

12 Datenbank-Programmierung

12.1 JDBC – Datenbankzugriff in Java

Installation des JDBC Treibers

Laden Sie sich den Datenbanktreiber `mysql-connector-java-5.1.30-bin.jar` von der Kurshomepage herunter¹. Kopieren Sie die Datei in einen beliebigen Ordner auf Ihrem Computer.

Für jedes Projekt, in dem Sie den Datenbanktreiber verwenden wollen, öffnen Sie in Eclipse den Menüpunkt Project/Properties. Wählen Sie "Java Build Path". Ergänzen Sie den Build-Path unter "Libraries" mit "Add External JARs" um das heruntergeladene jar File. Damit ist der JDBC-Treiber für das Projekt zugreifbar.

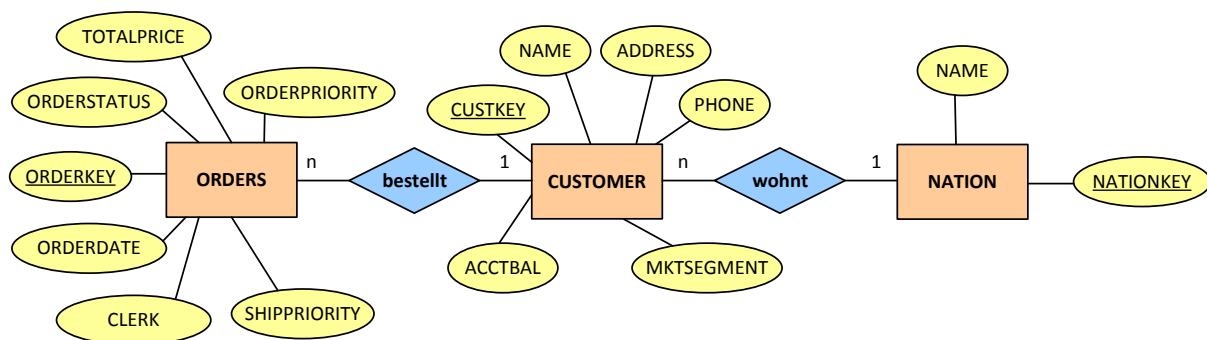
Wenn obige Schritte erfolgreich durchgeführt wurden, so muss folgender Code

```
1  try{
2      Class.forName("com.mysql.jdbc.Driver");
3  } catch (ClassNotFoundException e) {
4      e.printStackTrace();
5  }
```

in Ihrem Projekt ohne Auftreten einer Exception durchlaufen. Sie können den Code der Vorlesung, welcher ja, wie immer, online verfügbar ist, für Ihre Übung als Template verwenden.

Query

Wir betrachten erneut das Schema aus Aufgabe 10.3.2:



Wir betrachten folgende Query:

"Wie ist der Name des Landes, für welches der Wert der nicht ausgelieferter Bestellungen maximal ist?"

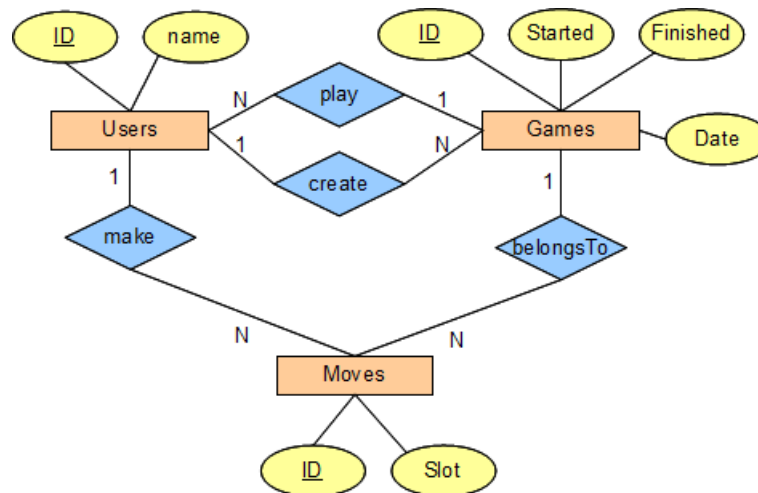
Implementieren Sie die Query unter Verwendung des MySQL Datenbankservers, den Sie in Aufgabe 10.3 eingerichtet haben. Die Verbindung zum Datenbankserver wird dabei über das JDBC Interface hergestellt. Das Resultat soll auf der Konsole erscheinen.

¹Alternativ beziehen Sie den Datenbanktreiber von <http://dev.mysql.com/downloads/connector/j/> ("Platform Independent") und extrahieren aus der erhaltenen Datei (tar oder zip) das genannte jar-File

12.2 Vier Gewinn

In dieser Übung geht es um die gemeinsame Nutzung einer Datenbank durch mehrere Nutzer. Wir haben ein kleines Spiel ("Vier Gewinn") in Java implementiert. Dieses Spiel benutzt eine Datenbank, welche wir auch vorgängig eingerichtet haben. Es ist konzeptuell möglich, dass beliebig viele Spieler gleichzeitig spielen, wobei Gruppen von maximal 8 Spielern gegeneinander spielen können.

Das folgende Schema modelliert die Zusammenhänge des Spieles.



(a) Konstruieren Sie, den Regeln aus der Vorlesung folgend, zuerst ein relationales Modell zu obigem Schema.

(b) Laden Sie sich das Java-Projekt des Spiels von der Kurshomepage herunter. Es sollte möglich sein, das Projekt zu kompilieren und das Spiel zu starten. Wir haben aber die Datenbankabfragen aus dem Spiel entfernt, d.h. Sie können das Spiel *noch nicht* wirklich spielen.

(c) Verschaffen Sie sich zuerst einen Überblick über die Implementation des Spieles, indem Sie das Klassendiagramm zum Java-Programm VierGewinnt zeichnen. Hinweis: jede java-Datei im Projekt enthält eine eigene Klasse.

(d) Fügen Sie dann die nötigen SQL-Anweisungen in der Klasse DataBase.java ein. Wir haben alle Stellen, an denen eine solche Anweisung vorkommt mit einem "STUDENT" Kommentar versehen. Ausserdem haben wir die jeweiligen Anweisungen in natürlicher Sprache formuliert, wobei wir auf Variablen oder Parameter aus Java mit eckigen Klammern Bezug nehmen, z.B. so: <gameID>. Um Ihnen einen Eindruck zu vermitteln, geben wir nachfolgend Beispiele solcher Sätze hier wieder:

- a) Query: Zeilen der Tabelle Users mit Name=<name>
- b) Einfügen: neue Zeile (Started, Finished, Creator) in Tabelle Games mit Werten ("0", "0", <userId>)
- c) Update: Spalte GameID neuer Wert <gameID> in Tabelle Users der Zeilen mit ID = <userId>

Hinweise:

- In MySQL werden "0" und "1" für die Boolean-Werte true und false von Java verwendet.
- Werte werden in MySQL mit doppelten Hochkommas " eingefasst.
- Um MySQL-Strings in Java-Strings darstellen zu können, werden doppelte Hochkommas in Java Strings mittels Backslash maskiert, also mit \" anstelle von " dargestellt.

(e) Spielen Sie gegeneinander. Viel Spass!