

Zugriff auf SQL-Datenbanken mit Java

Zugriff auf eine vorhandene Datenbank (z.B. Nordwind DB, Tabelle ARTIKEL)

Wenn Java auf eine Datenbank zugreifen will, wird das Paket JDBC benötigt. Darin ist der notwendige JDBC-ODBC Treiber enthalten, mit dem man auf die meisten Datenbanken zugreifen kann. (Es gibt insgesamt 4 Klassen für Datenbanktreiber (Class 1-4)).

Der Treiber-Manager im JDBC regelt den Zugriff von Java auf die Datenbank. Die Art der Datenbank spielt dabei für den Java-Programmierer keine Rolle

Damit für die Kommunikation mit einer Access-Datenbank Access nicht mehr benötigt wird, muss die Datenbank im ODBC-Manager angemeldet werden.

1. ODBC-Treiber registrieren

Dazu aufrufen: **Systemsteuerung – Verwaltung – ODBC-Datenquellen**

Benutzer-DSN – Hinzufügen . Dann Microsoft-Access-Treiber auswählen.

Der Datenquelle einen beliebigen Namen geben, unter dem sie später in Java angesprochen wird. Z.B. **mydb**. Dann **Auswählen** und die entsprechende Datenbank auswählen. Alles mit OK bestätigen.

2. Das Java-Programm schreiben:

```
import java.sql.*;
```

Laden des Datenbanktreibers:

```
Class.forName(“sun.jdbc.odbc.JdbcOdbcDriver“);
```

Exception mit try und catch abfangen

3. Verbindung zur Datenbank herstellen

Dazu wir ein Object der Klasse Connection benötigt.

```
Connection conn;
```

```
conn=DriverManager.getConnection(“jdbc:odbc:mydb“);
```

allgemein: DriverManager.getConnection(url, user, password)

url ist immer wie folgt aufgebaut: jdbc:odbc:Datenquelle

4. Eine SQL-Anfrage absetzen

Um Anfragen zu stellen, muss ein Objekt der Klasse Statement erzeugt werden.

```
Statement stm = conn.createStatement();
```

Das Ergebnis der Anfrage wird einem Objekt des Interface ResultSet zugewiesen:

```
ResultSet rs=stm.executeQuery(“Select * from Artikel“); oder
```

```
ResultSet rs=stm.executeQuery(“Select Artnr, Artbez from Artikel WHERE Artnr >100“);
```

Artikel sei hier eine Tabelle aus der Datenbank, die unter mydb angesprochen wurde.

5. Das Ergebnis der SQL-Anfrage auswerten

Die Ergebnistabelle wurde in der Variablen rs gespeichert und kann nun ausgewertet werden.

Zum Auswerten kann der Datenbankzeiger verwendet werden. Es gibt:

rs.next(); nächster Datensatz
rs.previous() vorheriger Datensatz
rs.first() erster Datensatz
rs.last() letzter Datensatz

Ausgabe: Gesamte Tabelle mit der **getString-Methode**:

```
while(rs.next( ) )  
    System.out.println(rs.getString(Artnr)+ " "+rs.getString(Artbez));
```

Parameter der getString-Methode ist entweder die Spaltenüberschrift über der Spaltenindex.

Bei der get-String-Methode wird das Ergebnis in einen String umgewandelt:

Für andere Datentypen gibt es andere Methoden wie:

getInt(); getDouble(), getDate(), getBoolean() usw.

Update einer Datenbank:

Connection c:

PreparedStatement ps= c.prepareStatement("Insert into Artikel Values(1010, 'Bügeleisen');

ps.setInt(1,1010); setzt Spalte 1 auf den Integerwert 1010

ps.setString(2,"Bügeleisen");

ps.executeUpdate();

Beispiel:

```
import java.sql.*; //SQL-Treiber laden  
public class jdb100  
{  
    public static void main(String[] args)  
    {  
        try {  
            //Installation des odbc-Treibers  
            Class.forName("sun.jdbc.odbc.JdbcOdbcDriver");  
            System.out.println("Treiber geladen");  
        }  
        catch (ClassNotFoundException e)  
        {  
            System.out.println("Treiber nicht gefunden");  
        }  
        try {  
            Connection conn;  
            conn = DriverManager.getConnection("jdbc:odbc:kunden");  
            System.out.println("Datenbank gefunden");  
            //eine Abfrage auf die tabelle KUNDEN erzeugen  
            Statement stm=conn.createStatement();  
            ResultSet rs = stm.executeQuery("Select * From Kunden"); //eine Tabelle aus der Datenbank  
            while(rs.next())  
            {  
                System.out.println(rs.getString(1)+" "+rs.getString(2)+" "+rs.getString(3));  
            }  
        }  
        catch (SQLException ex)  
        {  
            System.out.println("Fehler");  
        }  
    }  
}
```

Zugriff auf SQL.doc