

7. Übungsblatt

(Testwoche: 1. - 3. Juni 2010)

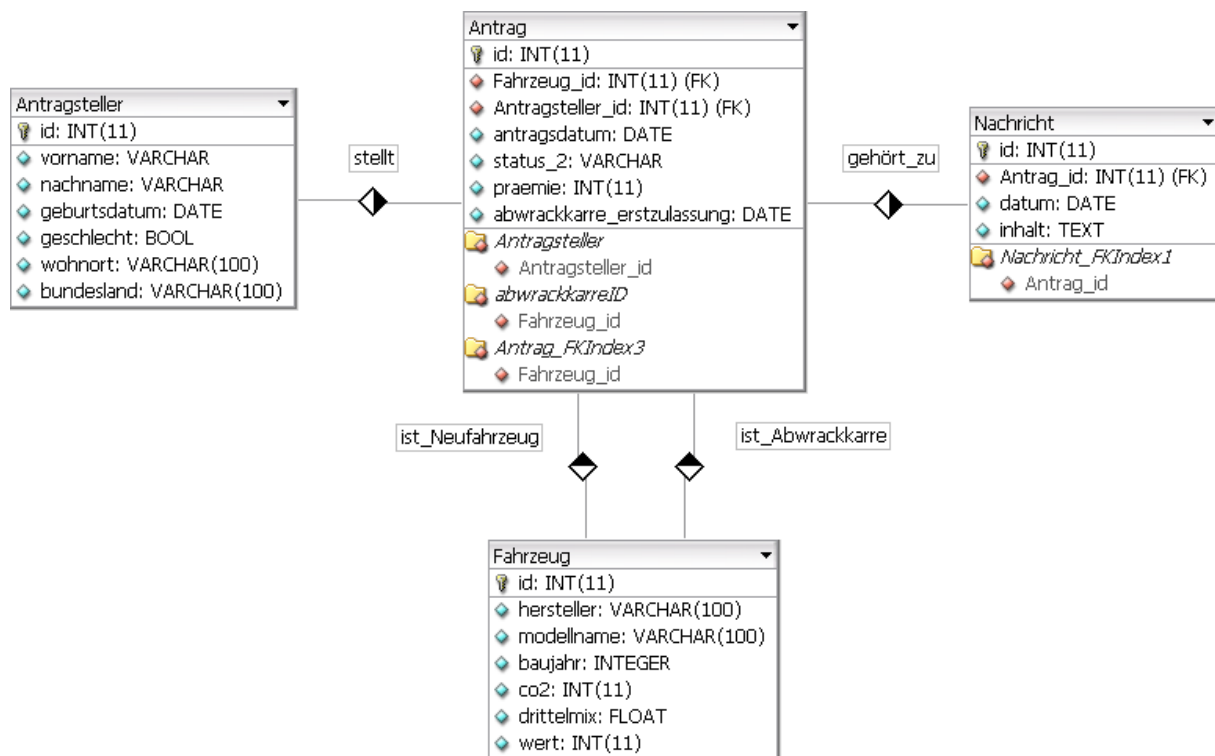
Einführung in Datenbanksysteme
Datenbanken für die Bioinformatik

Heinz Schweppe, Katharina Hahn

Aufgabe 1 (Funktionen, Trigger)

3+4+4 Punkte

Im Jahr 2009 wurde die Umweltprämie („Abwrackprämie“) im Rahmen des Konjunkturpakets II in Deutschland eingeführt. Käufer konnten bei Kauf eines Neuwagens und gleichzeitiger Verschrottung ihres alten PKWs die Prämie beantragen. Im Jahr nach der „Abwrackprämie“ soll nun Bilanz gezogen werden. Betrachten Sie dazu das folgende Schema für die „Abwrackdatenbank“.



SQL-Anweisungen zum Erzeugen des Schemas stehen Ihnen sowohl für Oracle als auch Postgres auf der Übungsseite zur Verfügung. Schreiben Sie folgende Funktionen in PL/SQL bzw. PL/pgSQL:

- Die Funktion *CO2BILANZ(NUMBER konstante)* → *NUMBER* berechnet die CO₂-Bilanz der Umweltprämie. Dazu wird die Differenz des CO₂-Ausstoß zwischen allen durch die Umweltprämie geförderten Neufahrzeugen und allen „Abwrackkarren“ bestimmt. Für die frühzeitige Herstellung eines Neufahrzeuges wird die Konstante *konstante* als zusätzlicher CO₂-Ausstoß eingerechnet. Die Differenz unter Berücksichtigung der Konstante wird zurückgegeben.
- Die Kosten (ggf. der Gewinn), die der Regierung durch die Gewährung der Umweltprämie entstanden sind, sollen durch die Funktion *KOSTEN(INTEGER*

steuer_vorteil_pauschal, *FLOAT* *verbrauch_schwellwert*) → *VARCHAR* berechnet werden. Die Kosten für den Staat sind bei jedem gewährten Antrag die Prämie. Gewinne ergeben sich aus der erhaltenen Mehrwertsteuer beim Verkauf eines Neuwagens (19% des Wertes des Neuwagens). Liegt der Verbrauch (*drittelmix*) des Neuwagens unter dem angegebenen Schwellwert (*verbrauch_schwellwert*), müssen zusätzlich als Kosten noch KFZ-Steuer ausfälle von *steuer_vorteil_pauschal* addiert werden. Sind dem Staat Kosten entstanden, so gibt die Funktion aus: „Durch die Gewährung der Prämie sind Kosten von ??? € entstanden.“ Ansonsten gibt die Funktion aus: „Der Staat hat durch die Abwrackprämie einen Gewinn von sage und schreibe ??? € gemacht.“

Damit Antragsteller sich über den Status ihres Antrags informieren können, werden Nachrichten in die Tabelle *Nachricht* geschrieben. Schreiben Sie einen Trigger, der die folgende Aufgabe erfüllt:

- c) Wenn ein Sachbearbeiter einen Antrag bearbeitet, ändert er aus Gründen der Nachvollziehbarkeit den *status* des Antrags. Ein Antrag kann folgende Status haben: „eingegangen“, „in Bearbeitung“, „abgelehnt“, „wird gewährt“, „überwiesen“. Schreiben Sie einen Trigger, der bei der Änderung des Status Antrags eine Nachricht in die *Nachrichten* Tabelle schreibt. Die Nachricht soll einen sinnvollen, natürlichsprachlichen Text ihrer Wahl enthalten, der den Status erläutert (z.B. „Ihr Antrag ist in Bearbeitung und wird in Kürze erledigt.“).

Aufgabe 2 (JDBC)

1+3+4 Punkte

- a) Was versteht man unter dem *Impedance Mismatch*?

Schreiben Sie ein Anwendungsprogramm, das über JDBC auf die *Mondial* Datenbank auf *esel* zugreift. Laden Sie sich dazu das Archiv *ue07.jar* von der Übungsseite runter. Vergessen Sie nicht, den JDBC-Treiber des von Ihnen gewählten Datenbanksystems dem Klassenpfad hinzuzufügen. Die Klasse *BaseDAO* enthält bereits Methoden, um eine Verbindung zu einer Datenbank herzustellen bzw. zu schließen. Wählen Sie die URL und den Klassennamen entsprechend des gewählten Datenbanksystems aus. Studieren Sie, wie diese zusammengesetzt sind. Schreiben Sie Ihre eigene(n) Klasse(n), die folgende Methoden enthält (enthalten). Verwenden Sie dabei *getConnection()* und *closeConnection()* der *BaseDAO*.

- b) Implementieren Sie die Methode *void printAllNeighbors(String country)*, die den Namen eines Landes als Eingabe erhält und die Namen aller Nachbarn alphabetisch sortiert ausgibt. Verwenden Sie dazu nicht mehr als eine SQL-Anfrage.
- c) Implementieren Sie die Methode *List<City> listBigCities(String country, int threshold)*, die die Funktionen *BIG_CITIES* der Aufgabe 3 des Übungsblatts 6 aufruft. Implementieren Sie dazu eine entsprechende Klasse *City*. Geben Sie die von der Funktion zurückgegebenen Werte zeilenweise aus.

Informationen zur Verwendung der JDBC Schnittstelle zum Ausführen von SQL-Anfragen und Handhabung der Rückgabe finden Sie in den Dokumentationen der Datenbanksysteme, z.B. <http://www.postgresql.org/files/documentation/books/pghandbuch/html/jdbc.html>.

Aufgabe 3 (Indexstrukturen, B-Bäume)

3+1+1+2 Punkte

- a) Erstellen Sie einen B-Baum mit maximaler Füllgröße $N=4$. Fügen Sie nacheinander in den anfänglich leeren Baum Elemente mit den folgenden Schlüsseln in der angegebenen Reihenfolge ein: 2, 3, 5, 7, 11, 17, 19, 23, 29, 31. Stellen Sie den resultierenden Baum grafisch dar.
- b) Ist der von Ihnen erstellte B-Baum der einzig mögliche für die enthaltenen Schlüsselwerte? Begründen Sie Ihre Antwort.
- c) Das Einfügen wie vieler Elemente in den Baum führt dazu, dass sich die Höhe um eins erhöht? Begründen Sie ihre Antwort. Geben Sie ein untere Schranke für die Anzahl der Elemente an.
- d) Löschen Sie das Element mit dem Schlüsselwert 3 aus dem Baum, den sie in Teilaufgabe a) erstellt haben. Geben Sie den resultierenden Baum an.