

## Übung 11

### Advanced SQL 2: Division,

Für dieses Übungsblatt nutzen Sie bitte das vorbereitete SQL Script  
DB - KH -Uebung10 Testdaten.sql aus der Community.

#### Aufgabe 1: Division

Geben Sie mit Hilfe der Division alle Veranstaltungen aus, für die jeder Student eine Note bekommen hat. Also anders formuliert: Alle Veranstaltungen, die vollständig bewertet wurden.

#### Aufgabe 2: Integrität

- a) Verändern Sie die Tabellendefinition von `Student_in_Veranstaltung` so, dass wenn ein Student exmatrikuliert wird, automatisch auch alle Noten entfernt werden.
- b) Verändern Sie die Tabellendefinition von `Veranstaltungen` so, dass wenn eine Dozent kündigt, seine Veranstaltungen keinen Dozenten mehr haben.
- c) Verändern Sie die Tabellendefinition von `Veranstaltungen` so, dass wenn der Name eines Dozenten sich durch Heirat verändert, sich der Name auch automatisch in der Tabelle `Veranstaltungen` verändert.
- d) Verändern Sie die Tabellendefinition von `Student_in_Veranstaltung` so, dass wenn der Name einer Veranstaltung sich ändert, automatisch auch der foreign key verändert wird.

#### Aufgabe 3: Trigger

Erstellen Sie einen Trigger, der jedes mal, wenn ein Dozent erfolgreich in die Tabelle `Dozenten` eingetragen wurde, automatisch eine Veranstaltung mit dem Titel „Einführung <Dozentname>“ erstellt. Sie können die Datumsfunktionen verwenden, um die Veranstaltung in dem nächsten Semester ab Einfügedatum stattfinden zu lassen oder als Vereinfachung `ss99` annehmen.

#### Aufgabe 4: Sichten und Zugriffskontrolle

- a) Speichern Sie eine Sicht in der Datenbank die Sie `Veranstaltungsplan` nennen und aus der für alle Studenten ersichtlich wird, wann sie wo eine Veranstaltung mit wem haben.
- b) Gestatten Sie lesenden Zugriff für die Benutzergruppe `public` und entziehen Sie der Benutzergruppe `public` die Rechte Einfügen und Löschen für die Sicht `Veranstaltungsplan`. Melden Sie sich mit dem Benutzer `public` an ihrer Datenbank an und überprüfen Sie das Ergebnis.