

Aufgabe 1

Worin besteht der Mehraufwand des kostenbasierten Optimizers gegenüber dem regelbasierten? Warum werden heute trotzdem kostenbasierte Optimizer eingesetzt?

Aufgabe 2

Der folgende Befehl gibt alle Auftragspositionen aus, die der Kunde *Fahrrad Shop* in Auftrag gegeben hat:

```
SELECT A.Bezeichnung, AP.Anzahl, Datum
FROM  Auftrag NATURAL INNER JOIN Auftragsposten AP
      INNER JOIN KUNDE K  ON Nr=Kundnr
      INNER JOIN Artikel A ON Anr=Artnr
WHERE Trim(K.Name) = 'Fahrrad Shop' ;
```

Wie ändert der Optimizer diesen Befehl ab? Geben Sie den geänderten Befehl an.

Aufgabe 3

Versehen Sie das Attribut *Bezeichnung* der Relation *Artikel* aus der Beispieldatenbank *Bike* mit einem Index. Ist ein eindeutiger Index mittels des Bezeichners *Unique* sinnvoll?

Aufgabe 4

In einem großen Dienstleistungsunternehmen werden zentral Aufträge entgegengenommen und an Sachbearbeiter weitergeleitet, deren Einsatzort in der Nähe des Kunden liegt. Dazu existiere in der Datenbank eine Relation *Personal*, die die benötigten Daten der Sachbearbeiter speichert, insbesondere auch den Namen und den Einsatzort. Wegen einer großen Anzahl von Sachbearbeitern ist die Suche nach möglichen Sachbearbeitern in der Nähe des Kunden zeitaufwendig. Schreiben Sie einen Index, um die alphabetische Suche nach Sachbearbeitern an bestimmten Einsatzorten zu beschleunigen.

Aufgabe 5

Die Range-Partitionierung ist sehr weit verbreitet. Welche zwei großen Vorteile bietet diese Partitionierung bei sehr großen Relationen?

Worin liegt der Vorteil einer Hash-Partitionierung?

Aufgabe 6

Worin liegt der große Vorteil der Referenzpartitionierung? Wie könnte diese Partitionierung in SQL Server und MySQL nachgebildet werden?

Aufgabe 7

Materialisierte Sichten müssen aktuell gehalten werden. Dies bedingt einen hohen Aufwand. Wo werden daher diese Sichten fast ausschließlich eingesetzt und warum?

Aufgabe 8

Der Optimizer änderte den Befehl in Aufgabe 2 ab. Warum wohl?

Aufgabe 9

Wann ist ein Merge-Join gegenüber Nested-Loop-Join und Hash-Join von Vorteil?

Aufgabe 10

Warum beeinflusst eine Group-By-Klausel die Performance so negativ? Warum kann in der Regel ein existierender Index auf das zu gruppierende Attribut nicht verwendet werden?

Aufgabe 11

Schreiben Sie eine Stored Procedure *Preisnachlass*, die den Preis aller Artikel um den als Parameter angegebenen Prozentsatz senkt. Dies gilt auch für die Preise aller Aufträge. Die Prozedur überprüft, dass der Nachlass nicht größer als 10 Prozentpunkte beträgt.

Aufgabe 12

Unsere Firma *Bike* hätte einen sehr großen Lieferantenstamm. Gesucht werden häufig Lieferanten, die bestimmte Artikel liefern. Nacheinander sollen dann zu einer Liste von Artikeln die entsprechenden Lieferanten angezeigt werden. Ziel ist es, dass der Parser nur so oft wie nötig ausgeführt wird. Wie könnte ein dazugehöriger PHP-Programmausschnitt aussehen?