

Dokumentation JDBC

Helmuth Brunner 4ahit, 2013

Inhaltsverzeichnis

Aufgabenstellung.....	3
Abreitsaufwand.....	5
Technologiebeschreibung.....	6
Testfälle.....	7

Aufgabenstellung

Erstelle eine Anwendung in Java im Package "deinnachname" (bitte durch Nachnamen ersetzen), die es Benutzern erlaubt, beliebige Datentabellen zu durchsuchen.

Dein Programm soll Kommandozeilenparameter für den Server, den Benutzernamen, das Passwort und die zu verwendende Datenbank verarbeiten - d.h. dein Programm **muss** sich wie folgt aufrufen lassen:

```
java deinnachname.jdbcuel -h projekte.tgm.ac.at -u  
benutzer -p password -d datenbankname
```

Daraufhin soll eine Swing-Eingabemaske angezeigt werden. Diese enthält die gezeigten Eingabefelder vorausgefüllt mit den Werten aus den Parametern, sowie für einen SQL-Befehl und eine noch leere Tabelle (siehe dazu auch das Interface TableModel und seine implementierenden Klassen). Der Befehl wird von deinem Programm an das DBMS geschickt werden und die Ergebnisse in einer Tabelle angezeigt.

The screenshot shows a Java Swing window titled "Countries". It contains several input fields and a table. The fields are labeled "Server", "Datenbank", "Benutzername", "Passwort", and "SQL-Befehl". The table has two columns, both labeled "Head", and three rows of "Cell" entries. The bottom row of the table is highlighted in blue.

Head	Head
Cell	Cell
Cell	Cell
Cell	Cell

Achtung: damit eine Datenmanipulation ausgeschlossen wird, muss dein Programm den eingegebenen SQL-Befehl prüfen: es darf nur EIN Befehl sein
es dürfen nur SELECT, SHOW oder DESCRIBE-Befehle verwendet werden INSERT, DELETE, CREATE ... müssen von deinem Programm mit einer sinnvollen Fehlermeldung abgelehnt werden.

Strichpunkt vorhanden?

Liefert das RDBMS eine Fehlermeldung zurück, muss diese sorgfältig verarbeitet und dem Benutzer eine entsprechende Meldung gezeigt werden.

Alle, die diese (oder eine ähnliche) Aufgabe im letzten Jahr schon gemacht haben:

Erstelle eine GUI, die zusätzlich zu den Verbindungsparametern (Kommandozeile, Felder wie oben) ein Feld für eine Combobox für alle in der DB vorhandenen Tabellen zeigt. Wird eine vom Benutzer ausgewählt, zeigt euer Interface jeweils ein Feld für alle vorhandenen Attribute in dieser Tabelle. Gibt der Benutzer dort Daten ein, werden diese in einem neuen Datensatz in die Datenbank gespeichert.[1]

Abreitsaufwand

Arbeitspacke	geschätzt Zeit	Tatsächliche Zeit
CLI	5 min	5 min
GUI	1 Stunde	3 Stunden
mysql-Connector	2 Stunden	2 Stunden
Fehlermeldung/Benutzerfreundliche Fehlermeldungen	1 Stunde	2 Stunden

Habe sehr viel Zeit in den Connector gesteckt weil ich mich am Anfang mit dieser Methode der Datenabfragen gar nicht bis wenig zu recht gefunden habe. Weiters habe ich für die JTable auch einiges an Zeit verloren sowie das Auslesen der Daten aus dem Result-Set und das diese auch schön in der Tabelle ausgegeben werden.

Technologiebeschreibung

JDBC (Java Database Connector) ermöglicht Java Programmen auf eine SQL-Datenbank zugreifen zu können.

Der Vorteil von JDBC ist dieser das man viele SQL-Datenbanken, sprich mysql oder psql Datenbanken ansprechen kann.

Testfälle

Testfälle die ich durchgeführt habe:

- Ausführen des Programms über die Kommandozeile: Funktioniert.
- Ausführen des Programms über die Kommandozeile bei nicht korrekten Parametern. Ausgabe einer Fehlermeldung was falsch ist. Parameter können neu gesetzt werden in folge auf den „Connect-Button“ klicken. Diesen Schritt so lange wiederholen bis keine Fehlermeldungen mehr angezeigt werden.
- Eingabe von SQL-Befehlen: select, show und describe Befehle werden als gültige Befehle angesehen und können mit der Richtigen SQL-Syntax ausgeführt werden.
- Eingabe von SQL-Befehlen: insert, drop, delete, update und alter Befehle werden als nicht gültige Befehle angesehen. Ausgabe eines Hinweises das dies nicht möglich ist.
- SQL-Befehle die mit einem Strichpunkt enden werden nicht ausgeführt. Ausgabe einer Meldung die den Benutzer informiert keinen Strichpunkt anzuhängen.

Quellen:

[1]: Angabe von elearning.tgm.ac.at, Aufgabe: „JDBC:Daten abfragen“