

Datenbanksysteme 2, 3. Übung

Einführung JDBC

Rekursive Abfragen mit CTEs

Generelles zu dieser Übung

Mit dieser Aufgabe sollen Sie sich mit JDBC vertraut machen und sicherstellen, dass Sie auf Ihrem Rechner eine passende Umgebung zum Entwickeln von JDBC-Programmen haben. In der letzten Aufgabe sollen Sie noch rekursive Abfragen mit CTEs üben.

Aufgabe 3.1 Datenbankzugriff, erstes JDBC-Programm

Ziel dieser Aufgabe ist es, das erste Beispiel aus der Vorlesung auf Ihrem Rechner laufen zu lassen. Dazu muss der Datenbankzugriff über einen SSH-Tunnel auf Ihrem Rechner eingerichtet sein, und Sie benötigen die passende Software (SQL Developer, JDBC-Treiber, Java-Installation, Eclipse).

Überprüfen Sie, ob Sie sich (noch) mit der Oracle 12 Datenbank der Abteilung Informatik verbinden können. Dazu müssen Sie einen SSH-Tunnel zum Datenbankserver aufgebaut haben. Entsprechende Informationen finden Sie in Moodle.

Verwenden Sie zum Zugriff auf die Datenbank den Oracle SQL Developer oder ein vergleichbares Werkzeug.

Verwenden Sie folgende Einstellungen:

- **Benutzername:** Ihre Netz-Kennung (z.B. xyz-i9a-t4)
- **Kennwort:** Ihr Oracle 12 Passwort
- **Hostname:** localhost (bei Verwendung eines Tunnels)
oder dboracleserv.inform.hs-hannover.de (im Pool)
- **Port:** 1521
- **SID:** db01

Sollten Sie Ihr Passwort nicht mehr kennen, wenden Sie sich bitte an Andreas Holitschke (andreas.holitschke@hs-hannover.de) oder kontaktieren Sie Ihren Dozenten in der Übung. Auch bei sonstigen Problemen mit dem SSH-Tunnel oder dem Datenbankzugriff wenden Sie sich bitte in der Übung an Ihren Dozenten!

Erstellen Sie jetzt ein JAVA-Projekt (z.B. mit Eclipse), um per JDBC auf die Datenbank zuzugreifen. Sie finden den Source-Code für ein einfaches Beispiel sowie den Oracle JDBC-Treiber auf Moodle. Dort finden Sie auch ein SQL-Skript, um die für das Beispiel benötigte Tabelle EMPLOYEE einzurichten.

Geben Sie im Programm (Klasse ConnectionManager) Ihren Benutzernamen und Ihr Passwort ein. Testen Sie das Programm. Es werden zunächst 2 Datenbank-Einträge eingefügt und dann wird der Inhalt der Tabelle EMPLOYEE auf dem Bildschirm ausgegeben.

Aufgabe 3.2: Erweiterungen des JDBC-Programms

Ziel dieser Aufgabe ist es, dass Sie Übung mit der Programmierung in JDBC bekommen. Dazu sollen Sie das Beispiel-Programm aus der Vorlesung erweitern.

Schreiben Sie zunächst eine Methode, die Datenbankeinträge wieder löscht. Die Methode soll als Parameter die ID des Employee bekommen, der gelöscht werden soll. Testen Sie die Methode, indem Sie den Eintrag mit der ID 2 wieder löschen (Aufruf aus der main-Methode).

Schreiben Sie dann eine Methode, die einen Eintrag verändert. Nutzen Sie die Methode, um "Meyer" in ein neues Department zu versetzen.

Aufgabe 3.3: ID-Vergabe mittels Sequenzen

Ändern Sie die insertEmployee-Method so, dass die ID aus eine Sequenz erzeugt wird. Die Sequenz müssen Sie dazu vorab über den SQL-Developer anlegen. Die Methode soll die ID des neu erzeugten Employees zurückgeben. Testen Sie die Methode.

Hinweis: Zur Lösung müssen Sie in insertEmployee insgesamt 2 Statements an die Datenbank absetzen. Im ersten Statement ermitteln Sie per "SELECT <sequenz>.nextval FROM dual" den nächsten Wert der Sequenz. Das zweite Statement ist dann der eigentliche Insert, der die ermittelte ID verwendet.

Aufgabe 3.4: Ressourcenfreigabe

In der Methode showEmployees fehlt die Ressourcenfreigabe. Testen Sie die Auswirkungen: schreiben Sie eine Schleife, die 2000 mal die Methode "showEmployees" aufruft. Wann tritt ein Fehler auf?

Beheben Sie den Fehler, indem Sie eine korrekte Freigabe der Ressourcen (PreparedStatement, ResultSet) implementieren.

Aufgabe 3.5: Rekursive Abfragen mit CTEs

Erstellen Sie analog zum Beispiel aus der Vorlesung zwei rekursive Abfragen mithilfe von CTEs:

- a) Alle Filme, die von der "Rocky Horror Picture Show" (ID 959170) parodiert werden, sowie alle von diesen Filmen parodierten Filme, usw.
- b) Alle Filme, welche die "Rocky Horror Picture Show" (ID 959170) parodieren, sowie alle Filme, die dieser Filme parodieren, usw.