|  |  |  |  |
| --- | --- | --- | --- |
| **Übungsprotokoll**  **INSY** | | | |
|  | **Übungsdatum:**  KW 36 – 02 | **Klasse:**  3AHIT | **Name:**  Felix Schneider |
| **Abgabedatum:**  13.01.2022 | **Gruppe:**  INSY | **Note:** |
| **Leitung:**  DI (FH) Alexander MESTL | **Mitübende:**  **Clemens Schlipfinger** | | |
| **Übungsbezeichnung**:  Arbeiten mit SQL | | | |

**Inhaltsverzeichnis:**

[1 Aufgabenstellung 2](#_Toc93518524)

[2 Theoretische Grundlagen 2](#_Toc93518525)

[3 Übungsdurchführung 2](#_Toc93518526)

[3.1 Einloggen, Erstellen, Loslegen 2](#_Toc93518527)

[3.2 select und zugehörige Klausel (viele) 4](#_Toc93518528)

[3.2.1 order by, concat(), where 5](#_Toc93518529)

[3.2.2 like, not, in, between, limit 6](#_Toc93518530)

[3.2.3 join 7](#_Toc93518531)

[3.2.3.1 Gedankenexperiment 8](#_Toc93518532)

[3.2.4 count, coalesce, min, max 9](#_Toc93518533)

[3.2.5 group by, having(, max, min ), avg 10](#_Toc93518534)

[3.2.6 Kombinationen 12](#_Toc93518535)

[3.2.7 Zusatzaufgabe für die treuen LeserInnen 14](#_Toc93518536)

[4 Kommentar 14](#_Toc93518537)

# Aufgabenstellung

SQL-Beispiele aus dem Aufgabendokument mit der angefügten Datenbank durchführen.

# Theoretische Grundlagen

Als Datenbankprogramm wird maria-db verwendet.

# Übungsdurchführung

## Einloggen, Erstellen, Loslegen

Damit man sich bei der Datenbank einloggen und loslegen kann, benötigt man folgenden Befehl und das Passwort von einem Benutzer, der die richtigen Rechte hat, um mit dieser Datenbank auch sinnvoll arbeiten zu können (notfalls einfach root):

-u gibt den Benutzer an, mithilfe dessen man sich einloggen möchte.

-p gibt an, dass ein Passwort benötigt wird.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Mit „show databases;“ kann man sich alle Datenbanken anzeigen lassen.Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

Mit „show tables;“ kann man sich alle Tabellen einer Datenbank anzeigen lassen. Damit dies funktioniert muss auch eine Datenbank ausgewählt sein (folgt unten!).

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

Damit man eine neue leere Datenbank erstellt, verwendet man „create database <Name>;“.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Wie oben angekündigt, kann man mit „use <Name>;“ die Datenbank auswählen.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Um eine Datenbank zu löschen, braucht man „drop database <Name>;“.



Mit dem „source <Datei>;“-Befehl kann man sich eine Datenbank importieren (diese kann bereits vollkommen funktionsfähig sein, sprich nicht leer). Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Um sich alle Spalten einer Tabelle einer Datenbank anzeigen zu lassen, muss man 1. die Datenbank usen und 2. diesen Befehl ausführen (EMPLOYEES ist hier die Datenbank): Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

## select und zugehörige Klausel (viele)

Mit dem select-Befehl kann man sich jegliche Daten von einer Datenbank ausgeben lassen, die man will. Dieser Befehl gehört zur DQL (Data Query Language) und umfasst die meisten Klausel der gesamten SQL (denke ich).Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

### order by, concat(), where

Mithilfe der „order by“ Klausel kann man sich die Ausgabe nach einer oder mehreren Kriterien sortieren lassen (Zahlen oder Alphabet).Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

Man kann absteigend und aufsteigend sortieren. Standardmäßig sortiert man immer aufsteigend (0-9, A-Z), wenn man dies erzwingen will, schreibt man ASC hinter die „order by“-Bedingung. Wenn man absteigend sortieren will (9-0, Z-A), schreibt man DESC dahinter.

Ein Bild, das Text enthält.

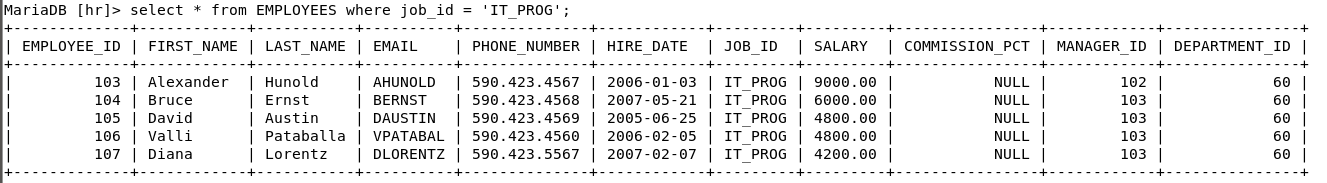
Automatisch generierte Beschreibung

Mit der „concat()“-Methode kann man sich Daten in einen String zusammenfassen. Mit dieser Technik könnte man auch sinnvolle Sätze bilden.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

„Das „where“ in SQL ist wie das „if“ in C#“ sagte einmal ein weiser Mann (nicht ich 😊). Sprich: Mithilfe der „where“ Klausel kann man eine Bedingung festlegen. Zum Beispiel kann man nur Employees ausgeben, die in den Programmierer-Abteil arbeiten gehen:



Natürlich kann man auch mehrere Bedingungen festlegen. Genau wie in C# gibt es && 🡪 AND und || 🡪 OR.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Diesen Befehl sollten Sie nun bereits interpretieren können:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### like, not, in, between, limit

Mithilfe von „like“ können Sie nach bestimmten Mustern mit der „where“-Klausel suchen. Das Prozentzeichen steht hierbei für beliebig viele Buchstaben oder Zahlen.

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

NOT verneint eine Klausel. Ob NOT NOT die Klausel nicht verneint? 😊

Ein Bild, das Text enthält.

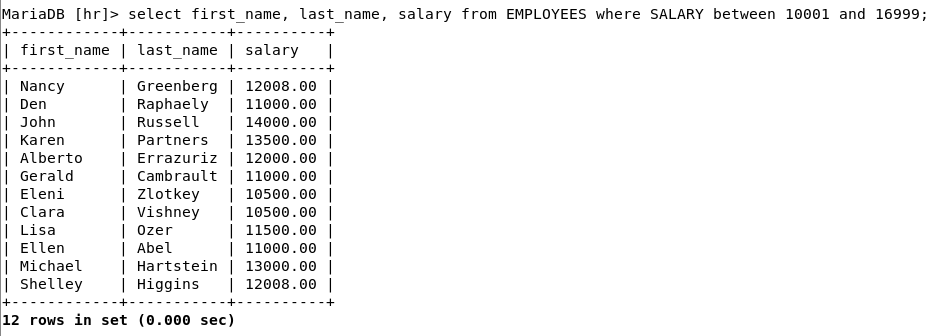
Automatisch generierte Beschreibung

Mithilfe von IN können Sie nach verschiedenen Zeichenketten filtern. Sozusagen ein „where“ mit ganz vielen AND nur kürzer!

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

BETWEEN – AND schreibt nur Zahlen die zwischen (inklusive, inklusive) einem Bereich sind. Dass die beiden angegebenen Zahlen inklusive sind, macht laut Definition der Wörter between und and eigentlich keinen Sinn. Jedoch macht es ITlern den Job wahrscheinlich einfacher, weil man nicht 999999 sondern 1000000 schreiben muss…



Mit LIMIT kann man einen Maximalanzahl an Reihen des Outputs festlegen (in manchen Datenbankprogrammen heißt es FETCH).

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### join

Der INNER JOIN fügt zwei Tabellen nach einer Bedingung zusammen, sodass man auf Attribute von beiden Tabellen zugreifen kann.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Der NATURAL JOIN ist wie der INNER JOIN, nur dass man die ON Bedingung weglasst, weil man einfach die Schlüssel vergleicht(, was in den meisten Fällen am meisten Sinn macht).

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Man kann auch geJOINte Tabellen JOINen (REKURSION)

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

#### Gedankenexperiment

Wenn man obriges (letztes) Beispiel ohne der „where“-Klausel ausführt, werden insgesamt 106 Reihen, sprich Employees ausgegeben.



Insgesamt gibt es allerdings 107 Employees…



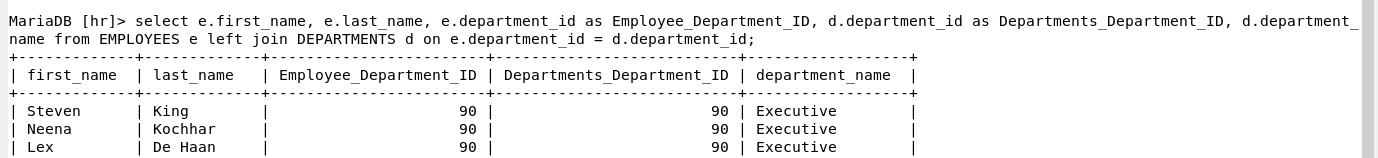
Warum werden bei dem JOIN also nur 106 Employees ausgegeben?

Antwort: Ein Employee hat keine Department\_id, weil dieser Employee wahrscheinlich Chef ist, weshalb der INNER JOIN diesen Employee nicht mit in die Ausgabe packt.

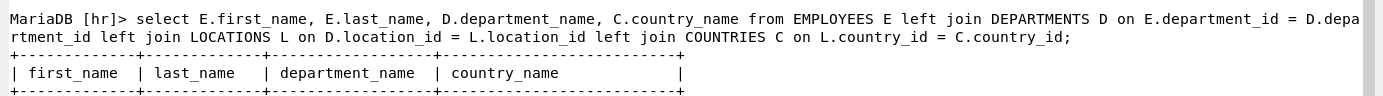
Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

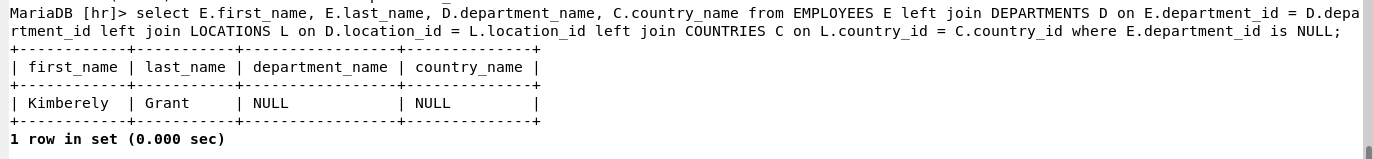
Ein LEFT JOIN nimmt alle Werte der linken Tabelle (auch wenn diese NULL sind) und vergleicht diese mit den Werten der rechten Tabelle.











Ein RIGHT JOIN funktioniert genau anders herum. Lustig ist zu erwähnen, dass LEFT und RIGHT wirklich links und rechts entsprechen, nicht 1 und 2 oder so…

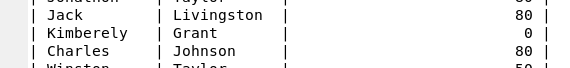
### count, coalesce, min, max

Mit COUNT() kann man Einträge zählen.Ein Bild, das Text enthält.

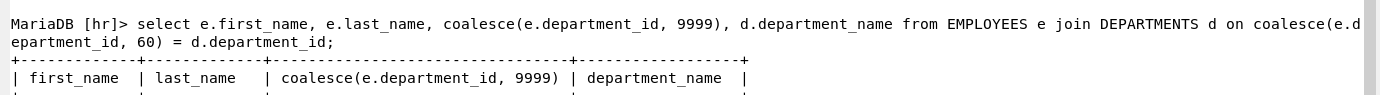
Automatisch generierte Beschreibung

Mittels COALESCE() kann man NULLWERTE ersetzen, sodass immer die angegebene Zahl dortsteht.





Dies betrifft allerdings NUR die Ausgabe:





Mit MIN() bzw. MAX() kann man das Minimum bzw. das Maximum einer Zahl ermitteln.

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### group by, having(, max, min ), avg

Mit GROUP BY kann man bestimmte Einträge gruppieren, sodass man zum Beispiel den maximalen bzw. minimalen Gehalt in jeder Abteilung berechnen kann:

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

Mittels AVG kann man den Durchschnitt einer Zahl berechnen:

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

„Having ist das Where von Group By” sagte einmal ein weiser Mann (diesmal wars unser Herr Professor…). Diese Aussage beschreibt Having so einfach, wie es ist. Will man nur bestimmte Einträge gruppieren, so kann man dies mit HAVING. SQL benötigt hier wahrscheinlich einen anderen Namen/Befehl, damit es unterscheiden kann, ob man nun die Gruppierung oder die Ausgabe beschränken will.

Ein Bild, das Text enthält.

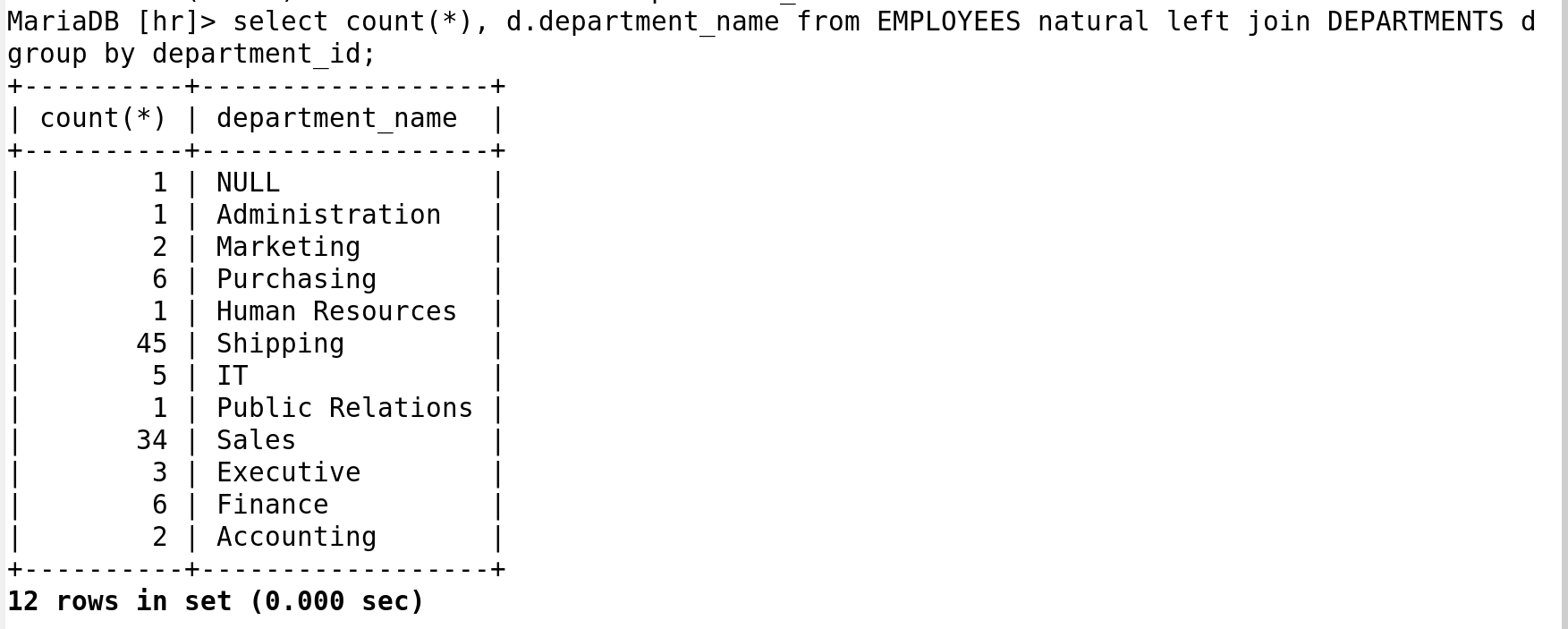
Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### Kombinationen

Um die Anzahl an Mitarbeiter je Abteilung auszugeben, muss man COUNT() und JOIN verwenden.



Manchmal ist es notwendig einen SELECT Befehl in einem SELECT Befehl durchzuführen, wenn nicht sogar rekursiv noch mehr SELECT Befehle (komplexere Abfragen). Diese Art von Befehl nennt man dann SUBSELECT:

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

Das Schlüsselwort DISTINCT schließt doppelte Einträge aus.

Ein Bild, das Tisch enthält.

Automatisch generierte Beschreibung

Alle Befehle lassen sich miteinander kombinieren, um jedes mögliche Ergebnis zu erreichen, welches man möchte (Voraussetzung: die Datenbank bietet dieses Ergebnis).

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

### Zusatzaufgabe für die treuen LeserInnen

Geben Sie für jedes Land aus, wie viele Abteilungen es dort gibt! Wenn in einem Land 0 Abteilungen sind, soll 0 ausgegeben werden (bei Lösungsvorschlägen bitte melden!)!

Die Lösung des 1. Satzes ist aus Sicherheitsgründen verkehrt herum (NICHT SCHUMMELN! 😊):

Ein Bild, das Text enthält.

Automatisch generierte Beschreibung

# Kommentar

Dieses Protokoll beschäftigt sich hauptsächlich mit DQL (Data Query Language).