

4BHEL, FSST, Übungsaufgabe (inkl. Abgabe)

JPA/Hibernate mit Servlets

Erstelle ein Java Projekt mit JPA/Hibernate Unterstützung und einem Tomcat-Webserver für eine JSP/Servlet Applikation, um aus einer Datenbank User, Passwörter und Kommentare abzufragen und zu bearbeiten.

Setze dafür folgende Aufgaben um:

- Importiere die Datenbank `logindb` aus dem Self-contained Backupfile und analysiere die Tabellen. Stelle fest, welche Daten gespeichert sind und welche Relationen bestehen.
- Erstelle ein Java Projekt in IntelliJ mit Hibernate Unterstützung und konfiguriere es entsprechend.
- Erstelle geeignete Entity- und Dao-Klasse(n), um die User und die Kommentare der User aus der Datenbank abzufragen und zu löschen > Operationen 1 - 5. Verwende Association Mappings (One-To-Many/Many-To-One) und Listen für Beziehungen zwischen Entities.
- Erstelle eine Klasse `LoginHandler` mit einer main-Methode (Konsolen-Applikation). Führe in der main-Methode **folgende Operationen (1 – 5)** aus:
 1. Gib alle User nach Namen sortiert aus.
 2. Gib alle User aus, die eine HTL-Emailadresse haben (d.h. ...@htl-klu.at)
 3. Gib alle Kommentare des Users mit der id ==8 aus.
 4. Erstelle in der DB einen neuen Kommentar für den User **Victor Mühleder** mit einem beliebigen, aber sinnvollen Text.
 5. Lösche alle Kommentare zu einem User unter Angabe seiner ID.
- Erstelle nun eine **index.jsp** - Seite die alle Daten in HTML strukturiert (Tabellen) und mit CSS formatiert zu den Operationen 1 – 3 darstellt/visualisiert.
- Erstelle nun eine Tomcat- Webapplikation (JSP+SERVLET), wobei:
 - ...für die Operation 4 in einer eigenen **insertcomment.jsp** - Seite ein Eingabe-Formular bestehen, womit die Daten per GET-Request zu einem Servlet übermittelt werden, das dann das Insert über die DAO mit den Daten durchführt. Nach dem Insert Vorgang, wird wieder auf die index.jsp Seite umgeleitet.
 - ...für die Operation 5 in einer eigenen **deletecomments.jsp** - Seite ein Eingabe-Formular bestehen, womit die ID per GET-Request zu einem Servlet übermittelt wird, das dann das Delete über die DAO durchführt. Nach dem Delete Vorgang, wird wieder auf die index.jsp Seite umgeleitet.
- Erweitere die DAO - Klasse um die Methode `boolean checkPassword(String user, String password)` welche `true` oder `false` zurück gibt, je nachdem ob das übergebene Passwort zum übergeben User in der Datenbank passt oder nicht.
 - Teste diese Methode in main-Methode Klasse.
 - Erstelle eine JSP + Servlet Seite, die bei einer Eingabe von einer korrekten Username+Passwort Kombination die zum User gehörenden Comments strukturiert und formatiert ausgibt, bei fehlerhafter Kombi, wird eine Fehlermeldung ausgegeben.