

# freiwilliges semesterbegleitendes Übungsblatt

## Aufgabe 1: Das Datenmodell als UML-Klassendiagramm

Ihr Team ist ein Unternehmen, das ein Datenmodell „Buchungssystem für Seminare“ zu erstellen hat. Für die Datenhaltung dieses Systems müssen Sie nun ein vollständiges UML-Klassendiagramm mit Java-Datentypen in angemessener Komplexität strikt nach UML-Notationsübersicht 2.5 entwerfen. Stimmen Sie die nötigen konkreten Inhalte rechtzeitig mit Ihrem Kunden Prof. Dopatka ab. Rahmenbedingungen sind:

- Es sind mind. 6 komplexe Hauptklassen mit Navigation und Multiplizitäten zu erstellen.
- Es müssen mind. 2 Enums vorhanden sein.
- Es muss mind. eine Vererbung auftreten zwischen einer Oberklasse und mind. zwei Unterklassen.
- Es muss mind. eine 1:1-Beziehung realisiert werden.
- Es muss mind. eine 1:\*-Beziehung realisiert werden.
- Es muss mind. eine \*:\*-Beziehung realisiert werden.
- Es muss mind. eine Aggregation oder Komposition realisiert werden.

## Aufgabe 2: Die Datenbank & phpMyAdmin

2a) Überführen Sie das Datenmodell aus Aufgabe 1 in ein ER-Diagramm strikt gemäß der Notation, die in den Vorlesungsfolien vorgestellt wurde. Beschreiben Sie zusätzlich textuell und anhand von Beispieldaten, wie Sie die Vererbung umsetzen wollen. Welchen vorgestellten Weg aus den Vorlesungsfolien wollen Sie gehen?

2b) Realisieren Sie nun das abgenommene Datenmodell via phpMyAdmin in einer MySQL-/MariaDB-Datenbank. Erstellen Sie die nötigen Tabellen/Relationen mit sinnvollen Datentypen und referenziellen Integritäten strikt gemäß Ihres Datenmodells. Füllen Sie die Tabellen mit einer großen Anzahl an sinnvollen Testdaten.

2c) Definieren Sie je 6 einfache und 6 komplexe sinnvolle SQL-Statements und beschreiben Sie jedes einzelne Statement sehr präzise in einem deutschen Fließtext. Ein komplexes Statement ist mindestens über 3 Tabellen übergreifend, besser über mehr. Mindestens 50% der einfachen und der komplexen Statements müssen parametrisiert sein. Die SQL-Statements müssen sich direkt via phpMyAdmin auf ihren Testdaten ausführen lassen.

## Aufgabe 3: Zugriff über eine Anwendung

Erstellen Sie eine Anwendung, mit der man problemlos und bequem über verschiedene SQL-Befehle innerhalb der Anwendung Abfragen aus der Datenbank ausführen kann. Dabei sind die einfachen und komplexen SQL-Statements aus Aufgabe 2 umzusetzen. Alle Parameter müssen über Prepared Statements an die Datenbank übergeben werden. Zusätzlich dazu müssen Sie die CRUD-Operationen implementieren; dies können Sie sinnvollerweise zur vollständigen Verwaltung einer sinnvollen Haupttabelle realisieren. Die Ausgabe von SELECT-Statements müssen tabellarisch aufbereitet werden, beispielsweise unter Verwendung geeigneter Collections in Java.

Als Anwendung eignet sich beispielsweise eine PHP<sup>1</sup>- oder eine Node.js/Express.js-Webseite. Aber auch Anwendungen in Python, Java<sup>2</sup> als Konsolenanwendung oder Swing-GUI sind möglich. Die Java-Anbindung ist Gegenstand der DM-Vorlesung und daher relevant für den Test und die Klausur.

Zur Kontrolle muss man via phpMyAdmin direkt auf die Datenbankinhalte zugreifen können.

<sup>1</sup> <https://www.youtube.com/watch?v=BqF2bU6ESLU&list=PL35XGjy0AQAqb0LdnV-wWbhbu6ZM1uCrn&index=25>

<sup>2</sup> [https://www.youtube.com/watch?v=c\\_FG\\_3TFfds&list=PL35XGjy0AQAo-zLfpISlyLB3DqVl1sh3A&index=9](https://www.youtube.com/watch?v=c_FG_3TFfds&list=PL35XGjy0AQAo-zLfpISlyLB3DqVl1sh3A&index=9)