



Prof. Dr. Bernhard Seeger
Jana Holznigenkemper, M.Sc.
Andreas Morgen, M.Sc.

Übungen zur Vorlesung
Datenbanksysteme I

Abgabe: 03. 05. 2019,
bis **spätestens** 10:00 Uhr
über die ILIAS Plattform

Übung 1

Datenbankschema

Auf diesem Übungsblatt wird in jeder Aufgabe folgende Datenbank verwendet:

LIEFERANT (<u>LNr</u> , LName, Ort)	PROJEKT (<u>PNr</u> , PName, Ort)
LIEFERUNG (<u>LNr</u> , <u>TNr</u> , <u>PNr</u> , Menge)	TEIL (<u>TNr</u> , TName, Farbe, Gewicht)

Aufgabe 1.1: Relationale Algebra (1+1+2+3) (7 Punkte)

Geben Sie Ausdrücke der relationalen Algebra zur Beantwortung der folgenden Anfragen an:

- a) Ermitteln Sie alle blauen Teile, die weniger als 8kg wiegen.
- b) Wie heißen die Projekte aus Marburg?
- c) Geben Sie die Teilnummern derjenigen Teile an, die in keiner Lieferung stehen.
- d) Wie ist die Projektnummer aller Projekte, die genau einmal beliefert wurden?

Aufgabe 1.2: Apache Flink (2+4) (6 Punkte)

Apache Flink¹ ist ein Open-Source-Framework zur verteilten Verarbeitung von Streaming-Daten, vom Berliner Startup Data Artisans, welches im Januar diesen Jahres von der Alibaba Gruppe übernommen wurde. Apache Flink ermöglicht mit seiner Table-API Daten mit Operatoren der relationalen Algebra auszuwerten.

- a) Um sich in Flink einzuarbeiten lesen Sie sich in die Dokumentation der Table-API unter <https://ci.apache.org/projects/flink/flink-docs-stable/dev/table/tableApi.html> ein. Beantworten Sie die folgenden beiden Fragen:
 - i) Wie wird mit der Table-API eine Projektion durchgeführt?
 - ii) Wie wird ein Join (Inner Join) durchgeführt?
- b) Betrachten Sie erneut das Datenbankschema und die dazugehörigen Anfragen aus Aufgabe 1.1. Werten Sie die Anfragen in Flink aus. Laden Sie dazu das im Ilias bereitgestellte Maven-Projekt (Aufgabe1.2.zip) herunter, entpacken es und öffnen es anschließend in der IDE IntelliJ². Ergänzen Sie das Projekt an der markierten Stelle in der Datei AufgabeTableAPI.java um die oben beschriebenen Anfragen. Bei der Abgabe genügt es

¹<https://flink.apache.org/>

²<https://www.jetbrains.com/idea/>

diese Datei mit ihrem Namen als Autor abzugeben. Bei technischen Problemen können in den Tutorien Hilfestellungen gegeben werden.

Aufgabe 1.3: Relationale Algebra / Theorie (1+2+2)

(5 Punkte)

Seien R , S und T Relationen. Zeigen oder widerlegen Sie:

- a) $\sigma_{\theta_1}(\sigma_{\theta_2}(R)) = \sigma_{\theta_1 \wedge \theta_2}(R)$
- b) $\pi_{RS_R}(R \bowtie_{\theta} S) \bowtie_{\theta} S = R \bowtie_{\theta} S$
- c) $(R \bowtie_{\theta_1} S) \bowtie_{\theta_2} T = R \bowtie_{\theta_1} (S \bowtie_{\theta_2} T)$

Aufgabe 1.4: DBMS vs. AWP (2+1*)

(2 Punkte)

Nennen Sie zwei Gründe, wieso Integritätsbedingungen zentral vom DBMS verwaltet und nicht in den Anwendungsprogrammen selbst realisiert werden sollen.

Bonusfrage*: Was spricht dafür, dass Anwendungsprogramme die Integritätsüberprüfung durchführen und nicht das DBS?