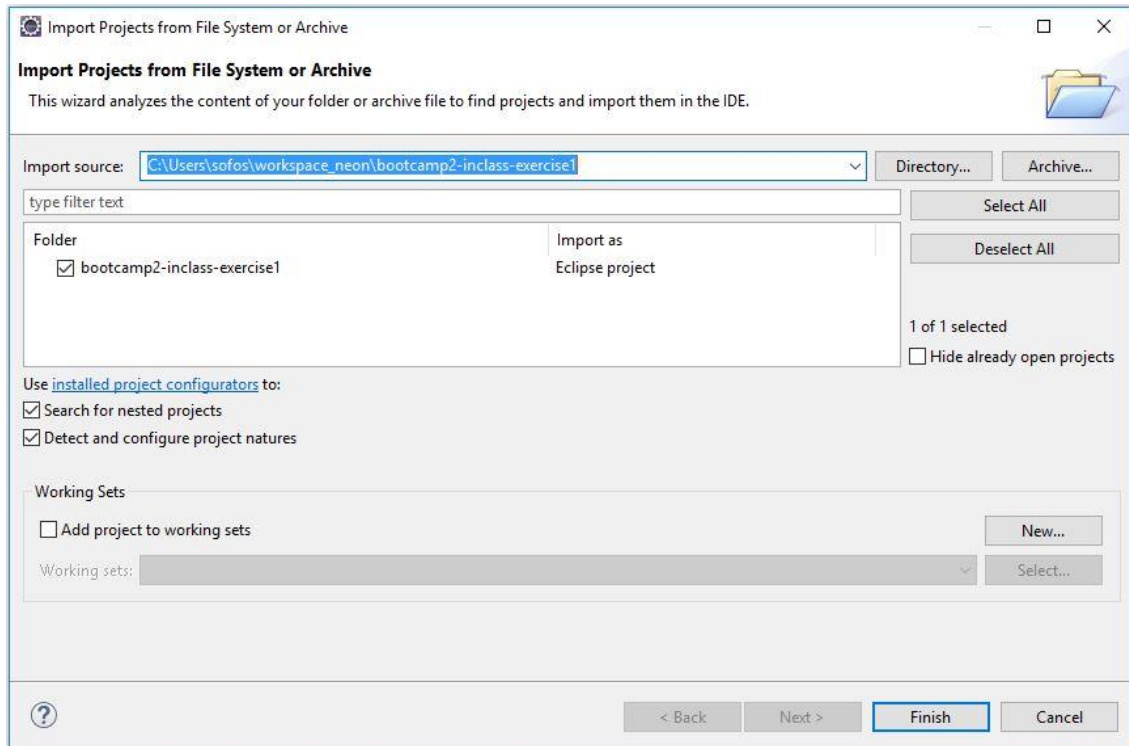
Προετοιμασία

Βήμα 1^ο: Εισαγωγή του project στο Eclipse

Για την υλοποίηση της ατομικής άσκησης σας δίνεται το αρχείο **bootcamp2-inclass-exercise1.zip** το οποίο περιέχει ένα ημιτελές web project. Κάντε unzip το αρχείο και θα δημιουργηθεί ένας φάκελος με όνομα **bootcamp2-inclass-exercise1**. Μεταφέρετε τον φάκελο στον **workspace** του eclipse σας και στην συνέχεια ανοίξτε το eclipse και επιλέξτε:

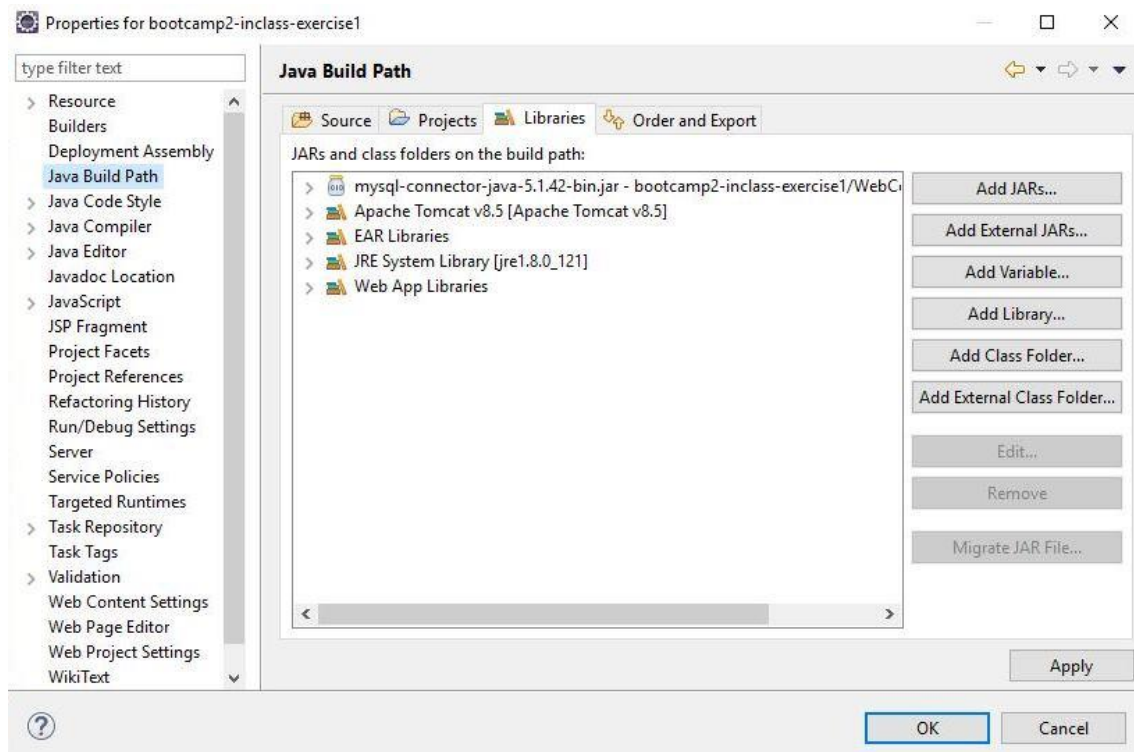
File -> Open Projects from File System

θα σας εμφανιστεί οθόνη όμοια με την παρακάτω:



Πατήστε **Directory** και εντοπίστε τον φάκελο **bootcamp2-inclass-exercise1** και κατόπιν πατήστε **Finish**.

Στη συνέχεια κάντε δεξί κλικ στο project **bootcamp2-inclass-exercise1** και επιλέξτε **Properties**, επιλέξτε Java Build Path και στην καρτέλα Libraries φροντίστε να είναι σύμφωνα με το δικό σας σύστημα (version Tomcat, JRE System Library)



Βήμα 2º: Δημιουργία Βάσης και Πίνακα

Συνδεθείτε (τοπικά) στον MySQL database server του υπολογιστή σας και είτε δημιουργήστε καινούργια **βάση** είτε σε μια υπάρχουσα δημιουργήστε έναν πίνακα με όνομα **exercise1_student** εκτελώντας το παρακάτω sql statement:

```
CREATE TABLE `exercise1_student` (  
  `am` varchar(45) COLLATE utf8_unicode_ci NOT NULL,  
  `name` varchar(255) COLLATE utf8_unicode_ci NOT NULL,  
  `surname` varchar(255) COLLATE utf8_unicode_ci NOT NULL,  
  PRIMARY KEY (`am`)  
) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8_unicode_ci;
```

Κατόπιν ανοίξτε την κλάση **DB** (πακέτο *bootcamp2.lesson4.exercise1.connection*) και βάλτε τα αντίστοιχα πεδία (όνομα βάσης, username, password).

Τέλος, εκτελέστε το project (Run on Server) για να θα ανοίξει ο browser στο URL:

<http://localhost:8080/bootcamp2-inclass-exercise1/>

Θα δείτε την παρακάτω οθόνη:

LESSON 4RegisterStudents

LESSON 4

Φόρμα Εγγραφής

Όνομα

το όνομα σου

Επώνυμο

το επώνυμο σου

Αρ.Μητρώου

ο Αριθμός Μητρώου σου

Καταχώρηση

© 2017 by INSERT YOUR NAME

Figure 1

TASK 1: **StudentDAO** class

Ανοίξτε την κλάση **StudentDAO** και υλοποιήστε τις μεθόδους **getStudents** και **saveStudent** ως εξής:

1. Η μέθοδος **getStudents**

Δεν δέχεται όρισμα, επιστρέφει λίστα (ArrayList) από αντικείμενα **Student** και παράγει εξαίρεση (Exception). Δημιουργεί ένα αντικείμενο της κλάσης **DB** και μέσω των μεθόδων (**open**, **getConnection**, **close**) θα συνδέεται στην βάση και θα εκτελεί αναζήτηση (query) στον πίνακα **exercise1_student** προκειμένου να λάβει όλους τους καταχωρημένους φοιτητές. Τέλος, θα επιστρέφει μια ArrayList από αντικείμενα **Student**.

Σε περίπτωση που συμβεί κάποιο σφάλμα θα παράγει εξαίρεση με τον κατάλληλο μήνυμα.

2. Η μέθοδος **saveStudent**

- Δέχεται όρισμα αντικείμενο τύπου **Student**, δεν επιστρέφει τιμή και παράγει εξαίρεση (Exception). Η μέθοδος αυτή θα δημιουργεί ένα αντικείμενο της κλάσης **DB** και μέσω των μεθόδων (**open**, **getConnection**, **close**) θα συνδέεται στην βάση και θα αποθηκεύει τα στοιχεία του φοιτητή (όνομα, επώνυμο & αρ. μητρώου) στον πίνακα **exercise1_student** στη βάση. Σε περίπτωση που συμβεί κάποιο σφάλμα θα παράγει εξαίρεση με τον κατάλληλο μήνυμα.

TASK 2: RegisterController Servlet

Ανοίξτε την κλάση **RegisterController** (*bootcamp2.lesson4.exercise1.controllers*) και υλοποιήστε την μέθοδο **doGet** ως εξής:

- a) Θα απαντά στην φόρμα και θα διαβάζει τις παραμέτρους (όνομα, επώνυμο & αρ. μητρώου)
- b) Θα ελέγχει εάν τα δεδομένα (φόρμα) είναι έγκυρα (όνομα και επώνυμο τουλάχιστον 2 χαρακτήρες, ο αρ. μητρώου από 8 ψηφία) και:
 1. Σε περίπτωση που τα δεδομένα δεν είναι έγκυρα τότε θα προσθέτει στο request ένα attribute με όνομα **errorMessage** με τιμή το (String):
Data not valid
και θα προωθεί το request πίσω στην φόρμα (**StudentFormServlet**)
 2. Εάν τα δεδομένα είναι έγκυρα τότε θα δημιουργεί ένα αντικείμενο της κλάσης **StudentDAO** και μέσω της μεθόδου **saveStudent** θα αποθηκεύει τα δεδομένα στην βάση.
Τέλος, θα προσθέτει ένα attribute στο request με όνομα **message** με τιμή το (String):
Registration completed successfully!
και θα προωθεί το request πίσω στην φόρμα (**StudentFormServlet**)

TASK 3: ViewStudentsServlet Servlet

Ανοίξτε την κλάση **ViewStudentsServlet** (*bootcamp2.lesson4.exercise1.views*) και υλοποιήστε την μέθοδο **doGet** ως εξής:

- a) Θα δημιουργεί ένα αντικείμενο της κλάσης **StudentDAO** και μέσω της μεθόδου **getStudents** θα λαμβάνει την λίστα με τους καταχωρημένους φοιτητές και τέλος θα τους τυπώνει σε πίνακα (bootstrap) (Figure 2)
- b) Σε περίπτωση που δεν βρεθεί κάποια καταχώρηση θα εμφανίζει κατάλληλο μήνυμα (Figure 3)

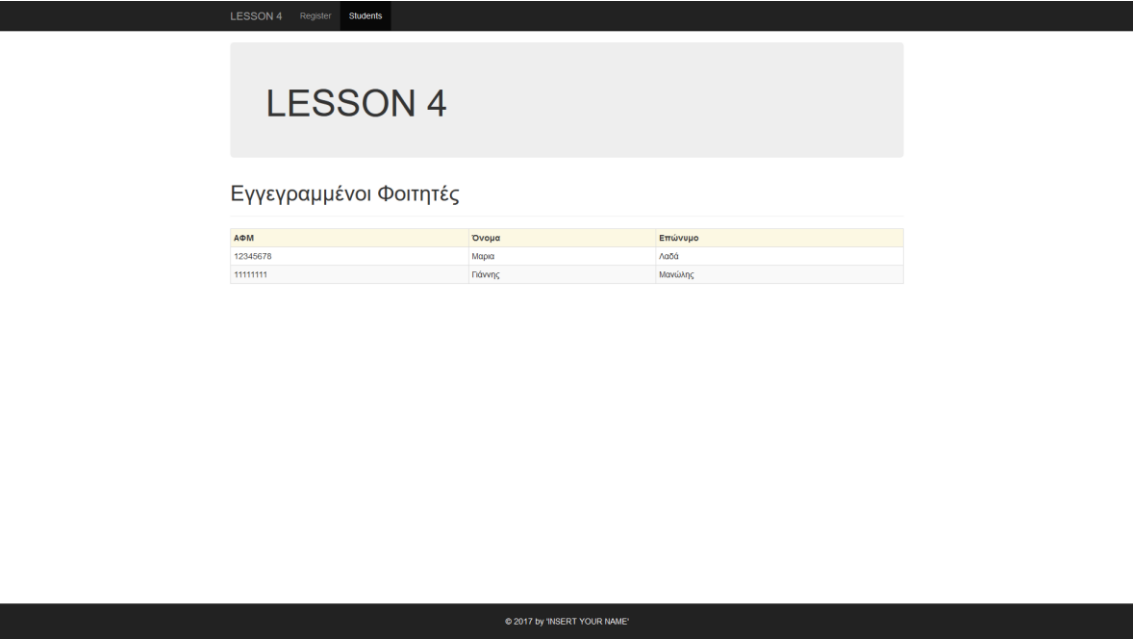


Figure 2

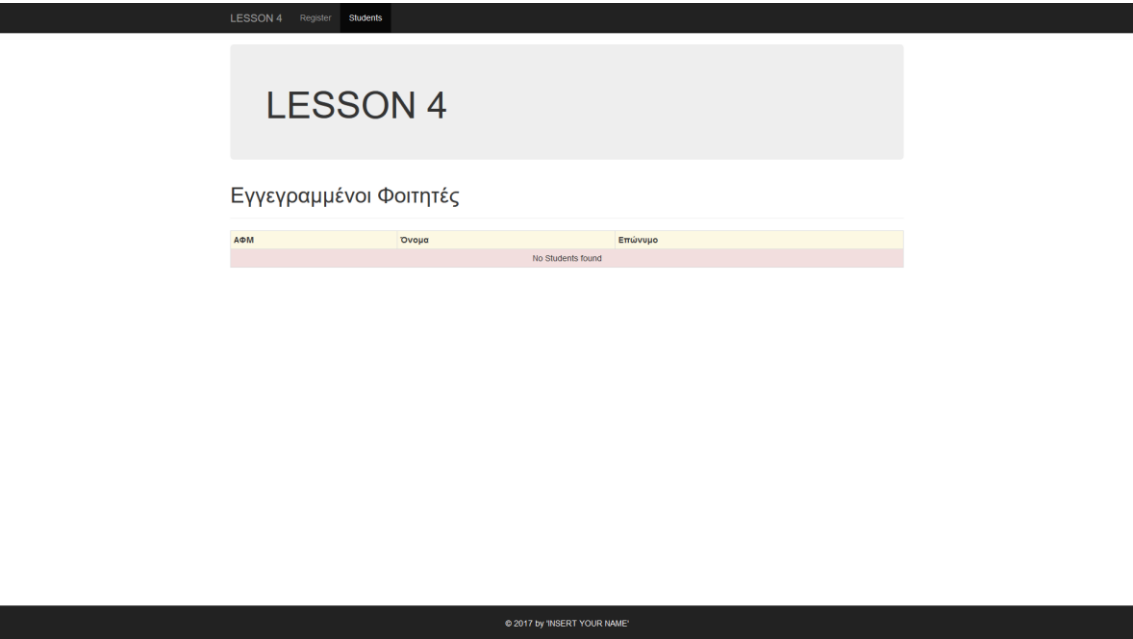


Figure 3