

## Nonstandard Datenbanken

Wintersemester 2012/2013

### 3. Übungsblatt

#### Aufgabe 1: Servlet mit JDBC-Datenbankanbindung

Für dieses Übungsblatt wird die Inbetriebnahme des Webservers **Apache Tomcat 7.x** (<http://tomcat.apache.org>) vorausgesetzt. Des weiteren kommt zur einfacheren Handhabung die Entwicklungsumgebung Eclipse zum Einsatz. Folgen Sie dem Tutorial unter <http://www.coreservlets.com/Apache-Tomcat-Tutorial/tomcat-7-with-eclipse.html> zur Einrichtung eines entsprechenden Systems.

In der folgenden Aufgabe soll ein Servlet erstellt werden, welches über eine JDBC - Datenbankverbindung eine Datenbank abfragt und dynamische HTML-Seiten erzeugt. Nennen Sie die Datei DB.java bzw. DB.class und verwenden Sie kein Package, um die Korrektur zu vereinfachen.

Die bisher verwendete Musik-Datenbank soll weiterverwendet werden. Achten Sie darauf, dass Ihre Tabelle **Album** folgendem Schema entspricht: *Album(Albumtitel, Interpret, Erscheinungsjahr, Bewertung)*.

Damit ein Servlet auf eine HSQLDB-Datenbank zugreifen kann, muss der JDBC-Treiber für den Tomcat-Webserver zugreifbar sein. Kopieren Sie dafür die Datei `hsqldb.jar` (`lib`-Ordner von HSQLDB) in den Ordner `lib` Ihrer Tomcat-Installation.

Sollte Ihre Eclipse-Installation u.a. die Typen `HttpServlet` und `HttpServletRequest` nicht auflösen können, so fügen Sie die Bibliothek `javax.servlet-api.jar` aus dem `lib`-Verzeichnis Ihrer Tomcat-Installation dem Classpath hinzu. Dies können Sie unter anderem direkt in Ihrem Eclipse-Projekt unter *Properties* → *Java Build Path* → *Libraries* → *Add External JARs* festlegen.

Das Servlet soll HTML-Seiten generieren, die folgenden Funktionsumfang abdecken:

- a) **Anzeigen eines Hauptmenüs:** Das Hauptmenü wird angezeigt und bietet Links

zum Anzeigen aller Alben und zum Hinzufügen und Entfernen eines Albums.

(5 Punkte)

- b) **Alle Alben anzeigen:** Es werden alle Alben, die in der Datenbank vorliegen, angezeigt. Die Anzeige soll mit einer Tabelle optisch formatiert werden. Es gibt einen Link, um zum Hauptmenü zurückzugelangen.

(5 Punkte)

- c) **Album hinzufügen:** Ein Formular, in das der Benutzer die Angaben eines Albums eingeben kann, wird angezeigt. Über eine Schaltfläche werden die Einträge in die Datenbank übertragen. Es gibt einen Link, um zum Hauptmenü zurückzugelangen.

(5 Punkte)

- d) **Album entfernen:** Der Benutzer gibt den Namen des zu löschenden Albums an. Wenn das Album existiert, wird es aus der Datenbank entfernt. Dem Benutzer wird angezeigt, ob das Album gefunden und gelöscht wurde. Es gibt einen Link, um zum Hauptmenü zurückzugelangen.

(5 Punkte)

Wenn bei der Ausführung Fehler (Exceptions) auftreten, so sollen diese in der HTML-Seite mit ausgegeben werden. Das Schreiben in die Systemkonsole (`System.out`) bietet sich nicht an, da diese nicht vom Benutzer auf der Seite des Clients/Browser gelesen werden kann.

#### **Technischer Hinweis:**

Es sollen alle Aktionen in einem einzigen Servlet realisiert werden. Die einzelnen Aktionen, die z.T. durch HTML-Seiten repräsentiert werden, sollen durch Methoden gekapselt sein. Da dasselbe Servlet bei jeder Aktion aufgerufen wird, muss die gewünschte Aktion in einem Parameter kodiert sein. Dieser Parameter befindet sich entweder als unsichtbarer Parameter im Formular oder wird - wie im Hauptmenü - einem Link angefügt. Folgendes Beispiel zeigt, wie ein Link um einen Parameter erweitert werden kann: `http://localhost/DB?aktion=anzeigen` Diesen Parameter wertet das Servlet bei jedem Aufruf als erstes aus und verzweigt anschliessend in die passende Methode.

**Achtung:** Nennen Sie die Datei `DB.java` bzw `DB.class` und verwenden Sie kein Package, um die Korrektur zu vereinfachen. Senden Sie beide Dateien per Email an Ihren Übungsleiter, und geben Sie die Lösung zusätzlich in Papierform ab.

Senden Sie ihre Servlet-Quelldateien (`*.java`) und den Bytecode (`*.class`) per Email an Ihren Übungsleiter.

---

**Abgabetermin: Montag, den 5. November 2012 bis 14 Uhr im IFIS**