



#### KN02: Datenmodellierung für MongoDB

- A) Konzeptionelles Datenmodell (40%)
- B) Logisches Modell für MongoDB (50%)
- C) Anwendung des Schemas in MongoDB (10%)

## KN02: Datenmodellierung für MongoDB

---

Beachten Sie die [allgemeinen Informationen zu den Abgaben](#).

In diesem Kompetenznachweis erstellen Sie ein Datenmodell für ihr gewähltes Thema. Die Grundlagen können Sie **in der Theorie nachlesen**.

**Wichtig:** In diesem Kompetenznachweise arbeiten Sie noch nicht mit JSON-Schemas. Das Thema in der Theorie also noch auslassen. Sie werden später damit arbeiten.

### A) Konzeptionelles Datenmodell (40%)

---

Das konzeptionelle Datenmodell kann als Grundlage für jeden Datenbank-Typ verstanden werden. Unterschiede ergeben sich erst im nächsten Schritt. Sie haben [im Modul 162 gelernt](#) wie Sie ein konzeptionelles Datenmodell erstellen.

Wählen Sie für sich persönlich ein Thema aus, **welches von den anderen Klassenkameraden abweicht**, z.B. Ihr Sport-Club, Musik-Band, Freizeitaktivitäten, etc. Sie werden dieses Thema im **gesamten** Modul verwenden.

Erstellen Sie nun ein konzeptionelles Datenmodell, welches Sie in allen zukünftigen Aufgaben als Basis verwenden. **Bedingungen:**

- Mindestens 4 Entitäten
- Mindestens eine netzwerkförmige Beziehung

#### Abgaben:

- Draw.io Diagramm (oder ähnliches Tool) des konzeptionellen Diagramms
- Bild des konzeptionellen Diagramms.
- Kurze textuelle Erklärungen zu den Entitäten und Beziehungen.

### B) Logisches Modell für MongoDB (50%)

---

In der [Theorie](#) finden Sie die Information wie Sie das das logische Datenmodell erstellen können.

**Achtung:** Zu diesem Zeitpunkt erstellen Sie noch **kein** JSON-Schema. Erstellen Sie eine Zeichnung der Verschachtelung Ihres Modells, so wie dies in der Theorie gemacht wurde. Beachten Sie, dass es zur visuellen Darstellung kein Standard gibt an den Sie sich halten können.

#### Bedingungen:

- Im Schnitt wird pro Entität 3 Attribute erwartet.
- Verschiedene Basis-Datentypen finden Einsatz (int, float, string, char).
- Mindestens einmal muss ein *date*-Datentyp verwendet werden.
- Es muss **mindestens eine Verschachtelung** enthalten. Falls dies inhaltlich keinen Sinn ergibt, überdenken Sie Ihr konzeptionelles Modell und passen Sie es an.

**Abgaben:**

- Ein Bild des logischen Datenmodells.
- Die Original-Datei des logischen Datenmodells (z.B. draw.io)
- Erklärung zu Verschachtelungen. wieso haben Sie Ihre Variante gewählt.

## C) Anwendung des Schemas in MongoDB (10%)

---

Erstellen Sie nun eine neue Datenbank mit einem sprechenden Namen, der zu Ihrem Thema passt. Es reicht, wenn Sie mit `use <Datenbankname>;` in Ihre Datenbank wechseln - ohne dass sie existiert. Sie müssen diesen Befehl einzeln aufrufen und nicht zusammen mit den folgenden.

Erstellen Sie die verschiedenen Collections mit [dem Befehl `createCollection`](#) `db.createCollection()`. Sie können eine .js-Datei (JavaScript) speichern und so später wiederverwenden.

**Abgaben:**

- Script mit den Befehlen zur Erstellung der Collections.